PC3D: Zero-Shot Cooperation Across Variable Rosters via Personalized Context Distillation

Anonymous Authors[1]

Abstract

Cooperative multi-agent reinforcement learning often assumes a fixed execution team, yet many decentralized systems must operate with varying numbers of active agents during deployment. We study this setting under episodic roster variation: each episode is executed by a set of homogeneous agents, with the team size varying across episodes. Agents act only from local histories, without execution-time communication, privileged coordinators, or online retraining. Therefore, effective cooperation requires each agent to recover relevant context about the active team and adapt its behavior accordingly. To this end, we propose **PC3D** (Personalized Central Coordination Context Distillation), a method for training decentralized policies to recover and use personalized coordination context from local interaction histories. During training, a set-structured centralized teacher compresses the active team into coordination tokens and personalizes them into agent-specific contexts, which are distilled into decentralized policies. At execution, each agent predicts its own context from local history and adaptively uses it to condition decision-making. Across three cooperative MARL benchmarks, PC3D achieves higher returns than the evaluated baselines with both seen and unseen roster sizes, and ablations attribute these gains to both context distillation and adaptive context use.

1. Introduction

Multi-agent reinforcement learning (MARL) studies how multiple decision-makers learn to act in shared environments, which makes it a natural framework for cooperative control problems that involve multiple learners with coordinated behavior (Zhang et al., 2021b; Oroojlooy & Hajinezhad, 2023; Gronauer & Diepold, 2022). As the application domains expand, cooperative multi-agent systems are increasingly expected to operate in settings where agents cannot rely on execution-time coordination mechanisms (Zhang et al., 2021a). Centralized training with decentralized execution (CTDE) has become the dominant framework for addressing this tension in cooperative MARL (CMARL) (Amato, 2024). Classical value-factorization methods such as VDN and QMIX improve decentralized control by constraining how a centralized training objective decomposes into per-agent utilities (Sunehag et al., 2018; Rashid et al., 2020). Centralized-critic methods such as MADDPG, COMA, and MAPPO instead use training-time information to stabilize policy learning while preserving decentralized execution (Lowe et al., 2017; Foerster et al., 2018; Yu et al., 2022).

Although CTDE methods have substantially advanced the field, they typically assume a fixed execution team. This leaves a structural gap for *open-team cooperation* (OTC), where the team size may vary during deployment. We refer to the set of agents active in a given episode as a *roster*, and study *episodic roster variation*: each episode is executed by a fixed roster, but the roster size may change across episodes. Our setting involves homogeneous agents and assumes fully decentralized execution under partial observability, without execution-time communication, global observations, or online retraining. OTC naturally arises in many decentralized control problems, including robot teams, which can be restructured to meet operational requirements (Rosenfeld et al., 2006; Portugal & Rocha, 2017); warehouse systems, where the infrastructure can be rescaled depending on corporate objectives (Rjeb et al., 2021); and autonomous vehicle routing, where the fleet size may evolve with changing demand and adoption rates (Boesch et al., 2016; Qu et al., 2022; Akman et al., 2025).

Several lines of work address variation in team composition in CMARL (Yuan et al., 2023) with different constraints on the execution model or the task structure. Agent–entity graph methods (Agarwal et al., 2020) learn policies over agents and entities by relying on graph message passing. SOG (Shao et al., 2022) organizes agents into temporary conductor-follower groups, exchanging summarized mes-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

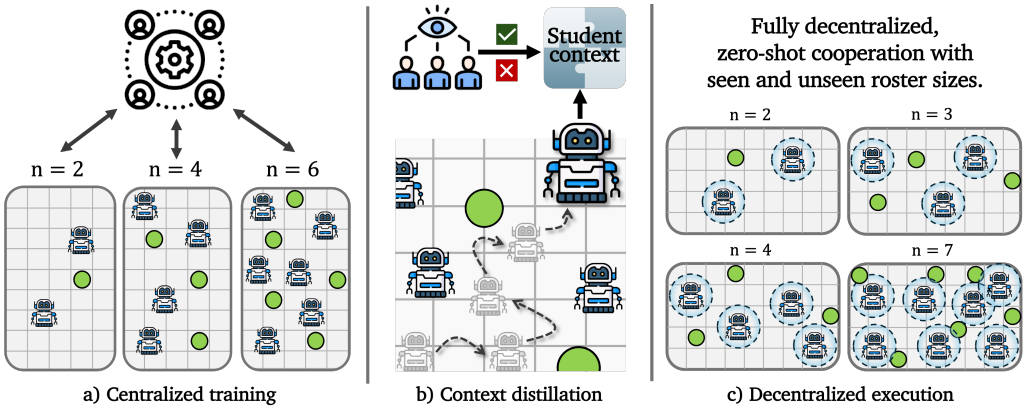


Figure 1. **PC3D at a glance.** PC3D trains with centralized information over a distribution of roster sizes for a given cooperative task (**a**). The centralized teacher provides personalized coordination contexts, which decentralized agents learn to recover from local interaction histories (**b**). At execution, agents act only from local histories, without communication or retraining, and coordinate across both seen and held-out roster sizes (**c**).

sages during execution. COPA (Liu et al., 2021) uses a privileged coach with an "omniscient" view to distribute strategies during both training and execution. MIPI (Wang et al., 2023), building on REFIL (Iqbal et al., 2021), regularizes reliance on team-related information by assuming that the designer could decompose agent states into team-related (s^-) and -unrelated (s^+) components. In contrast, we explore whether a method can address the OTC problem natively within the CTDE setting without changes to the execution model, with centralized information available only during training and execution relying solely on each agent's local history. By keeping the execution contract fixed and treating the method as the variable of interest, we ask whether methodological changes alone can improve zero-shot cooperation across roster sizes.

In this setting, the core challenge is not merely learning effective coordination for a given task, but maintaining it across different, possibly unseen roster sizes at execution time. This aspect resembles Ad Hoc Teamwork (AHT), which is an adjacent problem concerned with adapting learners to unfamiliar teammates (Stone et al., 2010; Rahman et al., 2023; Wang et al., 2024). Although the two settings share structural challenges, AHT primarily focuses on mitigating coordination failures caused by unfamiliar teammate policies, whereas OTC requires leveraging additional teammates when new cooperative opportunities arise. This makes CTDE methods a viable option to train such policies, although their standard form does not account for changing cooperation regimes across roster sizes. To that end, this study explores *whether CTDE methods can improve cooperation across varying team sizes and zero-shot generalization to unseen ones by leveraging a centralized team representation, which can provide personalized and locally recoverable coordination signals while preserving fully decentralized execution.*

Existing methods provide ingredients for this goal. *Teacher-student* methods such as CTDS and PTDE show that centralized guidance can be distilled into decentralized agents (Zhao et al., 2024) and that this guidance should be *agent-personalized* (Chen et al., 2024); however, they leave open how this signal should be formed, personalized, and used under the OTC setting. On scalability and architectural compatibility with changing roster sizes, attention-based and permutation-invariant critics have shown that centralized representations can handle unordered agent collections (Iqbal & Sha, 2019; Liu et al., 2020), with Deep Sets and Set Transformers providing the underlying design principles (Zaheer et al., 2017; Lee et al., 2019). However, these pieces do not, by themselves, solve OTC: a set-compatible critic improves the centralized training signal but does not automatically provide the decentralized policy with a reusable notion of coordination across roster sizes.

Building on these ideas, this paper introduces **PC3D** (**P**ersonalized **C**entral **C**oordination **C**ontext **D**istillation): a method for improving CTDE learners under episodic roster variation by (i) extracting a compact team-level coordination summary using a set-structured central module, (ii) personalizing that summary into locally recoverable contexts, and (iii) distilling it into decentralized policies that learn how and when to rely on it. We instantiate it on top of a MAPPO backbone, although the idea can be extended to other CTDE learners. During training, a centralized set critic embeds the active team as an unordered set, compresses it into a small number of *coordination tokens* via token-based cross-attention, and produces personalized *per-agent teacher contexts*. These teacher contexts are used for training agents to infer team context estimates from local interactions, while the coordination tokens support centralized value estimation. At execution, each agent still acts only on its local observation history while also predicting a

student coordination context to adaptively condition policy features. As illustrated in Figure 1, training PC3D over a distribution of roster sizes within the same cooperative task enables decentralized policies to recover relevant team context from local histories and coordinate under both seen and held-out roster sizes.

This research has been structured around our central hypothesis: **For a given task structure, compact team coordination representations can be personalized into locally recoverable agent contexts and distilled into decentralized executors, enabling enhanced team-context awareness at the agent level for stronger cooperation across seen rosters and better zero-shot generalization to unseen ones**. To rigorously study it, we first formalize the problem (Section 2), propose a method that reflects this methodological intent (Section 3), conduct evaluations tailored to confirm our hypothesis (Section 4.2), and perform ablations to strengthen our conclusions (Section 4.3). Our evaluations across three cooperative MARL benchmarks show that PC3D achieves the highest returns on both seen and held-out rosters, consistently improving its MAPPO backbone by a clear margin and outperforming the IPPO and PIC-MAPPO baselines. Moreover, ablations attribute these gains to the full distillation-adaptive conditioning mechanism, not merely to adding a stronger centralized critic.

Contributions.

- We provide a new formalization for the variable-roster cooperation problem (where each episode is executed by varying teams of homogeneous agents) using a family of cooperative tasks induced from a common template.
- We propose *personalized central coordination context distillation* as a solution for open-team cooperation and instantiate it on top of a MAPPO backbone.
- We evaluate our method across three cooperative MARL benchmarks with varying roster sizes. We highlight the added value of our method by comparing it to three MARL baselines.
- We perform ablations to test the marginal gains of distillation and adaptive policy conditioning. Moreover, we offer additional insights on whether the distilled context is recoverable from local history and is meaningfully used for decision-making.

2. Open-Team Cooperation

We study fully cooperative, partially observable tasks in which a set of agents must act autonomously to optimize a shared objective. Such tasks are generally formalized as a *Decentralized Partially Observable Markov Decision Process* (Dec-POMDP) (Bernstein et al., 2002). Although this formulation is useful for describing a task instance, it is insufficient to capture the higher-level OTC objective of learning generalizable cooperative policies in the presence of episodic roster variability.

We focus on tasks with homogeneous agents (sharing the same action and observation spaces) and refer to a set of agents admitted in an episode as a **roster**. We represent a cooperative task with different rosters using a family of Dec-POMDPs induced from a common **environment template**. An environment template E describes the shared structural properties of a space of Dec-POMDPs and can be formalized as a tuple:

$$E = (\mathcal{S}, A, O, \mathcal{U}, \mathcal{R}, \Gamma, \gamma),$$

where $\mathcal{S}$ is the shared state-description schema of the task, A and O are the shared per-agent action and observation spaces, $\mathcal{U}$ is the shared cooperative objective, $\mathcal{R}$ is the set of *admissible rosters*, Γ is the *roster-indexing mechanism* that instantiates roster-specific Dec-POMDPs with the task semantics defined by the template, and γ is the discount factor. For each roster $r \in \mathcal{R}$, the template induces a roster-indexed Dec-POMDP

$$\mathcal{M}_r = \Gamma(r) = (r, S_r, A, O, P_r, \Omega_r, R_r, \rho_r, \gamma),$$

where r is the roster (active agent set), S_r is the state space induced by the template for that roster, P_r is the transition kernel, Ω_r is the joint observation kernel, R_r is the shared reward function, and ρ_r is the initial-state distribution. Thus, Γ-induced Dec-POMDPs share the task semantics and cooperative objective of E, while allowing roster-dependent dynamics and observation/reward structure.

Optimality. To define optimality for a given environment template E, we use the notion of **policy generators**. A policy generator G is a mapping of the given environment template and roster pair to a joint decentralized policy ($G(E, r) = \pi_r \in \Pi_r$). This object describes the family of decentralized policies induced across rosters and not an execution-time coordinator. A policy generator G^* is optimal for the given environment template E if

$$G^*(E, r) \in \arg\max_{\pi_r \in \Pi_r} J_r(\pi_r), \qquad \forall r \in \mathcal{R},$$

where $J_r(\pi_r)$ is the expected discounted return of a decentralized joint policy π_r for the roster r. This defines the ideal roster-wise objective. Since training separate policies across $\mathcal{R}$ is often impractical as it scales with $|\mathcal{R}|$, we study whether a shared policy mechanism trained on $\mathcal{R}_s \subset \mathcal{R}$ can approximate this objective and generalize zero-shot to held-out rosters in $\mathcal{R} \setminus \mathcal{R}_s$.

3. PC3D: Personalized Central Coordination Context Distillation

We hypothesize that achieving strong generalization across different-roster task instances in a partially observable, fully

decentralized setting requires enhanced context awareness and adaptive decision-making. We therefore propose **PC3D**, a CTDE extension for open-team settings that preserves the practical conveniences that make CTDE attractive: centralized information can shape learning during training, while execution remains fully decentralized with the same local observation interface.

PC3D builds on the teacher-student CTDE idea of distilling centralized signals into decentralized executors (Zhao et al., 2024; Chen et al., 2024), but targets a distinct structural limitation. For instance, PTDE (Chen et al., 2024) distills global information into decentralized agents to improve local decision-making. While this is useful for fixed-roster cooperation, extending this idea to the OTC setting requires additional requirements, which we tailor PC3D to explicitly address: the centralized representation should be responsive to roster variability, transferable across roster-induced cooperation regimes, and personalized in a way that remains tied to agent-observable features to support recoverability. Therefore, the method employs components to (i) produce a global representation that compactly summarizes the coordination context for the active team, (ii) from which to produce per-agent teacher contexts that include useful and recoverable coordination cues for decision-making, (iii) use context distillation to recover these contexts from local information at execution time, and (iv) adaptively condition agent policies on the estimated context. This study introduces PC3D atop a MAPPO backbone (illustrated in Figure 2) with parameter-shared (to reuse across varying numbers of agents) and recurrent (using GRUs (Cho et al., 2014) to mitigate partial observability (Hausknecht & Stone, 2015; Rashid et al., 2020)) actor networks.

3.1. Centralized coordination context and personalization

We replace the fixed-width centralized critic with a permutation-invariant set critic for architectural compatibility with varying team sizes. At each training step, the set critic receives the observations of the active agents and encodes them individually with a shared encoder:

$$e_i^t = \phi_\psi(o_i^t), \qquad i \in r,$$

where r is the active roster and o_i^t is the observation of agent $i \in r$. Then, within the teacher module, a small number K of learned query vectors (q_k) attend to these observation encodings through a single-head cross-attention layer with identity projections to produce K coordination tokens (z_k^t):

$$\alpha_{kj}^t = \mathrm{softmax}_{j\in r}\left(\frac{q_k^\top e_j^t}{\sqrt{d}}\right),$$
$$z_k^t = \sum_{j\in r} \alpha_{kj}^t e_j^t, \qquad k = 1, \dots, K.$$

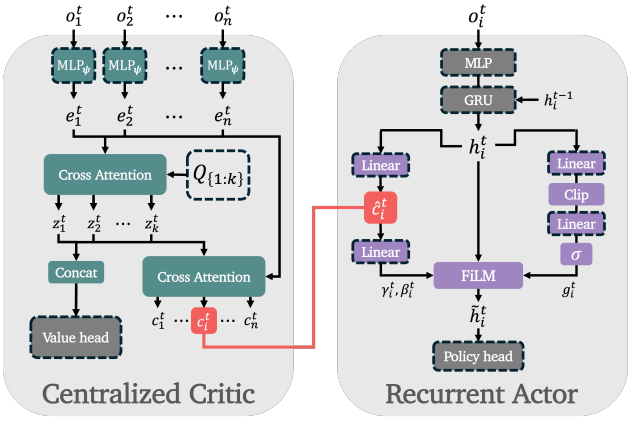


Figure 2. **PC3D-MAPPO architecture.** PC3D extends MAPPO with a critic/teacher module (left) and a context-conditioned actor (right). The critic encodes agent observations with a shared encoder, read by learned query tokens $Q_{1:k}$ through cross-attention to produce coordination tokens $z_{1:k}^t$, used for team value prediction. This representation is personalized in a secondary cross-attention into per-agent teacher contexts c_i^t. The actor uses recurrent features h_i^t to predict a student context $\hat{c}_i^t$ to FiLM-modulate policy features, controlled by the context-reliance gate g_i^t. The dashed boxes indicate trainable components and the **coral** connection denotes the distillation path.

By using cross-attention with a fixed number of trainable query vectors, we enforce an information bottleneck that yields a compact coordination summary. This is intended to make the representation more transferable across rosters by biasing the critic toward team-level factors most useful for value estimation rather than overly granular roster-specific details. The coordination tokens Z^t are concatenated into a fixed-width team representation and passed to the value head to predict the centralized team value. In parallel, the per-agent observation encodings attend back to the coordination tokens in a secondary cross-attention layer to produce per-agent teacher contexts (c_i^t), which are personalized coordination contexts used in context distillation:

$$\eta_{ik}^t = \mathrm{softmax}_{k=1,\dots,K}\left(\frac{(e_i^t)^\top z_k^t}{\sqrt{d}}\right),$$
$$c_i^t = \sum_{k=1}^{K} \eta_{ik}^t z_k^t, \qquad i \in r. \tag{1}$$

This construction is permutation-invariant for the value branch and permutation-equivariant for the per-agent teacher contexts, which allows for assigning one personalized context to each active agent independently of agent ordering. Implementing the teacher module within the centralized critic enables the value loss to shape the teacher's parameters to identify useful team features for value estimation. The learned query vectors (Q) first extract team-level factors from the set of agent observation embeddings (E^t), shaped by the value objective, so that the coordination tokens (Z^t) tend to encode compact patterns that matter at

the collective level. Then, the secondary attention provides each agent with a personalized context (c_i^t) by retrieving the subset of these latent factors most aligned with the agent's embedding (e_i^t). Using dot-product attention with identity projections keeps this readout *similarity-based*, which is a deliberate inductive bias intended to reduce the risk that the context is overly shaped by the value loss through unnecessary learnable flexibility and to make it more likely to remain structured, recoverable, and tied to agent-observable features.

3.2. Decentralized context recovery and adaptive conditioning

PC3D employs shared-parameter actor networks for reuse by a variable number of agents. To enable agents to recover and leverage teacher context under partial observability, we equip the actor networks with context-estimation and feature-modulation paths. First, recurrent actor features (h_i^t) undergo two linear transformations to produce the agent's context ($\hat{c}_i^t$) and context reliance control signal (ρ_i^t) estimates:

$$\begin{aligned} \hat{c}_i^t &= W_c h_i^t + b_c, \\ \rho_i^t &= \mathrm{clip}\big(w_u^\top h_i^t + b_u,\ \rho_{\min},\ \rho_{\max}\big)\,. \end{aligned} \tag{2}$$

We clip ρ_i^t to stabilize early training and reduce premature gate (g_i^t below) saturation. ρ_i^t is then converted into a gating scalar to control the modulation of the recurrent features in a *Feature-wise Linear Modulation* (FiLM) (Perez et al., 2018) with the agent's context estimation:

$$\begin{aligned} \big[\gamma_i^t; \beta_i^t\big] &= W_f \hat{c}_i^t + b_f, \\ g_i^t &= \sigma(a_g \rho_i^t + b_g), \\ \tilde{h}_i^t &= h_i^t \odot (1 + g_i^t \gamma_i^t) + g_i^t \beta_i^t, \end{aligned} \tag{3}$$

where γ_i^t and β_i^t are the scaling and shifting terms for the FiLM modulation; a_g and b_g are scale and offset control parameters for the context reliance gating. The resulting transformed hidden features ($\tilde{h}_i^t$) are then fed into the policy head.

We use feature modulation (instead of a concatenation such as $[h_i^t; \hat{c}_i^t]$) so that context estimation can adaptively influence the policy features without competing with them as a separate input stream. Moreover, the context estimate $\hat{c}_i^t$ is distilled from the teacher context c_i^t (Eq. 4) shaped for team-value estimation, so the way this information should affect action selection is not fixed a priori. The actor learns, through the policy objective, how (by γ_i^t and β_i^t) and to what extent (by g_i^t) the recovered context should shape policy features.

3.3. Training objective

PC3D-MAPPO retains the standard MAPPO optimization components, including PPO-style clipped policy updates, centralized value regression, entropy regularization, and GAE-based advantage estimation (Schulman et al., 2017; 2018; Yu et al., 2022; Ahmed et al., 2019), and extends the learning objective with a single distillation term.

Let $\bar{c}_i^t$ denote the *detached* personalized teacher context used as the distillation target for agent i at time t (from Eq. 1), and let $\hat{c}_i^t$ denote the student context predicted from local history (from Eq. 2). We train the student context with a smooth L_1 (Huber) distillation loss:

$$\mathcal{L}_{\text{distill}} = \frac{1}{|\mathcal{D}|} \sum_{(t,i)\in\mathcal{D}} \ell_{\text{Huber}}\big(\hat{c}_i^t,\ \bar{c}_i^t\big)\,, \tag{4}$$

where $\mathcal{D}$ is the set of valid agent-time decision pairs in the minibatch. Huber distillation loss allows the distillation to recover from large early-stage teacher-student misalignment without weakening regression in well-aligned contexts. In practice, we use the exponential moving average of the teacher to stabilize the distillation target during policy updates.

Then the full PC3D-MAPPO objective becomes

$$\mathcal{L} = \mathcal{L}_{\text{PPO}} + \lambda_V \mathcal{L}_V - \lambda_H \mathcal{L}_H + \lambda_{\text{distill}} \mathcal{L}_{\text{distill}}, \tag{5}$$

where $\mathcal{L}_{\text{PPO}}$, $\mathcal{L}_V$, and $\mathcal{L}_H$ are the standard MAPPO actor, value, and entropy terms, respectively.

The objective is optimized over a training distribution over a subset of admissible rosters for the same cooperative task. This exposes the learner to multiple roster sizes during training, while preserving the evaluation goal of decentralized execution on both seen and held-out rosters.

Importantly, the context-reliance estimation receives no direct supervision. It is optimized only with respect to the policy objective so that the model learns, via return maximization, how strongly and under what conditions the context estimates should influence action selection.

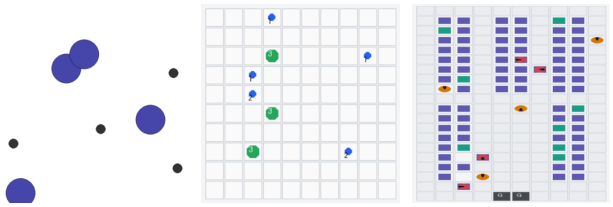
4. Results

4.1. Experimental setup

Benchmarks. We evaluate PC3D on three fully cooperative MARL environments (Figure 3). In each benchmark, the roster size varies across episodes, execution is decentralized, agents are homogeneous, and each agent acts based on its local observation history. We modify environments to use fixed-width local observations where possible, limiting exposure to trivial roster cues arising from changing observation dimensionality across roster sizes. The roster sizes we use in our training and evaluations are split into explicit training (seen during training), validation (unseen but

Table 1. **Evaluation performance across roster splits.** Returns (means ± standard deviations) across five seeded final checkpoints. For each seed, the mean is the average per-count evaluation returns within the corresponding train, validation, or test roster sizes. Higher is better for all tasks. LBF values are multiplied by 10^2 for readability. **Bold** indicates the best method in each column.

Method	Spread			LBF ($\times 1e2$)			RWARE		
	Train	Validation	Test	Train	Validation	Test	Train	Validation	Test
IPPO	-57.06 ± 1.6	-65.01 ± 2.0	-103.43 ± 2.6	6.84 ± 1.4	3.98 ± 0.7	8.24 ± 0.6	2.58 ± 1.5	2.58 ± 1.5	6.17 ± 3.0
MAPPO	-42.45 ± 0.8	-51.96 ± 1.3	-86.13 ± 1.7	5.61 ± 0.6	3.44 ± 0.5	7.91 ± 0.8	1.07 ± 0.6	0.99 ± 0.6	2.77 ± 1.6
PIC-MAPPO	-42.00 ± 0.5	-50.34 ± 0.9	-84.42 ± 1.3	6.84 ± 1.4	3.76 ± 0.7	8.40 ± 1.1	2.30 ± 1.4	2.22 ± 1.4	5.67 ± 2.8
PC3D-MAPPO	$\mathbf{-39.90 \pm 0.7}$	$\mathbf{-48.09 \pm 1.0}$	$\mathbf{-79.18 \pm 1.5}$	$\mathbf{7.91 \pm 0.7}$	$\mathbf{4.47 \pm 0.3}$	$\mathbf{8.98 \pm 0.1}$	$\mathbf{3.58 \pm 1.5}$	$\mathbf{3.53 \pm 1.5}$	$\mathbf{7.73 \pm 2.7}$



(a) Spread *(b)* LBF *(c)* RWARE

Figure 3. **Evaluation benchmarks.** We evaluate PC3D on Spread, LBF, and RWARE, adapting each benchmark to episodic roster variation under fixed local observation interfaces.

used for selection during hyperparameter search, reserved for intermediate values), and held-out test counts (unseen and used only for reporting, reserved for larger counts to demonstrate extrapolation).

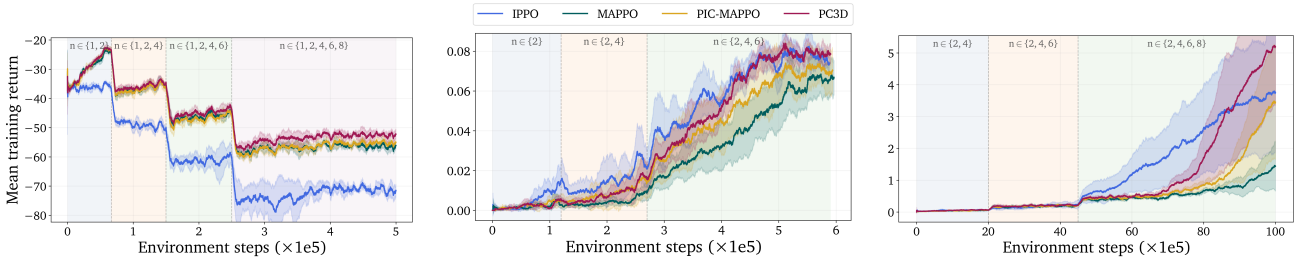
Simple Spread (Mordatch & Abbeel, 2018) is a standard MPE particle-world coverage task, with a two-dimensional arena in which agents must spread out to cover the landmarks while avoiding collisions (Figure 3a). Our version was built from PettingZoo's (Terry et al., 2021) `simple_spread_v3` environment with discrete actions and n landmarks for each n agent roster. We modify the environment interface with shared team rewards (negative-sum of distances between each landmark and the closest respective agent minus collision penalties), disabled communication channels, and fixed-width local observations (retaining only the agent's own velocity and position, the three nearest landmarks and teammates). We use training roster sizes $\{1, 2, 4, 6, 8\}$, validation roster sizes $\{3, 5, 7\}$, and held-out test roster sizes $\{9, 10\}$.

Level-based foraging (LBF) (Christianos et al., 2020; Papoudakis et al., 2021) is a grid-world mixed cooperative-competitive game (Figure 3b) where agents and food items have levels and a food item can be collected only when adjacent agents execute the loading action with a sufficient combined level. We use the cooperative variant `Foraging-2s-10x10-{n}p-{f}f-coop-v3`, with sight range 2. We report *normalized team returns*, computed as the native team reward divided by the active roster size. We scale the number of food items (`f`) with the active roster size (`n`). We replace the native observation with a fixed-width local entity encoding that does not grow with team size. We use training roster sizes $\{2, 4, 6\}$, validation roster sizes $\{3, 5\}$, and held-out test roster sizes $\{7, 8\}$.

Multi-Robot Warehouse (RWARE) (Papoudakis et al., 2021) is a robotic warehouse control benchmark in which robots move through aisles, pick up requested shelves, and deliver them to goal cells (`G`) (Figure 3c). We use the `rware-small-{n}ag-v2` layout (20×10). The reward type is set to `global`, so every robot receives the same reward when the team successfully delivers the requested shelves. RWARE is sparse and congestion-sensitive: larger teams can increase throughput, but they also congest the passages and interfere with shelf retrieval. We use training roster sizes $\{2, 4, 6, 8\}$, validation roster sizes $\{3, 5, 7\}$, and held-out test roster sizes $\{9, 10\}$.

Baselines. We compare PC3D against three MARL baselines chosen to evaluate its contribution under the same decentralized execution setting: agents act from local histories without execution-time communication, privileged coordinators, global observations, or problem-specific state decompositions. Our analyzes systematically evaluate the gains introduced by personalized context distillation and adaptive context use in isolation, rather than reporting an exhaustive benchmarking study. **IPPO** is the MARL adaptation of the Proximal Policy Optimization algorithm with an independent learning setting, where both training and execution are fully decentralized (de Witt et al., 2020; Yu et al., 2022). **MAPPO** extends it with a centralized critic and serves as our backbone method (Yu et al., 2022). In our variable-roster setting, MAPPO uses a fixed-width critic input based on the maximum admitted roster size, with inactive slots masked. **PIC-MAPPO** replaces the fixed-width centralized critic with a permutation-invariant set critic (Liu et al., 2020), but it does not distill personalized teacher contexts or adaptively condition the actor on the recovered context. All method implementations employ recurrent and shared-parameter actor networks so that the same policy can be reused across roster sizes.

Training and evaluation. We adopt a staged curriculum that gradually increases roster diversity (Long et al., 2020; Agarwal et al., 2020). We specify these stages in Figure 4. We report five seeded repetitions for each method-task pair.



(a) Spread *(b)* LBF *(c)* RWARE

Figure 4. **Training returns.** Curves show mean training returns (±95 CI) across seeds, with each colored patch corresponding to one curriculum stage and its active training roster set. Higher returns are better.

Table 2. **Ablation study.** We train three versions of each model with ablations. Entries report mean ± standard deviation across five seeded final checkpoints, using the same split-level aggregation as Table 1. PC3D-MAPPO row is taken from Table 1. **Bold** indicates the best variant in each column.

Method	Spread			LBF ($\times 1e2$)			RWARE		
	Train	Validation	Test	Train	Validation	Test	Train	Validation	Test
PC3D-MAPPO	-39.90 ± 0.7	$\mathbf{-48.09 \pm 1.0}$	$\mathbf{-79.18 \pm 1.5}$	$\mathbf{7.91 \pm 0.7}$	4.47 ± 0.3	$\mathbf{8.98 \pm 0.1}$	$\mathbf{3.58 \pm 1.5}$	$\mathbf{3.53 \pm 1.5}$	$\mathbf{7.73 \pm 2.7}$
Always Off Gate	-41.71 ± 0.8	-50.23 ± 0.7	-82.95 ± 2.3	7.09 ± 0.5	3.54 ± 0.6	8.39 ± 0.5	2.14 ± 0.9	2.00 ± 0.9	5.43 ± 2.1
Always On Gate	-41.40 ± 0.7	-49.67 ± 1.1	-82.20 ± 2.1	7.67 ± 1.0	4.28 ± 0.5	8.93 ± 0.3	3.19 ± 2.1	3.06 ± 2.0	7.22 ± 3.9
A-MAPPO	$\mathbf{-39.78 \pm 0.9}$	-48.24 ± 1.1	-80.65 ± 2.2	7.54 ± 0.7	$\mathbf{4.61 \pm 0.4}$	8.97 ± 0.4	2.69 ± 1.7	2.58 ± 1.7	6.14 ± 2.9

For the results reported in Tables 1 and 2, and Figure 5, the **final checkpoints** are evaluated on train, validation, and test agent-count splits, with 100 rollouts per count.

4.2. Main results

Table 1 reports final-checkpoint performance across the three benchmarks. PC3D-MAPPO obtains the strongest mean return in all tasks and splits, including held-out roster sizes that are never seen during training. The gains are clearest in Spread, where the PC3D actor improves substantially over the second-best method (PIC-MAPPO). LBF shows the same pattern on a different reward scale, with PC3D improving validation and test returns while preserving better training performance. RWARE results appear noisier (reflected in larger standard deviations), but display the same pattern: PC3D improves its backbone (MAPPO) by a clear margin and performs the best on train, validation, and test counts.

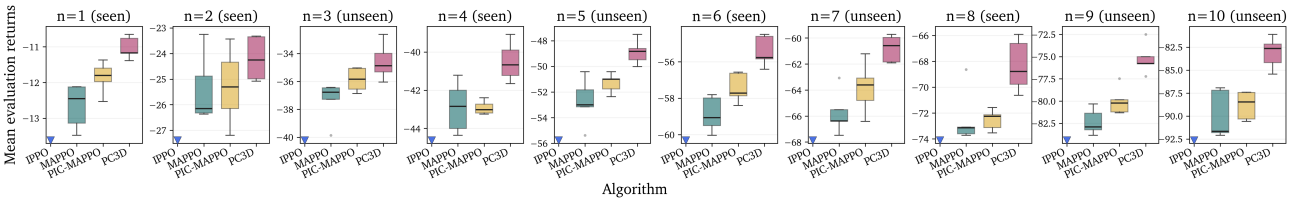
The learning curves in Figure 4 show optimization behavior under the active curriculum distribution. First, PC3D remains competitive throughout the curriculum, with more notable improvements over baselines in later stages as rosters grow and roster distribution becomes more diverse. This is consistent with the primary objective of PC3D: a method that extends the single-roster optimizers to generalize across diverse roster distributions.

Figure 5 shows the evaluation returns for each method and benchmark across the used roster sizes. These plots better highlight count-specific performances that are compressed in the split means we report in Table 1. PC3D generally shifts the return distribution upward across both seen and unseen counts, rather than improving only a single favorable roster size. In particular, evaluations on larger rosters show that PC3D widens the margin over baselines as coordination becomes less trivial.

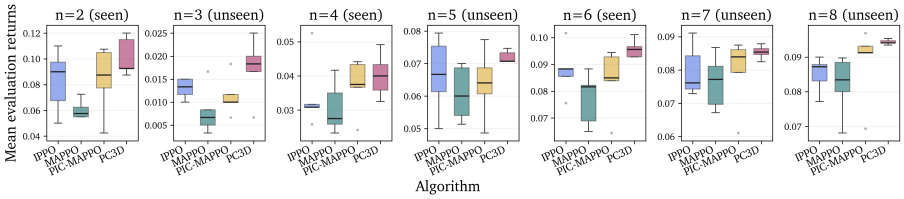
4.3. Ablations

We perform ablations to test whether the gains reported in Section 4.2 are correlated with our methodological objectives. Using the same PC3D runs (from Section 4.2), we ablate the two mechanisms that form the basis of the intuition behind PC3D: context distillation and adaptive context conditioning. **Always Off Gate** sets $g_i^t = 0$ (see Eq. 3), preventing context modulation; **Always On Gate** sets $g_i^t = 1$, forcing non-adaptive modulation; and **A-MAPPO** sets $\lambda_{\text{distill}} = 0$ (see Eq. 5), retaining the attention critic and feature conditioning path without teacher-student alignment. The ablations use the same training and evaluation protocol as the corresponding PC3D runs.

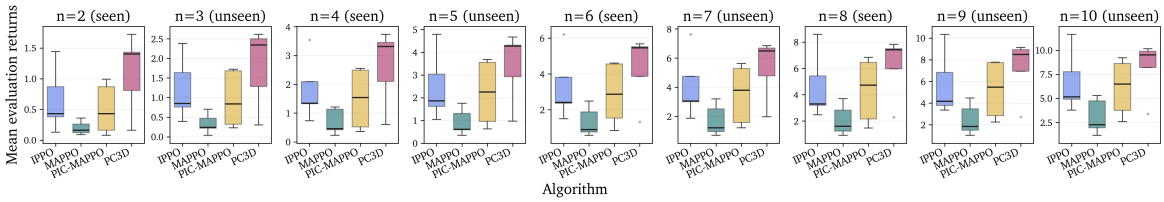
The results presented in Table 2 support three conclusions. First, turning the gate off consistently hurts performance, showing that the learned context pathway is not a passive auxiliary head. Second, forcing the gate on is competitive in LBF but notably weaker in Spread and RWARE, suggesting that adaptive reliance is most useful when roster diversity increases and task demands vary across roster sizes. Third, removing distillation can occasionally remain competitive (most notably Spread seen and LBF unseen splits), but it weakens generalization in Spread and substantially hurts RWARE across splits. Overall, these results suggest that the centralized teacher does not merely improve the critic; it



(a) Spread



(b) LBF



(c) RWARE

Figure 5. **Evaluation returns across roster sizes.** Final-checkpoint returns for each evaluated roster size. Each count is evaluated separately with 100 rollouts. Downward markers indicate methods whose returns fall below the displayed range.

provides a personalized signal that helps the recurrent actor recover and use coordination context under roster shift.

5. Conclusions

This study focused on open-team cooperation under episodic roster variation and partial observability, where fully decentralized agents cooperate across varying and unseen team sizes. We formalized this setting as a family of roster-indexed Dec-POMDPs induced by a shared template, and argued that standard CTDE methods lack an explicit mechanism for turning centralized coordination information into a reusable decentralized representation.

We introduced **PC3D** on top of a MAPPO backbone as a method that trains a set-structured centralized teacher to personalize its context and distill it into decentralized policies. The resulting actor recovers a student coordination context from local history and adaptively uses it through gated feature modulation. In contrast to approaches that address OTC by introducing structural assumptions, PC3D preserves the fully decentralized execution contract while providing the policy with a direct training signal to recover useful coordination context from local interactions, supporting zero-shot adaptation across varying roster sizes.

Across Spread, LBF, and RWARE, PC3D improves over IPPO, MAPPO, and PIC-MAPPO on both seen and unseen roster sizes. Furthermore, the ablations support that non-adaptive modulation or the removal of distillation weakens performance, especially under larger roster shifts. Generally, our results indicate that open-team cooperation should be treated not only as a robustness problem but also as a representation-transfer problem between centralized training and decentralized execution.

Extending PC3D to heterogeneous teams and in-episode roster changes is a natural next step. Moreover, testing it with value-factorization or off-policy critic CTDE methods would clarify the transferability of the coordination-distillation principle. PC3D is least compelling when execution permits communication or centralized observations, or when the task does not contain a reusable cooperative structure across rosters. It is intended for settings where centralized roster-dependent representations can be personalized and meaningfully guide decentralized execution.

PC3D aims to support more robust decentralized coordination in robotics, logistics, and distributed control. However, deployment in safety-critical settings requires additional validation, as failures in unseen team configurations could lead to unsafe collective behavior.

Accessibility. We make our codebase (with hyperparameters and environment configurations) publicly available in an online repository[1].

[1]Anonymized repository: `https://anonymous.4open.science/r/pc3d_anon`

References

Agarwal, A., Kumar, S., Sycara, K., and Lewis, M. Learning Transferable Cooperative Behavior in Multi-Agent Teams. In *Proceedings of the 19th International Conference on Autonomous Agents and Multi Agent Systems*, AAMAS '20, pp. 1741–1743, Richland, SC, 2020. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450375184.

Ahmed, Z., Le Roux, N., Norouzi, M., and Schuurmans, D. Understanding the Impact of Entropy on Policy Optimization. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 151–160. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/ahmed19a.html.

Akman, A. O., Psarou, A., Hoffmann, M., Łukasz Gorczyca, Łukasz Kowalski, Gora, P., Jamróz, G., and Kucharski, R. URB – Urban Routing Benchmark for RL-equipped Connected Autonomous Vehicles. In *Advances in Neural Information Processing Systems*, 2025.

Amato, C. An Introduction to Centralized Training for Decentralized Execution in Cooperative Multi-Agent Reinforcement Learning, 2024. URL https://arxiv.org/abs/2409.03052.

Bernstein, D. S., Givan, R., Immerman, N., and Zilberstein, S. The complexity of decentralized control of Markov decision processes. *Mathematics of operations research*, 27(4):819–840, 2002.

Boesch, P. M., Ciari, F., and Axhausen, K. W. Autonomous vehicle fleet sizes required to serve different levels of demand. *Transportation Research Record*, 2542(1):111–119, 2016.

Chen, Y., Mao, H., Mao, J., Wu, S., Zhang, T., Zhang, B., Yang, W., and Chang, H. PTDE: personalized training with distilled execution for multi-agent reinforcement learning. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, IJCAI '24, 2024. ISBN 978-1-956792-04-1. doi: 10.24963/ijcai.2024/4. URL https://doi.org/10.24963/ijcai.2024/4.

Cho, K., van Merrienboer, B., Bahdanau, D., and Bengio, Y. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches, 2014. URL https://arxiv.org/abs/1409.1259.

Christianos, F., Schäfer, L., and Albrecht, S. V. Shared Experience Actor-Critic for Multi-Agent Reinforcement Learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

de Witt, C. S., Gupta, T., Makoviichuk, D., Makoviychuk, V., Torr, P. H. S., Sun, M., and Whiteson, S. Is Independent Learning All You Need in the StarCraft Multi-Agent Challenge?, 2020. URL https://arxiv.org/abs/2011.09533.

Foerster, J. N., Farquhar, G., Afouras, T., Nardelli, N., and Whiteson, S. Counterfactual multi-agent policy gradients. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018. ISBN 978-1-57735-800-8.

Gronauer, S. and Diepold, K. Multi-agent deep reinforcement learning: a survey. *Artificial Intelligence Review*, 55(2):895–943, 2022.

Hausknecht, M. J. and Stone, P. Deep Recurrent Q-Learning for Partially Observable MDPs. In *AAAI fall symposia*, volume 45, pp. 141, 2015.

Iqbal, S. and Sha, F. Actor-attention-critic for multi-agent reinforcement learning. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2961–2970. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/iqbal19a.html.

Iqbal, S., De Witt, C. A. S., Peng, B., Boehmer, W., Whiteson, S., and Sha, F. Randomized entity-wise factorization for multi-agent reinforcement learning. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 4596–4606. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/iqbal21a.html.

Lee, J., Lee, Y., Kim, J., Kosiorek, A., Choi, S., and Teh, Y. W. Set Transformer: A Framework for Attention-based Permutation-Invariant Neural Networks. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3744–3753. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/lee19d.html.

Liu, B., Liu, Q., Stone, P., Garg, A., Zhu, Y., and Anandkumar, A. Coach-Player Multi-agent Reinforcement Learning for Dynamic Team Composition. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6860–6870. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/liu21m.html.

Liu, I.-J., Yeh, R. A., and Schwing, A. G. PIC: Permutation Invariant Critic for Multi-Agent Deep Reinforcement Learning. In Kaelbling, L. P., Kragic, D., and Sugiura, K. (eds.), *Proceedings of the Conference on Robot Learning*, volume 100 of *Proceedings of Machine Learning Research*, pp. 590–602. PMLR, 30 Oct–01 Nov 2020. URL https://proceedings.mlr.press/v100/liu20a.html.

Long, Q., Zhou, Z., Gupta, A., Fang, F., Wu, Y., and Wang, X. Evolutionary Population Curriculum for Scaling Multi-Agent Reinforcement Learning, 2020. URL https://arxiv.org/abs/2003.10423.

Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., and Mordatch, I. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/68a9750337a418a86fe06c1991a1d64c-Paper.pdf.

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

Oroojlooy, A. and Hajinezhad, D. A review of cooperative multi-agent deep reinforcement learning. *Applied Intelligence*, 53(11):13677–13722, 2023.

Papoudakis, G., Christianos, F., Schäfer, L., and Albrecht, S. V. Benchmarking Multi-Agent Deep Reinforcement Learning Algorithms in Cooperative Tasks. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS)*, 2021.

Perez, E., Strub, F., de Vries, H., Dumoulin, V., and Courville, A. FiLM: Visual Reasoning with a General Conditioning Layer. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018. doi: 10.1609/aaai.v32i1.11671. URL https://ojs.aaai.org/index.php/AAAI/article/view/11671.

Portugal, D. and Rocha, R. P. Performance Estimation and Dimensioning of Team Size for Multirobot Patrol. *IEEE Intelligent Systems*, 32(6):30–38, 2017. doi: 10.1109/MIS.2017.4531222.

Qu, B., Mao, L., Xu, Z., Feng, J., and Wang, X. How Many Vehicles Do We Need? Fleet Sizing for Shared Autonomous Vehicles With Ridesharing. *IEEE Transactions on Intelligent Transportation Systems*, 23(9):14594–14607, 2022. doi: 10.1109/TITS.2021.3130749.

Rahman, A., Carlucho, I., Höpner, N., and Albrecht, S. V. A General Learning Framework for Open Ad Hoc Teamwork Using Graph-based Policy Learning. *Journal of Machine Learning Research*, 24(298):1–74, 2023. URL http://jmlr.org/papers/v24/22-099.html.

Rashid, T., Samvelyan, M., de Witt, C. S., Farquhar, G., Foerster, J., and Whiteson, S. Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning. *Journal of Machine Learning Research*, 21 (178):1–51, 2020. URL http://jmlr.org/papers/v21/20-081.html.

Rjeb, A., Gayon, J.-P., and Norre, S. Sizing of a homogeneous fleet of robots in a logistics warehouse. *IFAC-PapersOnLine*, 54(1):552–557, 2021. ISSN 2405-8963. doi: https://doi.org/10.1016/j.ifacol.2021.08.169. URL https://www.sciencedirect.com/science/article/pii/S2405896321009393. 17th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2021.

Rosenfeld, A., Kaminka, G. A., and Kraus, S. *A Study of Scalability Properties in Robotic Teams*, pp. 27–51. Springer US, Boston, MA, 2006. doi: 10.1007/0-387-27972-5_2. URL https://doi.org/10.1007/0-387-27972-5_2.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal Policy Optimization Algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Schulman, J., Moritz, P., Levine, S., Jordan, M., and Abbeel, P. High-dimensional continuous control using generalized advantage estimation, 2018. URL https://arxiv.org/abs/1506.02438.

Shao, J., Lou, Z., Zhang, H., Jiang, Y., He, S., and Ji, X. Self-Organized Group for Cooperative Multi-agent Reinforcement Learning. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 5711–5723. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/25b040c97a75021e57100648a20b1e10-Paper-Conference.pdf.

Stone, P., Kaminka, G., Kraus, S., and Rosenschein, J. Ad Hoc Autonomous Agent Teams: Collaboration without Pre-Coordination. *Proceedings of the AAAI Conference on Artificial Intelligence*, 24(1):1504–1509, Jul. 2010. doi: 10.1609/aaai.v24i1.7529. URL https://ojs.aaai.org/index.php/AAAI/article/view/7529.

Sunehag, P., Lever, G., Gruslys, A., Czarnecki, W. M., Zambaldi, V., Jaderberg, M., Lanctot, M., Sonnerat, N., Leibo, J. Z., Tuyls, K., and Graepel, T. Value-Decomposition Networks For Cooperative Multi-Agent Learning Based On Team Reward. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '18, pp. 2085–2087, Richland, SC, 2018. International Foundation for Autonomous Agents and Multiagent Systems.

Terry, J., Black, B., Grammel, N., Jayakumar, M., Hari, A., Sullivan, R., Santos, L. S., Dieffendahl, C., Horsch, C., Perez-Vicente, R., et al. Pettingzoo: Gym for multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 34:15032–15043, 2021.

Wang, J., Li, Y., Zhang, Y., Pan, W., and Kaski, S. Open Ad Hoc Teamwork with Cooperative Game Theory. In Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F. (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 50902–50930. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/wang24an.html.

Wang, W., Ye, D., and Lu, Z. Mutual-Information Regularized Multi-Agent Policy Iteration. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 2617–2635. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/079949 2e7be38b66d10ead5e8809616d-Paper-Conference.pdf.

Yu, C., Velu, A., Vinitsky, E., Gao, J., Wang, Y., Bayen, A., and WU, Y. The surprising effectiveness of ppo in cooperative multi-agent games. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24611–24624. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/9c1535a02f0ce079433344e14d910597-Paper-Datasets_and_Benchmarks.pdf.

Yuan, L., Zhang, Z., Li, L., Guan, C., and Yu, Y. A survey of progress on cooperative multi-agent reinforcement learning in open environment. *arXiv preprint arXiv:2312.01058*, 2023.

Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R. R., and Smola, A. J. Deep Sets. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/f22e4747da1aa27e363d86d40ff442fe-Paper.pdf.

Zhang, K., Yang, Z., and Başar, T. Decentralized multi-agent reinforcement learning with networked agents: Recent advances. *Frontiers of Information Technology & Electronic Engineering*, 22(6):802–814, 2021a.

Zhang, K., Yang, Z., and Başar, T. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handbook of reinforcement learning and control*, pp. 321–384, 2021b.

Zhao, J., Hu, X., Yang, M., Zhou, W., Zhu, J., and Li, H. CTDS: Centralized Teacher With Decentralized Student for Multiagent Reinforcement Learning. *IEEE Transactions on Games*, 16(1):140–150, 2024. doi: 10.1109/TG.2022.3232390.