PECAN: Leveraging Policy Ensemble for Context-Aware Zero-Shot Human-AI Coordination

Xingzhou Lou[*]
Institute of Automation, Chinese
Academy of Sciences
Beijing, China
School of Artificial Intelligence,
University of Chinese Academy of
Sciences
Beijing, China
louxingzhou2020@ia.ac.cn

Jiaxian Guo
The University of Sydney
Sydney, Australia
jguo5934@uni.sydney.edu.au

Junge Zhang[†]
Institute of Automation, Chinese
Academy of Sciences
Beijing, China
School of Artificial Intelligence,
University of Chinese Academy of
Sciences
Beijing, China
jgzhang@nlpr.ia.ac.cn

Jun Wang
University College London
London, United Kingdom
j.wang@cs.ucl.ac.uk

Kaiqi Huang
Institute of Automation, Chinese
Academy of Sciences
Beijing, China
School of Artificial Intelligence,
University of Chinese Academy of
Sciences
Beijing, China
kqhuang@nlpr.ia.ac.cn

Yali Du[†]
King's College London
London, United Kingdom
yali.du@kcl.ac.uk

ABSTRACT

Zero-shot human-AI coordination holds the promise of collaborating with humans without human data. Prevailing methods try to train the ego agent with a population of partners via self-play. However, these methods suffer from two problems: 1) The diversity of a population with finite partners is limited, thereby limiting the capacity of the trained ego agent to collaborate with a novel human; 2) Current methods only provide a common best response for every partner in the population, which may result in poor zero-shot coordination performance with a novel partner or humans. To address these issues, we first propose the policy ensemble method to increase the diversity of partners in the population, and then develop a context-aware method enabling the ego agent to analyze and identify the partner's potential policy primitives so that it can take different actions accordingly. In this way, the ego agent is able to learn more universal cooperative behaviors for collaborating with diverse partners. We conduct experiments on the Overcooked environment, and evaluate the zero-shot human-AI coordination performance of our method with both behavior-cloned human proxies and real humans. The results demonstrate that our method significantly increases the diversity of partners and enables ego agents to learn more diverse behaviors than baselines, thus achieving state-of-the-art performance in all scenarios. We also open-source a human-AI

coordination study framework on the Overcooked for the convenience of future studies. Codes and demo videos are available at https://sites.google.com/view/pecan-overcooked.

KEYWORDS

Zero-shot Human-AI Coordination; Multi-agent; Reinforcement Learning

1 INTRODUCTION

Reinforcement learning (RL) has shown remarkable success in various domains, such as gaming AI [10, 16, 42, 43, 51], robotic manipulation [13, 24], traffic control [11, 12], etc. However, a significant challenge remains in constructing agents that can collaborate effectively with unseen partners, which is especially important under human-AI coordination. Many real-world applications of human-AI coordination, such as cooperative games [14], self-driving vehicles [28, 37] and AI assistants [1, 22], can be modeled as zero-shot human-AI coordination tasks. By avoiding the expensive human data collection and human involvement during training, zero-shot human-AI coordination holds the promise of more accessible AI systems that can enhance human capabilities. In this approach, an ego agent is trained with partner agents and later interacts with human proxy models or real humans.

Existing methods mainly vary in how the partner agents are acquired and how the ego agent is trained. Self-play methods tried

---

[*]Work done while visiting King's College London.
[†]Correspondence.

to train ego agents through self play [3, 4, 43, 44], in which the ego agent is trained to collaborate with a copy of itself. However, this approach has been shown to result in over-fitting to a single cooperative pattern [19, 26] and poor generalization to real human collaborations. To address this issue, population-based training (PBT) [26, 46, 56] has been employed, where a population of diverse partner agents is used to train the ego agent. The variety of behaviors exhibited by these diverse partners can prevent over-fitting to a single cooperative pattern and improve generalization ability of the ego agent when cooperating with real humans.

However, there are still two limitations of current PBT methods: 1) The finite number of partners in the population restricts the behavioral diversity, making it difficult for the ego agent to coordinate with new partners, particularly those with unique behaviors. Although increasing the population size can address this issue, it requires significant computational resources and decreases the learning efficiency of the ego agent. 2) The ego agent learns a common best response (BR) for every partner, regardless of their behavior patterns. This partner-specific common BR can lead to unsatisfactory human-AI coordination performance as the ego agent lacks the ability to adapt its policy based on the partner's type and behavior pattern.

To address these issues, we propose **P**olicy **E**nsemble **C**ontext-**A**ware zero-shot human-AI coordinatio**N** (PECAN), where the policy ensemble method is proposed to increase the diversity of partners without increasing the population size, and the context-aware module is proposed to identify whether the partner is good or poor at the given task, *i.e.* the level of coordination skills. Thus, the ego agent is able to learn level-based common BR rather than common BR for specific partners in the population, which allows the ego agent to acquire more universal coordination behaviors and better coordinate with novel partners.

Specifically, the proposed policy ensemble can generate a new partner whose policy is the weighted average of policy primitives in the population. Since the weights are randomly generated, the policy-ensemble partners are distinct in each iteration. This increases partner diversity and improves the ego agent's ability to collaborate with unseen partners. Additionally, we find that partners created by mixing policies from the same level display better behavioral diversity than those created from the entire population (see Fig. 4 and section 5.2). The context-aware module in PECAN is designed to identify the partner's level of coordination skills based on past trajectories. This is achieved through supervised learning. The training data is collected by rolling out various policy ensembles and assigning the corresponding levels of the ensembles' policy primitives as labels. During evaluation, the ego agent updates its recognized context at the start of each episode based on the past trajectory and uses this context to condition its actions.

Our proposed approach, PECAN, presents three main contributions to zero-shot human-AI coordination. 1) PECAN trains generalizable agents without relying on human data. 2) The use of level-based policy-ensemble partners and a context-aware module enhances population diversity without increasing the population size, allowing the ego agent to learn level-based common best responses. 3) PECAN achieves superior performance compared to state-of-the-art baselines in the Overcooked environment [14], as demonstrated by our experimental results and additional studies.

2 RELATED WORK

Zero-shot Coordination Zero-shot coordination (ZSC) has been studied in multiple previous studies [7, 38, 48]. In ZSC framework introduced by [19], two independently trained agents are paired together to fulfill a common purpose in a cooperative game. The paired agents will never encounter each other during training. Thus, the agents must employ compatible policies and should not overfit to any arbitrary partners or cooperative patterns. Training the agents with diverse partners is effective to alleviate over-fitting to specific partners and improve ZSC performance. Population-based training (PBT) methods [26, 46, 56] have achieved state-of-the-art performance in ZSC. In Fig. 1(b), by maintaining a diverse population of training partners, the ego agent in PBT is able to collaborate with a diverse set of partners. FCP [46] trains a diverse population by setting different random seeds and including partners of level of cooperation skills and architectures. TrajeDi [26] and MEP [56] adopt explicit diversity objective to generate diverse policies as partners and achieve state-of-the-art ZSC performance. Our PECAN also maintains a population of policies. But the population is used to provide policy primitives for the policy ensemble rather than partners for the ego agent.

Human-AI Coordination Much previous work in human-AI coordination focuses on planning and learning with human models [5, 23, 31, 39]. However, human-AI coordination can be naturally modelled as ZSC tasks, because humans are usually not involved in the training process. And humans usually prefer adaptive AI partners [46]. Thus, an adaptive agent with strong ZSC performance is more likely to succeed in human-AI coordination tasks [46, 56].

Diversity in RL Diversity is a widely discussed issue in RL. SAC [15] encourages maximum entropy over the action distributions to improve a single policy's diversity. [18] maximizes the KL divergence between the current policy and some recent policy to encourage diverse action choices. Instead of the diversity of a single policy, many works also focus on diversity of a policy group. [27] shows that a team of weak yet diverse agents can even defeat teams of strong but uniform agents under certain conditions. [9] proposed to use generative model to generate diverse policies. MEP [56] maximizes the population entropy to encourage diversity among a group of policies. EPPO [53] uses mean inner product as diversity enhancement regularization and obtains a mutually distinct policy group. Besides diversity over action distributions, multiple previous studies also focus on the diversity over the induced trajectories. DIPG [29] adopts maximum mean discrepancy (MMD) of the induced trajectories as the metric for diversity among policies and encourage qualitatively distinct behaviors. Other measures of diversity such as Jensen-Shannon Divergence [26] and mutual information [6] are also adopted to improve diversity among policies and agents in (multi-agent) reinforcement learning. The partner diversity in PECAN is provided by 1) diversity enhancement regularization when generating policy primitives; 2) random selection as well as the random weights of policy primitives when generating policy-ensemble partners.

Policy Ensemble Policy ensemble (mixture of experts) [20] is the mixture of a group of policy primitives [47], which is able to work individually in the target task. PMOE [36] models policy ensemble as a Gaussian Mixture Model (GMM) with learnable weights

and practically shows the improved diversity over both action distribution and induced trajectories. EPPO [53] uses the arithmetic mean of the policy primitives as the policy ensemble. The learned policy ensemble achieves both high sample-efficiency and strong performance. Instead of learnable weights or arithmetic mean, the weights of policy primitives in PECAN are randomly generated to further improve diversity of the policy ensemble.

3 PROBLEM SETTING

Two-player Markov Decision Process We model the problem as a two-player Markov Decision Process (MDP) $\mathcal{M} = \langle \mathcal{S}, \{A^i\}, \mathcal{P}, \gamma, R \rangle$ [2]. $\mathcal{S}, \gamma$ are the state space and discount factor. A^i is the action space for the i-th agent where $i \in \{1, 2\}$. $\mathcal{P} : \mathcal{S} \times A^1 \times A^2 \times \mathcal{S} \to [0, 1]$ is the transition dynamics. The objective is to maximize the expected reward sum over trajectories $\mathbb{E}_\tau [R(\tau)] = \mathbb{E}_\tau \left[\sum_t^T R(s, a^1, a^2) \right]$, where $a^1 \in A^1, a^2 \in A^2$. In our setting, each agent has its own observation and decentralized policy.

Cooperative Tasks Different from competitive tasks [32, 43, 51] where agents tend to exploit each other, cooperative tasks require agents to collaborate rather than compete and fulfill a shared purpose. In this paper, we consider the problem of common-payoff two-player cooperative games. See the formal representation of two-player cooperative games in [30] and an intuitive comparison between zero-sum competitive games and common-payoff cooperative games in [5]. MARL can be applied to both competitive [17, 32, 43, 52] and cooperative tasks[8, 25, 35, 54]. In our cooperative setting, MARL learns a joint-policy $\boldsymbol{\pi} = [\pi_1, \pi_2]$ of the two agents so that their expected reward sum over the induced trajectories $\mathbb{E}_{\tau \sim \boldsymbol{\pi}} [R(\tau)]$ is maximized. In PECAN, both the agents in the population and the ego agent are trained by PPO [41].

4 METHODOLOGY

In this section, we will give the details of PECAN. First, we will introduce the population of policy primitives in PECAN. Then, we will introduce the proposed policy-ensemble module and the context-aware module respectively.

4.1 Population in PECAN

Like previous PBT methods [26, 46, 56], PECAN first maintains a population of training partners. To obtain a population, multiple diversity enhancement regularizations have been proposed such as mean inner-product among policies [53], KL divergence among policies [18] and maximum mean discrepancy [29]. In our paper, we directly use the MEP method [56] to constitute the population considering its computational efficiency and effectiveness.

Specifically, the agent in the population is trained by self-play to learn cooperation ability and maximize the population entropy (*PE*) in [56], which quantifies the diversity of the population. The formulation of PE is

$$PE(\{\pi^1, \pi^2, ..., \pi^n\}, s_t) = \mathcal{H}(\bar{\pi}(\cdot|s_t)) \tag{1}$$

where $\bar{\pi}(\cdot|s_t) = \frac{1}{n} \sum_{i=0}^n \pi^i(\cdot|s_t)$ is the mean policy of the population, and $\mathcal{H}(\bar{\pi}(\cdot|s_t))$ is the entropy of the mean policy.

For agent i, the objective of self-play training $J(\pi^i)$ is

$$J(\pi^i) = \sum_t \mathbb{E}_{(s_t, \boldsymbol{a}_t) \sim \boldsymbol{\pi}} [R(s_t, \boldsymbol{a}_t) + \alpha \mathcal{H}(\bar{\pi}(\cdot|s_t))] \tag{2}$$

where $\boldsymbol{\pi} = [\pi^i, \pi^i]$ is the joint policy, $\boldsymbol{a}_t$ is the joint-action sampled from $\boldsymbol{\pi}$, R is the reward function and α is the temperature parameter controlling the relative importance of entropy maximization. By maximizing Eq. 2, agent i will master high-level cooperation skills, and its policy π^i will be encouraged to diversify the current population.

In previous PBT methods [5, 46, 56], after the population of training partners is obtained, the ego agent will select a partner from the population for training in each iteration. The partner is selected by either uniform sampling [5, 46] or prioritized sampling [56]. However, although the diversity enhancement regularization is adopted, the population diversity is still limited because the population is finite. Moreover, in their training procedure, the ego agent only learns a common BR to every partner in the population, which limits its capacity to coordinate with a novel partner in zero-shot evaluation, since the novel partner is never exposed to the ego agent during training.

4.2 Diverse Partners with Policy Ensemble

In this subsection, we will introduce how we adopt policy ensemble to improve the diversity of partners without increasing the population size. Figure. 1(c) gives the idea of our policy ensemble.

Instead of directly using policy π^i, $i \in [1, n]$ in a policy set $G : \{\pi^1, \pi^2, ..., \pi^n\}$, we use an ensembled policy π_p over this set G as the training partner for the ego agents. Specifically, π_p is the weighted average of policies in G, *i.e.*,

$$\pi_p(\cdot|s) = \sum_{\pi^i \in G} \omega_i \pi^i(\cdot|s)$$

$$where \ \omega_i \geq 0 \ and \ \sum_{i=1}^{|G|} \omega_i = 1 \tag{3}$$

where ω_i is the weight of π^i. Intuitively, ω_i decides the contribution of π^i to the policy ensemble π_p. In particular, if the weight of one policy is 1 and the weights of the other policies are 0, the policy ensemble will degenerate into a single policy in the population (as previous PBT method did [26, 46, 56]). In contrast to this special case, we randomly assign different weights to each training iteration, and we observe that the generated partners typically exhibit different policy behaviour than the policy in the population (refer to Fig. 4). In this way, the partners generated by our method are more diverse than those generated by the vanilla PBT method, even if the population size is the same.

We split the population into three groups G_1, G_2 and G_3 for low, medium and high level of agents according to their self-play performance. Specifically, the checkpoints from different training stages (initial, middle and final) to the population of the corresponding level following [46, 56]. For example, the low-level group consists of the initial models from the population, the medium-level group consists of the middle models and the high-level group consists of the final models. And we also provide a self-play group G_4 as in [56], which only contains a copy of the ego agent itself, to improve the ego agent's self-play performance.
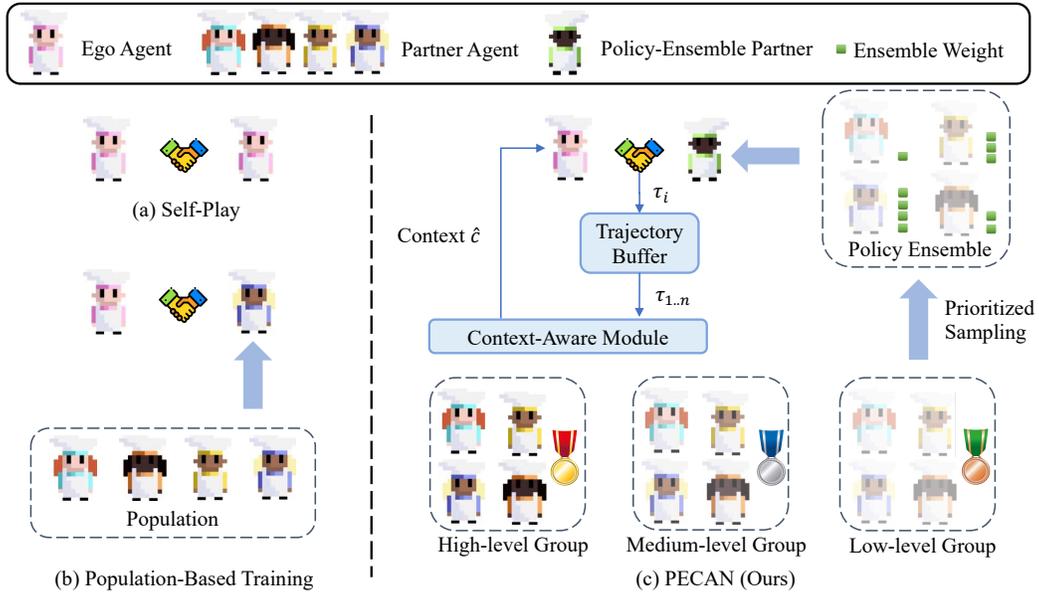
Figure 1: (a) Self-play training (SP). The ego agent is trained with a copy of itself. (b) Population-based training (PBT). The ego agent is trained with a population of partners. A partner is sampled at each iteration to cooperate with the ego agent. (c) The proposed PECAN method. A different policy-ensemble partner is generated at each iteration. The ego agent will collect trajectories during collaborating with the partner and recognize the partner's level-based context at the beginning of each episode with a pretrained context-aware module introduced in section 4.3.

The mixture of policy primitives from the same group tends to remain within the same group, giving partners greater control over their levels. If the ego agent wishes to learn the level-based common BR introduced in a later section, the controllable level of partners is essential. Please refer to section 5.2 for empirical experiments validating that level-based grouping can provide partners with controllable levels. It is interesting that we also discovered that level-based grouping can increase more partner diversity . It is consistent with the previous results in [46] that level of skills is an important factor of partner diversity.

In each training iteration, one group is chosen to generate a policy ensemble for training the collaboration ability of the ego agent. Instead of randomly sampling the group, we employ a group-level prioritised sampling strategy that selects groups based on the average performance of the ego agent with each group $J(G_i)$. This can stabilise the training process and guarantee that the trained ego agent can collaborate effectively with all groups. Specifically, we use rank-based prioritized sampling to assign higher priority to the group with which the ego agent is hard to cooperate as in [52, 56]. The probability of group G_i being sampled is

$$p(G_i) = rank\left(\frac{1}{J(G_i)}\right)^{\beta} \bigg/ \sum_{j=1}^{4} rank\left(\frac{1}{J(G_j)}\right)^{\beta} \qquad (4)$$

where β is the hyperparameter controlling the strength of prioritization. When $\beta = 0$, the prioritized sampling degenerates to uniform sampling as all groups have the same priority. And when $\beta \rightarrow \infty$, the group with the worst average performance will be selected with probability 1. As a smooth approximation of the "maximize minimal" paradigm, group-level prioritized sampling helps the ego agent learns to coordinate with partners at the level with the worst coordination performance, and thus can avoid the problem of over-exploiting easy-to-cooperate partners [56].

With the randomly assigned weight ω for each policy primitives and the rank-based prioritized group sampling, the policy-ensemble partners generated in each training iteration are different by design, and thus more diverse than previous methods. To stabilize the learning process, at the beginning of training, the partners have a high probability to be agents directly from the population rather than policy ensembles, during which the ego agent can learn basic cooperation skills. As training goes on, the partners are more likely to be policy ensembles, and the ego agent will collaborate with partners that have more diverse behaviours.

4.3 Context-Aware Ego Agent

In current methods, the ego agent only learns a common BR to every partner in the population, which limits its capacity to coordinate with an unfamiliar novel partner. Regardless of diversity of partners, this problem occurs whenever the novel partner in zero-shot evaluation is different from those during training. To alleviate this problem, we propose to learn level-based common BR for the ego agent instead of common BR to specific partners.

In order to learn a level-based BR, we devise a context encoder f to help the ego agent analyze and predict the level of its partner as policy context. Context identification by trajectories has been previously studied [34]. But different from their context whose distribution is Gaussian, our contexts are class labels indicating the corresponding group $G_{\{1,2,3\}}$ for low, medium and high level partners, which is determined by their training time as in [46]. Therefore, we model the context encoder as a classifier in Fig. 3.
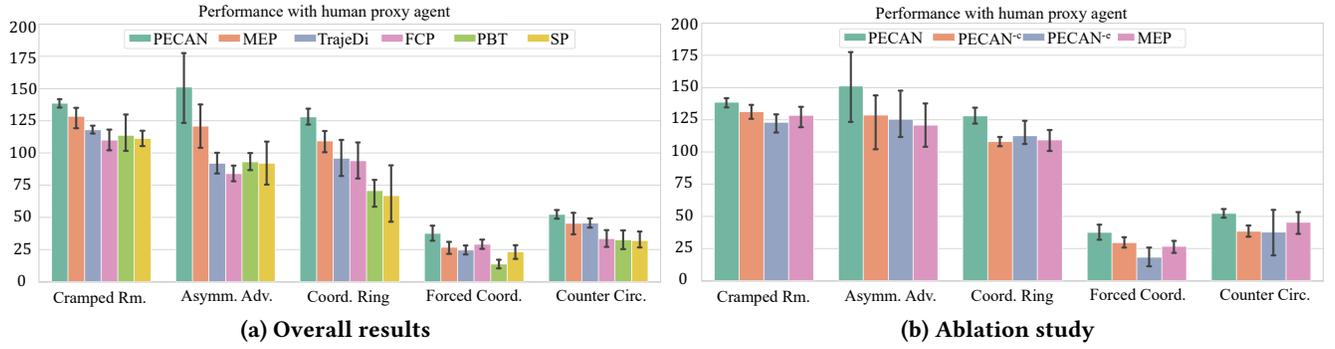
Figure 2: (a) Overall results with a human proxy model. PECAN outperforms the baselines on all five layouts. (b) Results of the ablation study. The performance drops when we ablate each component from PECAN, and in some layouts the performance is even worse than the baseline method.
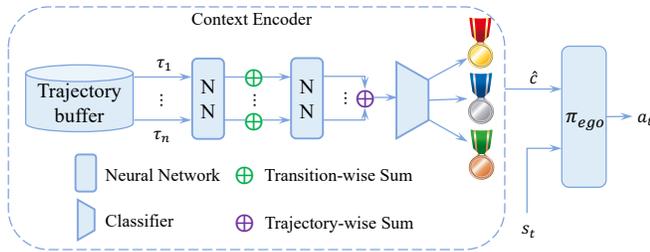

Figure 3: The context-aware module predicts the partner's level-based context $\hat{c}$. τ_i are past trajectories with the same partner from the trajectory buffer. Transition-wise and Trajectory-wise sum are taken respectively to make sure the predictions are irrelevant w.r.t. their orders. The ego agent π_{ego} conditions on $\hat{c}$ to make decisions. The context encoder predicts $\hat{c}$ at the beginning of every episode.

After the population is obtained, we generate many policy-ensemble partners of different levels and roll out the partners to collect training data $\tau_{\{1,..,N\}}$ and corresponding level-based labels $c_{\{1,..,N\}}$ for the context encoder. The network is updated by gradient descent to predict context $\hat{c}$ with the one-hot label c. And the loss function $\mathcal{L}$ is the cross-entropy between $\hat{c}$ and c in Eq. 5.

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{3} c_{ij}\log \hat{c}_{ij} \tag{5}$$

where N is the batchsize. Each input τ in the batch is a set of trajectories with the same label, and $\hat{c}_{ij} = f_j(\tau_i)$ is the probability of group j with input τ_i.

We predict the context from past trajectories. The sequence of transitions in the trajectory are encoded by a self-attention [50] module followed by multi-layer perceptrons (MLP). The order of transitions and trajectories is irrelevant to the partner's level. Therefore, we take sum over both encoded transitions within a trajectory and encoded trajectories, so that the prediction result is permutation-invariant w.r.t. transitions in a trajectory and trajectories in the buffer. This architecture has the capacity to represent any permutation-invariant function [55]. More implementation details of PECAN are given in the supplementary material.

By conditioning on the predicted context $\hat{c}$ indicating the partner's level, the ego agent's policy $\pi_{ego}(\cdot|s,\hat{c})$ learns level-specific coordination skills, which are more universal than previous partner-specific coordination skills.

The ego agent will learn to coordinate with partners based on their level-based context, and thus reach a level-based common BR to partners of different levels. Different from common BR to specific partners, the level-based common BR is more universal and enables the ego agent to better coordinate with unfamiliar novel partners by analyzing their level of skills and taking actions accordingly.

During zero-shot evaluation with a novel partner, we use the collected trajectories to infer context of the partner. Akin to posterior sampling [33, 34, 45], as more trajectories are collected, the ego agent's belief narrows, and the prediction becomes more accurate. But we would like to note that the process is different from posterior sampling, since we model the context identification as a mapping from trajectories to classes rather than a distribution. Although the context inference uses past trajectories, we do not fine-tune or update any parameters during evaluation. Thus, the evaluation with a novel partner is still zero-shot.

5 EXPERIMENTS

In this section, we will first introduce the tasks, baselines and procedures that we adopt to evaluate zero-shot human-AI coordination performance of PECAN. Then, experimental results of coordinating with human proxy models and real human players will be given. Finally, case studies are conducted to show the adaptiveness of PECAN intuitively.

5.1 Experimental Setting

Tasks We follow the evaluation protocol proposed in [5] and evaluate the proposed method on a challenging collaborative game Overcooked [5, 14]. Five layouts (*Cramped Room, Asymmetric Advantages, Coordination Ring, Forced Coordination* and *Counter Circuit*) in Overcooked are adopted to evaluate the ego agent's ability to coordinate with some novel partners. See more details of the layouts in [5]. Each layout exhibits a unique challenge, which can be overcome if the players coordinate well with each other. The players are required to put three onions in a pot, collect an onion soup from the pot after 20 timesteps and deliver the dish to a counter.

(a) Effect of policy ensemble on diversity of partners

(b) Effect of the recognized context c
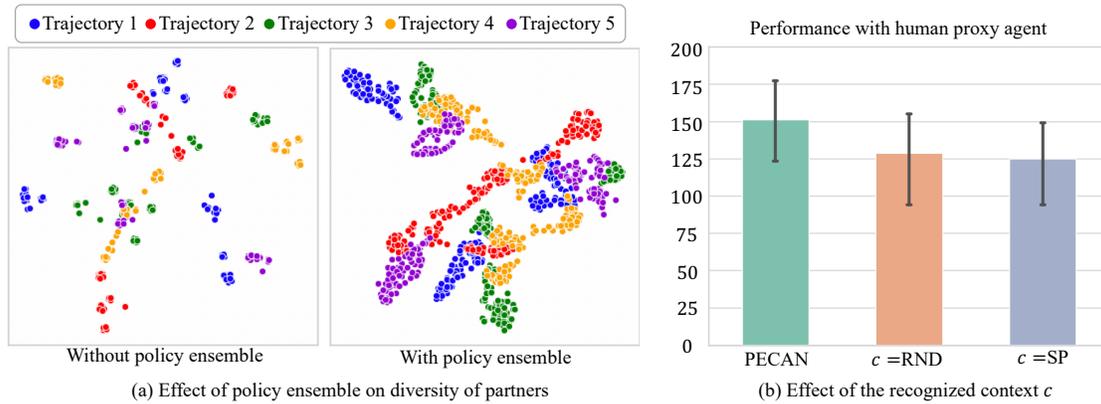
Figure 4: Effects of each component in PECAN. (a) the t-SNE results of partners' policy $\pi_p(\cdot|s)$ in each episode. The results show policy ensemble is able to improve partners' diversity. (b) $c = RND$ means feeding a random context to the ego agent and $c = SP$ means manually assigning context for the ego agent to indicate that the partner is from the self-play group G_4. The results show that only with the correct context, the ego agent can perform well, which means the ego agent's policy is context-aware.

(a) Trajectory 1   (b) Trajectory 2   (c) Trajectory 3   (d) Trajectory 4
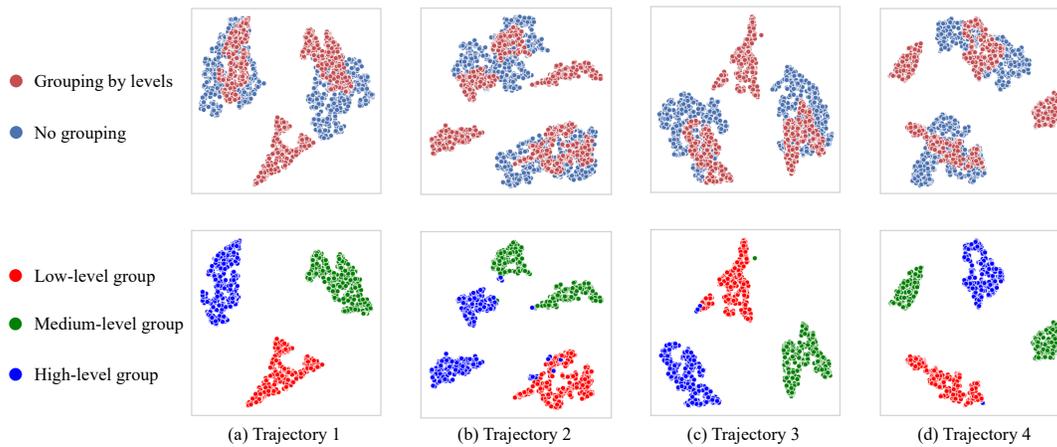
Figure 5: Comparison between policy ensembles with and without level-based grouping. The ensembles with level-based grouping demonstrate larger diversity (upper row) and their levels of coordination skills are controllable based on the levels of their policy primitives (lower row). Colors in the lower row indicate which group the partner's policy primitives belong to.

The agents will receive 20 points for each dish served. The objective is to serve as many dishes as possible in 1 minute.

Baselines The baseline methods include Self-play PPO (SP) [5, 41], population-based training (PBT) [5, 21], Fictitious Co-Play (FCP) [46], TrajeDi [26] and Maximum-Entropy Population-based training (MEP) [56].

Procedure First, we pair the agents with a human proxy model, a behavior-cloning agent that mimics human's behaviors, to test their coordination performance. The effect of each proposed component of PECAN is studied in the ablation study. Besides, we design other experiments to reinforce our claim that (a) policy ensemble is able to improve partners' diversity and (b) the PECAN agent learns a context-aware policy. Then, we recruit human players to evaluate the human-AI coordination ability of PECAN. The human players are required to give their subjective ratings to the agents. Finally, two case studies are conducted to demonstrate the adaptiveness

of PECAN in human-AI coordination. More experiment details are given in the supplementary material.

5.2 Experiments with Human Proxy Model

Overall Result Fig. 2(a) shows the overall coordination performance of PECAN, MEP, SP, PBT, FCP and TrajeDi agents when paired with a human proxy model. We run PECAN and MEP for 4 times with different random seeds and report their average performance. The results of SP, PBT, FCP and TrajeDi are taken from [56]. For a fair comparison, MEP and PECAN have the same population size. And we adopt the recommended hyperparameter settings for MEP in their paper [56]. From the results, we can see that PECAN outperforms baseline methods on all five layouts. Especially in Asymm. Adv., the best score by PECAN agents exceeds the baselines by a very large margin (+26.8%). But in layouts that require less coordination like Cramped Rm., PECAN has relatively marginal performance advantage than the baselines, which indicates that

(a) Evaluation results with human players

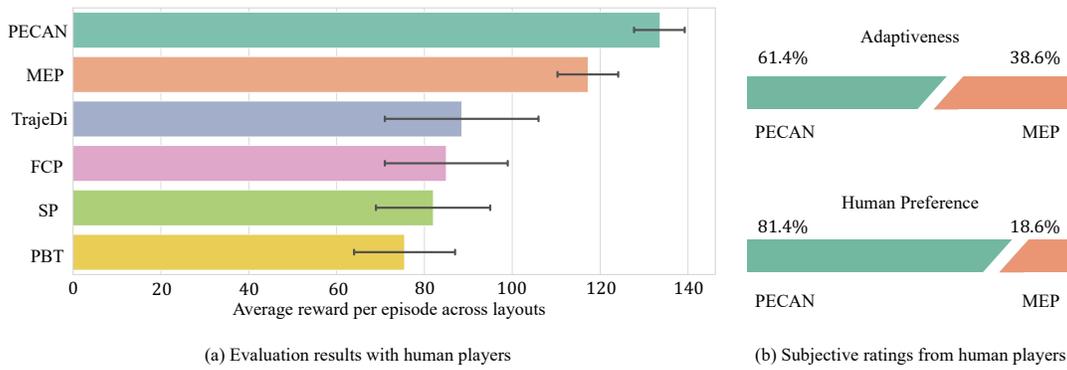(b) Subjective ratings from human players

Figure 6: (a) The coordination performance of each method with real humans. PECAN outperforms the baselines on human-AI coordination. (b) Human players' subjective ratings of the adaptiveness and their preference of the PECAN and the MEP agents. Human players give higher subjective ratings to PECAN on both adaptiveness and personal preferences.
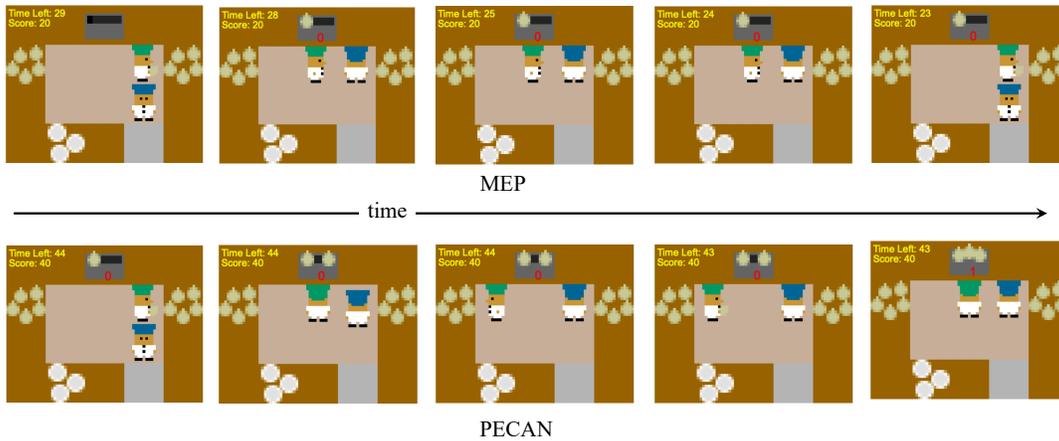

Figure 7: The MEP agent only learns to pick up onions from the right side. If we intentionally block its way to the onions on the right, it will be stuck and wait until we move away so that it can pick up onions there again. But for the PECAN agent, if we block its way to the onions on the right side, it will turn around to pick up onions on the left with no hesitation. Green agent is the ego agent and blue agent is the human player.

PECAN effectively improves the ego agent's ability to coordinate with its partner rather than to accomplish the task by itself.

Ablation Study To study the effect of each component, we ablate policy ensemble and context encoder of PECAN respectively. In PECAN^{-e}, policy ensemble is removed, and the partners are chosen the same as in MEP to train the ego agent. In PECAN^{-c}, we remove the context encoder and make the ego agent's policy no longer condition on context c.

Fig. 2(b) shows that without policy ensemble and context encoder, PECAN^{-e} and PECAN^{-c} have similar or worse performance than MEP (average performance drop -18.1 for PECAN^{-e} and -14.4 for PECAN^{-c}), while PECAN consistently outperforms the baseline. The result validates the effectiveness of the two proposed modules.

Policy Ensemble and Partner Diversity We will empirically validate our previous claim that policy ensemble is able to improve the diversity of partners. We randomly sample some states s from 5 trajectories and partners with/without policy ensemble. Then, we plot the distribution of partners' actions $\pi_p(\cdot|s)$ by t-SNE [49].

Fig. 4(a) gives the visualization results. Each point represents the action distribution $\pi_p(\cdot|s)$ of some partner p over state s. There are the same number of data points in the graphs with and without policy ensemble. But because there are limited number of partners without policy ensemble, the points excessively overlap with each other, indicating limited diversity. On the contrary, the action distributions become much more diverse with policy ensemble, which shows that policy ensemble is able to improve partner diversity effectively. The diverse partners allow the ego agent to learn more universal coordination behaviors, and therefore have stronger zero-shot human-AI coordination performance.

Study of the Context-aware Policy We claim that the ego agent's policy $\pi_e(\cdot|s, \hat{c})$ is context-aware, which means that the ego agent's policy conditions on context $\hat{c}$. The context encoder in PECAN will recognize the partner's context based on past behaviors and help the ego agent take actions accordingly. If incorrect context is fed to the ego agent, it may make improper decisions. There will be a performance gap between the true context and incorrect

Table 1: The MEP agent has very strong preference to collect dishes from Pot 1 and overlooks dishes in Pot 2, while the PECAN agent shows no such preference and serves much more dishes than the MEP agent.

Method	Usage		Dishes served by agent
	Pot 1	Pot 2	
PECAN	57.1%	42.9%	12.25
MEP	88.9%	11.1%	6.75

context, if the ego agent's policy is context-aware. Thus, we design an experiment to test whether the ego agent's policy is context-aware by manually creating context mismatch.

Specifically, we feed a random or manually-assigned context to the ego agent and pair them with a human proxy model to compare their performance with PECAN, where the context is the recognized context. Fig. 4(b) gives the results with context mismatch on *Asymmetric Advantages*. $c = RND$ represents replacing the real context with a random context, and $c = SP$ represents forcing the context to indicate the ego agent is collaborating with a partner from the self-play group G_4, which is a clear mismatch. The results demonstrates significant performance drop with context mismatch, which means the ego agent's policy is context-aware and the recognized context is effective. And it is worth noting that $c = SP$ has very poor performance. This confirms the conclusions from previous studies [5, 46] that agents with a self-play cooperative pattern have significantly different behaviors from an agent with strong human-AI coordination performance.

Effect of Level-based Grouping Similar to section 5.1.3, we randomly sample some states s and plot the action distribution $\pi_p(\cdot|s)$ by t-SNE. Fig. 5 gives the visualization results. It can been seen that the policy ensembles with level-based grouping are more diverse and their levels are controllable based on the level of their policy primitives, while the policy of partners without grouping always tends to form two clusters, which may be the result of compromise between high-level and low-level partners and limits the diversity of partners. Therefore, the result confirms that the level-based grouping is able to improve partner diversity and provide partners with more controllable levels. And the controllable level of partners allows the ego agent to learn level-based common BR more easily.

5.3 Human-AI Coordination

We follow the Human-AI coordination test protocol proposed in [5] and recruit 15 human players to participate in the study. We evaluate the average performance across layouts of PECAN and state-of-the-art method MEP and reuse the evaluation results of other baselines in [56]. The results are compatible because the test procedure is consistent. For the convenience of human-AI coordination experiments on Overcooked, we integrate models from [5] with PantheonRL [40], a newly released library for dynamic training. The code is available here[1].

Fig. 6 gives the result of our human study. PECAN outperform all other baselines, and the recruited human players give higher

adaptiveness rating to PECAN and noticeably prefer coordinating with PECAN than MEP.

Case Study To show the adaptiveness of PECAN, we give two case studies in our human-AI coordination experiments. Demo videos are available at https://sites.google.com/view/pecan-overcooked.

Case Study 1 See Fig. 7. In Cramped Room, we (blue chef) intentionally block the agents'(green chef) way to the onions to see how the agents will react. The MEP agent gets stuck and stand still until we move aside, while our PECAN agent makes adjustments immediately and turn around to pick up onions from the other side. It shows the PECAN agents are more adaptive and capable of adjusting its policy according to the human player's behaviors.

Case Study 2 We record the pot usage by the agent in Asymmetric Advantages (see the layout in Fig. 8). The trick for the agent (green chef) is to pick up dishes from both pots and serve, because it's much nearer to the serving counter than the human player (blue chef). Table 1 shows that the PECAN agent has no clear preference over pots than the MEP agent, which has very strong preference for Pot 1. The MEP agent adopts a non-adaptive strategy which has poor generalization to typical human behaviors of using both pots, which further leads to worse performance. However, the PECAN agent's policy is diverse and adaptive, which helps it better coordinate with real humans and serve much more dishes than the MEP agent.
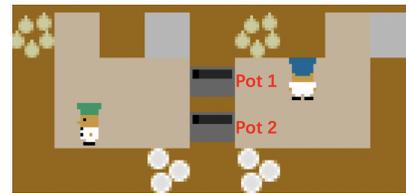
Figure 8: *Asymmetric Advantages.* **AI agents play the chef with green hat. Pot 1 is 1 step nearer to the serving counter.**

6 CONCLUSION AND FUTURE WORK

In this paper, we propose a new method (PECAN) for zero-shot human-AI coordination. Policy-ensemble partners and a context encoder are proposed to improve diversity of partners and help the ego agent learn more universal coordination behaviors. We evaluate PECAN with a human proxy model on Overcooked and shows that PECAN is able to outperform all baselines. Ablation studies, further studies and visualization experiments are conducted to demonstrate each component in PECAN. We also organize a human study to evaluate the proposed method's capability of human-AI coordination. The results indicate that PECAN outperforms all other baselines on performance as well as subjective ratings.

Our future work is to study how to analyze and identify the human player's behavior pattern as the ego agents' context (rather than the level-based context in the current method) with the population and no human data during training. In this way, the ego agent can take actions accordingly and better coordinate with humans. This is a very challenging research subject because it requires the ego agent to comprehend human behaviors given only the population of AI agents. And since PECAN is a two-stage method, another future direction is to study how to train it in an end-to-end manner.

---

[1]https://github.com/LxzGordon/pecan_human_AI_coordination

ACKNOWLEDGMENTS

REFERENCES

[1] OpenAI: Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Jozefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, et al. 2020. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research* 39, 1 (2020), 3–20.

[2] Craig Boutilier. 1996. Planning, learning and coordination in multiagent decision processes. In *TARK*, Vol. 96. 195–210.

[3] Noam Brown and Tuomas Sandholm. 2018. Superhuman AI for heads-up no-limit poker: Libratus beats top professionals. *Science* 359, 6374 (2018), 418–424.

[4] Noam Brown and Tuomas Sandholm. 2019. Superhuman AI for multiplayer poker. *Science* 365, 6456 (2019), 885–890.

[5] Micah Carroll, Rohin Shah, Mark K Ho, Tom Griffiths, Sanjit Seshia, Pieter Abbeel, and Anca Dragan. 2019. On the utility of learning about humans for human-ai coordination. *Advances in neural information processing systems* 32 (2019).

[6] Li Chenghao, Tonghan Wang, Chengjie Wu, Qianchuan Zhao, Jun Yang, and Chongjie Zhang. 2021. Celebrating diversity in shared multi-agent reinforcement learning. *Advances in Neural Information Processing Systems* 34 (2021), 3991–4002.

[7] Brandon Cui, Hengyuan Hu, Luis Pineda, and Jakob Foerster. 2021. K-level Reasoning for Zero-Shot Coordination in Hanabi. *Advances in Neural Information Processing Systems* 34 (2021), 8215–8228.

[8] Christian Schroeder de Witt, Tarun Gupta, Denys Makoviichuk, Viktor Makoviychuk, Philip HS Torr, Mingfei Sun, and Shimon Whiteson. 2020. Is independent learning all you need in the starcraft multi-agent challenge? *arXiv preprint arXiv:2011.09533* (2020).

[9] Kenneth Derek and Phillip Isola. 2021. Adaptable Agent Populations via a Generative Model of Policies. *Advances in Neural Information Processing Systems* 34 (2021), 3902–3913.

[10] Yali Du, Lei Han, Meng Fang, Tianhong Dai, Ji Liu, and Dacheng Tao. 2019. LIIR: learning individual intrinsic reward in multi-agent reinforcement learning. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems (NeurIPS)*. 4403–4414.

[11] Yali Du, Bo Liu, Vincent Moens, Ziqi Liu, Zhicheng Ren, Jun Wang, Xu Chen, and Haifeng Zhang. 2021. Learning Correlated Communication Topology in Multi-Agent Reinforcement learning. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*. 456–464.

[12] Yali Du, Chengdong Ma, Yuchen Liu, Runji Lin, Hao Dong, Jun Wang, and Yaodong Yang. 2022. Scalable Model-based Policy Optimization for Decentralized Networked Systems. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 1654–1656.

[13] Meng Fang, Tianyi Zhou, Yali Du, Lei Han, and Zhengyou Zhang. 2019. Curriculum-guided hindsight experience replay. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems (NeurIPS)*. 12623–12634.

[14] Ghost Town Games. 2016. Overcooked. *https://store.steampowered.com/app/448510/Overcooked/*. (2016).

[15] Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. 2018. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905* (2018).

[16] Lei Han, Peng Sun, Yali Du, Jiechao Xiong, Qing Wang, Xinghai Sun, Han Liu, and Tong Zhang. 2019. Grid-wise control for multi-agent reinforcement learning in video game ai. In *International Conference on Machine Learning (ICML)*. PMLR, 2576–2585.

[17] He He, Jordan Boyd-Graber, Kevin Kwok, and Hal Daumé III. 2016. Opponent modeling in deep reinforcement learning. In *International conference on machine learning*. PMLR, 1804–1813.

[18] Zhang-Wei Hong, Tzu-Yun Shann, Shih-Yang Su, Yi-Hsiang Chang, Tsu-Jui Fu, and Chun-Yi Lee. 2018. Diversity-driven exploration strategy for deep reinforcement learning. *Advances in neural information processing systems* 31 (2018).

[19] Hengyuan Hu, Adam Lerer, Alex Peysakhovich, and Jakob Foerster. 2020. "Other-Play" for Zero-Shot Coordination. In *International Conference on Machine Learning*. PMLR, 4399–4410.

[20] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. 1991. Adaptive mixtures of local experts. *Neural computation* 3, 1 (1991), 79–87.

[21] Max Jaderberg, Valentin Dalibard, Simon Osindero, Wojciech M Czarnecki, Jeff Donahue, Ali Razavi, Oriol Vinyals, Tim Green, Iain Dunning, Karen Simonyan, et al. 2017. Population based training of neural networks. *arXiv preprint arXiv:1711.09846* (2017).

[22] Zahi M Kakish, F RodrÃguez-Lera, D Bischel, A Mosquera, R Boumghar, S Kaczmarek, T Seabrook, P Metzger, and JL Galanche. 2019. Open-source AI assistant for cooperative multi-agent systems for lunar prospecting missions. In *8th European Conference for Aeronautics and Space Sciences (EUCASS)*.

[23] Ilias Kazantzidis, Timothy J Norman, Yali Du, and Christopher T Freeman. 2022. How to Train Your Agent: Active Learning from Human Preferences and Justifications in Safety-critical Environments. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*. 1654–1656.

[24] Runze Liu, Fengshuo Bai, Yali Du, and Yaodong Yang. 2022. Meta-Reward-Net: Implicitly Differentiable Reward Learning for Preference-based Reinforcement Learning.. In *Advances in Neural Information Processing Systems (NeurIPS)*.

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

[26] Andrei Lupu, Brandon Cui, Hengyuan Hu, and Jakob Foerster. 2021. Trajectory diversity for zero-shot coordination. In *International Conference on Machine Learning*. PMLR, 7204–7213.

[27] Leandro Soriano Marcolino, Albert Xin Jiang, and Milind Tambe. 2013. Multi-agent team formation: Diversity beats strength?. In *Twenty-Third International Joint Conference on Artificial Intelligence*.

[28] Stefano Mariani, Giacomo Cabri, and Franco Zambonelli. 2021. Coordination of autonomous vehicles: taxonomy and survey. *ACM Computing Surveys (CSUR)* 54, 1 (2021), 1–33.

[29] Muhammad A Masood and Finale Doshi-Velez. 2019. Diversity-inducing policy gradient: Using maximum mean discrepancy to find a set of diverse policies. *arXiv preprint arXiv:1906.00088* (2019).

[30] John Nash. 1953. Two-person cooperative games. *Econometrica: Journal of the Econometric Society* (1953), 128–140.

[31] Stefanos Nikolaidis and Julie Shah. 2013. Human-robot cross-training: computational formulation, modeling and evaluation of a human team training strategy. In *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 33–40.

[32] OpenAI. 2019. OpenAI Five finals. *https://openai.com/blog/openai-five-finals/* (2019).

[33] Ian Osband, Daniel Russo, and Benjamin Van Roy. 2013. (More) efficient reinforcement learning via posterior sampling. *Advances in Neural Information Processing Systems* 26 (2013).

[34] Kate Rakelly, Aurick Zhou, Chelsea Finn, Sergey Levine, and Deirdre Quillen. 2019. Efficient off-policy meta-reinforcement learning via probabilistic context variables. In *International conference on machine learning*. PMLR, 5331–5340.

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

[36] Jie Ren, Yewen Li, Zihan Ding, Wei Pan, and Hao Dong. 2021. Probabilistic mixture-of-experts for efficient deep reinforcement learning. *arXiv preprint arXiv:2104.09122* (2021).

[37] Cinjon Resnick, Ilya Kulikov, Kyunghyun Cho, and Jason Weston. 2018. Vehicle communication strategies for simulated highway driving. *arXiv preprint arXiv:1804.07178* (2018).

[38] João G Ribeiro, Cassandro Martinho, Alberto Sardinha, and Francisco S Melo. 2022. Assisting Unknown Teammates in Unknown Tasks: Ad Hoc Teamwork under Partial Observability. *arXiv preprint arXiv:2201.03538* (2022).

[39] Dorsa Sadigh, Shankar Sastry, Sanjit A Seshia, and Anca D Dragan. 2016. Planning for autonomous cars that leverage effects on human actions.. In *Robotics: Science and systems*, Vol. 2. Ann Arbor, MI, USA, 1–9.

[40] Bidipta Sarkar, Aditi Talati, Andy Shih, and Dorsa Sadigh. 2022. PantheonRL: A MARL Library for Dynamic Training Interactions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 13221–13223.

[41] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[42] Zijing Shi, Meng Fang, Yunqiu Xu, Ling Chen, and Yali Du. 2023. Stay Moral and Explore: Learn to Behave Morally in Text-based Games. In *International Conference on Learning Representations*. https://openreview.net/forum?id=CtS2Rs_aYk

[43] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. 2017. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815* (2017).

[44] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. 2018. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science* 362, 6419 (2018), 1140–1144.

[45] Malcolm Strens. 2000. A Bayesian framework for reinforcement learning. In *ICML*, Vol. 2000. 943–950.

[46] DJ Strouse, Kevin McKee, Matt Botvinick, Edward Hughes, and Richard Everett. 2021. Collaborating with humans without human data. *Advances in Neural Information Processing Systems* 34 (2021), 14502–14515.

[47] Richard S Sutton, Doina Precup, and Satinder Singh. 1999. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence* 112, 1-2 (1999), 181–211.

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

[49] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).

[50] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[51] Oriol Vinyals, Igor Babuschkin, Junyoung Chung, Michael Mathieu, Max Jaderberg, Wojciech M Czarnecki, Andrew Dudzik, Aja Huang, Petko Georgiev, Richard Powell, et al. 2019. Alphastar: Mastering the real-time strategy game starcraft ii.

[52] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* 575, 7782 (2019), 350–354.

[53] Zhengyu Yang, Kan Ren, Xufang Luo, Minghuan Liu, Weiqing Liu, Jiang Bian, Weinan Zhang, and Dongsheng Li. 2022. Towards Applicable Reinforcement Learning: Improving the Generalization and Sample Efficiency with Policy Ensemble. *arXiv preprint arXiv:2205.09284* (2022).

[54] Chao Yu, Akash Velu, Eugene Vinitsky, Yu Wang, Alexandre Bayen, and Yi Wu. 2021. The surprising effectiveness of ppo in cooperative, multi-agent games. *arXiv preprint arXiv:2103.01955* (2021).

[55] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. 2017. Deep sets. *Advances in neural information processing systems* 30 (2017).

[56] Rui Zhao, Jinming Song, Hu Haifeng, Yang Gao, Yi Wu, Zhongqian Sun, and Yang Wei. 2021. Maximum Entropy Population Based Training for Zero-Shot Human-AI Coordination. *arXiv preprint arXiv:2112.11701* (2021).

DeepMind blog 2 (2019).