Robust Asymmetric Learning in POMDPs

Andrew Warrington [*1] **J. Wilder Lavington** [*2 3] **Adam Ścibior** [2 3] **Mark Schmidt** [2 4] **Frank Wood** [2 3 5]

Abstract

Policies for partially observed Markov decision processes can be efficiently learned by imitating expert policies learned using asymmetric information. Unfortunately, existing approaches for this kind of imitation learning have a serious flaw: the expert does not know what the trainee cannot see, and may therefore encourage actions that are sub-optimal or unsafe under partial information. To address this flaw, we derive an update that, when applied iteratively to an expert, maximizes the expected reward of the trainee's policy. Using this update, we construct a computationally efficient algorithm, adaptive asymmetric DAgger (A2D), that jointly trains the expert and trainee policies. We then show that A2D allows the trainee to safely imitate the modified expert, and outperforms policies learned either by imitating a fixed expert or direct reinforcement learning.

1. Introduction

Consider the stochastic shortest path problem (Bertsekas & Tsitsiklis, 1991) where an agent learns to cross a frozen lake while avoiding patches of weak ice. The agent can either cross the ice directly, or take the longer, safer route circumnavigating the lake. The agent is provided with aerial images of the lake, which include color variations at patches of weak ice. To cross the lake, the agent must learn to identify its own position, goal position, and the location of weak ice from the images. Even for this simple environment, high-dimensional inputs and sparse rewards can make learning a suitable policy computationally expensive and sample inefficient. Therefore one might instead efficiently learn, in simulation, an omniscient *expert*, conditioned on a low-dimensional vector which fully describes the state of the world, to complete the task. A *trainee*, observing only images, can then learn to mimic the actions of the expert using sample-efficient online imitation learning (Ross et al., 2011). This yields a high-performing trainee, conditioned on images, learned with fewer environment interactions overall compared to direct reinforcement learning (RL).

While appealing, this approach can fail in environments where the expert has access to information unavailable to the agent, referred to as *asymmetric information*. Consider instead that the image of the lake does not indicate the location of the weak ice. The trainee now operates under increased uncertainty. This results in a different optimal partially observing policy, as the agent should now circumnavigate the lake. However, imitating the expert forces the trainee to always cross the lake, despite being unable to locate and avoid the weak ice. Even though the expert is optimal under full information, the supervision provided to the trainee through imitation learning is poor and yields a policy that is not optimal under partial information. The key insight is that *the expert has no knowledge of what the trainee does not know*. Therefore, the expert cannot provide suitable supervision, and proposes actions that are not robust to the increased uncertainty under partial information. The main algorithmic contribution we present follows from this insight: the *expert* must be refined based on the behavior of the *trainee* imitating it.

Building on this insight, we present a new algorithm: adaptive asymmetric DAgger (A2D), illustrated in Figure 1. A2D extends imitation learning by refining the expert policy, such that the resulting supervision moves the trainee policy closer to the optimal *partially observed* policy. This allows us to safely take advantage of asymmetric information in imitation learning. Crucially, A2D can be easily integrated with a variety of different RL algorithms, does not require any pretrained artifacts, policies or example trajectories, and does not take computationally expensive and high-variance RL steps in the trainee policy network.

We first introduce asymmetric imitation learning (AIL). AIL uses an expert, conditioned on full state information, to supervise learning a trainee, conditioned on partial information. We show that the solution to the AIL objective is a posterior inference over the true state; and provide sufficient conditions for when the expert is guaranteed to provide cor-

[*]Equal contribution [1]Department of Engineering Science, University of Oxford [2]Department of Computer Science, University of British Columbia [3]Inverted AI [4]Alberta Machine Learning Intelligence Institute (AMII) [5]Montréal Institute for Learning Algorithms (MILA). Correspondence to: Andrew Warrington <andreww@robots.ox.ac.uk>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

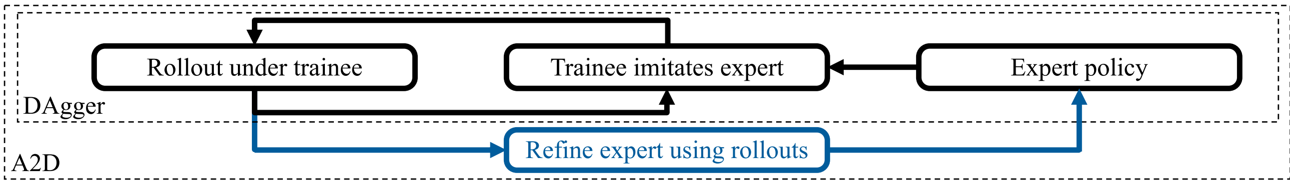


Figure 1: Flow chart describing adaptive asymmetric DAgger (A2D), introduced in this work, which builds on DAgger (Ross et al., 2011) by further refining the expert conditioned on the trainee's policy.

rect supervision. Using these insights, we then derive the theoretical A2D update to the expert policy parameters in terms of Q functions. This update maximizes the reward of the trainee implicitly defined through AIL. We then modify this update to use Monte Carlo rollouts and GAE (Schulman et al., 2015b) in place of Q functions, thereby reducing the dependence on function approximators.

We apply A2D to two pedagogical gridworld environments, and an autonomous vehicle scenario, where AIL fails. We show A2D recovers the optimal partially observed policy with fewer samples, lower computational cost, and less variance compared to similar methods. These experiments demonstrate the efficacy of A2D, which makes learning via imitation and reinforcement safer and more efficient, even in difficult high dimensional control problems such as autonomous driving. Code and additional materials are available at `https://github.com/plai-group/a2d`.

2. Background

2.1. Optimality & MDPs

An MDP, $\mathcal{M}_\Theta(\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T}_0, \mathcal{T}, \Pi_\Theta)$, is defined as a random process which produces a sequence $\tau_t := \{a_t, s_t, s_{t+1}, r_t\}$, for a set of states $s_t \in \mathcal{S}$, actions $a_t \in \mathcal{A}$, initial state $p(s_0) \in \mathcal{T}_0$, transition dynamics $p(s_{t+1}|s_t, a_t) \in \mathcal{T}$, reward function $r_t : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}$, and policy $\pi_\theta \in \Pi_\Theta : \mathcal{S} \rightarrow \mathcal{A}$ parameterized by $\theta \in \Theta$. The generative model, shown in Figure 2, for a finite horizon process is defined as:

$$q_{\pi_\theta}(\tau) = p(s_0) \prod\nolimits_{t=0}^{T} p(s_{t+1}|s_t, a_t)\pi_\theta(a_t|s_t). \quad (1)$$

We denote the marginal distribution over state $s_t \in \mathcal{S}$ at time t as $q_{\pi_\theta}(s_t)$. The objective of RL is to recover the policy which maximizes the expected cumulative reward over a trajectory, $\theta^* = \arg\max_{\theta \in \Theta} \mathbb{E}_{q_{\pi_\theta}}[\sum_{t=0}^{T} r_t(s_t, a_t, s_{t+1})]$. We consider an extension of this, instead maximizing the non-stationary, infinite horizon discounted return:

$$\theta^* = \arg\max\nolimits_{\theta \in \Theta} \ \mathbb{E}_{d^{\pi_\theta}(s)\pi_\theta(a|s)}[Q^{\pi_\theta}(a, s)], \quad (2)$$

$$\text{where } \ d^{\pi_\theta}(s) = (1-\gamma)\sum\nolimits_{t=0}^{\infty} \gamma^t q_{\pi_\theta}(s_t = s), \quad (3)$$

$$Q^{\pi_\theta}(a, s) = \mathop{\mathbb{E}}_{p(s'|s,a)}\Big[r(s, a, s') + \gamma \mathop{\mathbb{E}}_{\pi_\theta(a'|s')}[Q^{\pi_\theta}(a', s')]\Big], \quad (4)$$

where $d^{\pi_\theta}(s)$ is referred to as the *state occupancy* (Agarwal et al., 2020), and the *Q function*, Q^π, defines the expected discounted sum of rewards ahead given a state-action pair.

2.2. State Estimation and POMDPs

A POMDP extends an MDP by observing a random variable $o_t \in \mathcal{O}$, dependent on the state, $o_t \sim p(\cdot|s_t)$, instead of the state itself. The policy then samples actions conditioned on all previous observations and actions: $\pi_\phi(a_t|a_{0:t-1}, o_{0:t})$. In practice, a *belief state*, $b_t \in \mathcal{B}$, is constructed from $(a_{0:t-1}, o_{0:t})$, as an estimate of the underlying state. The policy, $\pi_\phi \in \Pi_\Phi : \mathcal{B} \rightarrow \mathcal{A}$, is then conditioned on this belief state (Doshi-Velez et al., 2013; Igl et al., 2018; Kaelbling et al., 1998). The resulting stochastic process, denoted $\mathcal{M}_\Phi(\mathcal{S}, \mathcal{O}, \mathcal{B}, \mathcal{A}, \mathcal{R}, \mathcal{T}_0, \mathcal{T}, \Pi_\Phi)$, generates a sequence of tuples $\tau_t = \{a_t, b_t, o_t, s_t, s_{t+1}, r_t\}$. As before, we wish to find a policy, $\pi_{\phi^*} \in \Pi_\Phi$, which maximizes the expected cumulative reward under the generative model:

$$q_{\pi_\phi}(\tau) = p(s_0) \prod\nolimits_{t=0}^{T} p(s_{t+1}|s_t, a_t) \times \\ p(b_t|b_{t-1}, o_t, a_{t-1}) p(o_t|s_t) \pi_\phi(a_t|b_t). \quad (5)$$

It is common to instead condition the policy on the last w observations and $w-1$ actions (Laskin et al., 2020a; Murphy, 2000), i.e. $b_t := (a_{t-w:t-1}, o_{t-w:t})$, rather than using the potentially infinite dimensional random variable (Murphy, 2000), defined recursively in Figure 2. This "windowed" belief state representation is used throughout this paper.

We also note that q_π is used to denote the distribution over trajectories under the subscripted policy ((1) and (5) for $\pi_\theta(\cdot|s_t)$ and $\pi_\phi(\cdot|b_t)$ respectively). The occupancies $d^{\pi_\phi}(s)$ and $d^{\pi_\phi}(b)$ define marginals of $d^{\pi_\phi}(s, b)$ in a partially observed processes (as in (3)). Later we discuss *MDP-POMDP pairs*, defined as an MDP and a POMDP with identical state transition dynamics, reward generating functions and initial state distributions. However, these process pairs can, and often do, have different optimal policies. This discrepancy is the central issue addressed in this work.

2.3. Imitation Learning

Imitation learning (IL) assumes access to either an expert policy capable of solving a task, or example trajectories gen-

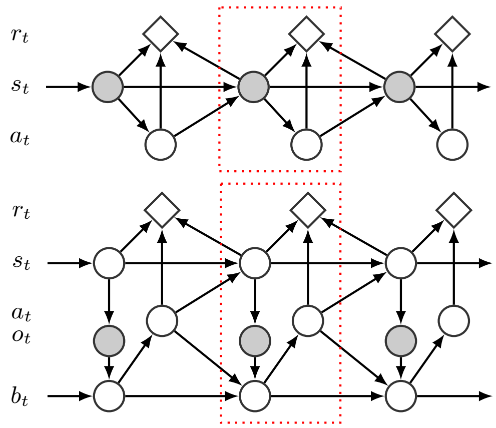


Figure 2: Graphical models of an MDP (top) and a POMDP (bottom) with identical initial and state transition dynamics, $p(s_t|s_{t-1}, a_t)$, $p(s_0)$, and reward function $R(s_t, a_t, s_{t+1})$.

erated by such an expert. Given example trajectories, the *trainee* is learned by regressing onto the actions of the expert. However, this approach can perform arbitrarily poorly for states not in the training set (Laskey et al., 2017). Alternatively, online IL (OIL) algorithms, such as DAgger (Ross et al., 2011), assume access to an expert that can be queried at any state. DAgger rolls out under a mixture of the expert π_θ and trainee π_ϕ policies, denoted π_β. The trainee is then updated to replicate the experts' actions at the visited states:

$$\phi^* = \arg\min_{\phi \in \Phi} \mathbb{E}_{d^{\pi_\beta}(s)} \left[\mathbb{KL} \left[\pi_\theta(a|s) || \pi_\phi(a|s) \right] \right], \quad (6)$$
$$\text{where} \quad \pi_\beta(a|s) = \beta \pi_\theta(a|s) + (1-\beta)\pi_\phi(a|s). \quad (7)$$

The coefficient β is annealed to zero during training. This provides supervision in states visited by the trainee, thereby avoiding compounding out of distribution error which grows with time horizon (Ross et al., 2011; Sun et al., 2017). While IL provides higher sample efficiency than RL, it requires an expert or expert trajectories, and is thus not always applicable. A trainee learned using IL from an imperfect expert can perform arbitrarily poorly (Sun et al., 2017), even in OIL. Addition of asymmetry in OIL can cause similar failures.

2.4. Asymmetric Information

In many simulated environments, additional information is available during training that is not available at test time. This additional *asymmetric information* can often be exploited to accelerate learning (Choudhury et al., 2018; Pinto et al., 2017; Vapnik & Vashist, 2009). For example, Pinto et al. (2017) exploit asymmetry to learn a policy conditioned on noisy image-based observations which are available at test time, but where the value function (or *critic*), is conditioned on a compact and noiseless state representation, only available during training. The objective function for this *asymmetric actor critic* (Pinto et al., 2017) algorithm is:

$$J(\phi) = \mathbb{E}_{d^{\pi_\phi}(s,b)} \left[\mathbb{E}_{\pi_\phi(a|b)} \left[A^{\pi_\phi}(s,a) \right] \right], \quad (8)$$
$$Q^{\pi_\phi}(a,s) = \mathbb{E}_{p(s'|s,a)} \left[r(s,a,s') + \gamma V^{\pi_\phi}(s') \right], \quad (9)$$
$$V^{\pi_\phi}(s) = \mathbb{E}_{\pi_\phi(a|b)} \left[Q^{\pi_\phi}(a,s) \right], \quad (10)$$

where the *asymmetric advantage* is defined as $A^{\pi_\phi}(s,a) = Q^{\pi_\phi}(a,s) - V^{\pi_\phi}(s)$, and $V^{\pi_\phi}(s)$ is the *asymmetric value function*. Asymmetric methods often outperform "symmetric" RL as $Q^{\pi_\phi}(a,s)$ and $V^{\pi_\phi}(s)$ are simpler to tune, train, and provide lower-variance gradient estimates.

Asymmetric information has also been used in a variety of other scenarios, including policy ensembles (Sasaki & Yamashina, 2021; Song et al., 2019), imitating attention-based representations (Salter et al., 2019), multi-objective RL (Schwab et al., 2019), direct state reconstruction (Nguyen et al., 2020), or privileged information dropout (Kamienny et al., 2020; Lambert et al., 2018). Failures induced by asymmetric information have also been discussed. Arora et al. (2018) identify an environment where a particular method fails. Choudhury et al. (2018) use asymmetric information to improve policy optimization in model predictive control, but do not solve scenarios such as "the trapped robot problem," referred to later as Tiger Door (Littman et al., 1995), and solved below. Notably, asymmetric environments are naturally suited to OIL (AIL) (Pinto et al., 2017):

$$\phi^* = \arg\min_{\phi} \ \mathbb{E}_{d^{\pi_\beta}(s,b)} \left[\mathbb{KL} \left[\pi_\theta(a|s) || \pi_\phi(a|b) \right] \right], \quad (11)$$
$$\text{where} \quad \pi_\beta(a|s,b) = \beta \pi_\theta(a|s) + (1-\beta)\pi_\phi(a|b). \quad (12)$$

As the expert is not used at test time, AIL can take advantage of asymmetry to simplify learning (Pinto et al., 2017) or enable data augmentation (Chen et al., 2020). However, naive application of AIL can yield trainees that perform arbitrarily poorly. Further work has addressed learning from imperfect experts (Ross & Bagnell, 2014; Sun et al., 2017; Meng et al., 2019), but does not consider issues arising from the use of asymmetric information. We demonstrate, analyze, and then address both of these issues in the following sections.

3. AIL as Posterior Inference

We begin by analyzing the AIL objective in (12). We first show that the optimal trainee defined by this objective can be expressed as posterior inference over state conditioned on the expert policy. This posterior inference is defined as:

Definition 1 (Implicit policy)**.** *For any state-conditional policy $\pi_\theta \in \Pi_\Theta$ and any belief-conditional policy $\pi_\eta \in \Pi_\Phi$ we define $\hat{\pi}_\theta^\eta \in \hat{\Pi}_\Theta$ as the implicit policy of π_θ under π_η as:*

$$\hat{\pi}_\theta^\eta(a|b) := \mathbb{E}_{d^{\pi_\eta}(s|b)} \left[\pi_\theta(a|s) \right], \quad (13)$$

When $\pi_\eta = \hat{\pi}_\theta^\eta$, we refer to this policy as the implicit policy of π_θ, denoted as just $\hat{\pi}_\theta$.

Note that a policy, or policy set, with a hat (e.g. $\hat{\pi}_\theta$), indicates that the policy or set is implicitly defined through composition of the original policy (e.g. π_θ) and the expectation defined in (13). The implicit policy defines a posterior predictive density, marginalizing over the uncertainty over state. We can then show that the solution to the AIL objective in (12) (for $\beta = 0$) is equivalent to the implicit policy:

Theorem 1 (Asymmetric IL target)**.** *For any fully observing policy π_θ and fixed policy π_η, and assuming $\hat{\Pi}_\Theta \subseteq \Pi_\Phi$, then the implicit policy $\hat{\pi}_\theta^\eta$, defined in Definition 1, minimizes the AIL objective:*

$$\hat{\pi}_\theta^\eta = \underset{\pi \in \Pi_\Phi}{\arg\min}\, \mathbb{E}_{d^{\pi_\eta}(s,b)} \left[\mathbb{KL} \left[\pi_\theta(a|s) || \pi(a|b) \right] \right]. \quad (14)$$

Proof. An extended proof is included in Appendix C.

$$\begin{aligned} &\mathbb{E}_{d^{\pi_\eta}(s,b)} \left[\mathbb{KL} \left[\pi_\theta(a|s) || \pi(a|b) \right] \right] \\ &= -\mathbb{E}_{d^{\pi_\eta}(b)} \left[\mathbb{E}_{d^{\pi_\eta}(s)} \left[\mathbb{E}_{\pi_\theta(a|s)} \left[\log \pi(a|b) \right] \right] \right] + K \\ &= -\mathbb{E}_{d^{\pi_\eta}(b)} \left[\mathbb{E}_{\hat{\pi}_\theta^\eta(a|b)} \left[\log \pi(a|b) \right] \right] + K \\ &= \mathbb{E}_{d^{\pi_\eta}(b)} \left[\mathbb{KL} \left[\hat{\pi}_\theta^\eta(a|b) || \pi(a|b) \right] \right] + K' \end{aligned}$$

Since $\hat{\pi}_\theta^\eta \in \Pi_\Phi$, it follows that

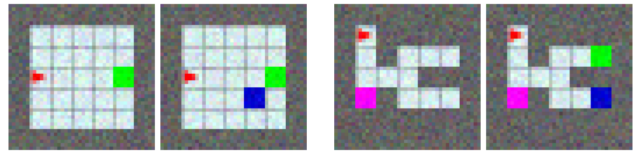
$$\begin{aligned} \hat{\pi}_\theta^\eta &= \underset{\pi \in \Pi_\Phi}{\arg\min} \underset{d^{\pi_\eta}(b)}{\mathbb{E}} \left[\mathbb{KL} \left[\hat{\pi}_\theta^\eta(a|b) || \pi(a|b) \right] \right] && (15) \\ &= \underset{\pi \in \Pi_\Phi}{\arg\min} \underset{d^{\pi_\eta}(s,b)}{\mathbb{E}} \left[\mathbb{KL} \left[\pi_\theta(a|s) || \pi(a|b) \right] \right]. \quad \square && (16) \end{aligned}$$

Theorem 1 shows that the implicit policy compactly defines the solution to the AIL objective. This allows us to specify the dependence of the learned trainee through AIL on the expert policy. We will in turn leverage this solution to derive the update applied to the expert parameters. We note that this definition and theorem are closely related to a result also derived by Weihs et al. (2020).

However, drawing multiple state samples from a single conditional occupancy, $d^{\pi_\eta}(s \mid b)$, is not generally tractable without access to a model of $\mathcal{T}$ and $\mathcal{T}_0$. This is because sampling from $d^{\pi_\eta}(s \mid b)$ requires resampling multiple trajectories that include the specified belief state b, which cannot be done through direct environment interaction. Therefore, generating the samples required to integrate (13) is not generally tractable. We are, however, able to draw samples from the joint occupancy, $d^{\pi_\eta}(s, b)$, simply by rolling out under π_η. Therefore, in practice, AIL instead learns a variational approximation to the implicit policy, $\pi_\psi \in \Pi_\Psi : \mathcal{B} \rightarrow \mathcal{A}$, by minimizing the following objective:

$$F(\psi) = \underset{d^{\pi_\eta}(s,b)}{\mathbb{E}} \left[\mathbb{KL} \left[\pi_\theta(a|s) || \pi_\psi(a|b) \right] \right], \quad (17)$$

$$\nabla_\psi F(\psi) = -\underset{d^{\pi_\eta}(s,b)}{\mathbb{E}} \left[\underset{\pi_\theta(a|s)}{\mathbb{E}} \left[\nabla_\psi \log \pi_\psi(a|b) \right] \right]. \quad (18)$$



(a) Frozen Lake.

(b) Tiger Door.

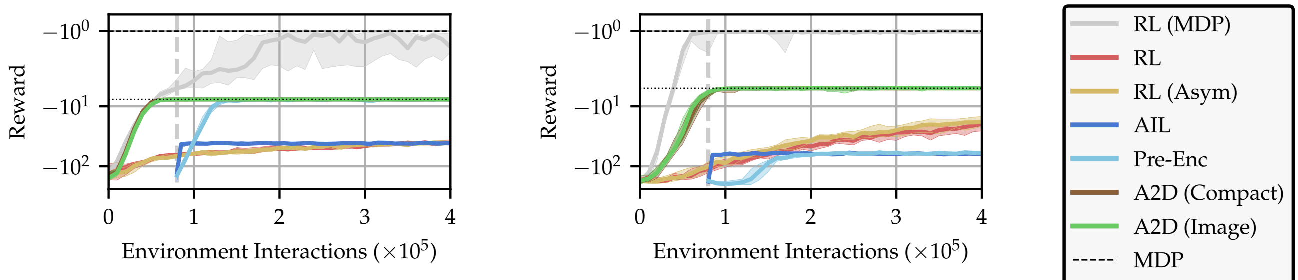
Figure 3: The two gridworlds we study. An agent (red) must navigate to the goal (green) while avoiding the hazard (blue). Shown are the raw, noisy 42×42 pixel observations available to the agent. The expert is conditioned on an omniscient compact state vector indicating the position of the goal and hazard. In Frozen Lake, the trainee is conditioned on the left image and cannot see the hazard. In Tiger Door, pushing the button (pink) illuminates the hazard.

Crucially, this approach only requires samples from the *joint* occupancy. This avoids sampling from the *conditional* occupancy, as required to directly solve (13). If the variational family is sufficiently expressive, there exists a $\pi_\psi \in \Pi_\Psi$ for which the divergence between the implicit policy and variational approximation is zero. In OIL, it is common to sample under the trainee policy by setting $\pi_\eta = \pi_\psi$, thereby defining a fixed point equation. Under sufficient expressivity and exact updates, an iteration solving this fixed point equation converges to the implicit policy (see Appendix C). In practice, this iterative scheme converges even in the presence of inexact updates and restricted policy classes.

4. Failure of Asymmetric Imitation Learning

We now reason about the failure of AIL in terms of *reward*. The crucial insight is that to guarantee that the reward earned by the trainee policy is optimal, the divergence between expert and trainee must go to exactly zero. The reward earned by policies with even a small (but finite) divergence may be arbitrarily low. This condition, referred to as *identifiability*, is formalized below. We leverage this condition in Section 5 to derive the update applied to the expert which guarantees the optimal partially observed policy is recovered under the assumptions specified by each theorem, and discussed in further detail in Appendix C.

However, to first motivate and explore this behavior, we introduce two pedagogical environments, referred to as "Frozen Lake" and "Tiger Door" (Littman et al., 1995; Spaan, 2012), illustrated in Figure 3. Both require an agent to navigate to a goal while avoiding hazards. The trainee is conditioned on an image of the environment where the hazard is not initially visible. The expert is conditioned on an omniscient compact state vector. Taking actions, reaching the goal, and hitting the hazard incurs rewards of -2, 20, and -100 respectively. In Frozen Lake, the hazard (weak ice) is in a random location in the interior nine squares. In Tiger



(a) Frozen Lake.

(b) Tiger Door.

Figure 4: Results for the gridworld environments. Median and quartiles across 20 random seeds are shown. TRPO (Schulman et al., 2015a) is used for RL methods. Broken lines indicate the optimal reward, normalized so the optimal MDP reward is -1 (*MDP*). All agents and trainees are conditioned on a image-based input, except *A2D (Compact)* which is conditioned on a partial compact state representation. All experts, and *RL (MDP)*, are conditioned on an omniscient compact state. *Pre-Enc* uses a fixed pretrained image encoder, trained on examples from the MDP. *AIL* and *Pre-Enc* begin when the MDP has converged, as this is the required expenditure for training. A2D is the only method that reliably and efficiently finds the optimal POMDP policy, and, in a sample budget commensurate with *RL (MDP)*. The convergence of A2D is also similar for *both* image-based (*A2D (Image)*) and compact (*A2D (Compact)*) representations, highlighting that we have effectively subsumed the image perception task. Configurations, additional results and discussions are included in the appendix.

Door, the agent can detour via a button, incurring additional negative reward, to reveal the goal location.

We show results for application of AIL, and comparable RL approaches, to these environments in Figure 4. These confirm our intuitions: RL in the MDP (*RL (MDP)*) is stable and efficient, and proceeds directly to the goal, earning maximum rewards of 10.66 and 6. Direct RL in the POMDP (*RL* and *RL (Asym)*) does not converge to a performant policy in the allocated computational budget. AIL (*AIL*) converges almost immediately, but, to a trainee that averages over expert actions. In Frozen Lake, this trainee averages the expert over the location of the weak patch, never circumnavigates the lake, and instead crosses directly, incurring an average reward of -26.6. In Tiger Door, the trainee proceeds directly to a possible goal location without pressing the button, incurring an average reward of -54. Both solutions represent catastrophic failures. Instead, the trainee should circumnavigate the lake, or, push the button and then proceed to the goal, earning rewards of 4 and 2 respectively.

These results, and insight from Theorem 1, lead us to define two important properties which provide guarantees on the performance of AIL:

Definition 2 (Identifiable Policies)**.** *Given an MDP-POMDP pair $\{\mathcal{M}_\Theta, \mathcal{M}_\Phi\}$, an MDP policy $\pi_\theta \in \Pi_\Theta$, and POMDP policy $\pi_\phi \in \Pi_\Phi$, we describe $\{\pi_\theta, \pi_\phi\}$ as an* ***identifiable policy pair*** *if and only if $\mathbb{E}_{d^{\pi_\phi}(s,b)}\left[\mathbb{KL}\left[\pi_\theta(a|s)||\pi_\phi(a|b)\right]\right] = 0$.*

Definition 3 (Identifiable Processes)**.** *If each optimal MDP policy, $\pi_{\theta^*} \in \Pi_{\Theta^*}$, and the corresponding implicit policy, $\hat{\pi}_{\theta^*} \in \hat{\Pi}_{\Theta^*}$, form an identifiable policy pair, then we define $\{\mathcal{M}_\Theta, \mathcal{M}_\Phi\}$ as an* ***identifiable process pair****.*

Identifiable policy pairs enforce that the partially observing implicit policy, recovered through application of AIL, can *exactly* reproduce the actions of the fully observing policy. These policies are therefore guaranteed to incur the same reward. Identifiable processes then extends this definition, requiring that such an identifiable policy pair exists for all optimal fully observing policies. Using this definition, we can then show that performing AIL using any optimal fully observing policy on an identifiable process pair is guaranteed to recover an optimal partially observing policy:

Theorem 2 (Convergence of AIL)**.** *For any identifiable process pair defined over sufficiently expressive policy classes, under exact intermediate updates, the iteration defined by:*

$$\psi_{k+1} = \underset{\psi \in \Psi}{\arg\min} \underset{d^{\pi_{\psi_k}}(s,b)}{\mathbb{E}} \left[\mathbb{KL}\left[\pi_{\theta^*}(a|s)||\pi_\psi(a|b)\right]\right], \quad (19)$$

where π_{θ^*} is an optimal fully observed policy, converges to an optimal partially observed policy, $\pi_{\psi^*}(a|b)$, as $k \rightarrow \infty$.

Proof. See Appendix C. □

Therefore, identifiability of processes defines a sufficient condition to guarantee that any optimal expert policy provides asymptotically unbiased supervision to the trainee. If a process pair is identifiable, then AIL recovers the optimal partially observing policy, and garners a reward equal to the fully observing expert. When processes are not identifiable, the divergence between expert and trainee is non-zero, and the *reward* garnered by the trainee can be arbitrarily sub-optimal (as in the gridworlds above). Unfortunately, identifiability of two processes represents a strong assumption, unlikely to hold in practice. Therefore, we propose an extension that modifies the *expert* on-line, such that the modified expert policy and corresponding implicit policy pair form an identifiable *and* optimal policy pair under partial information. This modification, in turn, guarantees that the expert provides asymptotically correct AIL supervision.

5. Correcting AIL with Expert Refinement

We now use the insight from Sections 3 and 4 to construct an update, applied to the expert policy, which improves the expected reward ahead under the implicit policy. Crucially, this update is designed such that, when interleaved with AIL, the optimal partially observed policy is recovered. We refer to this iterative algorithm as adaptive asymmetric DAgger (A2D). To derive the update to the expert, π_θ, we first consider the RL objective under the implicit policy, $\hat{\pi}_\theta$:

$$J(\theta) = \mathop{\mathbb{E}}_{d^{\hat{\pi}_\theta}(b)\hat{\pi}_\theta(a|b)} \left[Q^{\hat{\pi}_\theta}(a,b) \right], \quad \text{where} \tag{20}$$

$$Q^{\hat{\pi}_\theta}(a,b) = \mathop{\mathbb{E}}_{p(b',s',s|a,b)} \left[r(s,a,s') + \gamma \mathop{\mathbb{E}}_{\hat{\pi}_\theta(a'|b')} \left[Q^{\hat{\pi}_\theta}(a',b') \right] \right].$$

This objective defines the cumulative reward of the trainee in terms of the parameters of the expert policy. This means that maximizing $J(\theta)$ maximizes the reward obtained by the implicit policy, and ensures proper expert supervision:

Theorem 3 (Convergence of Exact A2D)**.** *Under exact intermediate updates, the following iteration converges to an optimal partially observed policy $\pi_{\psi^*}(a|b) \in \Pi_\Psi$, provided both $\Pi_{\Phi^*} \subseteq \hat{\Pi}_{\Theta^*} \subseteq \Pi_\Psi$:*

$$\psi_{k+1} = \mathop{\arg\min}_{\psi \in \Psi} \mathop{\mathbb{E}}_{d^{\pi_{\psi_k}}(s,b)} \left[\mathbb{KL}\left[\pi_{\hat{\theta}^*}(a|s) || \pi_\psi(a|b) \right] \right], \tag{21}$$

$$\textit{where} \quad \hat{\theta}^* = \mathop{\arg\max}_{\theta \in \Theta} \mathop{\mathbb{E}}_{\hat{\pi}_\theta(a|b) d^{\pi_{\psi_k}}(b)} \left[Q^{\hat{\pi}_\theta}(a,b) \right]. \tag{22}$$

Proof. See Appendix C. □

First, an inner optimization, defined by (22), maximizes the expected reward of the implicit policy by updating the parameters of the *expert* policy, under the current trainee policy. The outer optimization, defined by (21), then updates the trainee policy by projecting onto the updated implicit policy defined by the updated expert. This projection is performed by minimizing the divergence to the updated expert, as per Theorem 1.

Unfortunately, directly differentiating through $Q^{\hat{\pi}_\theta}$, or even sampling from $\hat{\pi}_\theta$, is intractable. We therefore optimize a surrogate reward instead, denoted $J_\psi(\theta)$, that defines a lower bound on the objective function in (22). This surrogate is defined as the expected reward ahead under the variational trainee policy Q^{π_ψ}. By maximizing this surrogate objective, we maximize a lower bound on the possible improvement to the implicit policy with respect to the parameters of the expert:

$$\max_{\theta \in \Theta} J_\psi(\theta) = \max_{\theta \in \Theta} \mathop{\mathbb{E}}_{\hat{\pi}_\theta(a|b) d^{\pi_\psi}(b)} \left[Q^{\pi_\psi}(a,b) \right] \tag{23}$$

$$\leq \max_{\theta \in \Theta} J(\theta) = \max_{\theta \in \Theta} \mathop{\mathbb{E}}_{\hat{\pi}_\theta(a|b) d^{\pi_\psi}(b)} \left[Q^{\hat{\pi}_\theta}(a,b) \right]. \tag{24}$$

To verify this inequality, first note that we assume that the implicit policy is capable of maximizing the expected reward ahead at every belief state (c.f. Theorem 3). Therefore, by definition, replacing the implicit policy, $\hat{\pi}_\theta$, with any *behavioral policy*, here π_ψ, cannot yield *larger* returns when maximized over θ (see Appendix C). Replacement with a behavioral policy is a common analysis technique, especially in policy gradient (Schulman et al., 2015a; 2017; Sutton, 1992) and policy search methods (see §4,5 of Bertsekas (2019) and §2 of Deisenroth et al. (2013)). This surrogate objective permits the following REINFORCE gradient estimator, where we define $f_\theta = \log \pi_\theta(a \mid s)$:

$$\begin{aligned} \nabla_\theta J_\psi(\theta) &= \nabla_\theta \, \mathbb{E}_{\hat{\pi}_\theta(a|b) d^{\pi_\psi}(b)} \left[Q^{\pi_\psi}(a,b) \right] \qquad (25) \\ &= \mathbb{E}_{d^{\pi_\psi}(b)} \left[\nabla_\theta \, \mathbb{E}_{d^{\pi_\psi}(s|b)} \left[\mathbb{E}_{\pi_\theta(a|s)} \left[Q^{\pi_\psi}(a,b) \right] \right] \right] \\ &= \mathbb{E}_{d^{\pi_\psi}(s,b)} \left[\mathbb{E}_{\pi_\theta(a|s)} \left[Q^{\pi_\psi}(a,b) \nabla_\theta f_\theta \right] \right] \\ &= \mathbb{E}_{d^{\pi_\psi}(s,b)\pi_\psi(a|b)} \left[\frac{\pi_\theta(a|s)}{\pi_\psi(a|b)} Q^{\pi_\psi}(a,b) \nabla_\theta f_\theta \right]. \qquad (26) \end{aligned}$$

Equation (26) defines an importance weighted policy gradient, evaluated using states sampled under the variational agent, which is equal to the gradient of the implicit policy reward with respect to the expert parameters. For (26) to provide an unbiased gradient estimate we (unsurprisingly) require an unbiased estimate of $Q^{\pi_\psi}(a,b)$. While, this estimate can theoretically be generated by directly learning the Q function using a universal function approximator, in practice, learning the Q function is often challenging. Furthermore, the estimator in (26) is *strongly* dependent on the quality of the approximation. As a result, imperfect Q function approximations yield biased gradient estimates.

This strong dependency has led to the development of RL algorithms that use Monte Carlo estimates of the Q function instead. This circumvents the cost, complexity and bias induced by approximating Q, by leveraging these rollouts to provide unbiased, although higher variance, estimates of the Q function. Techniques such as generalized advantage estimation (GAE) (Schulman et al., 2015b) allow bias and variance to be traded off. However, as a direct result of asymmetry, using Monte Carlo rollouts in A2D can bias the gradient estimator. Full explanation of this is somewhat involved, and so we defer discussion to Appendix B. However, we note that for most *environments* this bias is small and can be minimized through tuning the parameters of GAE.

The final gradient estimate used in A2D is therefore:

$$\nabla_\theta J_\psi(\theta) = \mathop{\mathbb{E}}_{\substack{d^{\pi_\beta}(s_t,b_t)\\ \pi_\beta(a_t|s_t,b_t)}} \left[\frac{\pi_\theta(a_t|s_t)}{\pi_\beta(a_t|s_t,b_t)} \hat{A}^{\pi_\beta} \nabla_\theta f_\theta \right], \quad (27)$$

$$\text{where} \quad \hat{A}^{\pi_\beta}(a_t, s_t, b_t) = \sum\nolimits_{t=0}^{\infty} (\gamma\lambda)^t \delta_t, \quad (28)$$

$$\text{and} \quad \delta_t = r_t + \gamma V^{\pi_\beta}(s_{t+1}, b_{t+1}) - V^{\pi_\beta}(s_t, b_t), \quad (29)$$

where (28) and (29) describe GAE (Schulman et al., 2015b). Similar to DAgger, we also allow A2D to interact under a mixture policy, $\pi_\beta(a|s,b) = \beta\pi_\theta(a|s) + (1-\beta)\pi_\psi(a|b)$, with Q and value functions defined as $Q^{\pi_\beta}(a,s,b)$ and $V^{\pi_\beta}(a,s,b)$ similarly. However, as was also suggested by (Ross et al., 2011), we found that aggressively annealing β, or even setting $\beta = 0$ immediately, often provided the best results. The full A2D algorithm, also shown in Algorithm 1, is implemented by repeating three individual steps:

1. **Gather data** (Alg. 1, Ln 8): Collect samples from $q_{\pi_\beta}(\tau)$ by rolling out under the mixture policy, as defined in (5).
2. **Refine Expert** (Alg. 1, Ln 11): Update expert policy parameters, θ, with importance weighted policy gradient as estimated in (27). This step also updates the trainee and expert value function parameters, ν_p and ν_m.
3. **Update Trainee** (Alg. 1, Ln 12): Perform an AIL step to fit the (variational) trainee policy parameters, ψ, to the expert policy using (18).

As the gradient used in A2D, defined in (27), is a REINFORCE-based gradient estimate, it is compatible with any REINFORCE-based policy gradient method, such as TRPO or PPO (Schulman et al., 2015a; 2017). Furthermore, A2D does not require pretrained experts or example trajectories. In the experiments we present, all expert and trainee policies are learned from scratch. Although using A2D with pretrained expert policies is possible, such pipelined approaches are susceptible to suboptimal local minima.

Algorithm 1 Adaptive Asymmetric DAgger (A2D)

1: **Input:** MDP $\mathcal{M}_\Theta$, POMDP $\mathcal{M}_\Phi$, Annealing schedule `AnnealBeta`(n, β).
2: **Return:** Variational trainee parameters ψ.
3: $\theta, \psi, \nu_m, \nu_p, \leftarrow$ `InitNets`$(\mathcal{M}_\Theta, \mathcal{M}_\Phi)$
4: $\beta \leftarrow 1, \ D \leftarrow \emptyset$
5: **for** $n = 0, \ \ldots, \ N$ **do**
6: $\quad \beta \leftarrow$ `AnnealBeta`(n, β)
7: $\quad \pi_\beta \leftarrow \beta\pi_\theta + (1-\beta)\pi_\psi$
8: $\quad \mathcal{T} = \{\tau_i\}_{i=1}^{\mathcal{I}} \sim q_{\pi_\beta}(\tau)$
9: $\quad D \leftarrow$ `UpdateBuffer`$(D, \mathcal{T})$
10: $\quad V^{\pi_\beta} \leftarrow \beta V_{\nu_m}^{\pi_\theta} + (1-\beta) V_{\nu_p}^{\pi_\psi}$
11: $\quad \theta, \nu_m, \nu_p \leftarrow$ `RLStep`$(\mathcal{T}, V^{\pi_\beta}, \pi_\beta)$
12: $\quad \psi \leftarrow$ `AILStep`$(D, \pi_\theta, \pi_\psi)$
13: **end for**

Algorithm 1: Adaptive asymmetric DAgger (A2D) algorithm. Additional steps we introduce beyond DAgger (Ross et al., 2011) are highlighted in blue, and implement the feedback loop in Figure 1. `RLStep` is a policy gradient step, updating the expert, using the gradient estimator in (27). `AILStep` is an AIL variational policy update, as in (18).

6. Experiments

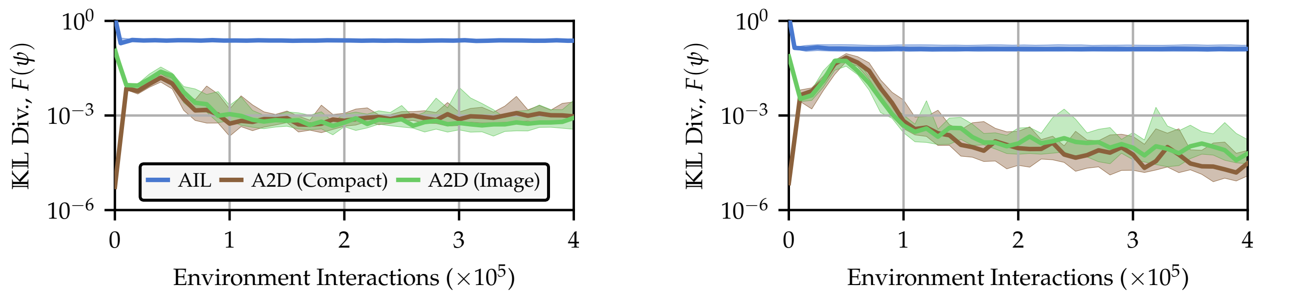
6.1. Revisiting Frozen Lake & Tiger Door

We evaluate A2D on the gridworlds introduced in Section 3. The results are shown in Figures 4 and 5. Figure 4 shows that A2D converges to the optimal POMDP reward in a similar number of environment interactions as the best-possible convergence (*RL (MDP)*), whereas the other methods fail for one, or both, gridworlds. Similar convergence rates are observed for both high-dimensional images (*A2D (Image)*) and low-dimensional compact representations (*A2D (Compact)*). We note that many of the hyperparameters are largely consistent between A2D and RL in the MDP, which is easy to tune. However, A2D did often benefit from increased entropy regularization and reduced λ (see Appendix B). The IL hyperparameters are largely independent of the RL hyperparameters, further simplifying tuning overall.

Figure 5 shows the divergence between the expert and trainee policies during learning. *AIL* saturates to a high divergence, indicating that the trainee is unable to replicate the expert. The divergence in A2D increases initially, as the expert learns using the full-state information. This rise is due to the non-zero value of β, imperfect function approximation, slight bias in the gradient estimator, and the tendency of the expert to initially move towards a higher reward policy not representable under the agent. As the learning develops, and $\beta \rightarrow 0$, the expert is forced to optimize the reward of the trainee. This, in turn, drives the divergence towards zero, producing a policy that can be represented by the agent. A2D has therefore created an identifiable expert and implicit policy pair (Definition 2), where the implicit policy is also optimal under partial information.

6.2. Safe Autonomous Vehicle Learning

Autonomous vehicle (AV) simulators (Dosovitskiy et al., 2017; Wymann et al., 2014; Kato et al., 2015) allow safe virtual exploration of driving scenarios that would be unsafe to explore in real life. The inherent complexity of training AV controllers makes exploiting efficient AIL an attractive opportunity (Chen et al., 2020). The expert can be provided with the exact state of other actors, such as other vehicles, occluded hazards and traffic lights. The trainee is then provided with sensor measurements available in the real world, such as camera feeds, lidar and the egovehicle telemetry.



(a) Frozen Lake. (b) Tiger Door.

Figure 5: The evolution of the policy divergence, $F(\psi)$. Shown are median and quartiles across 20 random seeds. *AIL* converges to a high divergence, whereas A2D achieves a low divergence for both representations, indicating that the trainee recovered by A2D is faithfully imitating the expert (see Figure 4 for more information).

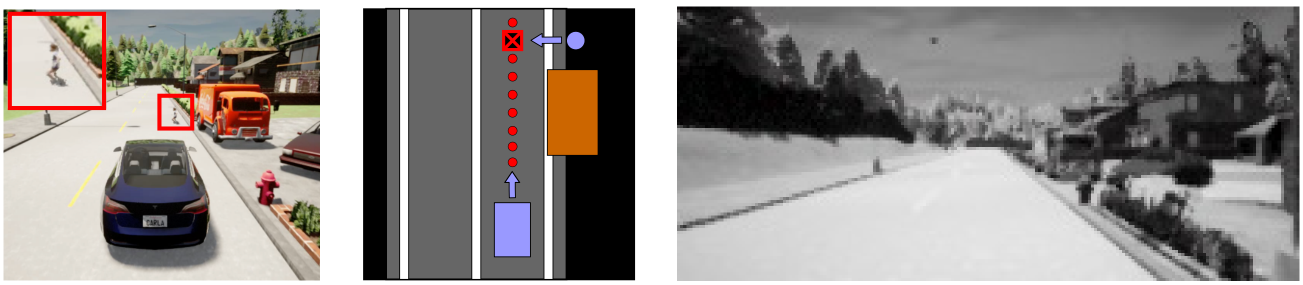


Figure 6: Visualizations of the AV scenario. Left: third-person view showing the egovehicle and child running out. Center: top-down schematic of the environment and asymmetric information. Right: front-view camera input provided to the agent.

The safety-critical aspects of asymmetry are highlighted in context of AVs. Consider a scenario where a child may dart into the road from behind a parked truck, illustrated in Figure 6. The expert, aware of the position and velocity of the child from asymmetric information, will only brake if there is a child, and will otherwise proceed at high speed. However, the trainee is unable to distinguish between these scenarios, before the child emerges from, just the front-facing camera. As the expected expert behavior is to accelerate, the implicit policy also accelerates. The trainee only starts to brake once the child is visible, by which time it is too late to guarantee the child is not struck. The expert should therefore proceed at a lower speed so it can slow down or evade the child once visible. This cannot be achieved by naive application of AIL.

We implement this scenario in the CARLA simulator (Dosovitskiy et al., 2017), which is visualized in Figure 6. A child is present in 50% of trials, and, if present, emerges with variable velocity. The action space consists of the steering angle and amount of throttle/brake. As an approximation to the optimal policy under privileged information, we used a hand-coded expert that completes the scenario driving at the speed limit if the child is absent, and slows down when approaching the truck if the child is present. The differentiable expert is a small neural network, operating on a six-dimensional state vector that fully describes the simulator state. The agent is a convolutional neural network that operates on grayscale images from the front-view camera.

Results comparing A2D to four baselines are shown in Figure 7. *RL (MDP)* uses RL to learn a policy conditioned on the omniscient compact state, only available in simulation, and hence does not yield a usable agent policy. This represents the absolute best-case convergence for an RL method, achieving good, although not optimal, performance quickly and reliably. *RL* learns an agent conditioned on the camera image, yielding poor, high-variance results within the experimental budget. *AIL* uses asymmetric DAgger to imitate the hand-coded expert using the camera image, learning quickly, but converging to a sub-optimal solution. We also include *OIL (MDP)*, which learns a policy conditioned on the omniscient state by imitating a hand-coded expert, and converges quickly to the near-optimal solution (*MDP*). As expected, *A2D* learns more slowly than AIL, since RL is used to update to the expert, but achieves higher reward than *AIL* and avoids collisions. This scenario, as well as any future asymmetric baselines, are distributed in the repository.

7. Discussion

In this work we have discussed learning policies in POMDPs. Partial information and high-dimensional observations can make direct application of RL expensive and

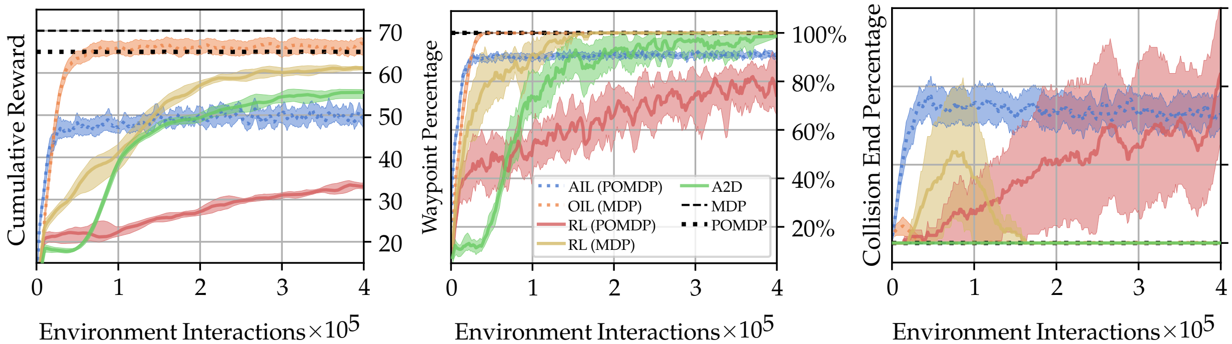


Figure 7: Performance metrics for the AV scenario, introduced in Section 6. We show median and quartiles across ten random seeds. Left: average cumulative reward. Center: average percentage of waypoints collected, measuring progress along route. Right: percentage of trajectories ending in a child collision. Optimal MDP and POMDP solutions are shown by dashed and dotted lines respectively. In methods marked as MDP the agent uses an omniscient compact state, including the child's state. AIL (*AIL (MDP)*) and RL (*RL (MDP)*) learn a performant (high reward and waypoint percentage, low collision percentage) policy quickly and reliably. In methods marked as POMDP the agent uses the high-dimensional monocular camera view. Therefore, *AIL* leads to a high collision, and the perception task makes RL in the POMDP (*RL (POMDP)*) slow and variable (low reward and waypoint percentage, high collision percentage). *A2D* solves the scenario (high reward and waypoint percentage, low collision percentage) in a budget commensurate with the best-case convergence of *RL (MDP)*.

unreliable. Asymmetric learning uses additional information to improve performance beyond comparable symmetric methods. Asymmetric IL can efficiently learn a partially observing policy by imitating an omniscient expert. However, this approach requires a pre-existing expert, and, critically, assumes that the expert can provide suitable supervision – a condition we formalize as identifiability. The learned trainee can perform arbitrarily poorly when this is not satisfied. We therefore develop adaptive asymmetric DAgger (A2D), which adapts the expert policy such that AIL can efficiently recover the optimal partially observed policy. A2D also allows the expert to be learned online with the agent, and hence does not require any pretrained artifacts.

There are three notable extensions of A2D. The first extension is investigating more conservative updates for the expert and trainee which take into consideration the limitations or approximate nature of each intermediate update. The second extension is studying the behavior of A2D in environments where the expert is not omniscient, but observes a superset of the environment relative to the agent. The final extension is integrating A2D into differentiable planning methods, exploiting the low dimensional state vector to learn a latent dynamics model, or, improve sample efficiency in sparse reward environments.

We conclude by outlining under what conditions the methods discussed in this paper may be most applicable. If a pretrained expert or example trajectories are available, AIL provides an efficient methodology that should be investigated first, but, that may fail catastrophically. If the observed dimension is small, and no reliable expert is available, direct application of RL is likely to perform well. If the observed dimension is large, and trajectories which adequately cover the state-space are available, then pretraining an image encoder can provide a competitive and flexible approach. Finally, if a compact state representation is available alongside a high dimensional observation space, A2D offers an alternative that is robust and expedites training in high-dimensional and asymmetric environments.

8. Acknowledgements

We thank Frederik Kunstner for invaluable discussions and reviewing preliminary drafts; and the reviewers for their feedback and improvements to the paper. AW is supported by the Shilston Scholarship, University of Oxford. JWL is supported by Mitacs grant IT16342. We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), the Canada CIFAR AI Chairs Program, and the Intel Parallel Computing Centers program. This material is based upon work supported by the United States Air Force Research Laboratory (AFRL) under the Defense Advanced Research Projects Agency (DARPA) Data Driven Discovery Models (D3M) program (Contract No. FA8750-19-2-0222) and Learning with Less Labels (LwLL) program (Contract No.FA8750-19-C-0515). Additional support was provided by UBC's Composites Research Network (CRN), Data Science Institute (DSI) and Support for Teams to Advance Interdisciplinary Research (STAIR) Grants. This research was enabled in part by technical support and computational resources provided by WestGrid (https://www.westgrid.ca/) and Compute Canada (www.computecanada.ca).

References

Achille, A. and Soatto, S. Information dropout: Learning optimal representations through noisy computation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2897–2905, 2018.

Agarwal, A., Kakade, S. M., Lee, J. D., and Mahajan, G. Optimality and approximation with policy gradient methods in Markov decision processes. In *Proceedings of Thirty Third Conference on Learning Theory*. PMLR, 2020.

Andrychowicz, O. A. M., Baker, B., Chociej, M., Józefowicz, R., McGrew, B., Pachocki, J., Petron, A., Plappert, M., Powell, G., Ray, A., Schneider, J., Sidor, S., Tobin, J., Welinder, P., Weng, L., and Zaremba, W. Learning dexterous in-hand manipulation. *International Journal of Robotics Research*, 39(1):3–20, 2020.

Arora, S., Choudhury, S., and Scherer, S. Hindsight is only 50/50: Unsuitability of mdp based approximate pomdp solvers for multi-resolution information gathering. *arXiv preprint arXiv:1804.02573*, 2018.

Bertsekas, D. P. Approximate policy iteration: A survey and some new methods. *Journal of Control Theory and Applications*, 9(3):310–335, 2011.

Bertsekas, D. P. *Reinforcement learning and optimal control*. Athena Scientific Belmont, MA, 2019.

Bertsekas, D. P. and Tsitsiklis, J. N. An analysis of stochastic shortest path problems. *Mathematics of Operations Research*, 16(3):580–595, 1991.

Biewald, L. Experiment Tracking with Weights and Biases, 2020. Software available from wandb.com.

Chen, D., Zhou, B., Koltun, V., and Krähenbühl, P. Learning by cheating. In *Conference on Robot Learning*, pp. 66–75. PMLR, 2020.

Chevalier-Boisvert, M., Willems, L., and Pal, S. Minimalistic gridworld environment for OpenAI gym. `https://github.com/maximecb/gym-minigrid`, 2018.

Choudhury, S., Bhardwaj, M., Arora, S., Kapoor, A., Ranade, G., Scherer, S., and Dey, D. Data-driven planning via imitation learning. *The International Journal of Robotics Research*, 37(13-14):1632–1672, 2018.

Deisenroth, M. P., Neumann, G., Peters, J., et al. A survey on policy search for robotics. *Foundations and trends in Robotics*, 2(1-2):388–403, 2013.

Doshi-Velez, F., Pfau, D., Wood, F., and Roy, N. Bayesian nonparametric methods for partially-observable reinforcement learning. *IEEE transactions on pattern analysis and machine intelligence*, 37(2):394–407, 2013.

Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., and Koltun, V. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, pp. 1–16, 2017.

Engstrom, L., Ilyas, A., Santurkar, S., Tsipras, D., Janoos, F., Rudolph, L., and Madry, A. Implementation matters in deep rl: A case study on ppo and trpo. In *International Conference on Learning Representations*, 2020.

Finn, C., Tan, X. Y., Duan, Y., Darrell, T., Levine, S., and Abbeel, P. Deep spatial autoencoders for visuomotor learning. *Proceedings - IEEE International Conference on Robotics and Automation*, 2016-June:512–519, 2016.

Igl, M., Zintgraf, L., Le, T. A., Wood, F., and Whiteson, S. Deep variational reinforcement learning for POMDPs. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2117–2126. PMLR, 2018.

Kaelbling, L. P., Littman, M. L., and Cassandra, A. R. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2):99–134, 1998.

Kamienny, P.-A., Arulkumaran, K., Behbahani, F., Boehmer, W., and Whiteson, S. Privileged information dropout in reinforcement learning. *arXiv:2005.09220*, 2020.

Kang, B., Jie, Z., and Feng, J. Policy optimization with demonstrations. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*. PMLR, 2018.

Kato, S., Takeuchi, E., Ishiguro, Y., Ninomiya, Y., Takeda, K., and Hamada, T. An Open Approach to Autonomous Vehicles. *IEEE Micro*, 35(6):60–68, 2015.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Könönen, V. Asymmetric multiagent reinforcement learning. *Web Intelligence and Agent Systems: An international journal*, 2(2):105–121, 2004.

Lambert, J., Sener, O., and Savarese, S. Deep learning under privileged information using heteroscedastic dropout. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8886–8895, 2018.

Laskey, M., Lee, J., Fox, R., Dragan, A., and Goldberg, K. Dart: Noise injection for robust imitation learning. *arXiv preprint arXiv:1703.09327*, 2017.

Laskin, M., Lee, K., Stooke, A., Pinto, L., Abbeel, P., and Srinivas, A. Reinforcement learning with augmented data. *arXiv preprint arXiv:2004.14990*, 2020.

Laskin, M., Srinivas, A., and Abbeel, P. Curl: Contrastive unsupervised representations for reinforcement learning. *Proceedings of the 37th International Conference on Machine Learning, Vienna, Austria, PMLR 119*, 2020. arXiv:2004.04136.

Levine, S., Finn, C., Darrell, T., and Abbeel, P. End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17:1–40, 2016.

Littman, M. L., Cassandra, A. R., and Kaelbling, L. P. Learning policies for partially observable environments: Scaling up. *Seventh International Conference on Machine Learning*, pp. 362–370, 1995.

Maei, H. R., Szepesvari, C., Bhatnagar, S., Precup, D., Silver, D., and Sutton, R. S. Convergent temporal-difference learning with arbitrary smooth function approximation. In *NIPS*, pp. 1204–1212, 2009.

Meng, Z., Li, J., Zhao, Y., and Gong, Y. Conditional teacher-student learning. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6445–6449. IEEE, 2019.

Murphy, K. P. A survey of POMDP solution techniques. *environment*, 2:X3, 2000.

Nguyen, H., Daley, B., Song, X., Amato, C., and Platt, R. Belief-grounded networks for accelerated robot learning under partial observability. *arXiv preprint arXiv:2010.09170*, 2020.

Pinto, L., Andrychowicz, M., Welinder, P., Zaremba, W., and Abbeel, P. Asymmetric actor critic for image-based robot learning. *arXiv preprint arXiv:1710.06542*, 2017.

Ross, S. and Bagnell, J. A. Reinforcement and imitation learning via interactive no-regret learning. *arXiv preprint arXiv:1406.5979*, 2014.

Ross, S., Gordon, G. J., and Bagnell, J. A. A reduction of imitation learning and structured prediction to no-regret online learning. *Journal of Machine Learning Research*, 15:627–635, 2011.

Salter, S., Rao, D., Wulfmeier, M., Hadsell, R., and Posner, I. Attention-privileged reinforcement learning. *arXiv preprint arXiv:1911.08363*, 2019.

Sasaki, F. and Yamashina, R. Behavioral cloning from noisy demonstrations. In *International Conference on Learning Representations*, 2021.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. Trust region policy optimization. In *International conference on machine learning*, pp. 1889–1897, 2015.

Schulman, J., Moritz, P., Levine, S., Jordan, M., and Abbeel, P. High-dimensional continuous control using generalized advantage estimation. *arXiv:1506.02438*, 2015.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Schwab, D., Springenberg, J. T., Martins, M. F., Neunert, M., Lampe, T., Abdolmaleki, A., Hertweck, T., Hafner, R., Nori, F., and Riedmiller, M. A. Simultaneously learning vision and feature-based control policies for real-world ball-in-a-cup. In *Robotics: Science and Systems XV*, 2019.

Song, J., Lanka, R., Yue, Y., and Ono, M. Co-training for policy learning. *35th Conference on Uncertainty in Artificial Intelligence*, 2019.

Spaan, M. T. J. *Partially Observable Markov Decision Processes*, pp. 387–414. Springer Berlin Heidelberg, 2012. ISBN 978-3-642-27645-3.

Sun, W., Venkatraman, A., Gordon, G. J., Boots, B., and Bagnell, J. A. Deeply AggreVaTeD: Differentiable imitation learning for sequential prediction. In *Proceedings of the 34th International Conference on Machine Learning*. PMLR, 2017.

Sun, W., Bagnell, J. A., and Boots, B. Truncated horizon policy search: Combining reinforcement learning & imitation learning. *6th International Conference on Learning Representations*, pp. 1–14, 2018.

Sutton, R. *Reinforcement Learning*. The Springer International Series in Engineering and Computer Science. Springer US, 1992. ISBN 9780792392347.

Vapnik, V. and Vashist, A. A new learning paradigm: Learning using privileged information. *Neural networks*, 22 (5-6):544–557, 2009.

Weihs, L., Jain, U., Salvador, J., Lazebnik, S., Kembhavi, A., and Schwing, A. Bridging the imitation gap by adaptive insubordination. *arXiv preprint arXiv:2007.12173*, 2020.

Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

Wymann, B., Espie, C. G., Dimitrakakis, C., Coulom, R., and Sumner, A. TORCS: The Open Racing Car Simulator, 2014.

Yarats, D., Kostrikov, I., and Fergus, R. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. In *International Conference on Learning Representations*, 2021.