Interactive Knowledge Distillation with Adaptive Teachers in Cooperative Multi-Agent Reinforcement Learning

Minwoo Cho†, Batuhan Altundas, and Matthew Gombolay
Institute for Robotics and Intelligent Machines
Georgia Institute of Technology, Atlanta, Georgia 30332
{mcho318, batuhan}@gatech.edu, matthew.gombolay@cc.gatech.edu

***Abstract*—Knowledge distillation (KD) has the potential to accelerate multi-agent reinforcement learning (MARL) by employing a centralized teacher for decentralized students. However, centralized teachers in MARL often fail because decentralized student exploration induces out-of-distribution (OOD) state distributions the teacher was never trained on, compounded by partial observability, which creates observation mismatches between teacher and students at execution time. We propose HINT (Hierarchical INteractive Teacher-based transfer), a novel KD framework for MARL in a centralized training, decentralized execution setup. By leveraging hierarchical RL, HINT provides a scalable, high-performing teacher. Pseudo off-policy RL treats student trajectories as additional training data for the teacher, allowing it to adapt its policy to student-induced state distributions. Performance-based filtering removes teacher guidance that depends on centralized observations unavailable to decentralized students, retaining only outcome-relevant signals. Across FireCommander and MARINE, HINT consistently outperforms state-of-the-art online MARL baselines, improving task success rates by 60%–165%.**

I. Introduction

Multi-agent reinforcement learning (MARL) has emerged as a promising approach for coordinating autonomous robot teams across various real-world applications, including search and rescue [27, 5], replenishment at sea [8, 3], and warehouse management [18, 33, 2]. As task complexity and team size increase, there is growing demand for methods that enable effective coordination without sacrificing individual agent autonomy. Despite many advancements, deploying MARL remains non-trivial due to challenges including partial observability, non-stationary dynamics, and effective credit assignment.

To address these challenges, centralized training with decentralized execution (CTDE) has become a dominant paradigm in MARL. CTDE utilizes global state information or joint observations during the training phase to stabilize learning, while preserving scalability by restricting agents to local observations during execution. Building on this framework, numerous CTDE-based algorithms have shown success in cooperative benchmarks [28, 35, 46, 19]. Building further, communication-based approaches [36, 34, 32] have been developed to enhance coordination by enabling agents to share information, infer shared context, and adapt to dynamic conditions. While this extension greatly improves cooperation, it also complicates training: agents must simultaneously optimize both decision-making and communication, which increases sensitivity to noisy signals and policy gradient variance [11]. To overcome these limitations, we propose a novel framework that integrates hierarchical supervision with interactive knowledge distillation to enable robust MARL with minimal coordination overhead.

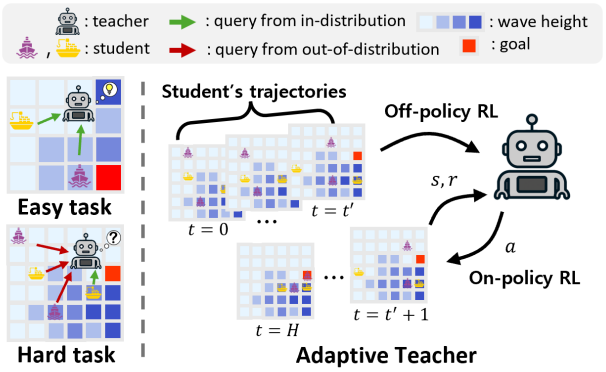

Fig. 1. **Bridging the distribution gap between teacher and student.** As task complexity increases, teachers trained offline provide unreliable guidance on student trajectories that diverge from the training distribution; we close this gap via adaptive refinement. (Example environment: MARINE)

Recently, knowledge distillation (KD) [14], which allows a simpler student model to learn by mimicking the outputs of a larger teacher model, has emerged as a promising alternative to alleviate unstable training signals and the credit assignment problem in MARL [47, 4, 15, 22]. In this paradigm, a centralized teacher with access to global states supervises student policies by minimizing the divergence between their respective policy distributions. However, existing distillation approaches face key bottlenecks. As task complexity increases, decentralized students explore out-of-distribution (OOD) states that lie outside the teacher's training distribution, resulting in inconsistent or poor-quality demonstrations (see Fig. 1, Sec. IV). Furthermore, teachers rely on centralized observations that are inaccessible to decentralized students under partial observability, leading to observation mismatches at execution time—teachers must adapt guidance dynamically to align with student context.

To address these limitations, we propose HINT (Hierarchical INteractive Teacher-based transfer) – a novel KD

†Corresponding author: Minwoo Cho (mcho318@gatech.edu). Our code is available at https://github.com/CORE-Robotics-Lab/HINT

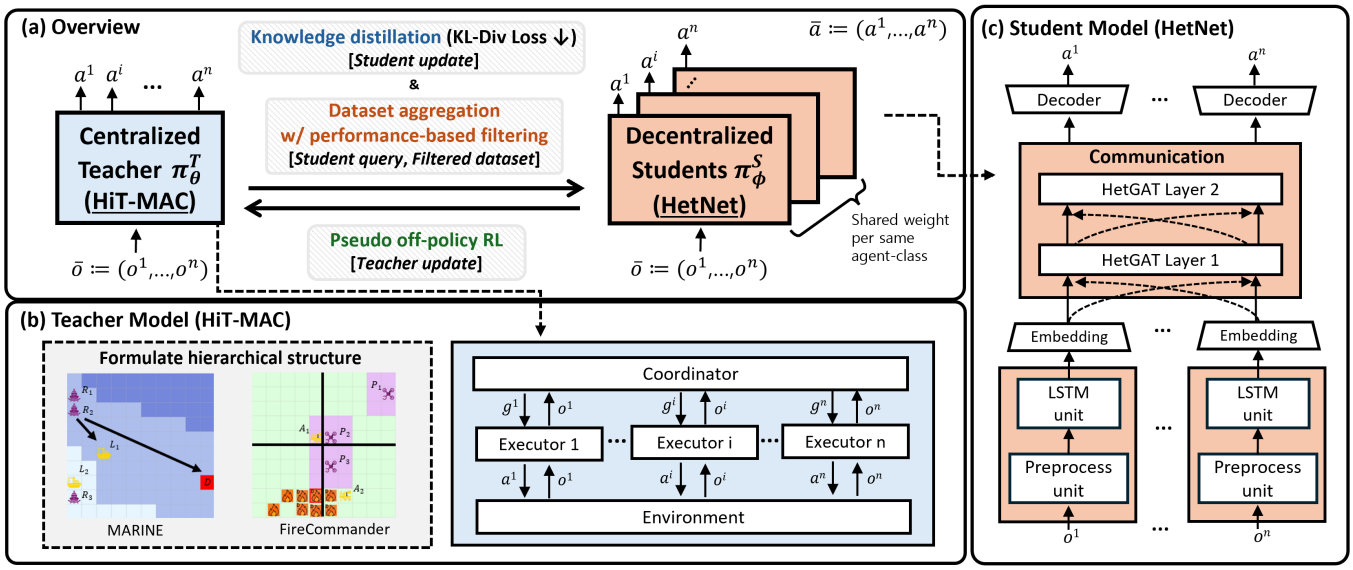

Fig. 2. **Overview of HINT.** (a) A centralized teacher guides decentralized students via three mechanisms: knowledge distillation for student updates, pseudo off-policy RL for teacher refinement, and dataset aggregation with performance-based filtering to support student queries and build a high-quality dataset. (b) The teacher operates hierarchically, with a high-level coordinator assigning subgoals to low-level executors based on a task-specific hierarchical structure. This structure enables temporal abstraction by decoupling strategic and tactical decisions. (c) Each student includes a preprocessing unit, an LSTM encoder for temporal abstraction, and a decoder for action selection, while HetGAT layers enable inter-agent communication.

framework for effective cooperative MARL (Fig. 2). We first pre-train a centralized teacher with hierarchical reinforcement learning (RL), decomposing decisions across two levels to enhance scalability and performance. Then, decentralized student policies are trained via KD, augmented by online expert queries to closely align training data with the student's test-time distribution. A key feature of HINT is *pseudo off-policy RL*, in which the teacher is updated using both its own and student trajectories. This improves awareness of student behavior and enables richer guidance in OOD states. Performance-based filter further improves dataset quality by retaining only outcome-relevant demonstrations, reducing observation mismatches. We test HINT in two challenging environments (e.g., FireCommander (FC) [30] for resource allocation, MARINE [13] for tactical combat), where stochasticity and time-varying dynamics pose additional training difficulties. Experiments show that HINT consistently outperforms baselines, validating its robustness in complex domains. **Our contributions are as follows:**

- We design a hierarchical teacher that decomposes multi-agent coordination into high-level goal assignment and low-level execution, enabling scalable and practical distillation.
- We propose pseudo off-policy RL that updates the teacher on mixed student-teacher trajectories and a performance-based filter that validates demonstrations via forward simulation, addressing distribution shift and observation mismatch. These interactive mechanisms relax the reliance on oracle-level teachers assumed in prior KD methods.
- We demonstrate that HINT outperforms competitive CTDE and KD baselines with 60%–165% gains in task success rate, indicating enhanced coordination and robustness.
- We validate HINT on physical multi-robot systems, confirming that the proposed methods transfer to real-world settings.

II. Related Work

A. CTDE in Cooperative MARL

CTDE has been widely adopted in cooperative MARL to reduce instability from non-stationarity by leveraging centralized value functions during training while enabling decentralized execution. These methods fall into off-policy [28, 35, 16, 40, 23] and on-policy categories [12, 46, 21, 19], balancing sample efficiency and learning stability. For instance, QMIX [28] and QTRAN [35] use experience replay buffers for scalability, while MAPPO [46] and DAE [21] rely on trust-region updates to mitigate unstable gradients.

A persistent and fundamental challenge in CTDE is fairly assigning credit to agents according to their individual contributions to team rewards. Various strategies have been proposed to address credit assignment: value decomposition (QMIX [28], QTRAN [35]), attention-based critics (MAAC [16]), and explicit fairness estimators like Shapley value (SQDDPG [40]) or counterfactual baselines (COMA [12]). More recent methods like HAPPO [19] and DAE [21] further refine gradient flows by decomposing joint advantages or leveraging difference rewards.

To supplement centralized training and enable adaptive coordination during decentralized execution, some CTDE methods include communication mechanisms. These include continuous channels (CommNet [36]), conditional activation (IC3Net [34]), and attention-based messaging (TarMAC [7], HetNet [30, 32]), which allow agents to share contextual cues and prioritize relevant interactions.

Despite these advancements, CTDE methods continue to struggle with unstable training due to fundamental architectural constraints in shared-value decomposition and inter-agent coordination. To overcome these challenges, we adopt

a centralized hierarchical RL framework, which decomposes complexity and allows for coarse-to-fine temporal abstraction (Sec. III-A) and enable decentralized execution through knowledge distillation (Secs. III-B and V-A).

B. Multi-Agent Learning with KD

Recently, KD has emerged in MARL as a promising method for transferring coordination strategies from centralized to decentralized agents. One of the main advantages of KD is the provision of stable training signals by leveraging a teacher model with access to global states or joint observations, implicitly addressing challenges such as credit assignment and non-stationarity. This benefit, however, often assumes an oracle-like teacher that consistently offers optimal guidance.

Distillation strategies vary by (1) what is distilled (e.g., value [47, 48] vs. policy [22]), (2) who teaches (e.g., planners [6, 20], humans [31, 45]), and (3) how interactivity is handled (e.g., one-shot [25, 38] vs. iterative [22, 15]). However, these methods often rely on static, oracle-level supervision, which is brittle in open-ended environments. Our empirical findings suggest that this assumption of optimality breaks down in complex, dynamic cooperative tasks, particularly when teachers are likely to encounter unfamiliar states (Sec. IV). To overcome this limitation, we enable adaptive supervision through a hierarchical teacher that continuously refines its policy based on student behaviors (Sec.V-B) and applies a performance-based filtering to selectively distill high-quality demonstrations (Sec.V-C).

III. Preliminaries

A. Hierarchical Target-oriented Multi-Agent Coordination

We employ hierarchical target-oriented multi-agent coordination (HiT-MAC) [44] as our centralized teacher because of its robust performance in dynamic, complex environments. Specifically, HiT-MAC facilitates efficient multi-agent training through hierarchical agent-target structures. In Fig. 2b, a high-level coordinator assigns subgoals based on joint observations, while each agent acts as a low-level executor, selecting primitive actions using its local observation and assigned subgoal.

Technically, HiT-MAC operates under a centralized training and centralized execution (CTCE) paradigm, leveraging a self-attention mechanism [39] over joint observations to capture inter-agent dependencies and evaluate the relative importance of agent-target pairs. This design supports effective credit assignment via Shapley value approximation, quantifying each pair's contribution to overall performance. Through this mechanism, HiT-MAC not only resolves challenging credit assignment problems in CTDE methods but also establishes itself as a reliable, high-performing teacher for decentralized students. Empirically, HiT-MAC achieves over 80% success rates in our challenging benchmark scenarios, outperforming tested CTDE and KD baselines. To accommodate agent heterogeneity, we extend the original design with class-specific encoders that handle diverse observation modalities and information access. Further design details are provided in Appendix B.

B. Heterogeneous Policy Network (HetNet)

To support decentralized execution among heterogeneous agents, we adopt HetNet [32, 30] as our student model. HetNet is specifically designed to support decentralized coordination by modeling agent-specific observation and action modalities.

Fig. 2c shows that each agent processes its local observations through a preprocessing module, followed by an LSTM [17], which helps mitigate partial observability by retaining relevant historical information. To enable structured communication, we employ heterogeneous graph attention (HetGAT) layers that assign attention weights conditioned on both agent types and their relational context. This mechanism enables agents to selectively attend to relevant teammates and share task-specific information. Finally, a decoder integrates the local features with aggregated graph messages to produce the action probability. Further architectural details are provided in Appendix B.

C. Notation for Teacher and Student Policies

The teacher policy π_θ^T is defined over the joint observations $\bar{o} = (o^1, \ldots, o^n)$ and joint actions $\bar{a} = (a^1, \ldots, a^n)$ as

$$\pi_\theta^T(\bar{a}|\bar{o}) = \sum_{\bar{g}} \pi_\theta^{T_H}(\bar{g}|\bar{o}) \prod_{i=1}^{n} \pi_\theta^{T_{L_i}}(a^i|o^i, g^i) \tag{1}$$

where $\pi_\theta^{T_H}$ is a high-level policy that assigns subgoals $\bar{g} = (g^1, \ldots, g^n)$ based on the joint observations $\bar{o}$, and $\pi_\theta^{T_{L_i}}$ is a low-level policy for agent i, selecting action a^i using its observation o^i and the assigned subgoal g^i. The teacher policy π_θ^T is parameterized by θ, and the corresponding value function $V_\psi^T(\bar{o})$ is parameterized by ψ.

The joint student policy π_ϕ^S is defined over joint observations $\bar{o} = (o^1, \ldots, o^n)$ and joint actions $\bar{a} = (a^1, \ldots, a^n)$ as

$$\pi_\phi^S(\bar{a} \mid \bar{o}) = \prod_{i=1}^{n} \pi_\phi^{S_i}(a^i \mid o^i) \tag{2}$$

where each $\pi_\phi^{S_i}$ is a decentralized policy that selects a^i based on local observation o^i. The policy π_ϕ^S is parameterized by ϕ.

IV. Student State Distribution When Scaling Environments

This section investigates how student state distributions shift as the environment scales and team size grows. Focusing on student rollouts with a 10–30% success rate, we compare their trajectories with teacher demonstrations by projecting both into a shared latent space (see Fig. 3). Here, PCA [1] is utilized to visualize differences in a common latent space. In smaller settings, student distributions generally align with those of the teacher, suggesting that the teacher may still provide high-quality demonstrations. However, in more complex environments with larger agent teams, students diverge (e.g., higher KL-divergence) and result in more OOD states, diminishing the teacher's ability to offer consistent and optimal guidance. To address this challenge, we propose two key components: pseudo off-policy RL and performance-based

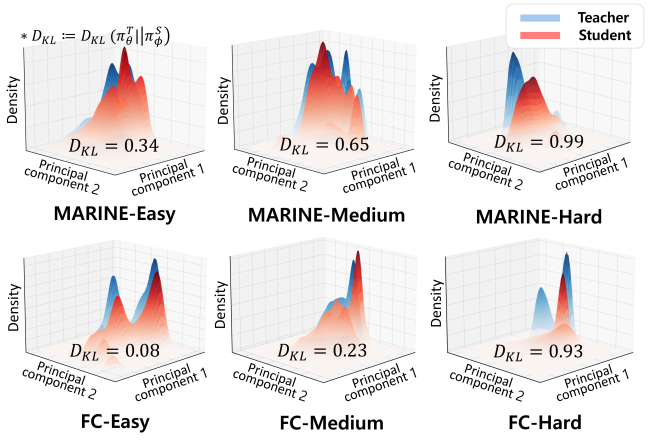

Fig. 3. Comparison of teacher (blue) and student (red) state distributions projected into a shared latent space for MARINE and FC. Each setting corresponds to student rollouts with a 10–30% success rate. As task complexity increases, the gap between student and teacher distributions widens (measured by KL-divergence), indicating that teachers are increasingly exposed to out-of-distribution (OOD) states.

filtering (Secs. V-B and V-C). Their effectiveness is evaluated in Secs. VI-B and VI-C.[1]

V. Method

In this section, we propose HINT, a framework for interactive knowledge distillation designed to support robust multi-agent learning (see Fig. 2a). HINT integrates three components: (1) knowledge distillation, (2) pseudo off-policy RL, and (3) performance-based filtering. At its core, the teacher refines its policy using student trajectories to provide more informative guidance. Meanwhile, the student runs its policies and selectively queries the teacher for improved actions, ensuring that only outcome-relevant feedback is retained. This cycle repeats until student policies converge. Please note that HINT adopts a multi-threaded structure throughout all submodules to enhance computational efficiency.

A. Knowledge Distillation

Compared to prior KD methods in MARL, we provide more efficient and robust demonstrations using a centralized, hierarchical teacher, because it enables systematic decomposition of decision-making, which improves tractability in complex multi-agent tasks. Since the teacher can access all agents' observations and infer the global context, it can efficiently address credit assignment among agents. However, to enable decentralized execution of agents, transferring knowledge from the teacher to student is necessary. The overall procedure is summarized in Alg. 1. Note that while we follow the typical KD procedure, our approach is distinct in that the dataset $\mathcal{D}$ is continuously augmented with an adaptive teacher, thereby providing richer guidance to the students.

[1]Additional examples of suboptimal demonstrations and the procedure for Fig. 3 are given in Appendix C, with environment details in Sec. VI-A and Appendix A.

Algorithm 1: Knowledge Distillation

Input: Dataset $\mathcal{D} = \{\tau = (\bar{o}_t, \bar{a}_t)_{t=1}^{H}\}$, Buffer $\mathcal{B}$
1 **Initialize** ϕ as the parameters of π_ϕ^S
2 **for** τ in $\mathcal{D}$ **do**
3 **for** $t = 1$ **to** H **do**
4 Compute $\log \pi_\phi^S(\bar{a}_t|\bar{o}_t)$, and $\mathcal{H}(\pi_\phi^S(\cdot \mid \bar{o}_t))$ and store in $\mathcal{B}$;
5 **if** $\mathcal{B}$ is full **then**
6 Compute loss $\mathcal{L}_\phi$ using Eq. 3; // KL-div. with entropy
7 $\phi \leftarrow \phi - \lambda_\phi \nabla_\phi \mathcal{L}_\phi$;
8 Clear buffer $\mathcal{B}$;

Algorithm 2: Pseudo Off-Policy RL

Input: Teacher π_θ^T, Student π_ϕ^S, Buffer $\mathcal{B}$, Episodes N_{pseudo}
1 Freeze π_ϕ^S;
2 **for** $e = 1$ **to** N_{pseudo} **do**
3 Sample switch point $t' \sim \text{Uniform}(1, H)$;
4 **for** $t = 0$ **to** H **do**
5 **if** $t \leq t'$ **then**
6 $\bar{a}_t \sim \pi_\phi^S(\cdot \mid \bar{o}_t)$; // Student explores
7 **else**
8 $\bar{a}_t \sim \pi_\theta^T(\cdot \mid \bar{o}_t)$; // Teacher resumes
9 Store $(\bar{o}_t, \bar{a}_t)$ in $\mathcal{B}$;
10 **if** $\mathcal{B}$ contains n steps **then**
11 Compute v_t^T (Eq. 4); // Correct off-policy
12 Compute value loss $\mathcal{L}_\psi$ (Eq. 5);
13 $\psi \leftarrow \psi - \lambda_\psi \nabla_\psi \mathcal{L}_\psi$; // Update teacher's value
14 Compute policy objective J_θ (Eq. 6);
15 $\theta \leftarrow \theta + \lambda_\theta \nabla_\theta J_\theta$; // Update teacher's policy
16 Clear $\mathcal{B}$;

Our policy distillation objective minimizes the difference between teacher policy π_θ^T and student policy π_ϕ^S by combining KL-divergence with entropy regularization to encourage both imitation and diversity. The loss is weighted by coefficient α in Eq. 3, and Alg. 1 updates ϕ via $\nabla_\phi \mathcal{L}_\phi$ (line 7).

$$\mathcal{L}_\phi = \mathbb{E}_{(\bar{o}_t, \bar{a}_t) \sim \pi_\theta^T} \Big[\big(\log \pi_\theta^T(\bar{a}_t \mid \bar{o}_t) - \log \pi_\phi^S(\bar{a}_t \mid \bar{o}_t) \big) + \alpha \mathcal{H}(\pi_\phi^S(\cdot \mid \bar{o}_t)) \Big] \tag{3}$$

To ensure stable training, samples are stored in a replay buffer and updated using batch-based gradient optimization. We empirically found this formulation yields stable gradients across tested benchmarks (see Appendix F).

B. Pseudo Off-Policy Reinforcement Learning

Our pseudo off-policy RL presents a new perspective on how the teacher and student co-evolve in multi-agent learning. The overall procedure is shown in Alg. 2. Rather than treating student and teacher data separately, we combine them within a single trajectory: the student explores initially, then the teacher completes the path (lines 5-8). This joint rollout structure (Fig.4) bridges the gap between teacher and student, allowing the teacher to learn not only what to do, but how to recover from student mistakes (lines 11-15). This approach yields a more resilient teacher and facilitates student learning through richer, context-aware guidance—especially in complex settings where students often deviate from training distributions and reactive corrections alone are insufficient.

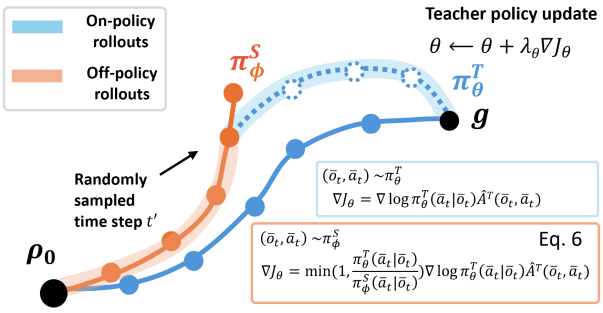

Fig. 4. Teacher policy (π_θ^T) is refined using its on-policy rollouts (shaded blue) and off-policy rollouts (shaded orange) from the student (π_ϕ^S), with policy gradients corrected via importance sampling.

To stably update the teacher from these hybrid trajectories, we employ V-trace clipping from IMPALA [9], which provides stable convergence through bounded off-policy corrections. This setup allows us to estimate corrected value targets, v_t^T, from a mixture of student and teacher trajectories (line 11).

$$v_t^T = V_\psi^T(\bar{o}_t) + \sum_{j=t}^{t+n-1} \gamma^{j-t} \left(\prod_{i=t}^{j-1} c_i \right) \delta_j V \tag{4}$$

In Eq. 4, $\delta_j V = \rho_j \left(r_j + \gamma V_\psi^T(\bar{o}_{j+1}) - V_\psi^T(\bar{o}_j) \right)$ is a temporal difference error scaled by truncated importance weights, $\rho_j = \min\left(1, \frac{\pi_\theta^T(\bar{a}_j|\bar{o}_j)}{\mu(\bar{a}_j|\bar{o}_j)}\right)$ and $c_i = \min\left(1, \frac{\pi_\theta^T(\bar{a}_i|\bar{o}_i))}{\mu(\bar{a}_i|\bar{o}_i))}\right)$. Here, μ denotes the behavior policy, either π_θ^T or π_ϕ^S. Note that while we utilize the V-trace without modification, we are unaware of prior work addressing mismatched teacher-student distributions in multi-agent distillation setups.

We train the teacher's value function $V_\psi^T(\bar{o}_t)$ by minimizing the squared error between its predictions and the corrected V-trace targets in Eq. 5 (line 12), with ψ updated via gradient descent on $\mathcal{L}_\psi$ (line 13). This value target v_t^T helps stabilize training even when student trajectories deviate significantly from the teacher, which could otherwise lead to instability.

$$\mathcal{L}_\psi = \mathbb{E}_{(\bar{o}_t, \bar{a}_t) \sim \{\pi_\phi^S, \pi_\theta^T\}} \left[\left(V_\psi^T(\bar{o}_t) - v_t^T \right)^2 \right] \tag{5}$$

Similarly, the policy update integrates trajectories from both teacher and student. For student-generated samples, we apply bounded off-policy corrections using the importance weight $\rho_t = \min\left(1, \frac{\pi_\theta^T(\bar{a}_t|\bar{o}_t)}{\pi_\phi^S(\bar{a}_t|\bar{o}_t)}\right)$, while teacher-generated samples remain on-policy ($\rho_t = 1$). The advantage estimate $\hat{A}^T(\bar{o}_t, \bar{a}_t)$ is computed via the V-trace target $r_t + \gamma v_{t+1}^T - V_\psi^T(\bar{o}_t)$; this off-policy correction with augmented advantage yields a robust and stable training signal without being misled by off-policy drift. The resulting policy objective is given in Eq. 6 (line 14), and Alg. 2 updates θ via gradient ascent on J_θ (line 15).

$$J_\theta = \mathbb{E}_{(\bar{o}_t, \bar{a}_t) \sim \{\pi_\phi^S, \pi_\theta^T\}} \left[\rho_t \log \pi_\theta^T(\bar{a}_t|\bar{o}_t) \hat{A}^T(\bar{o}_t, \bar{a}_t) \right] \tag{6}$$

The objectives in Eq. 5 and Eq. 6 allow adaptive refinement of the teacher using both on-policy and off-policy data (lines 12-15). This hybrid learning scheme enhances generalization while preserving policy consistency. During this adaptive phase, the teacher's low-level policy remains fixed, as it was pretrained on a sufficiently diverse distribution of states. This design choice simplifies the optimization process and focuses adaptation on strategic goal assignment.

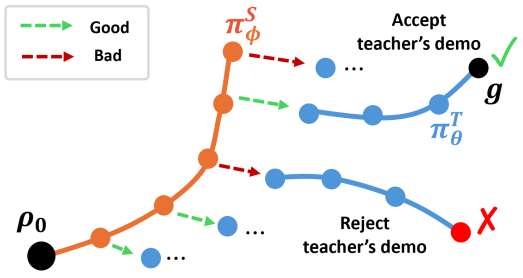

Fig. 5. Performance-based filter is applied during dataset aggregation. High-quality teacher demonstrations are accepted (green, ✓), while suboptimal ones are rejected (red, ✗).

Theoretical Support: Our use of V-trace is motivated by the theoretical properties established in IMPALA. In our formulation, the behavior policy, μ, corresponds to a mixture of student and teacher policies induced by the switching mechanism, while the target policy remains the teacher, π_θ^T. Specifically, Theorem 1 of IMPALA shows that the V-trace operator is a contraction mapping with a unique fixed point, and Theorem 2 further guarantees that the online updates converge to this fixed point almost surely under *Robbins-Munro conditions*. This follows from the clipped importance weights in Eq. 4, which bound the off-policy corrections. Since HINT applies V-trace corrections when the student policy generates trajectories for teacher training, the corrected value targets provide stable updates even when trajectories are partially generated by the student. While we do not re-derive convergence guarantees for this mixed-policy setting, the formulation follows the same conditions under which V-trace has been shown to yield stable and convergent value estimates in practice.

C. DAgger with Performance-Based Filtering

Algorithm 3: DAgger with Performance-Based Filter

Input: Teacher π_θ^T, Student π_ϕ^S, Dataset $\mathcal{D}$, Total Episodes N_{query}

for $e = 1$ **to** N_{query} **do**
  Initialize environment, set trajectory $\tau \leftarrow [\,]$;
  for $t = 1$ **to** H **do**
    Execute student action $\bar{a}_t \sim \pi_\phi^S(\cdot|\bar{o}_t)$; // Student explores
    Query teacher action $\bar{a}_t^* \sim \pi_\theta^T(\cdot|\bar{o}_t)$; // Teacher guides
    Simulate teacher policy π_θ^T from $(\bar{o}_t, \bar{a}_t^*)$ to terminal state $\bar{o}_H$;
    if terminal state $\bar{o}_H$ satisfies success condition **then**
      Append $(\bar{o}_t, \bar{a}_t^*)$ to trajectory τ ; // Performance-based filter
  Add τ to dataset $\mathcal{D}$; // Aggregate dataset

DAgger (Dataset Aggregation) [29] addresses distributional shift by iteratively collecting expert-labeled actions at states visited by the student. The student is then trained on this aggregated dataset, which better matches the test-time distribution. However, DAgger usually assumes access to a reliable

teacher. In practice, especially in open-ended environments (Fig. 3), our hierarchical teacher policy may face unfamiliar or unseen states during interactive queries, leading to inconsistent demonstrations and compounding observation mismatches between the centralized teacher and decentralized students.

To address this challenge, we propose a performance-based filtering mechanism that operates during data aggregation to enhance the quality of expert actions. The overall procedure is described in Alg. 3. As in Fig. 5, when the student queries the teacher at each timestep (line 5), we simulate the teacher's trajectory from that point to the end of the episode (line 6). Suppose the resulting terminal state meets the task-specific success criterion (line 7). In that case, we consider the initially queried action as a proxy for a good action (green) and include it in the dataset (lines 8 and 9). Otherwise, the expert action is discarded. While the teacher provides a heuristic proxy for optimal behavior, our mechanism adds a second layer of evaluation to assess the trustworthiness of this proxy by simulating its long-term consequences.

This domain-agnostic filtering approach makes the data aggregation more robust to noisy or suboptimal demonstrations. By selectively incorporating only expert actions empirically linked to successful outcomes, the student policy is trained on higher-quality data. Also, to stabilize training and support the evolving nature of the teacher policy, we retain a fixed number of initial demonstrations and periodically sample recent interactions. While this strategy is similar to experience replay with curriculum bias, our method uniquely blends static expert supervision with adaptive feedback, providing both reliable foundations and fresh guidance that aligns with the student's current capabilities.

VI. Results and Discussion

In this section, we evaluate the performance of HINT by benchmarking against baselines (Sec. VI-B) and validating the contribution of each submodule via ablation studies (Sec. VI-C). We additionally present a physical robot demonstration (Sec. VI-D). Detailed training configurations, hyperparameters, complete benchmark results, and ablation studies are provided in Appendices D, E, and F. Code is available in the supplementary material for reproducibility.

A. Environments

We selected two multi-agent domains that are highly dynamic, partially observable, and heterogeneous. These domains represent real-world challenges requiring timely and effective coordination, making them ideal for testing HINT. Performance is measured using two key metrics: success rate, defined as the proportion of episodes completed successfully, and average steps taken, indicating the number of steps required to complete each task. For further details about the environments, refer to Appendix A

MARINE [13]: A maritime logistics environment where routing agents navigate dynamic ocean conditions, while logistic agents provide mid-sea refueling. The environment uses WaveWatch III forecast data [37], introducing weather-induced uncertainty into planning and coordination. Routing agents must reach destinations before fuel depletion, requiring timely rendezvous with logistic agents. Three difficulty levels are used: easy (5×5, 2*R*/1*L*), medium (10×10, 3*R*/2*L*), and hard (20×20, 6*R*/4*L*), where *R* and *L* denote routing and logistic agents, respectively.

FireCommander (FC) [30]: A grid-based wildfire environment inspired by FARSITE [10], a widely used simulator modeling spatial and temporal fire behavior via differential equations. In FC, action agents are responsible for extinguishing fires but do not have perception capabilities, while perception agents monitor fire spread but cannot put out fires. This division of roles encourages collaboration in a stochastic environment shaped by dynamic fire propagation. The environment features three difficulty levels: easy (5×5, 2*P*/1*A*), medium (10×10, 3*P*/2*A*), and hard (21×21, 6*P*/4*A*), where *P* and *A* denote perception and action agents, respectively.

B. Benchmark Test

1) Online CTDE Baselines: We evaluate HINT against online CTDE baselines, including non-communicative (MAPPO [46], HAPPO [19]) and communicative methods (TarMAC [7], IC3Net [34], CommNet [36], HetNet [32]). For fairness, all baselines are trained using the same total budget as HINT, including the timesteps used to pre-train our teacher policy. Detailed training settings are provided in Tables VI and VII in Appendix D. To further validate the effectiveness of interactive distillation, we compare HINT with RL methods that employ warm-starting. Warm-starting initializes the RL baselines via behavior cloning (BC) from our teacher policy, rather than training from scratch. Based on Table I, HAPPO and HetNet are selected as strong baselines for MARINE and FireCommander, respectively; we warm-start these methods with 25%/50% of total timesteps allocated to BC.

Table I shows that our proposed method consistently outperforms other CTDE baselines in both MARINE and FC across medium and hard tasks. Specifically, a high success rate coupled with fewer steps taken indicates that our approach yields agents that are both reliable and efficient, even in challenging scenarios. Moreover, HINT achieves notable improvements over warm-started HAPPO in the MARINE-Hard task. We suspect that MARINE's terminal condition—fuel depletion of routing agents—hinders exploration and leads agents to become easily trapped in local minima, even with warm-starting. In the FC-Hard task, HINT outperforms warm-started HetNet by 20-45%, while also exhibiting a much smaller standard deviation, reflecting a more stable training process.

These improvements can be attributed to the structured communication in our framework. However, compared to HetNet and other warm-started variants, which employ the same student policy as HINT or are initialized with the teacher's demonstrations, we can observe how stable training signals and adaptive teacher's guidance contribute to final performance. This informative guidance comes from our centralized hierarchical teacher and an adaptive mechanism guided by pseudo off-policy RL and performance-filtered data. Together,

TABLE I

QUANTITATIVE RESULTS FOR MARINE AND FIRECOMMANDER (FC) IN THE ONLINE CTDE BENCHMARK, REPORTED AS THE MEAN (± STANDARD DEVIATION) ACROSS THREE RANDOM SEEDS. HERE, *WS* DENOTES WARM STARTING RL WITH OUR TEACHER POLICY.

Method	MARINE-Medium (10x10, 5 agents)		MARINE-Hard (20x20, 10 agents)		FC-Medium (10x10, 5 agents)		FC-Hard (21x21, 10 agents)	
	Success Rate (%) ↑	Steps Taken ↓	Success Rate (%) ↑	Steps Taken ↓	Success Rate (%) ↑	Steps Taken ↓	Success Rate (%) ↑	Steps Taken ↓
MAPPO	40.67 ± 51.39	69.23 ± 37.31	0.67 ± 1.15	198.88 ± 1.94	6.00 ± 3.46	96.33 ± 1.19	4.00 ± 0.00	202.89 ± 0.86
TarMAC	94.00 ± 10.39	**25.84 ± 14.77**	0.00 ± 0.00	200.0 ± 0.00	5.33 ± 5.03	96.89 ± 3.97	2.67 ± 3.06	207.51 ± 2.48
IC3Net	88.00 ± 10.58	34.97 ± 17.99	0.00 ± 0.00	200.0 ± 0.00	6.67 ± 6.11	97.07 ± 2.79	2.67 ± 3.06	207.08 ± 2.68
CommNet	0.00 ± 0.00	100.00 ± 0.00	0.00 ± 0.00	200.0 ± 0.00	3.33 ± 3.06	97.81 ± 1.92	1.33 ± 2.31	207.96 ± 1.77
HAPPO	98.67 ± 2.31	27.81 ± 0.46	0.00 ± 0.00	200.0 ± 0.00	0.67 ± 1.15	99.35 ± 1.13	0.67 ± 1.15	208.61 ± 2.40
→ 25% *WS*	-	-	24.00 ± 3.46	169.94 ± 9.25	-	-	-	-
→ 50% *WS*	-	-	38.00 ± 6.93	154.73 ± 8.30	-	-	-	-
HetNet	61.33 ± 53.27	60.33 ± 34.64	0.00 ± 0.00	200.0 ± 0.00	82.00 ± 14.00	49.43 ± 16.00	19.33 ± 7.02	177.93 ± 11.91
→ 25% *WS*	-	-	-	-	-	-	35.33 ± 15.28	152.35 ± 18.78
→ 50% *WS*	-	-	-	-	-	-	42.67 ± 38.28	139.01 ± 62.69
HINT (Ours)	**98.67 ± 1.15**	26.49 ± 4.92	**53.33 ± 9.02**	**143.28 ± 4.47**	**84.00 ± 2.00**	**47.95 ± 3.73**	**51.33 ± 5.03**	**136.40 ± 3.86**

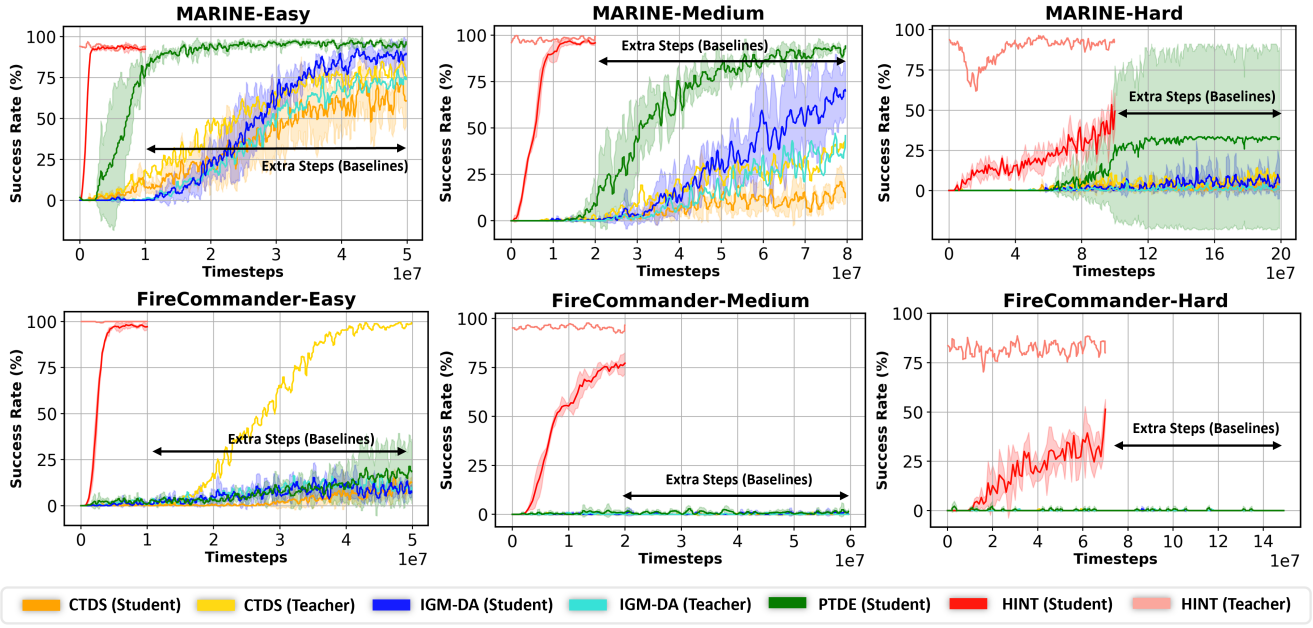

Fig. 6. Learning curves for MARINE and FireCommander in the online KD benchmark, reported as the mean (± standard deviation) across three random seeds and three difficulty settings. HINT consistently outperforms all baselines in both domains.

these components enable robust policy learning under partially observable environments and OOD states, while also reducing observation mismatches between teacher and students. For the full comparisons and training curves, find Appendix E.

2) Online KD Baselines: Here, we evaluate HINT against three online knowledge distillation (KD) methods:

1. CTDS [47] - It employs a centralized teacher that has access to global information and trains decentralized students through imitation learning.
2. IGM-DA [15] - Similar to CTDS, but the centralized teacher is trained on data collected by students, augmented with global information, and distills knowledge back to students via DAgger.
3. PTDE [4] - Global Information Personalization (GIP) module is introduced as a teacher network to distill agent-personalized global knowledge into the agent's local information.

All KD baselines are designed for value-based MARL. Hence, we adopt QMiX [28] as the base policy for both teacher and student, as it achieved the best performance in the original KD studies. To mitigate convergence issues arising from the co-evolution of teacher and student networks, we train the KD baselines for more timesteps than the CTDE baselines.

Fig. 6 shows that HINT outperforms other KD baselines in both MARINE and FC environments. In FC, since QMiX lacks a communication module, KD baselines struggle to achieve good performance. Nonetheless, the poor performance of CTDS and IGM-DA teachers, even with access to global information, underscores the robustness of our hierarchical teacher in more challenging settings. (Since the teacher in PTDE functions as an auxiliary module rather than a policy, its performance is not shown in Fig. 6.) In MARINE, only PTDE achieves performance comparable to HINT in easy and medium tasks, but HINT surpasses PTDE by 60% on hard tasks. Similar to Sec. VI-B1, HINT's success is attributed to its structured communication, but the results support the importance of an adaptive hierarchical teacher in our entire pipeline. For complete quantitative results, find Appendix E.

C. Ablation Study

1) Effect of Key Modules: To evaluate the impact of each key component in HINT, we conduct an ablation study comparing the full model against three variants: one without pseudo off-policy RL, one without performance-based filtering, and one with both components removed, effectively reducing the method to standard DAgger. Throughout training, both the teacher and student policies are evaluated based on success rate and average steps taken. For the teacher, we also report the suboptimal demonstration rate, which serves as a proxy for guidance quality by measuring whether the trajectory generated by the teacher, after responding to a student query, leads to a successful outcome from the query point onward.

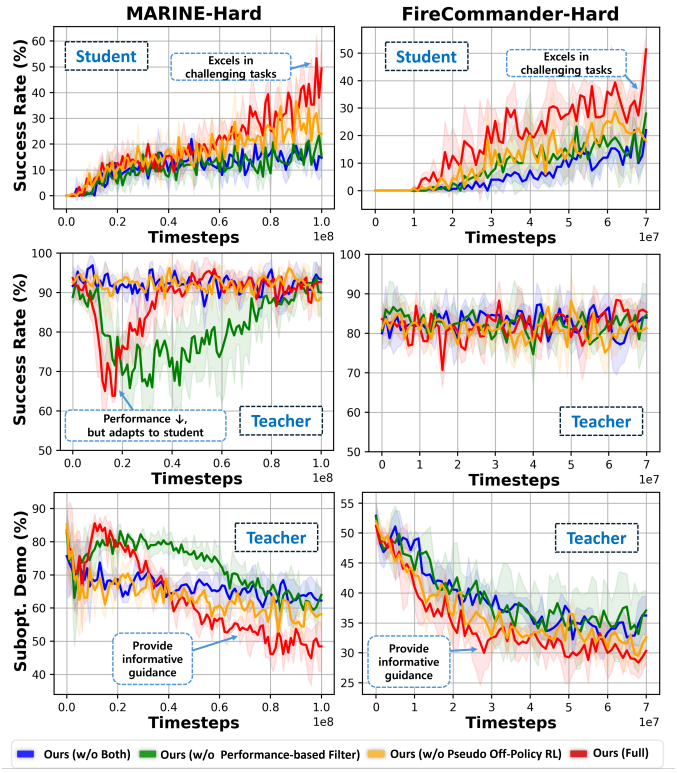

Fig. 7. Ablation of key modules on MARINE-Hard and FC-Hard. Each row corresponds to the student's success rate, teacher's success rate, and teacher's suboptimality demo rate, reported as the mean (± standard deviation) over three random seeds.

Fig. 7 shows that the full model consistently outperforms all ablated variants, achieving notable gains on both MARINE-Hard and FC-Hard. In particular, we observe that our full model exhibits a lower suboptimal demo rate than other variants, resulting in higher student performance. In MARINE-Hard, the teacher's initial performance may temporarily decline as it adapts to diverse student-induced state distributions that differ from its pre-training data; however, this adaptation yields fewer suboptimal demonstrations and more informative guidance. Together, these findings confirm that HINT's adaptability and filtering mechanisms are essential for robust, high-quality policy learning. We further validate

TABLE II
HIERARCHICAL VS. FLAT CENTRALIZED TEACHER PERFORMANCE ACROSS MARINE AND FC

Method	MARINE-Medium	MARINE-Hard	FC-Medium	FC-Hard
HiTMAC	100.00	98.00	100.00	96.00
Sable	100.00	0.00	12.50	9.38
MAT	100.00	0.00	12.50	6.25

TABLE III
ABLATION RESULT OF STUDENT STRUCTURE ON MARINE-MEDIUM AND FC-MEDIUM, REPORTED AS THE MEAN (± STANDARD DEVIATION) ACROSS THREE RANDOM SEEDS.

Method	MARINE-Medium		FC-Medium	
	Success Rate (%) ↑	Steps Taken ↓	Success Rate (%) ↑	Steps Taken ↓
Ours (LSTM)	92.67 ± 3.06	36.28 ± 6.31	0.00 ± 0.00	100.00 ± 0.00
Ours (LSTM + GNN)	98.00 ± 2.00	27.80 ± 0.97	66.00 ± 8.72	56.91 ± 6.17
Ours (HetNet)	**98.67 ± 1.15**	**26.49 ± 4.92**	**84.00 ± 2.00**	**47.95 ± 3.73**

HINT's robustness via sensitivity analyses on key hyperparameters in FC-Medium, demonstrating stable performance across reasonable parameter ranges (see Appendix F for full ablation and sensitivity analysis results).

2) Hierarchical vs. Flat Centralized Teachers: To justify our choice of a hierarchical centralized teacher, we compare HiTMAC with flat centralized models, including MAT [42] and Sable [24]. This comparison allows us to isolate the effect of hierarchical structure from centralized training alone. For a fair comparison, both MAT and Sable are trained using HiTMAC's learning rate and a 3× larger training budget. In Table II, even strong baselines struggle to find high-performing solutions in the hardest settings, whereas HiTMAC consistently achieves the best performance. These results suggest that hierarchical organization provides more effective teacher guidance than flat centralized policies in our settings.

3) Student Structure: To better understand the contributions of different architectural components in our student model, we conducted ablation studies on MARINE-Medium and FC-Medium tasks. Our proposed student model, HetNet, integrates two key modules: LSTM to address partial observability, and HetGAT mechanism for structured inter-agent communication. We compared HetNet against two ablated variants: (i) **LSTM-Only**, a student model equipped with an LSTM but no inter-agent communication; and (ii) **LSTM+GNN**, a student model with an LSTM and a GNN-based communication module [20].

As shown in Table III, HetNet consistently outperforms both ablated baselines, with the largest gains in FC-Medium—improving the success rate by 84% compared to LSTM-only, and by 18% compared to LSTM+GNN. In this setting, structured communication is essential due to the limited sensing capabilities of action agents. The performance gap between the LSTM+GNN and HetNet highlights the value of heterogeneous attention in our architecture.

D. Real-World Robot Demonstration

We demonstrate HINT on the Robotarium [43], a remote swarm robotics research platform. Given the testbed's scale and the physical limitations of the robots, we evaluate easy

versions of the MARINE and FC tasks by deploying the learned student policy across randomly generated configurations (see Fig. 8). At deployment, student policies operate fully decentralized—using only local observations and received messages without centralized state access. Students select task-relevant goals, while a goal-based planner handles low-level execution and collision avoidance via barrier functions. Further details are provided in the supplementary video.

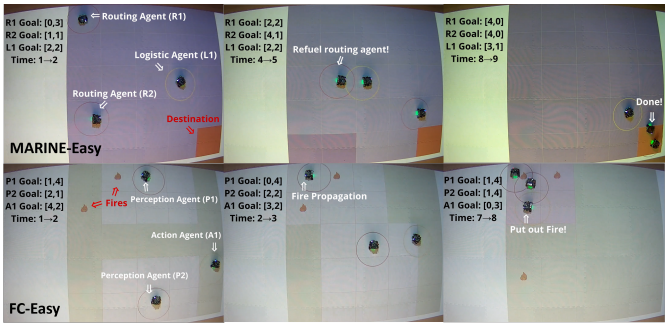

Fig. 8. Robot demonstration: MARINE-Easy (top) and FC-Easy (bottom)

VII. Limitations and Future Works

Since our teacher policy follows the CTCE paradigm, its scalability is fundamentally limited. Nevertheless, we choose a fully centralized teacher due to its high performance and try to improve scalability via hierarchical RL. In practice, HINT scales to 20×20 grid environments with up to 10 agents, which may appear modest in scale but remains challenging when jointly learning a communication protocol.

Another limitation is the computational overhead introduced by our additional modules, particularly the forward-simulation-based filtering step. We use this filtering strategy instead of value-based weighting because value estimates can be unreliable early in training, especially for out-of-distribution states explored by the student. Although this step increases computational cost, our multi-threaded pipeline keeps the overall runtime comparable to competitive baselines (e.g., HINT: 146 hours vs. PTDE: 161 hours on MARINE-Hard; HINT: 90 hours vs. HetNet: 179 hours on FC-Hard). Further details are provided in Appendix G. A promising future direction is to combine forward simulation with value-based weighting, for example by bootstrapping value estimates after n-step forward simulation.

Finally, HINT relies on human expertise to define appropriate hierarchical structures. While this design provides practical flexibility and is easy to implement, fully end-to-end training could yield more adaptive behaviors and reveal emergent capabilities. Future work could explore automating hierarchy construction through macro-action-based skill discovery and developing credit assignment mechanisms that operate asynchronously across agents based on the extracted skills.

VIII. Conclusion

We introduced HINT, a robust multi-agent learning framework leveraging hierarchical teacher-based adaptive knowledge distillation. By integrating pseudo off-policy RL and performance-based filtering, our approach enables adaptive and robust knowledge transfer in dynamic, complex cooperative tasks. Empirical evaluations demonstrate that HINT consistently achieves robust performance even as environment complexity and team size increase. Our findings highlight the critical importance of addressing teacher policy suboptimality, thereby paving the way for future research on adaptive and robust multi-agent systems.

Acknowledgments

This work was supported by the Naval Research Laboratory under N00173-25-1-0050.

References

[1] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

[2] Batuhan Altundas, Shengkang Chen, Shivika Singh, Shivangi Deo, Minwoo Cho, and Matthew Craig Gombolay. Heterogeneous graph transformers for simultaneous mobile multi-robot task allocation and scheduling under temporal constraints. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

[3] Gerald G Brown, Walter C DeGrange, Wilson L Price, and Anton A Rowe. Scheduling combat logistics force replenishments at sea for the us navy. *Naval Research Logistics*, 64(8):677–693, 2017.

[4] Yiqun Chen, Hangyu Mao, Jiaxin Mao, Shiguang Wu, Tianle Zhang, Bin Zhang, Wei Yang, and Hongxing Chang. Ptde: Personalized training with distilled execution for multi-agent reinforcement learning. *arXiv preprint arXiv:2210.08872*, 2022.

[5] John R Cooper. Optimal multi-agent search and rescue using potential field theory. In *AIAA Scitech 2020 forum*, page 0879, 2020.

[6] Mehul Damani, Zhiyao Luo, Emerson Wenzel, and Guillaume Sartoretti. Primal _2: Pathfinding via reinforcement and imitation multi-agent learning-lifelong. *IEEE Robotics and Automation Letters*, 6(2):2666–2673, 2021.

[7] Abhishek Das, Théophile Gervet, Joshua Romoff, Dhruv Batra, Devi Parikh, Mike Rabbat, and Joelle Pineau. Tarmac: Targeted multi-agent communication. In *International Conference on machine learning*, pages 1538–1546. PMLR, 2019.

[8] Jared R Deiter. Statistical sensitivity analysis of the replenishment at sea planner. Master's thesis, Naval Postgraduate School, 2022.

[9] Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Vlad Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, et al. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. In *International conference on machine learning*, pages 1407–1416. PMLR, 2018.

[10] Mark Arnold Finney. *FARSITE, Fire Area Simulator–model development and evaluation*. Number 4. US De-

partment of Agriculture, Forest Service, Rocky Mountain Research Station, 1998.

[11] Jakob Foerster, Ioannis Alexandros Assael, Nando De Freitas, and Shimon Whiteson. Learning to communicate with deep multi-agent reinforcement learning. *Advances in neural information processing systems*, 29, 2016.

[12] Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Proc. AAAI conference on artificial intelligence*, volume 32, 2018.

[13] Byeolyi Han, Minwoo Cho, Letian Chen, Rohan Paleja, Zixuan Wu, Sean Ye, Esmaeil Seraj, David Sidoti, and Matthew Gombolay. Learning multi-agent coordination for replenishment at sea. *IEEE Robotics and Automation Letters*, 2024.

[14] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[15] Yitian Hong, Yaochu Jin, and Yang Tang. Rethinking individual global max in cooperative multi-agent reinforcement learning. *Advances in neural information processing systems*, 35:32438–32449, 2022.

[16] Shariq Iqbal and Fei Sha. Actor-attention-critic for multi-agent reinforcement learning. In *International conference on machine learning*, pages 2961–2970. PMLR, 2019.

[17] Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. An empirical exploration of recurrent network architectures. In *International conference on machine learning*, pages 2342–2350. PMLR, 2015.

[18] Aleksandar Krnjaic, Raul D Steleac, Jonathan D Thomas, Georgios Papoudakis, Lukas Schäfer, Andrew Wing Keung To, Kuan-Ho Lao, Murat Cubuktepe, Matthew Haley, Peter Börsting, et al. Scalable multi-agent reinforcement learning for warehouse logistics with robotic and human co-workers. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 677–684. IEEE, 2024.

[19] Jakub Grudzien Kuba, Ruiqing Chen, Muning Wen, Ying Wen, Fanglei Sun, Jun Wang, and Yaodong Yang. Trust region policy optimisation in multi-agent reinforcement learning. In *International Conference on Learning Representations*, 2022.

[20] Qingbiao Li, Fernando Gama, Alejandro Ribeiro, and Amanda Prorok. Graph neural networks for decentralized multi-robot path planning. In *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 11785–11792. IEEE, 2020.

[21] Yueheng Li, Guangming Xie, and Zongqing Lu. Difference advantage estimation for multi-agent policy gradients. In *International Conference on Machine Learning*, pages 13066–13085. PMLR, 2022.

[22] Alex Tong Lin, Mark Debord, Katia Estabridis, Gary Hewer, Guido Montufar, and Stanley Osher. Decentralized multi-agents by imitation of a centralized controller. In *Mathematical and Scientific Machine Learning*, pages 619–651. PMLR, 2022.

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

[24] Omayma Mahjoub, Sasha Abramowitz, Ruan de Kock, Wiem Khlifi, Simon du Toit, Jemma Daniel, Louay Ben Nessir, Louise Beyers, Claude Formanek, Liam Clark, et al. Sable: a performant, efficient and scalable sequence model for marl. *arXiv preprint arXiv:2410.01706*, 2024.

[25] Linghui Meng, Muning Wen, Chenyang Le, Xiyun Li, Dengpeng Xing, Weinan Zhang, Ying Wen, Haifeng Zhang, Jun Wang, Yaodong Yang, et al. Offline pre-trained multi-agent decision transformer. *Machine Intelligence Research*, 20(2):233–248, 2023.

[26] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937. PmLR, 2016.

[27] Aowabin Rahman, Arnab Bhattacharya, Thiagarajan Ramachandran, Sayak Mukherjee, Himanshu Sharma, Ted Fujimoto, and Samrat Chatterjee. Adversar: Adversarial search and rescue via multi-agent reinforcement learning. In *2022 IEEE International Symposium on Technologies for Homeland Security (HST)*, pages 1–7. IEEE, 2022.

[28] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder de Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Monotonic value function factorisation for deep multi-agent reinforcement learning. *Journal of Machine Learning Research*, 21(178):1–51, 2020.

[29] Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635. JMLR Workshop and Conference Proceedings, 2011.

[30] Esmaeil Seraj, Zheyuan Wang, Rohan Paleja, Daniel Martin, Matthew Sklar, Anirudh Patel, and Matthew Gombolay. Learning efficient diverse communication for cooperative heterogeneous teaming. In *Proc. International Conference on Autonomous Agents and Multiagent Systems*, page 1173–1182, 2022.

[31] Esmaeil Seraj, Jerry Xiong, Mariah Schrum, and Matthew Gombolay. Mixed-initiative multiagent apprenticeship learning for human training of robot teams. *Advances in Neural Information Processing Systems*, 36: 35426–35440, 2023.

[32] Esmaeil Seraj, Rohan Paleja, Luis Pimentel, Kin Man Lee, Zheyuan Wang, Daniel Martin, Matthew Sklar, John Zhang, Zahi Kakish, and Matthew Gombolay. Heterogeneous policy networks for composite robot team communication and coordination. *IEEE Transactions on Robotics*, 2024.

[33] Yi Shen, Benjamin McClosky, Joseph W Durham, and Michael M Zavlanos. Multi-agent reinforcement learning for resource allocation in large-scale robotic warehouse sortation centers. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, pages 7137–7143. IEEE, 2023.

[34] Amanpreet Singh, Tushar Jain, and Sainbayar Sukhbaatar. Learning when to communicate at scale in multiagent cooperative and competitive tasks. In *Proc. International Conference on Learning Representations*, 2018.

[35] Kyunghwan Son, Daewoo Kim, Wan Ju Kang, David Earl Hostallero, and Yung Yi. Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In *International conference on machine learning*, pages 5887–5896. PMLR, 2019.

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

[37] Hendrik L. Tolman, Bhavani Balasubramaniyan, Lawrence D. Burroughs, Dmitry V. Chalikov, Yung Y. Chao, Hsuan S. Chen, and Vera M. Gerald. Development and implementation of wind-generated ocean surface wave modelsat ncep. *Weather and Forecasting*, 17(2): 311 – 333, 2002.

[38] Wei-Cheng Tseng, Tsun-Hsuan Johnson Wang, Yen-Chen Lin, and Phillip Isola. Offline multi-agent reinforcement learning with knowledge distillation. *Advances in Neural Information Processing Systems*, 35:226–237, 2022.

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

[40] Jianhong Wang, Yuan Zhang, Tae-Kyun Kim, and Yunjie Gu. Shapley q-value: A local reward approach to solve global reward games. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7285–7292, 2020.

[41] Stanisław Węglarczyk. Kernel density estimation and its application. In *ITM web of conferences*, volume 23, page 00037. EDP Sciences, 2018.

[42] Muning Wen, Jakub Kuba, Runji Lin, Weinan Zhang, Ying Wen, Jun Wang, and Yaodong Yang. Multi-agent reinforcement learning is a sequence modeling problem. *Advances in Neural Information Processing Systems*, 35: 16509–16521, 2022.

[43] Sean Wilson, Paul Glotfelter, Li Wang, Siddharth Mayya, Gennaro Notomista, Mark Mote, and Magnus Egerstedt. The robotarium: Globally impactful opportunities, challenges, and lessons learned in remote-access, distributed control of multirobot systems. *IEEE Control Systems Magazine*, 40(1):26–44, 2020.

[44] Jing Xu, Fangwei Zhong, and Yizhou Wang. Learning multi-agent coordination for enhancing target coverage in directional sensor networks. *Advances in Neural Information Processing Systems*, 33:10053–10064, 2020.

[45] Pei Xu and Ioannis Karamouzas. Human-inspired multi-agent navigation using knowledge distillation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8105–8112. IEEE, 2021.

[46] Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. The surprising effectiveness of PPO in cooperative multi-agent games. In *Advances in Neural Information Processing Systems, Track on Datasets and Benchmarks*, 2022.

[47] Jian Zhao, Xunhan Hu, Mingyu Yang, Wengang Zhou, Jiangcheng Zhu, and Houqiang Li. Ctds: Centralized teacher with decentralized student for multiagent reinforcement learning. *IEEE Transactions on Games*, 16 (1):140–150, 2022.

[48] Yang Zhou, Siying Wang, Wenyu Chen, Ruoning Zhang, Zhitong Zhao, and Zixuan Zhang. Double distillation network for multi-agent reinforcement learning. *arXiv preprint arXiv:2502.03125*, 2025.