Mixed-Initiative Multiagent Apprenticeship Learning for Human Training of Robot Teams

Esmaeil Seraj
Georgia Institute of Technology
eseraj3@gatech.edu

Jerry Xiong
Georgia Institute of Technology
jxiong60@gatech.edu

Mariah Schrum
University of California Berkeley
mariahschrum@berkeley.edu

Matthew Gombolay
Georgia Institute of Technology
matthew.gombolay@cc.gatech.edu

Abstract

Extending recent advances in Learning from Demonstration (LfD) frameworks to multi-robot settings poses critical challenges such as environment non-stationarity due to partial observability which is detrimental to the applicability of existing methods. Although prior work has shown that enabling communication among agents of a robot team can alleviate such issues, creating inter-agent communication under existing Multi-Agent LfD (MA-LfD) frameworks requires the human expert to provide demonstrations for both environment actions and communication actions, which necessitates an efficient communication strategy on a known message space. To address this problem, we propose Mixed-Initiative Multi-Agent Apprenticeship Learning (MixTURE). MixTURE enables robot teams to learn from a human expert-generated data a preferred policy to accomplish a collaborative task, while simultaneously learning emergent inter-agent communication to enhance team coordination. The key ingredient to MixTURE's success is automatically learning a communication policy, enhanced by a mutual-information maximizing reverse model that rationalizes the underlying expert demonstrations without the need for human generated data or an auxiliary reward function. MixTURE outperforms a variety of relevant baselines on diverse data generated by human experts in complex heterogeneous domains. MixTURE is the first MA-LfD framework to enable learning multi-robot collaborative policies directly from real human data, resulting in 44% less human workload, and 46% higher usability score.

1 Introduction

In recent years, Multi-Agent Reinforcement Learning (MARL) has been predominantly used by researchers to optimize a reward signal and for learning multi-robot tasks. Nevertheless, RL generally suffers from key limitations such as difficulty in designing an expressive and suitable reward function for complex tasks [60, 2] which can lead to undesirable robot behavior [14, 60, 64], high sample complexity [29], and safety concerns due to direct robot-environment interactions for optimizing the policy [64]. These problems are further exacerbated in multi-robot scenarios where inter-robot interactions and environment dynamics can be more complex and task descriptions and objectives more ambiguous [60, 53, 52]. As such, accurate models of human strategies and behaviors achieved via imitation methods are increasingly important for safely and effectively deploying autonomous systems and aligning values motivating robot behaviors with human values [47, 22, 12].

Learning from Demonstration (LfD) attempts to learn the correct behavior (policy) from a set of expert-generated demonstrations rather than a reward function, which can result in lower sample

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

https://doi.org/10.52202/075280-1539

complexity and a learned policy that more closely reflects the human's preferred strategy [48, 47]. Unfortunately, extending existing single-agent LfD paradigms such as Behavioral Cloning (BC) [66] or Inverse Reinforcement Learning (IRL) [1] to multi-robot settings poses several challenges such as environment non-stationarity and existence of multiple equilibrium solutions (an agent's optimal policy depends on other agents' policies) [37, 27, 60]. One can adopt such frameworks directly in a centralized system. However, centralized systems are not scalable, are prone to single-node failure, and pose significant computation overhead, and therefore, decentralized approaches (e.g., limited-range communication and local computations) have been more desired in multi-robot systems [20].

Prior work has shown that enabling communication among agents of a robot team creates a shared mental model of joint action-spaces and therefore, allowing coordinated action decisions [38, 65, 55, 19, 51] to handle challenges such as partial observability and environment dynamicity [25, 54, 50]. Although LfD can resolve the high sample complexity and reward shaping problems in RL, the task of MA-LfD can be onerous for the humans as they have to control multiple robots and imagine and simulate an appropriate Theory-of-Mind to create a communication strategy for the robot team.

Introducing the inter-agent communication [32] under current state-of-the-art (SOTA) Multi-Agent LfD (MA-LfD) frameworks such as Multi-Agent Generative Adversarial Imitation Learning (MA-GAIL) [60] or Multi-Agent Adversarial IRL (MA-AIRL) [71] requires the human expert to provide demonstrations for both *environment actions* and *communication actions*. Such approaches assume that the human expert has access to an efficient communication strategy on a known finite message space in addition to a known strategy for taking environment actions [42]. This assumption, however, is invalid, since dynamic environments with multiple interacting agents will be too complex and unpredictable for humans to be able to develop cohesive and comprehensive inter-agent communication protocols. Even if such an efficient communication strategy exists, demonstrating both the task strategy as well as the communication strategy could potentially pose significant workload on the human expert, which in turn can affect the performance and quality of demonstrations. These issues become even more severe when the robot team is heterogeneous or of composite nature (i.e., agents with different observation- and action-spaces as well as different objectives) where agents must rely on communication to operate and fulfill their tasks correctly [54, 7, 41].

To address these challenges, we develop a distributed MA-LfD framework to efficiently incorporate a human expert's domain-knowledge of teaming strategies for collaborative robot teams and directly learn team coordination policies from expert human teachers. To this end, we propose Mixed-Initiative Multi-Agent Apprenticeship Learning (MixTURE) which enables robot teams to learn an expert's preferred strategy to act in an environment. MixTURE simultaneously learns end-to-end emergent communication for the robot team to enhance team coordination, without the need for human generated data or an auxiliary reward function. To improve the quality of the learned inter-agent communication protocol, we reduce the entropy of a generated message given joint states and actions by maximizing the Mutual Information (MI) between messages and joint states. We demonstrate through empirical evaluation and a human subject experiment that our LfD-based MixTURE outperforms RL based methods due to reward function independence and low sample complexity. Furthermore, MixTURE significantly alleviates the human demonstrators' workload and time required to provide demonstrations, increases system usability, and improves overall collaboration performance of the robot team. **Our key contributions are as follows**:

1. We propose the MixTURE framework for learning robot teaming strategies from human expert demonstrations while simultaneously learning inter-agent communication through online interactions during training, without the need for expert data or an auxiliary reward.
2. We develop an MI maximization-based emergent communication learning model which reduces the entropy of a generated message for an agent given joint state-observations.
3. We evaluate MixTURE on real, diverse human-generated data, collected in a human-subject user study, and show that, in a complex multi-agent domain with heterogeneous tasks, we are able to achieve $\sim 42\% - \sim 77\%$ higher performance and a significantly lower sample complexity. We also show that using MixTURE significantly improves workload and system usability relative to a benchmark MA-LfD framework. To best of our knowledge, this is the first work to train a MA-LfD framework on real human data.
4. We investigate the effects of demonstrating both environment actions and communication actions on a human expert's workload, demonstration quality, and system usability score. Our results show that a high-workload demonstration process in classic MA-LfD approaches

https://doi.org/10.52202/075280-1539

significantly ($p < .001$) reduces an expert's demonstration quality (measured by performance) and the system's usability score. MixTURE significantly improves these results; increasing a human's performance and experience engaging in MA-LfD.

2 Related Work

The literature for Multi-Agent LfD (MA-LfD) primarily aims to address the complexity of simultaneously training multiple agents under coordinated [43, 58, 46, 57, 68, 28] and uncoordinated tasks [6, 69, 40, 70]. In [61, 8] the MA-LfD problem is reduced into a single-agent problem by making the assumption that all agents share the same dynamics, observation spaces, and model parameters. In [34], a coordinated multi-agent IL approach is proposed which learns a latent coordination model along with the individual agent policies. In [60] the single-agent GAIL framework is extended for multi-agent scenarios along with a practical actor-critic method for multi-agent imitation. Similarly, in [71] the AIRL method was extended to the multi-agent settings. In [62] a scalable multi-agent LfD approach is proposed where a model-based heuristic method for automated swarm reorganization is leveraged to improve multi-agent task allocation problem. In [4] an expert feedback-based system is developed to address multi-agent path-finding problem. In [64] authors create an advising system to incorporate sub-optimal model-based heuristic policies to help improve MARL performance. Other prior work focused on learning human profile/behavior models for improved MA-LfD [45, 5]. More recently, Hoque et al. [26] proposed Fleet-DAgger , formalizing interactive fleet learning setting, in which multiple robots interactively query and learn from multiple human supervisors.

Nevertheless, applicability of these prior works in the collaborative multi-agent problems is considerably limited since none of these works explicitly address the inter-agent communication in dynamic and partially observable domains where agents not only need to take task-related actions, but also need to communicate and share information for coordination. Additionally, none of these prior work leverage real human-generated data for training to evaluate the approach against heterogeneity and diversity in human data. Our work addresses these limitations by eliminating the requirement for an expert to demonstrate a communication strategy. Instead, the human expert only needs to teach the robot team how to complete a task through demonstrations, and the robots will automatically learn a communication strategy that aligns with the expert's demonstrations. We also collect real human data to evaluate our method's ability to cope with variations in demonstration styles and strategies.

3 Problem Formulation: General MA-LfD with Heterogeneous Agents

We ground our problem formulation in a Markov Game (MG) [36] generalized to include partial observability and heterogeneous agents. We define a set of heterogeneous agents in a *composite* robot team (i.e., composed of different classes of robots) as agents that can have arbitrarily different state-, observation-, and action-spaces. The agents can also have different task objectives which, when enacted in coordination, enable the team to achieve a shared overarching mission. Accordingly, we define our generic MG as a 9-tuple: $\langle \mathcal{C}, \mathcal{N}, \{\mathcal{S}^{(c)}\}_{c\in\mathcal{C}}, \{\mathcal{A}^{(c)}\}_{c\in\mathcal{C}}, \{\Omega^{(c)}\}_{c\in\mathcal{C}}, \{\mathcal{O}^{(c)}\}_{c\in\mathcal{C}}, r, \mathcal{T}, \gamma\rangle$. $\mathcal{C}$ is set of all available agent classes in the composite robot team and the index $c \in \mathcal{C}$ denotes the agent class. $\mathcal{N} = \sum_{\langle c\in\mathcal{C}\rangle} N^{(c)}$ is the total number of collaborating agents where $N^{(c)}$ represents the number of agents in class c. $\{\mathcal{S}^{(c)}\}_{c\in\mathcal{C}}$ and $\{\mathcal{A}^{(c)}\}_{c\in\mathcal{C}}$ are discrete joint sets of state- and action-spaces, respectively. $\{\Omega^{(c)}\}_{c\in\mathcal{C}}$ is the joint set of observation-spaces, including class-specific observations. Agents of the same *class* have identical $\mathcal{S}$, $\mathcal{A}$, and Ω. $\gamma \in [0, 1)$ is the temporal discount factor for each unit of time and $\mathcal{T}$ is the state transition probability density function. At each timestep, t, each agent, j, of the c-th class can receive a partial observation $o_t^{c_j} \in \Omega^{(c)}$ according to some class-specific observation function $\{\mathcal{O}^{(c)}\}_{c\in\mathcal{C}} : o_t^{c_j} \sim \mathcal{O}^{(c)}(\cdot|\bar{s})$. Next, each agent, j, of class c, takes an action, $a_t^{c_j}$, forming a joint action vector $\bar{a} = \left(a_t^{1_1}, a_t^{1_2}, \cdots, a_t^{c_1}, \cdots, a_t^{c_j}\right)$. When agents take the joint action $\bar{a}$, in the joint state $\bar{s}$ and depending on the next joint-state, they receive an immediate reward, $r(\bar{s}, \bar{a}) \in \mathbb{R}$, shared by all agents regardless of classes. Each agent, j, of a class, c, achieves its own objective by sampling actions from a stochastic policy $\pi_j^{(c)}$. The objective of each agent is then to maximize the team return (expected sum of discounted rewards), i.e., $\mathbb{E}_\pi\left[\sum_{t=0}^{T} \gamma^t r_t\right]$.

35428 https://doi.org/10.52202/075280-1539

To directly learn the human's preferred strategy and resolve the reward specification problems [2] posed by RL, we leverage a demonstration dataset, D, provided by an expert, rather than the ground truth reward signal r employed in MARL. Unlike [71, 60] or [67], we do not assume multiple human experts in our MG to avoid the need for further coordination amongst the experts, which can be time consuming and expensive. D is a set of trajectories $\{\tau_j^c\}_{j=1}^{\mathcal{N}^{(c)}}$, where $\tau_j^c = \{(o_t^{c_j}, a_t^{c_j})\}_{t=1}^{T}$ is an expert trajectory collected by sampling $a_t^{c_j} \sim \pi_E(a_t^{c_j}|\bar{o}_t)$ in which π_E is the expert policy and $\bar{o}_t$ is the joint observation that the expert has access to at time t. We further assume that D contains the entire supervision to the learning algorithm (i.e., no online interactions during training). We build the MixTURE architecture on the generative adversarial training [21]. Our distributed GAIL objective underlying MixTURE is shown in Eq. 1 where $\mathcal{D}_\theta^{(c_j)}$ is a local discriminator that classifies expert and policy trajectories for agent j of a class c, and $\pi_\phi^{(c_j)}$ is the parameterized policy of agent j of a class c.

$$\mathcal{L}_{\mathcal{D}_\theta^{(c)}} = -\mathbb{E}_{\tau\sim\pi_E,(\bar{o},\bar{a})\sim\tau}\left[\log \mathcal{D}_\theta^{(c_j)}(\bar{o},\bar{a})\right] - \mathbb{E}_{\tau\sim\pi_\phi^{(c_j)},(\bar{o},\bar{a})\sim\tau}\left[\log\left(1-\mathcal{D}_\theta^{(c_j)}(\bar{o},\bar{a})\right)\right] \quad (1)$$

According to [21], under the GAIL objective in Eq. 1 and at optimality, the distribution of generated state-action pairs by $\bar{\pi}_\phi$ should match the distribution of demonstrated state-action pairs.

4 Mixed-Initiative Multi-Agent Apprenticeship Learning

Motivation and Problem Overview – Consider a generic composite team of robots including agents with heterogeneous characteristics and task objectives. Without loss of generality, consider a robot team composed of *perception-only* and *action-only* robots. Under our problem formulation in Section 3, perception robots and action robots create two separate *classes* of agents that need to collaborate on an overarching mission. For instance, in an application of wildfire fighting, robots of the perception class (e.g., quadcopters) need to search an environment for firespots, while the action robots (e.g., fire-extinguishing ground robots) who cannot sense the environment are required to extinguish the firespots found by the perception robots [54, 55, 52, 3]. Note that neither of the robot classes are capable of accomplishing the task without the other class.

To teach a collaborative policy to such a robot team, one can leverage demonstrations from a team of humans where each member is an expert. Using a team of human experts, however, poses further challenges: (1) simultaneous access to several human experts is expensive and can be time consuming, and (2) a communication strategy (e.g., natural language) among the human demonstrators is required for coordination, which can be challenging to translate to robot domain due to ambiguity, colloquialisms, and context-dependent use [30, 23]. Additionally, humans' communications might not make sense to the robot agents as humans could be unaware of all agents' full local state spaces.

Alternatively, we can leverage the demonstrations from a single human with domain-knowledge regarding the entire mission objective. For example, a trainer/coach can play a simulated game of firefighting using the aforementioned perception and action robots and provide expert demonstrations for how to efficiently distribute agents and prioritize tasks for searching the environment and putting out the fire. The challenge of using a single human expert, however, is that in this case the human would also need to demonstrate communication-actions (i.e., what information should an agent broadcast at each state) on top of the environment-actions (i.e., moving around or dousing fire). To this end, humans would have to create and maintain a Theory-of-Mind (ToM) of each agent under this increased action-space dimensionality, which significantly increases workload [15, 17].

To resolve this problem, we propose taking separate initiatives for teaching the robots in the team how to operate (environment actions, a_t, per observation, o_t) and how to communicate (communication message, z_t, per state, s_t) such that a human expert would only be required to provide environment-action demonstrations and the robot team would automatically infer on their own a suitable communication policy for the underlying expert demonstrations. We call our approach Mixed-Initiative Multi-Agent Apprenticeship Learning (MixTURE).

MixTURE Architecture – The proposed MixTURE architecture for human training of robot teams is shown in Fig. 1. At each timestep, each agent generates an embedding, representing agent's belief space, from its local observation. To handle agents' partial observability, the local observation embeddings are then passed into local recurrent policies for each agent. Each GRU policy receives the preprocessed features from the local observations as well as its own hidden state from previous timesteps. Therefore, the policy output depends only on the history of local observations and actions.

https://doi.org/10.52202/075280-1539

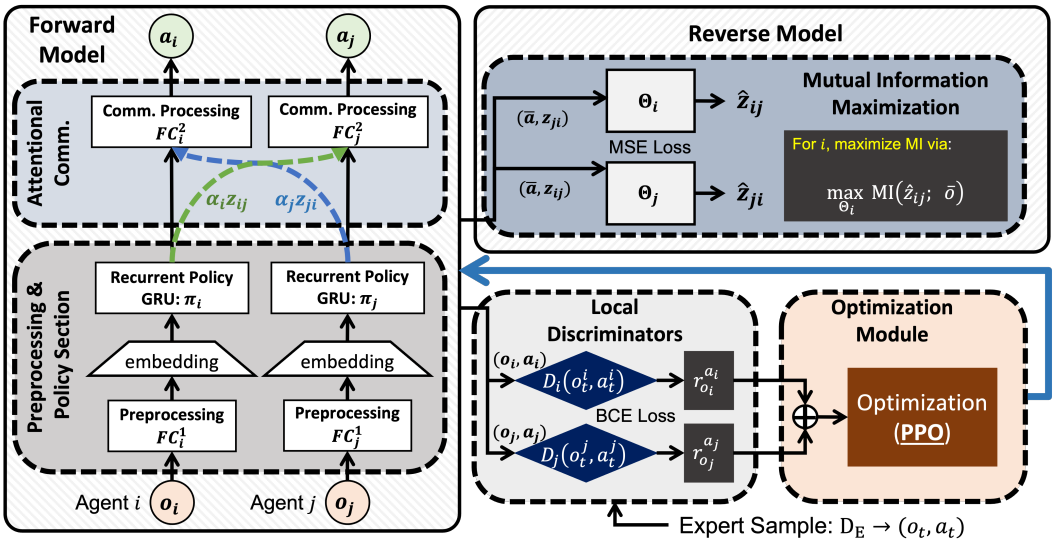

Figure 1: The proposed MixTURE architecture for a two-agent example scenario. At each timestep, each agent generates an embedding from its local observation, which is then passed into local recurrent policies for each agent. To learn from the expert demonstrations stored in the dataset D_E, we build a distributed multi-agent GAIL architecture. We enable inter-agent communication by adding an attentional communication module enhanced by a MI maximization reverse model.

To learn from the human expert demonstrations stored in the dataset D_E, as shown in Fig. 1, we build a distributed multi-agent GAIL architecture where each agent is equipped with a parametrized discriminator, $\mathcal{D}_\theta$. The discriminators are trained via a Binary Cross Entropy (BCE) loss to distinguish between state-action pair samples from the expert dataset and those generated by the generator (i.e., local policies). The output of the discriminators are treated as local rewards, which are then combined to encourage collaboration and teaming behaviour among agents [31]. To learn the agent policies through this shared reward signal, we leverage the Proximal Policy Optimization (PPO) [49].

We enable inter-agent communication by adding an attentional communication module in which each agent is equipped with a fully-connected network that processes action-embeddings generated by the recurrent policies of an agent and those generated by its local teammates (i.e., messages, z) to output an action-decision. We enable this message-passing by creating differentiable communication channels among agents. To maintain locality, these communication channels can be leveraged locally (i.e., a communication graph where edges only exist when robots are spatially within close proximity). For an agent, i, the action-embedding messages received from a teammate, j, are weighted by some learned attention coefficients, α_{ji}, to assign message importance. Therefore, the input message for agent i at time, t, can be computed as $m_t^{j\rightarrow i} = \sum_{j=1}^{\mathcal{N}_t(i)} \alpha_{ji} z_{ji}^t$ where $\mathcal{N}_t(i)$ represents the neighbors, j, of agent i at time t. Here, α_{ji} are the learned attention coefficients that are computed via $\alpha_{ji} = \text{softmax}_j(\sigma(\bar{W}_{att}[\omega \bar{h}_i \parallel \bar{m}^{j\rightarrow i}]))$. In this equation, $\bar{W}_{att}$ are the learnable weights of the attention network, $\parallel$ represents concatenation, σ is an activation function nonlinearity, and $\bar{h}$ represents the hidden states. The Softmax function is used to normalize the coefficients across all neighbors j. Such attentional communication can enhance the action-decision quality, particularly with increased number of agents or when the states may significantly vary in different parts of the environment [16]. Messages in our communication module are entirely learned via backpropagation.

Mutual Information Maximization-Based Differentiable Communication – A challenge with the communication model learned via the described end-to-end differentiable channels is that the distribution of the messages broadcasted by an agent, i, given the state-observations, $\rho(z_{ij}|\bar{o})$, can have a high variance. The desired behavior, instead, is that the agents employ a cohesive communication strategy in which an agent sends a consistent message when it observes relatively similar states.

To resolve this issue, we propose maximizing the Mutual Information (MI) between an agent's outgoing message and the joint state-observations. MI is a measure of the reduction in entropy of a probability distribution, X, given another probability distribution, Y, such that $I(X;Y) = H(X) - H(X|Y)$, where $H(X)$ denotes the entropy of X and $H(X|Y)$ is the conditional entropy

https://doi.org/10.52202/075280-1539

of X given Y [33]. In our work, by maximizing the MI between the distribution of an agent's message, $\rho(z_{ij}|\bar{o})$, and the joint observations, we reduce the entropy over messages and encourage the communication model to be more consistent. Maximizing the MI in this way encourages z_{ij} to correlate with features within the observation distribution (i.e., mode discovery) [13, 44].

Unfortunately, a direct MI Maximization (MIM) between the message distributions and joint observations as formulated above, $I(z_{ij};\bar{o}) = H(z_{ij}) - H(z_{ij}|\bar{o})$, is intractable as it requires access to the true posterior, $\rho(z_{ij}|\bar{o})$. Therefore, in keeping with prior work, we rely on the Evidence Lower Bound (ELBO) of the MI instead. As shown in prior work [13], by minimizing an MSE loss between a sample from the current message embedding and the approximate posterior, modeled as a normal distribution with constant variance, is equivalent to maximizing the likelihood of the posterior.

In practice, we build a distributed reverse model in the MixTURE architecture (shown in Fig. 1) to accommodate the mentioned MSE loss. The distributed reverse model for each agent has access to the local received messages as well as the global joint-actions taken by all agents such that the optimization results in a communication entropy-reduction mechanism at the team level. Since actions and generated messages are functions of the state-observations, each reverse model, Θ, can take in the joint actions, $\bar{a}$, and all received messages, z_{ji}, to estimate the outgoing message for agent, i, as $\hat{z}_{ij}^t = \Theta_i(\bar{a}, z_{ji}^t)$. The policy and reverse models are trained together in an end-to-end fashion to minimize the message reconstruction error as $\mathcal{L}_{\text{MIM}^{c_i}} = \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} \|\Theta_i(\bar{a}, z_{ji}) - z_{ij}\|^2$.

We note that while the addition of the MI maximization loss would make the overall loss function more complex, we believe there are further benefits that could be achieved via training the MIM reverse model, besides enhancing agent certainty in choosing actions. Particularly, we believe our MIM reverse model, when tuned well, has the potential to roughly cluster message embeddings based on observation-action pairs which can provide further useful information on the learned communication protocol for a task and further insight into interpreting the learned communication protocols.

Training and Execution – We build the MixTURE framework in a Centralized Training for Decentralized Execution (CTDE) paradigm [19] to accommodate for the global joint-action inputs to the MI maximizing reverse models during training. We note that, the MIM reverse model is only used during training and is cut during the execution, and therefore, the learned policies can be executed fully decentralized. To optimize the policies based on the reward signal generated by the discriminators, we leverage the PPO algorithm [49]. Moreover, to enhance and stabilize the training we propose combining an offline BC loss, $\mathcal{L}_{\text{BC}^{c_i}} = -\frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} \pi_i(a_t^{c_i}|o_t^{c_i}, z_{ji}^t)$, with the online GAIL loss. As shown by prior work [18], through this combination, the offline BC helps preserving ground knowledge that should be respected during training, while the online part helps with learning of new information encountered during execution. As such, putting together our distributed GAIL loss in Eq. 1, the standard clipped PPO loss [49], the offline BC loss, and the MIM loss introduced above, we present the full loss expression to train the MixTURE architecture as in Eq. 2, where ζ_π and ζ_{D_E} are minibatches of trajectory segments belonging to current policy, π, and demonstration dataset, D_E, respectively, $\mathcal{N}$ is the total number of agents and λ is a tunable scaling parameter.

$$\mathcal{L}_{\text{total}}(\zeta_\pi, \zeta_{D_E}) = \sum_{i=1}^{\mathcal{N}} (\mathcal{L}_{\mathcal{D}_\theta^{(c_i)}}(\zeta_\pi, \zeta_{D_E}) + \mathcal{L}_{\text{PPO}^{(c_i)}}(\zeta_\pi)) + \lambda_{\text{BC}} \mathcal{L}_{\text{BC}^{(c_i)}}(\zeta_{D_E}) + \lambda_{\text{MIM}} \mathcal{L}_{\text{MIM}^{(c_i)}}(\zeta_\pi) \quad (2)$$

5 Evaluation

We break the problem of evaluating our proposed architecture for teaching multi-agent coordination policies to (heterogeneous) robot teams into three research questions (RQ):

RQ1 Can MixTURE learn useful multi-agent coordination strategies from synthetic data (e.g, models of human experts / Oz-of-Wizard [63])?

RQ2 Can MixTURE learn from diverse data generated by real human experts?

RQ3 How challenging is it for a human expert to provide multi-agent demonstrations and does MixTURE alleviate these challenges (comparing workload and system usability)?

Environments – In keeping with prior work in MARL and MA-LfD [60, 54], we selected three multi-agent domains that are partially observable and require collaboration among heterogeneous agents (see Section 3). Please refer to the supplementary material for more details about the environments.

https://doi.org/10.52202/075280-1539

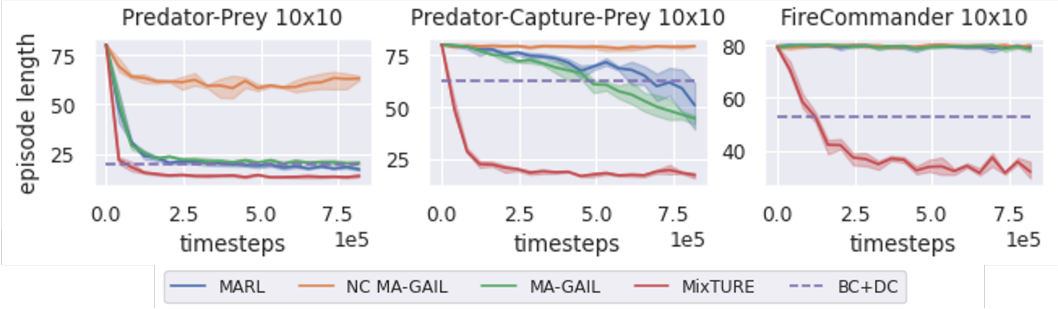

Figure 2: Evaluation results for MixTURE and the baselines in the medium case on synthetic dataset. MixTURE outperforms all baselines in both sample complexity and performance at convergence.

1. **Predator-Prey (PP)** [59]: the goal in this homogeneous (i.e., same class agents) domain is for $\mathcal{N}$ *predator* agents with limited vision to find a stationary prey and move to its location.
2. **Predator-Capture-Prey (PCP)** [54]: a new class of *capture*-agents, are introduced to the PP. In this heterogeneous domain, the goal of *predator* agents is the same, while *capture* agents must move to the prey location and capture it, without having any observation inputs.
3. **FireComander (FC)** [54, 56]: two classes of robots, *perception* and *action* robots, are required to collaborate to extinguish propagating firespots. Similar to the PCP domain, *perception* robots search the domain to find hidden firespots, and *action* robots with no observation must rely on communication to know where to put out the firespots using an extra action when on a fire. Unlike the PCP, in this complex domain firespots randomly spread over time and thus, the team must continue until all firespots are found and extinguished.

Baselines – To investigate our **RQ1**, we employ a variety of baselines, described below, all of which utilize the combined offline BC and online loss training scheme. To collect the synthetic dataset, the expert heuristic (Appendix 4) for environment-actions is a near-optimal search algorithm and for the communication, it is an anticipatory observation sharing mechanism inspired by prior work [11, 10].

- **MARL** [49]: MA-PPO optimizing both environment-action and communication policies.
- **BC+DC** [66]: MixTURE ablation trained only via offline BC loss with diff. communication.
- **NC MA-GAIL** [60]: Non-communicative ablation of the MA-GAIL [60] framework.
- **MA-GAIL** [60]: Full MA-GAIL trained on full dataset w/ demonstrated communication.

5.1 Human-Subject Experiment: Conditions and Procedure

To investigate our **RQ2** and **RQ3**, we conducted an IRB-approved human-subject user study in the FireCommander domain. For more experiment details please refer to the supplementary material.

Domain, Setup, and Procedure – Our experiments were conducted in the context of a simulated multi-robot task, leveraging the FireCommander (FC) domain under different difficulty levels and modes. Depending on the level of difficulty, there can be multiple initial firespots, hidden from the human, that propagate randomly based on a fixed wall-clock rate. The human expert was responsible to strategically move the simulated robots to find and extinguish all firespots as fast as possible. The human subject was shown a performance score at the end of each round, computed based on existing, found, and extinguished fires. After briefing, each subject began by filling in a pre-questionnaire form (demographic and prior videogame experience) followed by reading through a series of detailed game instructions. To minimize the learning effect, subjects were allowed to practice all conditions until they felt comfortable. Next, each subject played six different rounds of the game (i.e., two conditions and three difficulty levels) in a randomly selected order. The demonstration data was fully stored to be later used for training. Finally, each subject was asked to fill some post-measurement forms.

Participants – We recruited 55 participants (mean age $= 25.0 \pm 2.67$; 34.5% female). All participants were recruited through on-university-campus advertisement and were trained equally for the task.

Independent Variables and Conditions – In this study, we seek to determine if MixTURE can learn multi-agent collaborative policies from diverse human generated data without demonstrated communication and we compare its performance against the MA-GAIL [60] with expert demonstrated

https://doi.org/10.52202/075280-1539

Number of steps taken to win the game (lower is better).											
			Predator-Prey			Predator-Capture-Prey			FireCommander		
	Heu.	Diff.	easy	medium	hard	easy	medium	hard	easy	medium	hard
MARL	✓	-	15.57	17.93	47.40	23.38	56.04	79.76	78.77	79.09	80.00
BC+DC	-	✓	11.84	21.28	46.16	15.38	61.54	79.75	38.95	44.70	72.87
NC MA-GAIL	-	-	17.48	60.38	79.52	27.60	77.77	80.00	76.49	79.47	79.98
MA-GAIL	✓	-	11.68	20.39	49.08	16.43	44.84	79.86	77.46	79.47	80.00
MixTURE	-	✓	**11.16**	**13.15**	**28.73**	**13.08**	**17.27**	**36.41**	**22.31**	**34.82**	**56.49**

Table 1: Full evaluation results for all methods and all difficulty settings on synthetic dataset. *Heu.* and *Diff.* indicate a models access to heuristic or differentiable communication, respectively.

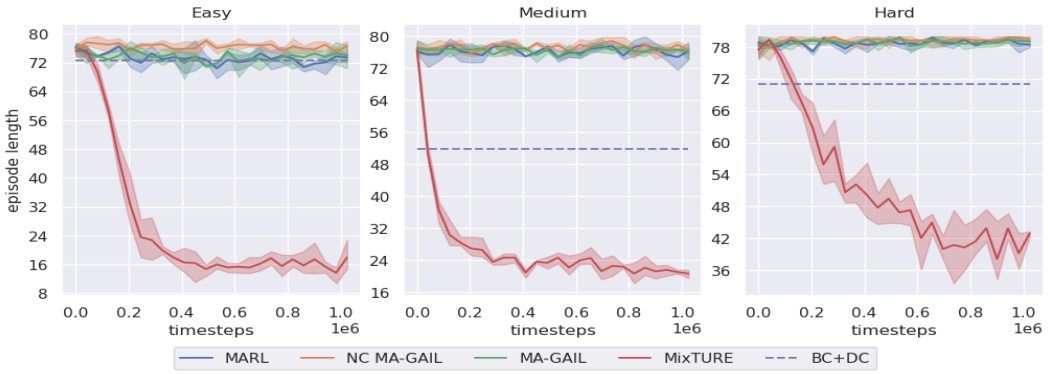

Figure 3: Evaluation results for MixTURE and MA-GAIL [60] on real human data. From left to right, the figures present test results for easy, medium, and hard scenarios with one initial firespot.

communication. We examine the performance across three levels of difficulty, i.e., easy, medium, and hard, which represent game complexity. As such, we utilize a 2×3 within-subjects experiment design varying across two abstractions: (1) only demonstrating environment actions for each robot at each time step (i.e., *noComm* condition), 2) demonstrating both environment actions and communication actions for each agent at each time step ((i.e., *withComm* condition)).

Metrics – To evaluate our **RQ2** and **RQ3**, the demonstration data collected from the human subjects for both *noComm* and *withComm* conditions are used to train MixTURE and the benchmark MA-GAIL [60], respectively. The algorithms are compared in terms of the learned policy performance (i.e., number of steps taken to win the game, where lower is better). Additionally, to address our **RQ3**, we leverage the NASA-TLX Workload Survey [24] and the System Usability Scale (SUS) [9].

5.2 Results and Discussion

RQ1: Evaluation on Synthetic Dataset – We perform our evaluations across three different difficulty levels: (1) easy (5×5 domain, 3 robots), (2) medium (10×10 domain, 6 robots), and (3) hard (20×20 domain, 10 robots). Table 1 presents the full evaluation results for collaborative policies learned by MixTURE and the baselines in terms of number of steps taken to win the game (lower is better). The test results show the mean performance values calculated over ten trials of running the best training models. As shown, MixTURE achieves significant improvement over all baselines and in all domains. Additionally, the learning curves in Fig. 2 (medium case), show significant improvement in sample complexity. We believe our model provides a strong step towards learning collaborative policies in multi-robot systems by setting a new SOTA in complex heterogeneous tasks.

We also performed several ablation studies, such as investigating scalability and effects of the MIM reverse model for message reconstruction, discriminator architecture, and the combined offline BC loss. Due to space constraints, all ablation results are available in the supplementary material.

RQ2: Training and Evaluation Results on Human-Subject Dataset – We train MixTURE and MA-GAIL [60] (and other baselines) on data collected in our human-subject user experiment to investigate our **RQ2**. The results are presented in Fig. 3. Please see the Appendix for environment

https://doi.org/10.52202/075280-1539

details. As shown, MixTURE can learn high-quality multi-robot collaboration policies from diverse human-generated demonstrations. This is while purely relying on human demonstrations [60], does not succeed in this task. We hypothesize that a key point underlying MixTURE's success despite diversity and heterogeneity in human data is that the MIM-based differentiable communication channels provide the model with ability to reason about the underlying human demonstrations and cope with trajectory distribution through automatically finding a suitable communication protocol. This result also shows that a demonstrated communication policy by a human expert is not enough to efficiently coordinate the robot team in complex domains as the human is unaware of all agents' local states, which leads to the failure of existing MA-LfD frameworks such as the MA-GAIL [60]. We hypothesize that MA-GAIL trained on demonstrated communications fails because: (1) humans are not fully aware of all agents' full local state spaces and therefore, the demonstrated communication could be severely inefficient and sub-optimal. MixTURE instead learns an efficient communication policy automatically through gradient updates; (2) The laborious task of providing both environment and communication actions for the human demonstrators significantly deteriorates the quality of demonstrated policies, which can be confirmed by our subject-study results in next section.

We believe that this strong result shows great potential for the MixTURE model to efficiently teach multi-agent coordination and collaboration policies to robot teams through human demonstrations.

RQ3: Statistical Analysis – We investigate our **RQ3** by quantifying the workload and SUS measures reported by the human subjects. We hypothesize that:

H1 Demonstrating both an environment-action and a communication-action strategy for the robot team increases the human expert's workload and decreases the system's usability score.

H2 Demonstrating both an environment-action and a communication-action strategy for the robot team negatively affects human performance and the demonstration quality.

H1: We test for normality and homoscedasticity and do not reject the null hypothesis in either case, using Shapiro-Wilk ($p > 0.32$ and $p > 0.96$) and Levene's ($p > 0.39$ and $p > 0.09$) tests for workload and SUS, respectively. For workload, we perform a paired t-test and find that using the MixTURE model w/o communication demonstration was rated statistically significantly lower than using MA-GAIL w/ demonstrated communication by the expert ($p < 0.001$) on NASA-TLX workload scale. For system usability, using a similar a paired t-test we find that using the MixTURE model w/o communication demonstration led to a statistically significantly higher SUS than using MA-GAIL w/ demonstrated communication by the expert ($p < 0.001$). As shown in Fig. 4, relaxing the need for demonstrating a communication strategy reduced the humans' workload by 44.3% and increased the systems' usability scale by 46.1% in our experiment.

H2: We examine: (1) human's performance score in the game, (2) total tasks completed (i.e., fires extinguished), and (3) average demonstration time per step. We apply a paired samples Wilcoxon test and confirm that (see Fig. 5, from left to right), relaxing the need for demonstrating a communication strategy through MixTURE leads to achieving significantly higher performance score ($p < 0.001$) by the human, better ability to scale to more complex scenarios with more tasks ($p < 0.001$), and significantly lower demonstration time per step ($p < 0.001$). Note that, the middle plot in Fig. 5 shows more tasks completed (i.e., more firespots killed) for *withComm* condition under easy and moderate scenarios, which indicates a worse human performance in this conditions since existence of more firespots means inefficiency in extinguishing the fire before it spreads too large. On the other hand, under the *noComm* condition, humans can easily scale to larger domain sizes and more initial firespots. Note that, lower tasks completed for *withComm* condition under the hard case is attributed to failing the task and losing the game (i.e., hard game cut-off when score drops below 50).

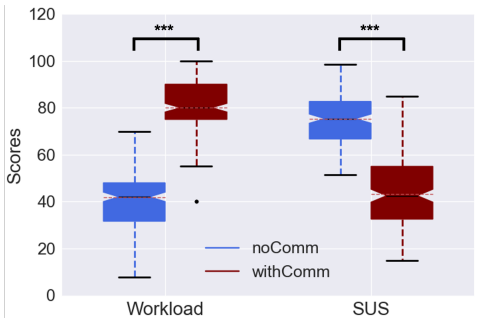

Figure 4: Workload and SUS results for **H1**.

Further Discussions and Limitations – Demonstrating multi-agent strategies can be considered a highly involved and high-workload task, which in turn can affect a human's situational awareness and

https://doi.org/10.52202/075280-1539

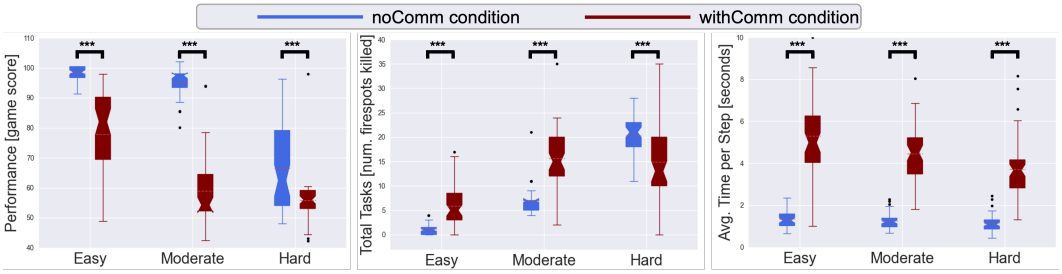

Figure 5: Objective human performance results supporting the **H2**. In summary, using the MixTURE framework leads to significantly better human performance and a lower demonstration time.

optimality of demonstrations. We posit that a human's decision-making for one agent is influenced by knowledge of other agents' observations. However, we note that, classic MA-LfD methods, such as MA-GAIL, necessitate awareness all robots' states and observations **and** they require providing both environment and communication actions at the same time. In MixTURE, we relax this assumption by tasking the human to only provide environment actions for robots one at a time. Our real human subject study demonstrates this relaxation's feasibility, enabling successful demonstrations provided by humans for large and heterogeneous teams of simulated robots. In practice, one can use existing multi-agent datasets (e.g., collaborative assembly in ROBOTURK [39, 67] or Meta's STARDATA for StarCraft II [35]) or readily create a demonstration dataset using existing tools (e.g., ROS, Gazebo). MixTURE's contribution comes in at this stage, where we can learn collaborative multi-agent policies from such datasets.

The addition of the MIM loss to Eq. 2 can complicate the tuning process. While we never observed any performance decay as a result of the MIM loss, tuning the MIM loss to achieve consistent and significant performance improvement seems to be challenging. Finally, MixTURE currently does not account for demonstration sub-optimality. An interesting future direction is then to modify the MixTURE architecture to address suboptimal human demonstrations.

Conclusion – We proposed the MixTURE model to learn multi-agent collaborative policies for a robot team, directly from human expert demonstrations. Using our method, a human expert can teach the robot team how to accomplish a task collaboratively via demonstrations and the team will automatically reason over and learn a communication strategy suitable for the underlying demonstrations. The learned communication helps the robot team to deal with the partial observability, reasoning about action-decisions to best respond to teammates' policies, and alleviate the effects of environment non-stationarity. We provided several empirical and experimental results, confirming MixTURE's strong ability to learn from expert heuristics and real diverse human generated data.

Broader Impact

Our experiments show that multi-agent learning from demonstration can be enabled for teams of cooperating heterogeneous robots via direct human teaching. This greatly assists robot collaboration through human domain knowledge and provides a strong next-step towards human-robot teaming. We note that our research could be applied for good or otherwise, particularly in an adversarial setting. Nonetheless, we believe democratizing this knowledge is for the benefit of society.

Acknowledgments

This work was supported by the Naval Research Laboratory (NRL) under grant number N00173-21-1-G009 and grant number N00014-22-1-2834. The authors also would like to thank Manisha Natarajan for being a sounding board and guiding our statistical analysis.

References

[1] Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In Proceedings of the International Conference on Machine Learning (ICML), page 1, 2004.

[2] David Abel, Will Dabney, Anna Harutyunyan, Mark K Ho, Michael Littman, Doina Precup, and Satinder Singh. On the expressivity of markov reward. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), 34, 2021.

https://doi.org/10.52202/075280-1539

[3] Ian F Akyildiz and Ismail H Kasimoglu. Wireless sensor and actor networks: research challenges. Ad hoc networks, 2(4):351–367, 2004.

[4] Ahmed Alagha, Rabeb Mizouni, Jamal Bentahar, Hadi Otrok, and Shakti Singh. Multi-agent deep reinforcement learning with demonstration cloning for target localization. IEEE Internet of Things Journal, 2023.

[5] Feryal Behbahani, Kyriacos Shiarlis, Xi Chen, Vitaly Kurin, Sudhanshu Kasewa, Ciprian Stirbu, Joao Gomes, Supratik Paul, Frans A Oliehoek, Joao Messias, et al. Learning from demonstration in the wild. In Proceedings of the International Conference on Robotics and Automation (ICRA), pages 775–781. IEEE, 2019.

[6] Sage Bergerson. Multi-agent inverse reinforcement learning: Suboptimal demonstrations and alternative solution concepts. arXiv preprint arXiv:2109.01178, 2021.

[7] Matteo Bettini, Ajay Shankar, and Amanda Prorok. Heterogeneous multi-robot reinforcement learning. arXiv preprint arXiv:2301.07137, 2023.

[8] Raunak P Bhattacharyya, Derek J Phillips, Blake Wulfe, Jeremy Morton, Alex Kuefler, and Mykel J Kochenderfer. Multi-agent imitation learning for driving simulation. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 1534–1539. IEEE, 2018.

[9] John Brooke et al. Sus-a quick and dirty usability scale. Usability evaluation in industry, 189(194):4–7, 1996.

[10] Abhizna Butchibabu. Anticipatory communication strategies for human robot team coordination. PhD thesis, Massachusetts Institute of Technology, 2016.

[11] Abhizna Butchibabu, Christopher Sparano-Huiban, Liz Sonenberg, and Julie Shah. Implicit coordination strategies for effective team communication. Human Factors, 58(4):595–610, 2016.

[12] Letian Chen, Rohan Paleja, Muyleng Ghuy, and Matthew Gombolay. Joint goal and strategy inference across heterogeneous demonstrators via reward network distillation. In Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI), pages 659–668, 2020.

[13] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), 29, 2016.

[14] Jack Clark and Dario Amodei. Faulty reward functions in the wild. Internet: https://blog. openai. com/faulty-reward-functions, 2016.

[15] Mary L Cummings and Carl E Nehme. Modeling the impact of workload in network centric supervisory control settings. In Neurocognitive and Physiological Factors During High-tempo Operations, pages 23–41. CRC Press, 2018.

[16] Abhishek Das, Théophile Gervet, Joshua Romoff, Dhruv Batra, Devi Parikh, Mike Rabbat, and Joelle Pineau. Tarmac: Targeted multi-agent communication. In Proceedings of the International Conference on Machine Learning (ICML), pages 1538–1546. PMLR, 2019.

[17] Birsen Donmez, Carl Nehme, and Mary L Cummings. Modeling workload impact in multiple unmanned vehicle supervisory control. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 40(6):1180–1190, 2010.

[18] Lydia Fischer, Barbara Hammer, and Heiko Wersing. Combining offline and online classifiers for life-long learning. In Proceedings of the International Joint Conference on Neural Networks (IJCNN), pages 1–8. IEEE, 2015.

[19] Jakob Foerster, Ioannis Alexandros Assael, Nando De Freitas, and Shimon Whiteson. Learning to communicate with deep multi-agent reinforcement learning. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), 29, 2016.

https://doi.org/10.52202/075280-1539

[20] Avinash Gautam and Sudeept Mohan. A review of research in multi-robot systems. In Proceedings of the IEEE 7th International Conference on Industrial and Information Systems (ICIIS), pages 1–5. IEEE, 2012.

[21] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. Communications of the ACM, 63(11):139–144, 2020.

[22] Nate Gruver, Jiaming Song, Mykel J Kochenderfer, and Stefano Ermon. Multi-agent adversarial inverse reinforcement learning with latent variables. In Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS), pages 1855–1857, 2020.

[23] Ibrahim A Hameed. Using natural language processing (nlp) for designing socially intelligent robots. In Proceedings of the Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob), pages 268–269. IEEE, 2016.

[24] Sandra G Hart and Lowell E Staveland. Development of nasa-tlx (task load index): Results of empirical and theoretical research. In Advances in Psychology, volume 52, pages 139–183. Elsevier, 1988.

[25] Guy Hoffman and Cynthia Breazeal. Collaboration in human-robot teams. In Proceedings of the AIAA 1st Intelligent Systems Technical Conference (IntelliSys), page 6434, 2004.

[26] Ryan Hoque, Lawrence Yunliang Chen, Satvik Sharma, Karthik Dharmarajan, Brijen Thananjeyan, Pieter Abbeel, and Ken Goldberg. Fleet-dagger: Interactive robot fleet learning with scalable human supervision. In Proceedings of the Conference on Robot Learning (CoRL), pages 368–380. PMLR, 2023.

[27] Junling Hu, Michael P Wellman, et al. Multiagent reinforcement learning: theoretical framework and an algorithm. In Proceedings of the International Conference on Machine Learning (ICML), volume 98, pages 242–250, 1998.

[28] Wonseok Jeon, Paul Barde, Derek Nowrouzezahrai, and Joelle Pineau. Scalable multi-agent inverse reinforcement learning via actor-attention-critic. arXiv preprint arXiv:2002.10525, 2020.

[29] Sham Machandranath Kakade. On the sample complexity of reinforcement learning. University of London, University College London (United Kingdom), 2003.

[30] Diksha Khurana, Aditya Koli, Kiran Khatter, and Sukhdev Singh. Natural language processing: State of the art, current trends and challenges. Multimedia Tools and Applications, pages 1–32, 2022.

[31] Daewoo Kim, Sangwoo Moon, David Hostallero, Wan Ju Kang, Taeyoung Lee, Kyunghwan Son, and Yung Yi. Learning to schedule communication in multi-agent reinforcement learning. In Proceedings of the International Conference on Learning Representations (ICLR), 2018.

[32] Sachin G Konan, Esmaeil Seraj, and Matthew Gombolay. Iterated reasoning with mutual information in cooperative and byzantine decentralized teaming. In Proceedings of the International Conference on Learning Representations (ICLR), 2021.

[33] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. Physical Review E, 69(6):066138, 2004.

[34] Hoang M Le, Yisong Yue, Peter Carr, and Patrick Lucey. Coordinated multi-agent imitation learning. In Proceedings of the International Conference on Machine Learning (ICML), pages 1995–2003. PMLR, 2017.

[35] Zeming Lin, Jonas Gehring, Vasil Khalidov, and Gabriel Synnaeve. Stardata: A starcraft ai research dataset. In Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AAAI), volume 13, pages 50–56, 2017.

https://doi.org/10.52202/075280-1539

[36] Michael L Littman. Markov games as a framework for multi-agent reinforcement learning. In Machine Learning Proceedings, pages 157–163. Elsevier, 1994.

[37] Ryan Lowe, Yi I Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), 30, 2017.

[38] Jean MacMillan, Elliot E Entin, and Daniel Serfaty. Communication overhead: The hidden cost of team cognition. 2004.

[39] Ajay Mandlekar, Yuke Zhu, Animesh Garg, Jonathan Booher, Max Spero, Albert Tung, Julian Gao, John Emmons, Anchit Gupta, Emre Orbay, et al. Roboturk: A crowdsourcing platform for robotic skill learning through imitation. In Proceedings of the Conference on Robot Learning (CoRL), pages 879–893. PMLR, 2018.

[40] Negar Mehr, Mingyu Wang, Maulik Bhatt, and Mac Schwager. Maximum-entropy multi-agent dynamic games: Forward and inverse solutions. IEEE Transactions on Robotics, 2023.

[41] Douglas De Rizzo Meneghetti and Reinaldo Augusto da Costa Bianchi. Towards heterogeneous multi-agent reinforcement learning with graph neural networks. arXiv preprint arXiv:2009.13161, 2020.

[42] Manisha Natarajan, Esmaeil Seraj, Batuhan Altundas, Rohan Paleja, Sean Ye, Letian Chen, Reed Jensen, Kimberlee Chestnut Chang, and Matthew Gombolay. Human-robot teaming: Grand challenges. Current Robotics Reports, pages 1–20, 2023.

[43] Sriraam Natarajan, Gautam Kunapuli, Kshitij Judah, Prasad Tadepalli, Kristian Kersting, and Jude Shavlik. Multi-agent inverse reinforcement learning. In Proceedings of the Ninth International Conference on Machine Learning and Applications (ICMLA), pages 395–400. IEEE, 2010.

[44] Rohan Paleja, Andrew Silva, Letian Chen, and Matthew Gombolay. Interpretable and personalized apprenticeship scheduling: Learning interpretable scheduling policies from heterogeneous user demonstrations. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), 33:6417–6428, 2020.

[45] Georgios Th Papadopoulos, Asterios Leonidis, Margherita Antona, and Constantine Stephanidis. User profile-driven large-scale multi-agent learning from demonstration in federated human-robot collaborative environments. In Proceedings of the Human-Computer Interaction. Technological Innovation: Thematic Area (HCI), pages 548–563. Springer, 2022.

[46] Peixi Peng, Junliang Xing, and Lili Cao. Hybrid learning for multi-agent cooperation with sub-optimal demonstrations. In In Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, pages 3037–3043, 2021.

[47] Lindsay Sanneman and Julie Shah. Transparent value alignment. In Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI), pages 557–560, 2023.

[48] Stefan Schaal. Learning from demonstration. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), 9, 1996.

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

[50] Esmaeil Seraj. Embodied team intelligence in multi-robot systems. AAMAS 2022 Doctoral Consortium, 2022.

[51] Esmaeil Seraj. Embodied, intelligent communication for multi-agent cooperation. In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), volume 37, pages 16135–16136, 2023.

[52] Esmaeil Seraj, Letian Chen, and Matthew C Gombolay. A hierarchical coordination framework for joint perception-action tasks in composite robot teams. IEEE Transactions on Robotics, 2021.

https://doi.org/10.52202/075280-1539

[53] Esmaeil Seraj, Andrew Silva, and Matthew Gombolay. Multi-uav planning for cooperative wildfire coverage and tracking with quality-of-service guarantees. Journal of Autonomous Agents and Multi-Agent Systems, 36(2):39, 2022.

[54] Esmaeil Seraj, Zheyuan Wang, Rohan Paleja, Daniel Martin, Matthew Sklar, Anirudh Patel, and Matthew Gombolay. Learning efficient diverse communication for cooperative heterogeneous teaming. In Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS), pages 1173–1182, 2022.

[55] Esmaeil Seraj, Zheyuan Wang, Rohan Paleja, Matthew Sklar, Anirudh Patel, and Matthew Gombolay. Heterogeneous graph attention networks for learning diverse communication. arXiv preprint arXiv:2108.09568, 2021.

[56] Esmaeil Seraj, Xiyang Wu, and Matthew Gombolay. Firecommander: An interactive, probabilistic multi-agent environment for heterogeneous robot teams. arXiv preprint arXiv:2011.00165, 2020.

[57] Andy Shih, Stefano Ermon, and Dorsa Sadigh. Conditional imitation learning for multi-agent games. In Proceedings of the 17th ACM/IEEE International Conference on Human-Robot Interaction (HRI), pages 166–175. IEEE, 2022.

[58] Marco AC Simões, Robson Marinho da Silva, and Tatiane Nogueira. A dataset schema for cooperative learning from demonstration in multi-robot systems. Journal of Intelligent & Robotic Systems, 99:589–608, 2020.

[59] Amanpreet Singh, Tushar Jain, and Sainbayar Sukhbaatar. Learning when to communicate at scale in multiagent cooperative and competitive tasks. arXiv preprint arXiv:1812.09755, 2018.

[60] Jiaming Song, Hongyu Ren, Dorsa Sadigh, and Stefano Ermon. Multi-agent generative adversarial imitation learning. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), 31, 2018.

[61] Adrian Šošic, Wasiur R KhudaBukhsh, Abdelhak M Zoubir, and Heinz Koeppl. Inverse reinforcement learning in swarm systems. In Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems (AAMAS), pages 1413–1421, 2017.

[62] William Squires and Sean Luke. Scalable heterogeneous multiagent learning from demonstration. In Proceedings of the International Conference on Practical Applications of Agents and Multi-Agent Systems, pages 264–277. Springer, 2020.

[63] Aaron Steinfeld, Odest Chadwicke Jenkins, and Brian Scassellati. The oz of wizard: simulating the human for interaction research. In Proceedings of the ACM/IEEE International Conference on Human Robot Interaction (HRI), pages 101–108, 2009.

[64] Sriram Ganapathi Subramanian, Matthew E Taylor, Kate Larson, and Mark Crowley. Multi-agent advisor q-learning. Journal of Artificial Intelligence Research, 74:1–74, 2022.

[65] Sainbayar Sukhbaatar, Rob Fergus, et al. Learning multiagent communication with backpropagation. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), 29, 2016.

[66] Faraz Torabi, Garrett Warnell, and Peter Stone. Behavioral cloning from observation. arXiv preprint arXiv:1805.01954, 2018.

[67] Albert Tung, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Yuke Zhu, Li Fei-Fei, and Silvio Savarese. Learning multi-arm manipulation through collaborative teleoperation. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), pages 9212–9219. IEEE, 2021.

[68] Hongwei Wang, Lantao Yu, Zhangjie Cao, and Stefano Ermon. Multi-agent imitation learning with copulas. In Machine Learning and Knowledge Discovery in Databases, pages 139–156. Springer, 2021.

[69] Xingyu Wang and Diego Klabjan. Competitive multi-agent inverse reinforcement learning with sub-optimal demonstrations. In Proceedings of the International Conference on Machine Learning (ICML), pages 5143–5151. PMLR, 2018.

[70] Fan Yang, Alina Vereshchaka, Changyou Chen, and Wen Dong. Bayesian multi-type mean field multi-agent imitation learning. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), 33:2469–2478, 2020.

[71] Lantao Yu, Jiaming Song, and Stefano Ermon. Multi-agent adversarial inverse reinforcement learning. In Proceedings of the International Conference on Machine Learning (ICML), pages 7194–7201. PMLR, 2019.

https://doi.org/10.52202/075280-1539