Apprenticeship Learning for Heterogeneous Multi-agent Coordination

Jerry Xiong

Acknowledgements

I would like to thank Esmaeil Seraj and Professor Matthew Gombolay, without whom this work would not have been possible.

Contents

1 Introduction **3**

2 Problem Description **4**

3 Literature Review **5**

4 Methods **9**

4.1 Algorithm 10

4.2 Discriminator Architecture 11

4.3 Communication Reconstruction 13

4.4 Behavioral Cloning 14

4.5 Full Objective 15

5 Evaluation **16**

5.1 Evaluation Domains 16

5.2 Baselines 17

5.3 Discriminator Architectures 19

5.4 Ablation: Reconstruction Objective 20

5.5 Ablation: Behavioral Cloning 21

6 Discussion **22**

7 Conclusion **23**

8 Limitations and Future Work **24**

A Hyperparameters **33**

B Expert Heuristics **33**

1 Introduction

Multi-agent teaming, a setting in which numerous, interacting entities cooperate to achieve common goals or behaviors, has a variety of applications including robot coordination [1], autonomous driving [2], and distributed control systems [3]. Under a paradigm known as multi-agent reinforcement learning, agents optimize their behavior in a trial-and-error process in order to maximize a reward signal—a scalar form of feedback which explicitly evaluates the quality of the actions taken [4]. However, this approach requires manually specifying a task-specific reward that effectively induces the desired coordination strategy. This can be an issue in certain domains as a poorly-specified reward can result in unintended or harmful behaviors [5]. Reinforcement learning approaches can also struggle to learn efficiently if the provided reward signal is sparse and only rarely provides useful feedback [6].

An alternative approach, known as apprenticeship learning, involves leveraging demonstrations which implicitly encode rich information about desirable behaviors [7]. Multi-agent teams trained using this paradigm no longer require the specification of a reward signal.

In this work, we specifically examine apprenticeship learning for environments involving coordination between heterogeneous agents. In other words, scenarios where different members of team are not restricted to having identical sensing or operational capabilities. Examples of such environments include wireless sensor and actor networks [8] in which low-cost, low-power sensor devices transmit information to resource rich actor nodes as part of a coordinated decision-making process.

We propose an algorithm for imitating a demonstrated multi-agent behavior while simultaneously learning to communicate from scratch. In other words, this approach enables learning a desired high-level coordination strategy without manual specification or demonstration of the low-level communication protocol required in order to enable such a strategy. We also propose several modifications to existing modeling architectures and optimization objectives which empirically demonstrate improved sample efficiency during learning, reducing the required amount of interactions with the environment. We show that these approaches

in combination can be effectively applied to a variety of difficult tasks, including some which involve simultaneous interaction with the environment and inter-agent communication. Through this work, we aim to reduce barriers of entry to deploying multi-agent teams in complex, real-world environments.

2 Problem Description

We consider the setting of decentralized partially observable Markov decision processes [9]. The environment configuration is described with a *state*, s, contained in a set of possible states, S. At each time step, every agent, $i \in 1, \ldots, n$, receives a local *observation*, o^i, as a probabilistic function of the state, $O^i(o^i \mid s)$, and takes an *action*, a^i, out of a set of possible actions, A^i. The set of all possible observations for an agent is called the agent's *observation space*, while the set of actions is called the *action space*. The *policy*, $\pi^i(a^i \mid o^i) : A^i \to \mathbb{R}$, represents the agent's distribution over actions taken in each state. At any timestep t, the system transitions from a state, s_t, to the next state, s_{t+1}, given the joint action, $\boldsymbol{a}_t = (a_t^1, \ldots, a_t^n)$, based on a transition probability function, $T(s_{t+1} \mid s_t, \boldsymbol{a}_t)$. The starting state is sampled from some distribution: $s_0 \sim \eta$.

In the context of standard multi-agent reinforcement learning, a *reward function*, $R(s, \boldsymbol{a})$, is defined. Fully-cooperative reinforcement learning involves a single scalar reward, shared between all agents. The quality of a set of policies, $\pi^1, \ldots, \pi^n$, is given by their expected *return*, $\sum_{t=0}^{T} \gamma^t r_t$, where $\gamma \in [0, 1]$ is the discount factor, r_t denotes the reward received at timestep t, and T is the episode time horizon. For algorithms which involve collecting data in an online fashion by interacting with the environment, *sample efficiency* refers to the amount of experience required during learning.

We define a trajectory, ζ, as a sequence of states and actions, $(s_0, \boldsymbol{a}_0, s_1, \boldsymbol{a}_1, \ldots, s_T)$. When discussing apprenticeship learning, we assume that the provided demonstrations satisfy $s_0 \sim \eta$, $\boldsymbol{a}_t = (a_t^1, \ldots, a_t^n)$ has $a_t^i \sim \pi_E^i(\cdot \mid s_t) \quad \forall i \in 1, \ldots, n$, and $s_{t+1} \sim T(s_t, \boldsymbol{a}_t) \quad \forall t \in 1, \ldots, T-1$.

The policies $\pi_E^1, \ldots, \pi_E^n$ are called the *expert* policies.

Apprenticeship learning is the task of learning to imitate a behavior using expert demonstrations [7]. In contrast to standard reinforcement learning, which involves learning a behavior policy given a reward function, a paradigm called *inverse reinforcement learning* (IRL) involves extracting a reward function from observed behavior [10]. Existing algorithms for both paradigms and the relevance of IRL to apprenticeship learning are discussed in Section 3.

In this work, we consider the problem of online apprenticeship learning in decentralized, partially observable multi-agent environments. In other words, given a set of trajectories, $\{\zeta_k^E\}_{k=1}^K$, generated by expert policies, $\pi_E^1, \ldots, \pi_E^n$, we attempt to learn policies, $\pi^1, \ldots, \pi^n$, which imitate the expert behavior. For this purpose, agents can interact with the environment online, potentially receiving a reward signal, R. This reward signal can itself be learned in order to induce the desired imitation behavior. We specifically consider the case in which agents may be heterogeneous, so different agents $i \neq j$ may have different observation spaces O^i, O^j or action spaces A^i, A^j.

3 Literature Review

Before we cover existing literature on multi-agent apprenticeship learning, we include a preliminary discussion of the corresponding approaches for the simpler, single-agent case.

One of the simplest approaches to apprenticeship learning is behavioral cloning, which directly trains the agent to maximize the likelihood of the demonstrated actions given the corresponding environment states [11]. However, behavioral cloning can often produce policies with poor performance, especially when dealing with limited data, due to compounding errors resulting in covariate shift [12]. Intuitively, the agents tend to have poor performance on states not seen in the demonstration dataset—and since actions taken in the current state affect future states, the errors tends to compound over the course of a trajectory.

One approach which addresses this issue, known as Dataset Aggregation (DAgger) [13], involves manually re-labeling visited states with expert actions during training. In a similar line of work, an approach called Training an Agent Manually via Evaluative Reinforcement (TAMER) [14] incorporates human-provided rewards into the training pipeline. In this work, we instead focus on approaches which do not require additional, online interaction with human experts.

Generative adversarial imitation learning (GAIL) [15] addresses the aforementioned issue with behavioral cloning by training the agent in such a way that steers entire trajectories closer to the desired behavior. To accomplish this, it is assumed that a simulator of the environment is available, and state-action pairs are collected from executing the learned policy. Then, a separate model known as the discriminator is trained to differentiate between state-action pairs collected by the learned agent and pairs provided in the demonstration dataset. The output of this discriminator is treated as a reward function and optimized over using standard reinforcement learning algorithms, encouraging the agent to match the expert in expectation over full trajectories. Notably, unlike direct reinforcement learning approaches, this reward signal is not manually specified but rather learned automatically.

GAIL is an approach which primarily aims to extract a policy which effectively imitates the demonstrated behavior. A paradigm known as inverse reinforcement learning (IRL) aims to extract a reward function which explains the demonstrated behavior [10]. Apprenticeship learning can be framed as an IRL problem, since a reward function under which the demonstrated policy is optimal can then be used to train an imitation policy with reinforcement learning [7].

Although the GAIL discriminator is used to provide a reward signal during training, a policy trained to optimality produces state-action pairs which are impossible to differentiate from the demonstrations. The discriminator tends to converge to a uniform output across all states and actions, which does not encode a particularly informative reward signal. Adversarial inverse reinforcement learning (AIRL) [16] is a framework that addresses this issue by placing

a specific structure on the discriminator which enables recovering a meaningful reward function even after convergence. The factor which makes AIRL particularly interesting in the context of apprenticeship learning is that by placing an additional constraint on the reward function—restricting the reward to depend only on the observation and not the action—the resulting policies were empirically found to be more robust than GAIL with respect to changes in environment dynamics.

Other approaches based on reward learning include maximum margin IRL, [10], maximum entropy IRL [17], soft Q imitation learning [18], and inverse soft-Q learning [19]. However, we will focus on approaches which have been directly extended to multi-agent environments in existing literature, namely GAIL and AIRL.

New difficulties arise when considering multi-agent environments. For instance, since each agent's policy—and therefore, the way they interact with other agents—changes over the course of training, the effect of any individual agent's action may also change, resulting in an effectively non-stationary environment [20]. Additionally, there may exist multiple strategy equilibria in multi-agent environments [21]. In other words, whether a policy is optimal for one agent can depend on the current policies of all other agents.

One body of existing work extends algorithms for single-agent apprenticeship learning to multiple agents using restrictive assumptions about the structure of the environment. For example, Šošić et al. [22] proposes an IRL algorithm for swarm systems based on the assumption that all agents share the same dynamics and observation spaces and that the demonstrated behavior depends only on a small neighborhood around each agent rather than the state of the entire group. This approach enables reducing the multi-agent problem to an equivalent single-agent problem but is not applicable to systems with any degree of agent heterogeneity. Bhattacharyya et al. [23] discuss an application of GAIL to a multi-agent driving simulator. However, the proposed algorithm is based on full parameter sharing between all agents, so it similarly cannot be applied to environments where it is desirable for different agents to learn different behaviors. Wang & Klabjan [24] propose a robust IRL

algorithm for two-player zero-sum games—in other words, games where one player's gain is always equivalent to the other player's loss—which rules out environments with inter-agent cooperation. In our work, we primarily focus on approaches relevant to heterogeneous teaming, where agents can differ in terms of observation spaces, action spaces, and/or policies.

Additional related works include the approach by [25], who examine the problem of adapting an agent's policy to the strategies of new partners during test time. In other words, the diversity in the joint strategy of all other agents is treated as diversity in the space of possible tasks, and the goal is to meta-learn a method to adapt to different partners, and thus different tasks, at test time. Agents do not interact directly through communication, and the adaptation is performed by observing the actions taken by the partner agents. In our work, we instead examine the setting where the policies for all agents, rather than just a single agent, are learned simultaneously, and the agents must coordinate by learning an inter-agent communication strategy.

Multi-agent generative adversarial imitation learning (MA-GAIL) [26] is one existing approach applicable to multi-agent apprenticeship learning in cooperative, heterogeneous environments. MA-GAIL is an extension of the aforementioned single-agent GAIL algorithm to multiple agents which introduces several variants of a multi-agent discriminator architecture. These variants are based on different kinds of prior knowledge about the structure of the environment's reward. For example, a fully centralized discriminator was utilized for scenarios where all agents should receive identical rewards, and a decentralized discriminator for environments which may have a mix of cooperative and competitive interactions. However, the empirical evaluations in this work were limited to relatively simple environments; for example, no task included an agent which was capable of simultaneously communicating with another agent and taking an action which affects the environment state directly.

Multi-agent adversarial inverse reinforcement learning (MA-AIRL) [27] is a similar algorithm based on extending the previously mentioned AIRL algorithm to multi-agent environments. Empirically, this approach was found to perform competitively with MA-GAIL

in terms of imitation learning performance while also extracting a reward function highly correlated with the ground-truth reward function used to train the demonstrating policies. Jeon et al. [28] builds off of MA-AIRL by examining the effects that the discriminator architectures proposed in the MA-GAIL paper as well as the observation-only variants in the original single-agent AIRL paper have on sample efficiency. In particular, it was found that an observation-only decentralized discriminator was shown to fail on environments where a blind listener agent acted solely based on the communications of a different, speaker agent. However, the empirical evaluations in these works are still limited to environments similar to those presented in the MA-GAIL paper, albeit scaled up to more agents in [28]. We experiment with complex environments involving simultaneous acting and communicating in Section 5.

To learn in environments with inter-agent communication, both MA-GAIL and MA-AIRL use demonstrations which include communication actions in the action space. In other words, the demonstration data needs to encode all the information shared between agents. This restriction rules out the possibility of leveraging a large body of work related to learning differentiable communication strategies—approaches which pass gradient information through messages sent between agents. One example of such an approach in the context of standard tabula rasa reinforcement learning is introduced in Sukhbaatar & Fergus [29]. We describe an algorithm which integrates a differentiable inter-agent communication channel with apprenticeship learning in Section 4.1, and compare the performance of this approach with those of several baseline communication methodologies in Section 5.2.

4 Methods

In this section, we describe an algorithm for apprenticeship learning in cooperative, multi-agent heterogeneous environments with inter-agent communication when the communication itself is not demonstrated. In Subsection 4.1, we give an overview of the learning algorithm

and modeling assumptions. We build on this foundation in Subsections 4.2, in which we discuss an alternative discriminator architecture designed to handle cooperation between heterogeneous agents. The remaining three subsections motivate auxiliary optimization objectives designed to further improve sample efficiency in difficult environments.

4.1 Algorithm

We consider a centralized training and decentralized execution (CTDE) paradigm [30] similar to the approaches used in MA-GAIL [26] and MA-AIRL [27]. Additional information is used during the training process to enable efficient learning of the agents' policies. However, during execution time, each agent only utilizes information provided in their own, local observations as well as the communicated messages received from other agents.

In our approach, we model communication by having each agent with index i broadcast a vector, $c_i \in \mathbb{R}^d$ for some fixed communication dimensionality d. Agent i's policy is conditioned on their own observation as well as the communications broadcast by every other agent: $a^i \sim \pi^i(o^i, \boldsymbol{c}^{-i})$ where $\boldsymbol{c}^{-i} = \{c_j \mid j \in 1, \ldots, n, j \neq i\}$ denotes the communications of all agents aside from agent i.

We note that the empirical evaluations discussed in the MA-GAIL and MA-AIRL papers involve environments in which every aspect of the current environment state is always observed by at least one agent. In this work, we experiment with environments that have a greater degree of partial observability. For instance, environments where each agent can only see within a local vision radius. This requires agents to not only share information with each other, but also integrate information across time. Thus, unlike, MA-GAIL and MA-AIRL, we modify all tested approaches, including existing baselines, by introducing recurrence into each agent's neural network policy. One impact of this change is that any kind of behavioral cloning pretraining (discussed further in Section 4.4) now involves some method to condition on and update stored hidden states, e.g. truncated back-propagation through time [31].

For simplicity, we use a similar baseline network architecture similar to the architectures

that appear in MA-GAIL for all agents, environments, and tested approaches. For the policy, each agent initially processes their local observation via a fully connected layer, followed by a Gated Recurrent Unit [32]. In the smaller environments, each agent simply receives the encoding of their own observation, concatenated with the communication vectors broadcast by the other agents, and passes this vector through a final linear layer in order to parameterize a standard softmax policy. In environments with a larger number of agents, we instead insert a self-attention mechanism as the penultimate layer in order to process the agents' communication embeddings. We use similar model sizes and hidden layer dimensionalities in all tested approaches; detailed hyperparameter settings can be found in Appendix A.

Following the CTDE paradigm, each agent is trained using a centralized critic. In particular, we condition the critic on the observations of all agents and the actions taken by all *other* agents when estimating any particular agent's state value function, which is consistent with the approach used in MA-GAIL. Identical critic architectures are used for all agents in all tested approaches.

During training, a discriminator learns to differentiate between observations and actions generated by the learned policies and those provided by expert demonstrations. The output of this discriminator provides a reward signal to the agents, who are then trained via standard reinforcement learning. Here, we use Proximal Policy Optimization (PPO) [33]. The pseudocode for the full, proposed algorithm is presented in 1.

4.2 Discriminator Architecture

MA-GAIL [26] introduces three three discriminator architectures, which they denote as centralized, decentralized, and zero-sum, intended for fully cooperative, mixed cooperative-competitive, and zero-sum games, respectively. In the decentralized architecture, each agent receives their own reward signal as a function of their current local observation and action. The centralized discriminator instead takes in the joint observations and actions of all agents and outputs a single shared reward. Although the centralized discriminator was intended for fully

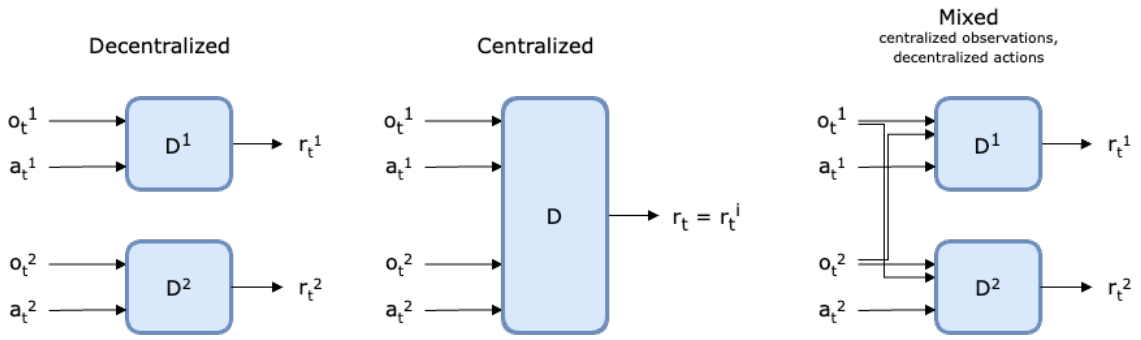

Figure 1: Discriminator Architecture Comparison for $n = 2$ agents

cooperative tasks, Jeon et al. [28] find the centralized discriminator to have poor scalability as the number of agents increases and recommend using the decentralized discriminator instead.

To illustrate a potential explanation for this phenomena, suppose we ignore the impact of function approximation and assume that we have a *tabular* discriminator which simply outputs 1 for observation-action pairs stored in the demonstration dataset and zero otherwise. For agent i, the likelihood that a single observation-action pair, (o^i, a^i), appears in the decentralized case, is necessarily at least the likelihood that the entire team's joint observation-action $(\boldsymbol{o}, \boldsymbol{a})$, appears in the centralized case, for o^i, a^i in $\boldsymbol{o}, \boldsymbol{a}$. Therefore, supposing that the set of demonstrated expert behaviors only represents a small portion of the space of all possible behaviors, such a tabular discriminator would provide a much sparser reward signal in the centralized case, reducing sample efficiency.

However, in order to enable cooperation among heterogeneous, communicating agents, we find that it may be necessary to utilize information from other agents in the discriminator in order to provide an informative learning signal. For example, if a particular agent does not have its own local observations and instead needs to learn to act solely based on information from other agents, a decentralized discriminator which only has access to the agent's local observation and action may not be sufficient.

We examine an alternative discriminator architecture which can access global observations while being restricted to only utilize local actions: $D^i : O^1 \times O^2 \times \cdots \times O^n \times A^i \to \mathbb{R}$. Since this architecture utilizes observations on a global scale with actions a local scale, we call it

the `mixed` discriminator architecture. The mixed architecture still maintains some of the strengths of the decentralized architecture: after all, a joint-observation, single-action pair $(\boldsymbol{o}, a^i)$ is still exponentially more likely than a joint-observation, joint-action pair $(\boldsymbol{o}, \boldsymbol{a})$ as the number of agents increases.

We conduct an ablation study, presented in Section 5.3, to analyze the empirical performance of the decentralized, centralized, and mixed discriminators in several homogeneous and heterogeneous multi-agent environments.

4.3 Communication Reconstruction

We introduce a reverse model for each agent that predicts the agent's most recently broadcasted communication given the team's joint action and the communications of all other agents. Denoting the reverse model g^i, the reconstructed communication is given by $\hat{c}_t^i = g^i(\boldsymbol{a}_t, \boldsymbol{c}^{-i})$ and the policy and reverse model are trained together in an end-to-end fashion to minimize the error in the reconstruction, $||c_t^i - \hat{c}_t^i||_2^2$. This approach bears similarity to the mutual information maximization approach presented in [34], but we apply the objective to embeddings for inter-agent communication rather than embeddings used to model demonstrator preferences. Additionally, in practice we do not sample from a posterior distribution over the latent variable, and the reverse model simply generates its prediction deterministically as a function of the agents' action logits.

Motivating the communication reconstruction objective is the notion that, given the effects of the communication (the actions taken by the other agents), we want to be able to predict what each agent's original communication was. This discourages distinct messages from encoding the same semantic meaning, as the reverse model can only generate a single point prediction for the original communication. Additionally, conditioning on the actions encourages messages which are semantically meaningful and have a concrete impact on the actions taken by other agents. The performance impact of this objective is analyzed in Section 5.4.

4.4 Behavioral Cloning

The mixed discriminator, unlike the global discriminator, does not condition the reward at timestep t for agent i on the actions taken by other agents on the same timestep, a_t^{-i}. This can potentially mitigate immediate credit assignment issues. However, the return, as a function of all future rewards, depends on all future states and thus still depends on actions taken by other agents. The size of the joint action space can increase exponentially in the number of agents, while it is possible for only a few joint actions to lead to a positive return. Thus, although the mixed discriminator can reduce the sparsity of the reward, the fundamental issue of credit assignment as we scale to a large number of agents still remains.

To reduce the necessity of exploring this exponentially large action space during online learning, we can leverage behavioral cloning in either an offline or an online fashion. One simple approach, suggested in [15], is to simply pretrain with behavioral cloning offline before fine-tuning with reinforcement learning during online learning. In some cases, existing work has introduced a constraint on the Kullback-Leibler divergence between the pretrained and online agent in order to help mitigate catastrophic forgetting [35]. However, the latter approach was intended for the problem setting of fine-tuning a sub-optimal initialization using online reinforcement learning rather than apprenticeship learning.

Given that the demonstration dataset is stored and accessed anyway during GAIL-style training, we examine the alternative of simply adding an auxiliary behavioral cloning objective to the loss function during online learning. This is similar to the single-agent imitation learning approach presented in [36], but is simpler as we find the importance sampling term to be unnecessary to induce effective learning in the tested environments. In summary, at each step, we update the policy and critic via reinforcement learning on the discriminator's reward signal applied to a minibatch of data generated by the current learned policy. Meanwhile, a minibatch sampled from the demonstration dataset is used to simultaneously update the discriminator and the policy, the former as positive samples in a binary cross-entropy objective, and the latter via maximum likelihood. An analysis of the impact of behavioral cloning used

Algorithm 1 Training Pseudocode

Obtain expert trajectories ζ_E^i for each agent $i = 1, \ldots, n$
Initialize policies π_θ^i and discriminators D_ϕ^i for each agent $i = 1, \ldots, n$.
while not converged **do**
    Collect trajectories $\zeta_\pi = \{(\boldsymbol{o}_t, \boldsymbol{a}_t, \boldsymbol{c}_t)\}_{t=1}^T$ by executing policies π_θ^i for $i = 1, \ldots, n$
    Predict rewards via discriminator, $r^i(\boldsymbol{o}_t, a_t^i) \leftarrow \log(D_\phi^i(\zeta_\pi))$
    Update θ, ϕ using ζ_π, ζ_E according to objective 4.5
end while

in offline pretraining compared to behavioral cloning used as an auxiliary online objective is ablated in Section 5.5.

4.5 Full Objective

The full objective function can be expressed as

$$\mathcal{L}_{\theta,\phi}(\zeta_\pi, \zeta_E) = \mathcal{L}_\theta^{\text{PPO}}(\zeta_\pi) + \mathcal{L}_\phi^{\text{BCE}}(\zeta_\pi, \zeta_E) + \lambda_1 \mathcal{L}_\theta^R(\zeta_\pi) + \lambda_2 \mathcal{L}_\theta^{\text{BC}}(\zeta_E), \tag{1}$$

where ζ_π is a minibatch of trajectory segments generated by the current policy; ζ_E is a minibatch of trajectory segments sampled from the demonstration dataset; $\mathcal{L}^{\text{PPO}}$ is the standard clipped PPO objective with MSE value loss as described in [33]; $\mathcal{L}^{\text{BCE}}$ is the binary cross-entropy loss

$$\mathcal{L}_\phi^{\text{BCE}}(\zeta_\pi, \zeta_E) = (1 - \log(D(\zeta_\pi))) + \log(D(\zeta_E)), \tag{2}$$

with $D(\zeta)$ being the mean predicted discriminator classification across trajectories ζ; $\mathcal{L}^R$ being the reconstruction objective (Section 4.3)

$$\mathcal{L}_\theta^R(\zeta_\pi) = \frac{1}{n} \sum_{i=1}^n ||g(\boldsymbol{a}, \boldsymbol{c}^{-i}) - c^i||^2, \tag{3}$$

with $\boldsymbol{a}, \boldsymbol{c}^{-i}$ for each i sampled from ζ_π; and $\mathcal{L}^{\text{BC}}$ being the online behavioral cloning objective

$$\mathcal{L}_\theta^{\text{BC}}(\zeta_E) = -\frac{1}{n} \sum_{i=1}^n \pi^i(a \mid o^i, \boldsymbol{c}^{-i}), \tag{4}$$

with $a, o^i, \boldsymbol{c}^{-i}$ sampled from ζ_E. The weights λ_1 and λ_2 are tuned as hyperparameters, and the selected values can be found in Appendix A. We provide the pseudocode of our algorithm, in terms of the objective defined above, in Algorithm 1.

5 Evaluation

5.1 Evaluation Domains

We empirically evaluate our approach on three environments previously studied in literature regarding learned multi-agent communication.

Predator-Prey, proposed in [37], is a grid environment in which predator agents attempt to find and move to a randomly placed prey. Each agent's observation consists of only the grid cells within a square vision radius. The environment terminates once every predator successfully moves onto the grid cell with the prey, or some maximum episode length is reached, whichever happens first.

The Predator-Capture-Prey environment, studied in [38], is a heterogeneous version of Predator-Prey in which a number of predator agents are replaced with blind 'capture' agents. These capture agents do not receive observations about their surrounding cells, so they must locate the prey based purely on the communications received from the predator agents.

FireCommander [39] is a difficult environment where a heterogeneous team of perception agents capable of sensing fires and blind action agents capable of extinguishing them must coordinate to put out a wildfire. Unlike Predator-Capture-Prey, the number of targets (fires, rather than prey) changes throughout an episode, as the fires spread or get extinguished. This environment terminates once all fires are extinguished, or again, if some maximum episode length is reached.

In contrast to the empirical evaluations presented in MA-GAIL and MA-AIRL, these three environments have the property that all agents can simultaneously take actions which directly influence the state in some way (such as moving, capturing, or extinguishing fires)

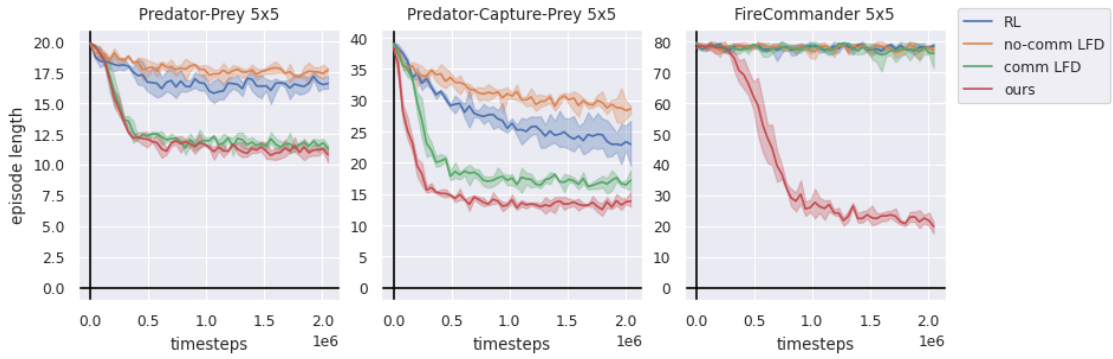

Figure 2: Learning curves comparing the proposed algorithm to several baseline approaches on the easy (5×5) versions of the environments. Training occurs fully online, without behavioral cloning pretraining and without the auxiliary behavioral cloning objective. Shaded regions correspond to bootstrapped 95% confidence intervals for the mean across 3 independent seeds.

and communicate with other agents on the team.

In order to generate demonstrations, the results presented in the original MA-GAIL [26] and MA-AIRL [27] papers collect data from a team of agents trained with reinforcement learning, leveraging a provided ground-truth reward signal. Unfortunately, due to the difficulty of the tasks evaluated in this work, state of the art reinforcement learning approaches typically do not learn near-optimal policies. For example, the best reported performance on FireCommander in [38] averages 46.40 steps per episode, while a heuristic-based policy we designed averages 14.44 steps per episode. Since MA-GAIL and MA-AIRL assume demonstration optimality, we use heuristic-based policies to generate all demonstrations for all presented experimental results.

5.2 Baselines

We first empirically validate the algorithm presented in Section 4.1 compared against several baseline approaches. In particular, one plausible alternative to introducing a differentiable communication channel is to integrate communication into the environment, similar to the setup described in MA-GAIL and MA-AIRL. However, this introduces extra communication observations and actions during online learning which are not available in the offline dataset. This asymmetry prevents a naive application of GAIL, as the discriminator requires consistent

observation and action spaces for both the generated and demonstrated data. In the `comm LFD` baseline, we remedy this issue by modifying the demonstrations to include additional communication actions using a heuristic. Each agent broadcasts a one-hot encoded vector based on their local observations. More detail on the heuristics for both the environment and communication actions are available in Appendix B. The `comm LFD` baseline contrasts with our proposed method in that instead of introducing new communications into the demonstrations, our method restricts the information available to the discriminator during online learning by abstracting the communication into a separate channel instead. In a third baseline, we attempt to sidestep the issues with the discriminator altogether by learning to communicate through the environment via the ground-truth reward signal. In other words, we simply use standard reinforcement learning, and correspondingly denote this method as `RL`. For the sake of completeness, we also compare against an MA-GAIL variant that simply ignores communication entirely, called `no-comm LFD`. All approaches utilize the same policy network architecture sizes where applicable, as well as identical hyperparameter sweep budgets and critic architectures.

Figure 2 depicts learning curves for each approach on the easy (5×5) versions of the three tasks. To isolate the impact of the communication architecture, we do not utilize behavioral cloning in any form for this experiment ($\lambda_2 = 0$). The proposed approach outperforms the baseline approaches in all environments in terms of final performance (fewest steps per episode) after 500 iterations of PPO, or approximately 2 million environment interactions.

On the medium (10×10) and hard (20×20) environments, we find the auxiliary behavioral cloning objective to be necessary for effective learning. For fairness, we include this objective for all tested methods in this experiment. The results are depicted in Figure 3. Due to the behavioral cloning objective, we find that sample efficiency is improved significantly, so we end training after 200 iterations of PPO, or approximately 800 thousand environment interactions. Again, we find that the proposed approach outperforms the baseline approaches in every case.

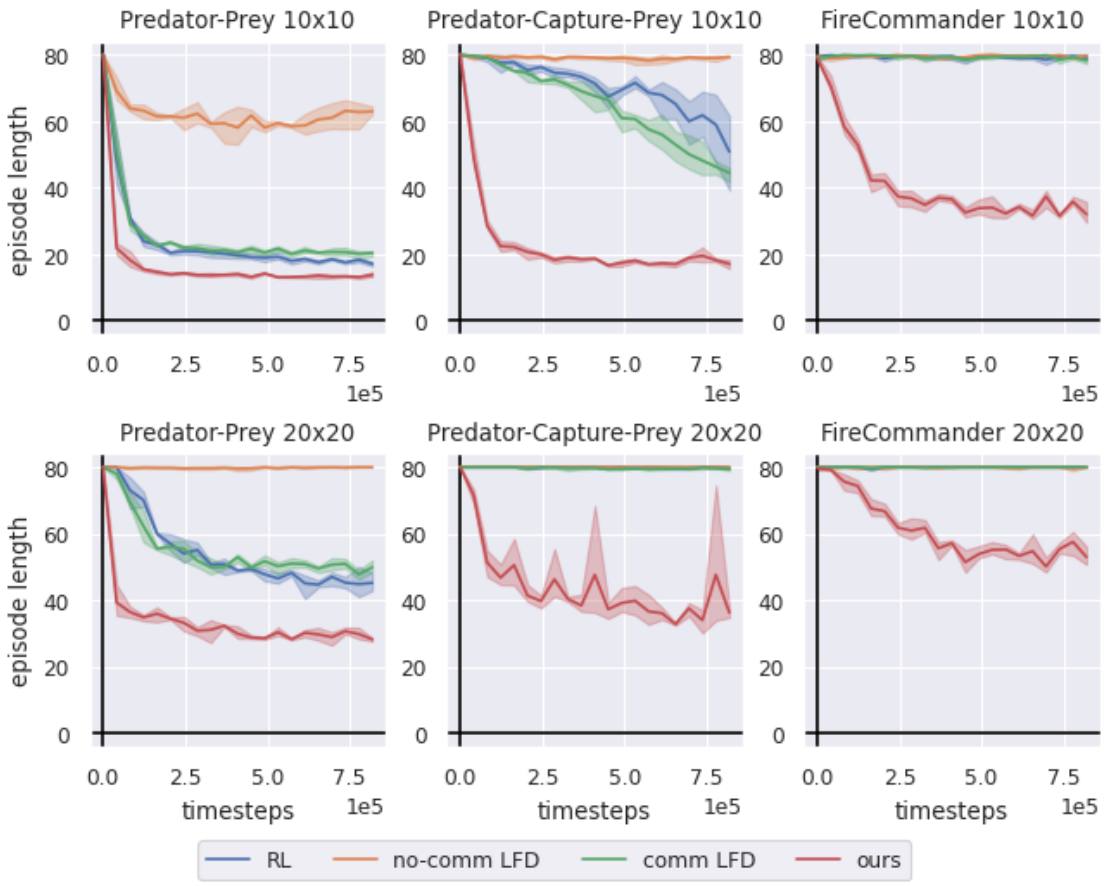

Figure 3: Learning curves comparing the proposed algorithm to several baseline approaches on the medium (10×10) and hard (20×20) versions of the environments. Training for all depicted methods utilizes the auxiliary behavioral cloning objective. Shaded regions correspond to bootstrapped 95% confidence intervals for the mean across 3 independent seeds.

5.3 Discriminator Architectures

We utilize the previously described differentiable communication channel architecture as a drop-in replacement for the policy network in MA-GAIL, and we now examine the impact of introducing our mixed discriminator architecture on the reward side of apprenticeship learning. We present baselines corresponding to two relevant existing discriminator architectures, namely the decentralized and centralized discriminators proposed in the original MA-GAIL paper. For fairness, all results presented in this section utilize the same policy and critic architectures, hyperparameter sweep budgets, and also leverage identical auxiliary reconstruction objectives.

In order to isolate the effects of the discriminator's reward signal, we do not train with

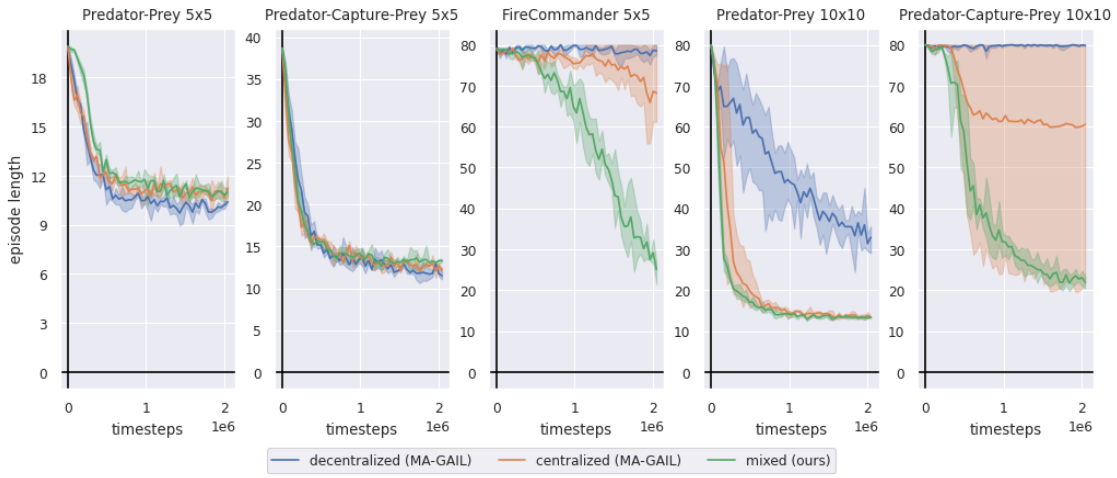

Figure 4: Learning curves comparing performance between different discriminator architectures on the 5 simplest environments, without behavioral cloning. Shaded regions correspond to bootstrapped 95% confidence intervals for the mean across 3 independent seeds.

behavioral cloning in this experiment. We evaluate the three discriminators on the five environments which demonstrate at least a reasonable degree of learning without behavioral cloning (see Section 5.5). As shown in Figure 4, the choice of discriminator architecture seems to have little to no impact on the easiest environments (5×5 Predator-Prey and Predator-Capture-Prey). However, on the difficult, heterogeneous environments (10×10 Predator-Capture-Prey, 5×5 FireCommander), the decentralized discriminator fails to learn any useful behavior. Meanwhile, the mixed discriminator architecture has performance which is at least competitive with or better than the performance of the next best architecture, typically being the centralized discriminator. We note that the centralized discriminator only demonstrated learning in one out of three seeds for 10×10 Predator-Capture-Prey.

5.4 Ablation: Reconstruction Objective

We compare performance with our method both with (tuned λ_1) and without ($\lambda_1 = 0$) the reconstruction loss, as defined in Equation 4.5. As depicted in Figure 5, although the reconstruction objective does result in a small improvement in sample efficiency, it does not explain the entirety of the performance improvement that the proposed algorithm

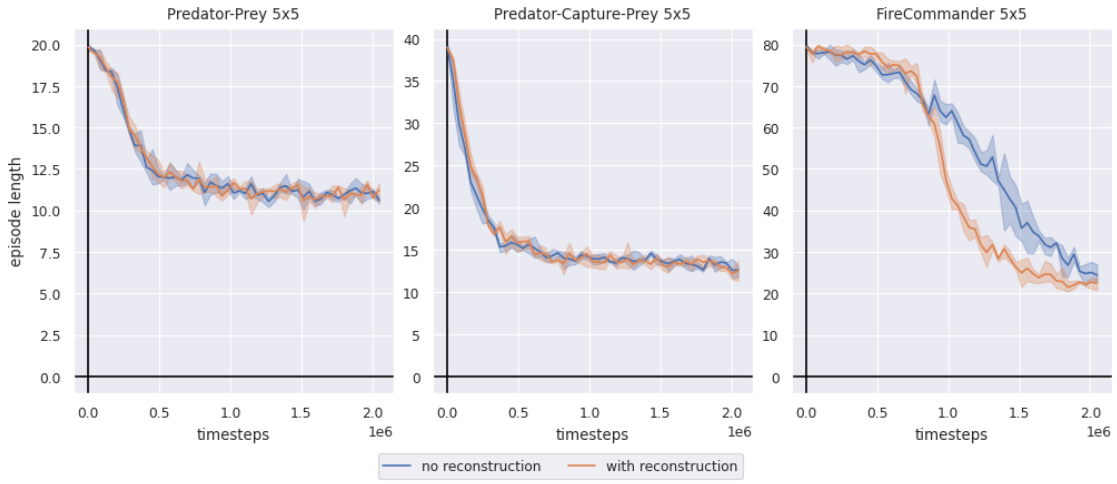

Figure 5: Ablation results for the communication reconstruction objective, trained without behavioral cloning. Shaded regions correspond to bootstrapped 95% confidence intervals for the mean across 3 independent seeds.

demonstrates in Section 5.2.

5.5 Ablation: Behavioral Cloning

Here, we analyze the performance impact of the different approaches to integrating behavioral cloning into our method. We compare three approaches. As a baseline, we include one approach without any behavioral cloning. In this case, the learning signal only comes from the discriminator's reward signal. This setup was is identical to the one used for the easy (5×5) results in Section 5.2. We also compare against the naive approach of pretraining with behavioral cloning offline in order to initialize the policy network, before continuing training online, again using only the discriminator's reward signal. A third approach, identical to the one used for the medium (10×10) and hard (20×20) results in Section 5.2, adds an auxiliary behavioral cloning term to the loss during online training. In other words, the minibatch of transitions sampled from the demonstration dataset is not only used to update the discriminator, but also update the current policy via maximum likelihood.

As depicted in Figure 6, although the approach which simply initializes via behavioral cloning does result in a performance improvement compared to no behavioral cloning at all,

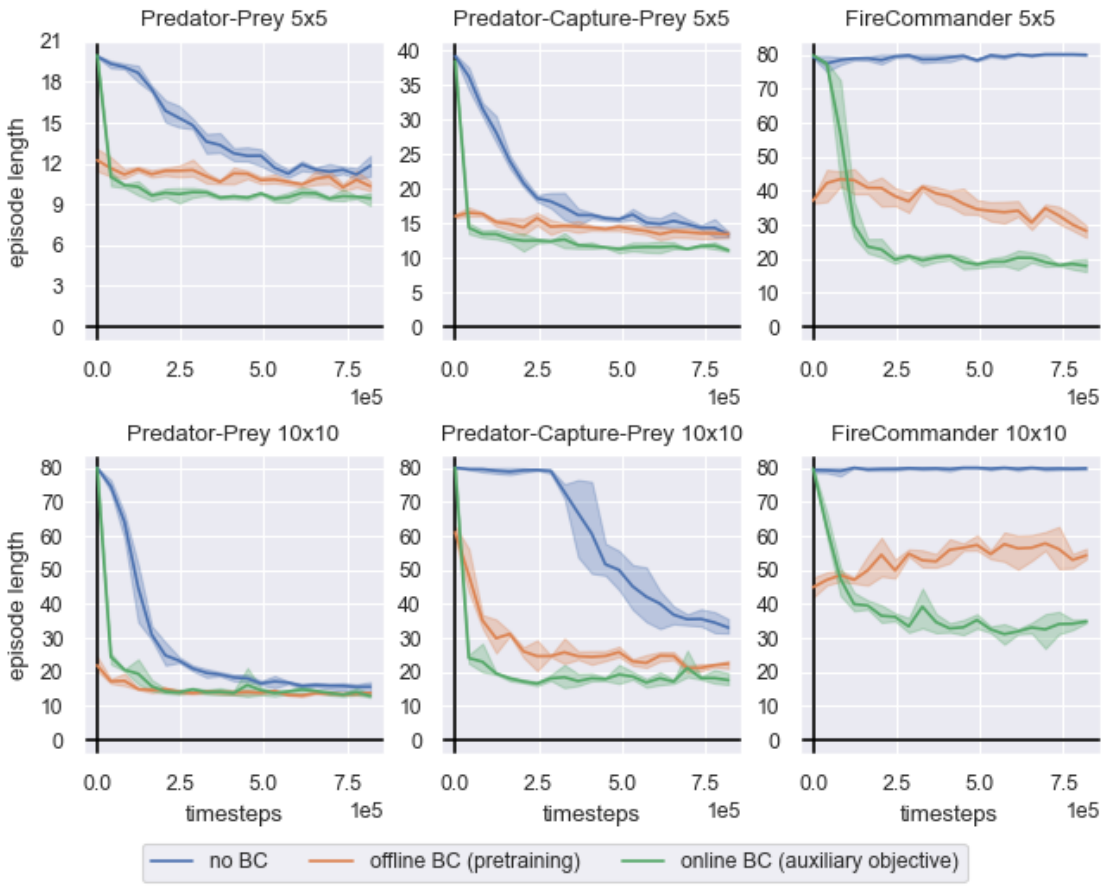

Figure 6: Ablation results for behavioral cloning, comparing a baseline without BC, a baseline which uses BC at initialization, and an approach which adds a BC term to the objective during online learning. Shaded regions correspond to bootstrapped 95% confidence intervals for the mean across 3 independent seeds.

the performance improves by at most only a small margin during online training, and even degrades over time in 10×10 FireCommander. Meanwhile, the online approach consistently learns a behavior which not only outperforms the initial performance of the pretrained approach, but also has the best final performance out of all methods.

6 Discussion

Of the three evaluation domains we experiment with, the simplest, homogeneous domain, Predator-Prey, primarily saw improvements in learning performance from introducing differentiable communication 5.2 and from introducing behavioral cloning as an auxiliary objective 5.5.

Indeed, these two approaches result in consistent improvements across all three domains. Meanwhile, we see from Sections 5.3 and 5.4 that the alternative discriminator architectures and the reconstruction objective generally see improvements only in the difficult heterogeneous domains, especially FireCommander. Thus, the former two contributions appear to be broadly useful across a variety of collaborative multi-agent settings, while the latter seem to be effective in specifically dealing with inter-agent heterogeneity.

One caveat to note is that in the Predator-Prey and Predator-Capture-Prey environments, agents are prevented from moving once they successfully reach a prey. Although this is not an issue in Predator-Prey, we note that this effectively grants capture agents a 1×1 vision radius, as, given a recurrent policy parameterization, an agent can simply check whether its last action resulted in a successful move to determine whether their last position contained the prey. This can, in practice, slightly reduce inter-agent heterogeneity, especially in the easy version of Predator-Capture-Prey, where the predator agents only have a 1×1 vision radius. We leave these dynamics unchanged to match the experimental settings appearing in prior work [38], but we note that a potential alternative approach may be to utilize a modified version of the environment in which capture agents are not prevented from moving, especially if the goal is specifically measure the impact of heterogeneity on a Predator-Prey-like environment. Though, for the purposes of this work, the FireCommander environment does not have this issue, and thus provides a sufficient setting for analyzing learning under heterogeneity.

7 Conclusion

In this work, we propose an approach for sample-efficient apprenticeship learning in cooperative multi-agent environments. By integrating MA-GAIL [26] with a differentiable, attention-based communication architecture, we create an algorithm that is capable of learning to imitate a demonstrated coordination strategy without a need for manual specification or demonstration of the necessary communication protocol.

Additionally, we introduce a novel mixed global/local discriminator architecture which is robust to heterogeneity in the agents' observation and action spaces, and we also propose several auxiliary optimization objectives which improve the scalability of our approach for difficult tasks with large numbers of agents. By combining these approaches, our algorithm demonstrates effective learning on a variety of difficult tasks that require communication between numerous heterogeneous agents.

8 Limitations and Future Work

Several potential avenues of future work exist. One is that although the proposed algorithm is designed to be robust to agent heterogeneity, the empirical evaluations in this work do not explicitly attempt to integrate our approach with multi-agent reinforcement learning algorithms designed to specifically leverage heterogeneity (or, to be precise, approaches which leverage the sparse homogeneity that exists within heterogeneous teams), such as the work presented in [38]. We also do not attempt to address scenarios in which communication may have limited bandwidth, noise, or sparse inter-agent connectivity.

Another avenue for future research is related to recent advancements in online, single-agent imitation learning algorithms which are discriminator-free, such as the approach by Garg et al. [19]. At the moment, it is unclear how our findings related to discriminator architecture design and its impacts on learning with communication between heterogeneous agents might be adapted to such approaches.

Since this work focuses on an apprenticeship learning setting without a provided ground-truth reward signal, we do not attempt to handle scenarios with sub-optimal demonstrations. Combining our work with approaches designed to handle suboptimality such as [40] is an additional area to potentially pursue in the future.

References

[1] L. Merino, F. Caballero, J. R. M. de Dios, J. Ferruz, and A. Ollero, "A cooperative perception system for multiple uavs: Application to automatic detection of forest fires," *Journal of Field Robotics*, vol. 23, no. 3-4, pp. 165–184, 2006. DOI: 10.1002/rob.20108. [Online]. Available: https://doi.org/10.1002/rob.20108.

[2] S. Shalev-Shwartz, S. Shammah, and A. Shashua, "Safe, multi-agent, reinforcement learning for autonomous driving," *ArXiv*, vol. abs/1610.03295, 2016. arXiv: 1610.03295. [Online]. Available: http://arxiv.org/abs/1610.03295.

[3] M. A. Wiering, "Multi-agent reinforcement learning for traffic light control," in *Proceedings of the Seventeenth International Conference on Machine Learning (ICML 2000), Stanford University, Stanford, CA, USA, June 29 - July 2, 2000*, P. Langley, Ed., Morgan Kaufmann, 2000, pp. 1151–1158.

[4] L. Buşoniu, R. Babuška, and B. D. Schutter, "Multi-agent reinforcement learning: An overview," *Innovations in multi-agent systems and applications-1*, pp. 183–221, 2010.

[5] D. Amodei, C. Olah, J. Steinhardt, P. F. Christiano, J. Schulman, and D. Mané, "Concrete problems in AI safety," *ArXiv*, vol. abs/1606.06565, 2016. arXiv: 1606.06565. [Online]. Available: http://arxiv.org/abs/1606.06565.

[6] A. Nair, B. McGrew, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Overcoming exploration in reinforcement learning with demonstrations," in *2018 IEEE International Conference on Robotics and Automation, ICRA 2018, Brisbane, Australia, May 21-25, 2018*, IEEE, 2018, pp. 6292–6299. DOI: 10.1109/ICRA.2018.8463162. [Online]. Available: https://doi.org/10.1109/ICRA.2018.8463162.

[7] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Machine Learning, Proceedings of the Twenty-first International Conference (ICML 2004), Banff, Alberta, Canada, July 4-8, 2004*, C. E. Brodley, Ed., ser. ACM Inter-

national Conference Proceeding Series, vol. 69, ACM, 2004. DOI: 10.1145/1015330.1015430. [Online]. Available: https://doi.org/10.1145/1015330.1015430.

[8] I. F. Akyildiz and I. H. Kasimoglu, "Wireless sensor and actor networks: Research challenges," *Ad Hoc Networks*, vol. 2, no. 4, pp. 351–367, 2004. DOI: 10.1016/j.adhoc.2004.04.003. [Online]. Available: https://doi.org/10.1016/j.adhoc.2004.04.003.

[9] F. A. Oliehoek, "Decentralized pomdps," in *Reinforcement Learning*, ser. Adaptation, Learning, and Optimization, M. A. Wiering and M. van Otterlo, Eds., vol. 12, Springer, 2012, pp. 471–503. DOI: 10.1007/978-3-642-27645-3_15. [Online]. Available: https://doi.org/10.1007/978-3-642-27645-3%5C_15.

[10] A. Y. Ng and S. Russell, "Algorithms for inverse reinforcement learning," in *Proceedings of the Seventeenth International Conference on Machine Learning (ICML 2000), Stanford University, Stanford, CA, USA, June 29 - July 2, 2000*, P. Langley, Ed., Morgan Kaufmann, 2000, pp. 663–670.

[11] D. Pomerleau, "Efficient training of artificial neural networks for autonomous navigation," *Neural Comput.*, vol. 3, no. 1, pp. 88–97, 1991. DOI: 10.1162/neco.1991.3.1.88. [Online]. Available: https://doi.org/10.1162/neco.1991.3.1.88.

[12] S. Ross and D. Bagnell, "Efficient reductions for imitation learning," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, Y. W. Teh and D. M. Titterington, Eds., vol. 9, 2010, pp. 661–668. [Online]. Available: http://proceedings.mlr.press/v9/ross10a.html.

[13] S. Ross, G. J. Gordon, and D. 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, AISTATS 2011, Fort Lauderdale, USA, April 11-13, 2011*, G. J. Gordon, D. B. Dunson, and M. Dudík, Eds., vol. 15, 2011,

pp. 627–635. [Online]. Available: http://proceedings.mlr.press/v15/ross11a/ross11a.pdf.

[14] W. B. Knox and P. Stone, "Interactively shaping agents via human reinforcement: The TAMER framework," in *Proceedings of the 5th International Conference on Knowledge Capture (K-CAP 2009), September 1-4, 2009, Redondo Beach, California, USA*, Y. Gil and N. F. Noy, Eds., ACM, 2009, pp. 9–16. DOI: 10.1145/1597735.1597738. [Online]. Available: https://doi.org/10.1145/1597735.1597738.

[15] J. Ho and S. Ermon, "Generative adversarial imitation learning," in *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, and R. Garnett, Eds., 2016, pp. 4565–4573. [Online]. Available: https://proceedings.neurips.cc/paper/2016/hash/cc7e2b878868cbae992d1fb743995d8f-Abstract.html.

[16] J. Fu, K. Luo, and S. Levine, "Learning robust rewards with adversarial inverse reinforcement learning," *ArXiv*, vol. abs/1710.11248, 2017. arXiv: 1710.11248. [Online]. Available: http://arxiv.org/abs/1710.11248.

[17] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, "Maximum entropy inverse reinforcement learning," in *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008, Chicago, Illinois, USA, July 13-17, 2008*, D. Fox and C. P. Gomes, Eds., AAAI Press, 2008, pp. 1433–1438. [Online]. Available: http://www.aaai.org/Library/AAAI/2008/aaai08-227.php.

[18] S. Reddy, A. D. Dragan, and S. Levine, "SQIL: imitation learning via reinforcement learning with sparse rewards," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, 2020.

[19] D. Garg, S. Chakraborty, C. Cundy, J. Song, and S. Ermon, "Iq-learn: Inverse soft-q learning for imitation," M. Ranzato, A. Beygelzimer, Y. N. Dauphin, P. Liang, and

J. W. Vaughan, Eds., pp. 4028–4039, 2021. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/210f760a89db30aa72ca258a3483cc7f-Abstract.html.

[20] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., pp. 6379–6390, 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/68a9750337a418a86fe06c1991a1d64c-Abstract.html.

[21] J. Hu and M. P. Wellman, "Multiagent reinforcement learning: Theoretical framework and an algorithm," in *Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998), Madison, Wisconsin, USA, July 24-27, 1998*, J. W. Shavlik, Ed., Morgan Kaufmann, 1998, pp. 242–250.

[22] A. Sosic, W. R. KhudaBukhsh, A. M. Zoubir, and H. Koeppl, "Inverse reinforcement learning in swarm systems," in *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2017, São Paulo, Brazil, May 8-12, 2017*, K. Larson, M. Winikoff, S. Das, and E. H. Durfee, Eds., ACM, 2017, pp. 1413–1421. [Online]. Available: http://dl.acm.org/citation.cfm?id=3091320.

[23] R. P. Bhattacharyya, D. J. Phillips, B. Wulfe, J. Morton, A. Kuefler, and M. J. Kochenderfer, "Multi-agent imitation learning for driving simulation," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2018, Madrid, Spain, October 1-5, 2018*, IEEE, 2018, pp. 1534–1539. DOI: 10.1109/IROS.2018.8593758. [Online]. Available: https://doi.org/10.1109/IROS.2018.8593758.

[24] X. Wang and D. Klabjan, "Competitive multi-agent inverse reinforcement learning with sub-optimal demonstrations," in *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, J. G. Dy and A. Krause, Eds., ser. Proceedings of Machine Learning Research,

vol. 80, PMLR, 2018, pp. 5130–5138. [Online]. Available: http://proceedings.mlr.press/v80/wang18d.html.

[25] A. Shih, S. Ermon, and D. Sadigh, "Conditional imitation learning for multi-agent games," in *ACM/IEEE International Conference on Human-Robot Interaction, HRI 2022, Sapporo, Hokkaido, Japan, March 7 - 10, 2022*, D. Sakamoto, A. Weiss, L. M. Hiatt, and M. Shiomi, Eds., IEEE / ACM, 2022, pp. 166–175. DOI: 10.1109/HRI53351.2022.9889671. [Online]. Available: https://doi.org/10.1109/HRI53351.2022.9889671.

[26] J. Song, H. Ren, D. Sadigh, and S. Ermon, "Multi-agent generative adversarial imitation learning," S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., pp. 7472–7483, 2018. [Online]. Available: https://proceedings.neurips.cc/paper/2018/hash/240c945bb72980130446fc2b40fbb8e0-Abstract.html.

[27] L. Yu, J. Song, and S. Ermon, "Multi-agent adversarial inverse reinforcement learning," in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, K. Chaudhuri and R. Salakhutdinov, Eds., ser. Proceedings of Machine Learning Research, vol. 97, 2019, pp. 7194–7201. [Online]. Available: http://proceedings.mlr.press/v97/yu19e.html.

[28] W. Jeon, P. Barde, D. Nowrouzezahrai, and J. Pineau, "Scalable multi-agent inverse reinforcement learning via actor-attention-critic," *ArXiv*, vol. abs/2002.10525, 2020. arXiv: 2002.10525. [Online]. Available: https://arxiv.org/abs/2002.10525.

[29] S. Sukhbaatar, A. Szlam, and R. Fergus, "Learning multiagent communication with backpropagation," D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, and R. Garnett, Eds., pp. 2244–2252, 2016. [Online]. Available: https://proceedings.neurips.cc/paper/2016/hash/55b1927fdafef39c48e5b73b5d61ea60-Abstract.html.

[30] J. N. Foerster, G. Farquhar, T. Afouras, N. Nardelli, and S. Whiteson, "Counterfactual multi-agent policy gradients," in *Proceedings of the Thirty-Second AAAI Conference*

on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, S. A. McIlraith and K. Q. Weinberger, Eds., AAAI Press, 2018, pp. 2974–2982. [Online]. Available: https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17193.

[31] R. J. Williams and J. Peng, "An efficient gradient-based algorithm for on-line training of recurrent network trajectories," *Neural Comput.*, vol. 2, no. 4, pp. 490–501, 1990. DOI: 10.1162/neco.1990.2.4.490. [Online]. Available: https://doi.org/10.1162/neco.1990.2.4.490.

[32] K. Cho, B. van Merrienboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, A. Moschitti, B. Pang, and W. Daelemans, Eds., ACL, 2014, pp. 1724–1734. DOI: 10.3115/v1/d14-1179. [Online]. Available: https://doi.org/10.3115/v1/d14-1179.

[33] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *ArXiv*, vol. abs/1707.06347, 2017. arXiv: 1707.06347. [Online]. Available: http://arxiv.org/abs/1707.06347.

[34] R. R. Paleja, A. Silva, L. Chen, and M. C. Gombolay, "Interpretable and personalized apprenticeship scheduling: Learning interpretable scheduling policies from heterogeneous user demonstrations," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds.,

2020. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/477bdb55b231264bb53a7942fd84254d-Abstract.html.

[35] O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev, J. Oh, D. Horgan, M. Kroiss, I. Danihelka, A. Huang, L. Sifre, T. Cai, J. P. Agapiou, M. Jaderberg, A. S. Vezhnevets, R. Leblond, T. Pohlen, V. Dalibard, D. Budden, Y. Sulsky, J. Molloy, T. L. Paine, Ç. Gülçehre, Z. Wang, T. Pfaff, Y. Wu, R. Ring, D. Yogatama, D. Wünsch, K. McKinney, O. Smith, T. Schaul, T. P. Lillicrap, K. Kavukcuoglu, D. Hassabis, C. Apps, and D. Silver, "Grandmaster level in starcraft II using multi-agent reinforcement learning," *Nat.*, vol. 575, no. 7782, pp. 350–354, 2019. DOI: 10.1038/s41586-019-1724-z. [Online]. Available: https://doi.org/10.1038/s41586-019-1724-z.

[36] R. Jena, C. Liu, and K. P. Sycara, "Augmenting GAIL with BC for sample efficient imitation learning," in *4th Conference on Robot Learning, CoRL 2020, 16-18 November 2020, Virtual Event / Cambridge, MA, USA*, J. Kober, F. Ramos, and C. J. Tomlin, Eds., ser. Proceedings of Machine Learning Research, vol. 155, PMLR, 2020, pp. 80–90. [Online]. Available: https://proceedings.mlr.press/v155/jena21a.html.

[37] A. Singh, T. Jain, and S. Sukhbaatar, "Learning when to communicate at scale in multiagent cooperative and competitive tasks," in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*, OpenReview.net, 2019. [Online]. Available: https://openreview.net/forum?id=rye7knCqK7.

[38] E. Seraj, Z. Wang, R. R. Paleja, D. Martin, M. Sklar, A. Patel, and M. C. Gombolay, "Learning efficient diverse communication for cooperative heterogeneous teaming," in *21st International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2022, Auckland, New Zealand, May 9-13, 2022*, P. Faliszewski, V. Mascardi, C. Pelachaud, and M. E. Taylor, Eds., International Foundation for Autonomous Agents

and Multiagent Systems (IFAAMAS), 2022, pp. 1173–1182. DOI: 10.5555/3535850.3535981. [Online]. Available: https://www.ifaamas.org/Proceedings/aamas2022/pdfs/p1173.pdf.

[39] E. Seraj, X. Wu, and M. C. Gombolay, "Firecommander: An interactive, probabilistic multi-agent environment for joint perception-action tasks," *ArXiv*, vol. abs/2011.00165, 2020. arXiv: 2011.00165. [Online]. Available: https://arxiv.org/abs/2011.00165.

[40] L. Chen, R. R. Paleja, and M. C. Gombolay, "Learning from suboptimal demonstration via self-supervised reward regression," in *4th Conference on Robot Learning, CoRL 2020, 16-18 November 2020, Virtual Event / Cambridge, MA, USA*, J. Kober, F. Ramos, and C. J. Tomlin, Eds., ser. Proceedings of Machine Learning Research, vol. 155, PMLR, 2020, pp. 1262–1277. [Online]. Available: https://proceedings.mlr.press/v155/chen21b.html.

[41] J. Schulman, P. Moritz, S. Levine, M. I. Jordan, and P. Abbeel, "High-dimensional continuous control using generalized advantage estimation," in *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2016. [Online]. Available: http://arxiv.org/abs/1506.02438.

A Hyperparameters

Table 1: Hyperparameters

Name	Value
hidden layer dimensionality	64 (easy), 256 (moderate/hard)
rollout steps	4096
T-BPTT [31] segment length	8
segments per minibatch	8 (easy), 32 (moderate/hard)
total minibatch size	64 (easy), 256 (moderate/hard)
PPO [33] clipping ϵ	0.2
PPO epochs	3
learning rate	$[10^{-4}, 10^{-3}]$
discount factor	0.99
GAE [41] lambda	0.5
Reconstruction coefficient (λ_1)	0.1 (easy), 0.01 (moderate/hard)
BC coefficient (λ_2)	0 (easy), $[10^{-1.5}, 10^{0}]$ (moderate/hard)
discriminator learning rate	10^{-5} (10^{-3} for centralized discriminator)
max gradient norm	5.0

We tuned hyperparameters via grid search up to a resolution of 2 steps per decade.

B Expert Heuristics

Table 2: Expert Heuristic Performance (episode length)

Difficulty	Predator-Prey	Predator-Capture-Prey	FireCommander
5×5	8.573 ± 2.175	9.677 ± 2.628	14.439 ± 8.712
10×10	12.221 ± 3.017	14.763 ± 3.858	16.160 ± 8.247
20×20	24.915 ± 5.512	27.701 ± 6.617	24.213 ± 13.721

For the communication heuristic, we used a baseline approach in which each agent generates a one-hot encoded representation as a function of their last k observations. To embed the observations into a one-hot vector, we simply perform K-means clustering over the observations in the demonstration dataset for each environment, and map each observation to the index of the nearest cluster center. For $k > 1$, we instead perform this procedure over the last k observation vectors, concatenated. We find $k = 2$ to have the best performance.