A GENERALISABILITY OF POLICY GRADIENT BASED RL ALGORITHMS

Policy gradient based RL algorithms minimise the cumulative discounted cost by directly optimising the policy parameters using gradient descent. We discuss in this section the ability of these algorithms to search for η-optimal policies. We distinguish two types of policy gradient approaches:

OPTIMISING THE GLOBAL OBJECTIVE DIRECTLY:

Many methods (such as Deterministic Policy Gradient Silver et al. (2014), and variants of the REINFORCE algorithm Williams (1992); Degris et al. (2012)) are rooted in the policy gradient theorem Sutton et al. (1999):

$$\nabla \mathbb{E}_{p_0,\pi}[\sum_t \gamma^t c(s_t,a_t)] = \int_{s_0,s} p_0(s_0)\rho_\pi(s|s_0)\mathbb{E}_{a\sim\pi}[Q^c_\pi(s,a)\nabla \log \pi(a|s)]$$
$$\text{with } \rho_\pi(s|s_0) = \sum_t \gamma^t \mathbb{P}_\pi(s_t = s|s_0)$$

where the policy π is a function of some parameter θ and all gradients are implicitly with respect to θ. In order to adapt these approaches for the search of η-optimal policies, the policy updates must take into account the future state distribution derivative. In fact, the gradient of the generalised criterion with respect to the policy induces an additional term as provided in proposition 5:

Proposition 5 *For any given distribution η:*

$$\nabla \mathbb{E}^\eta_{p_0,\pi}[\sum_t \gamma^t c(s_t,a_t)] = \underbrace{\int_{s_0,s_+} p_0(s_0) v^c_\pi(s_+)\nabla P^\eta_\pi(s_+|s_0)}_{\textbf{\textit{additional term}}}$$
$$+ \underbrace{\int_{s_0,s_+,s} p_0(s_0)P^\eta_\pi(s_+|s_0)\rho_\pi(s|s_+)\mathbb{E}_\pi[Q^c_\pi(s,a)\nabla \log \pi_\theta(a|s)]}_{\textbf{\textit{modified term}}}$$

where $P^\eta_\pi(s_+|s_0) = \sum_{n=0}^\infty \eta(n)\mathbb{P}_\pi(s_n = s_+|s_0)$.

Notice that the modified term has the same form as the original policy gradient theorem. This is not an issue as it's a matter of adapting the used estimators in practice. However, the additional term is not taken into account. Furthermore, in this current form, $\nabla P^\eta_\pi(s_+|s_0)$ is not tractable. This implies that current policy gradient approaches that rely on the policy gradient theorem can not search in a reliable way for η-optimal policies.

OPTIMISING A LOCAL VERSION OF THE GLOBAL OBJECTIVE:

On the other hand, recent policy gradient algorithms such as Trust Region Policy Optimisation (TRPO) Schulman et al. (2015) and Proximal Policy Optimisation (PPO) Schulman et al. (2017) iteratively search for a new policy π_n that improves the performances of an old policy π_o by optimising a local approximation of the right hand term in the following identity Kakade & Langford (2002):

$$\begin{aligned}\mathcal{L}^{\delta_0}_0(\pi_n,c) =& \mathcal{L}^{\delta_0}_0(\pi_o,c) + \mathbb{E}_{p_0,\pi_n}[\sum_t \gamma^t A^c_{\pi_o}(s_t,a_t)] \\ =& \mathcal{L}^{\delta_0}_0(\pi_o,c) + \int_{s_0,s} p_0(s_0)\rho_{\pi_n}(s|s_0)\int_a \pi_n(a|s)A^c_{\pi_o}(s,a)\end{aligned}$$

where $\mathcal{L}^{\delta_0}_0$ is the unregulated loss function, and $A^c_\pi(s,a)$ is the advantage function:

$$A^c_\pi(s,a) = Q^c_\pi(s,a) - v^c_\pi(s)$$

In principle, this approach is tractable for the search of η-optimal policies. In fact, the generalised setting verifies a similar formulation of this identity:

Proposition 6 *For any given distribution η:*

$$\begin{aligned}\mathcal{L}_0^\eta(\pi_n, c) &= \mathcal{L}_0^\eta(\pi_o, c) + \mathbb{E}^\eta_{p_0,\pi_n}[\sum_t \gamma^t A^c_{\pi_o}(s_t, a_t)] \\ &= \mathcal{L}_0^\eta(\pi_o, c) + \int_{s_0,s} p_0(s_0)\rho_{\pi_n}(s|s_0)\mathbb{E}^\eta_{\pi_n}[A^c_{\pi_o}(s_+, a_+)|s] \\ \textit{with} \qquad \mathbb{E}^\eta_{\pi_n}[A^c_{\pi_o}(s_+, a_+)|s] &= \int_{s_+,a_+} P^\eta_{\pi_n}(s_+, a_+|s) A^c_{\pi_o}(s_+, a_+)\end{aligned}$$

Propositions 6 lay the ground to adapt proximal policy gradient based RL algorithms to the search of η-optimal policies. However, we do not investigate this further in this paper as we focus on the Inverse problem.

B GENERALISATION OF CLASSICAL IRL ALGORITHMS

In this section, we discuss particular classical penalisation functions Ω and Ψ that lead to generalisation of well known IRL algorithms. In all cases, Ω is always chosen as the entropy regulariser, as in state of the art RL algorithms.

Let $\mathcal{C}$ be a subset of admissible cost functions, and the penalisation function defined as:

$$\psi(c) = \imath_{\mathcal{C}}(c) = \begin{cases} 0 & \textit{if } c \in \mathcal{C} \\ +\infty & \textit{if } c \notin \mathcal{C} \end{cases}$$

Two particular subsets are studied in details in the following as they lead to generalisations of classical IRL algorithms.

B.1 EMMA: EXPECTATION MATCHING - MAXIMUM ENTROPY

First, consider the set of linear interpolation of some finite basis set function $\{f_i(s,a),\ i \in \mathcal{I}\}$, i.e.,

$$\mathcal{C}_{\textit{linear}} = \Big\{ \sum_{i\in\mathcal{I}} w_i f_i, \ \ \textit{such that } \|w\|_2 \leq 1 \Big\}.$$

In the classical, non-generalised IRL problem (with $\eta = \delta_0$), this problem coincides with features expectation matching IRL algorithm Abbeel & Ng (2004), that minimises the l_2 expected feature vectors Ho & Ermon (2016):

$$L(\pi, c) + \Omega(\pi) = \max_{c\in\mathcal{C}_{\textit{linear}}} \mathbb{E}_{\rho_\pi}[c(s,a)] - \mathbb{E}_{\rho_{\bar\pi}}[c(s,a)] = \big\|\mathbb{E}_{\rho_\pi}[f] - \mathbb{E}_{\rho_{\bar\pi_E}}[f]\big\|_2,$$

where $f(s,a) = (f_i(s,a))_{i\in\mathcal{I}}$. Generalising this algorithm for any geometric distribution η simply consists in substituting the expectation with $\mathbb{E}^\eta_\pi$:

Proposition 7 *Under the assumptions of Proposition 2, and for $\psi = \imath_{\mathcal{C}_{\textit{linear}}}$, it holds that:*

$$\mathrm{RL}^\eta_\Omega \circ \mathrm{IRL}^\eta_\psi(\bar\pi) = \arg\min_\pi -\Omega(\pi) + \big\|\mathbb{E}^\eta_\pi[f] - \mathbb{E}^\eta_{\pi_E}[f]\big\|_2$$

The generalised version of this algorithm (which is actually $\mathrm{GIRL}_{\imath_{\mathcal{C}_{\textit{linear}}},H,\eta}$) that we called EMMA_η, is derived in the following as a generalisation of expectation matching IRL algorithm Abbeel & Ng (2004).

The optimal cost function c^*_π (given a previously learned policy π) must satisfy the following equality[4]:

$$\int_{s_0} p_0(s_0)\big[\mu_\pi(s_+, a_+|s_0) - \mu_{\pi_E}(s_+, a_+|s_0)\big] c^*_\pi(s_+, a_+) = \Big\|\mathbb{E}_{\mu_\pi}[f] - \mathbb{E}_{\mu_{\pi_E}}[f]\Big\|_2 \qquad (4)$$

[4] c.f. the proof of proposition 7 in Appendix F

As the objective solution is known, we propose to replace the cost optimisation step in the template procedure we provide in Algorithm 1 with the following simple quadratic loss:

$$\mathcal{L}_{\text{linear}}(w) = \Big(\mathbb{E}_{\mu_\pi}[wf^T] - \mathbb{E}_{\mu_{\pi_E}}[wf^T] - \|\mathbb{E}_{\mu_\pi}[f] - \mathbb{E}_{\mu_{\pi_E}}[f]\|_2\Big)^2 \tag{5}$$

Given that the set of feasible cost function is convex, we can update the loss using projected gradient updates. We also use an approximation of this loss in practice:

$$\bar{\mathcal{L}}_{\text{linear}}(w) = \Big(\sum_{S^+,A^+} wf^T(s,a) - \sum_{S_E^+,A_E^+} wf^T(s,a) - \|\sum_{S^+,A^+} f(s,a) - \sum_{S_E^+,A_E^+} f(s,a)\|_2\Big)^2 \tag{6}$$

The algorithm we propose is then defined as follows:

Algorithm 3 EMMA

1: **Input:** Expert trajectories $\tau_E \sim \pi_E$, initial policy π_{θ_0} and initial cost function w_0
2: **for** $e \in [1, N]$ **do**
3: Sample trajectories $\tau \sim \pi_{\theta_i}$
4: Sample states randomly $(S_t, A_t) \sim \tau$ and $(S^+, A^+) = (S_{t+k}, A_{t+k})$ where $k \sim \eta$
5: Sample states randomly $(S'_t, A'_t) \sim \tau_E$ and $(S_E^+, A_E^+) = (S'_{t+k}, A'_{t+k})$ where $k \sim \eta$
6: Update the cost weights w_i to minimise $\bar{\mathcal{L}}_{\text{linear}}(w_i)$
7: Project the cost weights on the feasible set $\mathcal{C}_{\text{linear}}$
8: Update θ_i using soft actor critic to minimise $w_{i+1}f^T$
9: **Return:** $(\pi_{\theta_N}, D_{w_N})$

B.2 WIEM: WORST INDIVIDUAL COST - ENTROPY MAXIMIZER:

We now consider convex combination of basis functions:

$$\mathcal{C}_{convex} = \Big\{\sum_{i\in\mathcal{I}} w_i f_i, \ \textit{with} \sum_{i\in\mathcal{I}} w_i = 1, \text{ and } w_i \geq 0, \forall i \in \mathcal{I}\Big\}$$

In the classical non-generalised IRL setting, this is equivalent to MWAL Syed & Schapire (2007) and LPAL Syed et al. (2008) where we minimise the worst-case excess cost among the basis functions Ho & Ermon (2016):

$$L(\pi, c) + \Omega(\pi) = \max_{c\in\mathcal{C}_{convex}} \mathbb{E}_{\rho_\pi}[c(s,a)] - \mathbb{E}_{\rho_{\bar{\pi}}}[c(s,a)] = \max_{i\in\mathcal{I}} \mathbb{E}_{\rho_\pi}[f_i] - \mathbb{E}_{\rho_{\bar{\pi}}}[f_i]$$

This setting is also simply generalised for any geometric η by takng the expectation w.r.t. $\mathbb{E}_\pi^\eta$:

Proposition 8 *Under the assumptions of Proposition 2, and for $\psi = \imath_{\mathcal{C}_{convex}}$, it holds that:*

$$\text{RL}_\Omega^\eta \circ \text{IRL}_\psi^\eta(\pi_E) = \arg\min_\pi -\Omega(\pi) + \max_i \mathbb{E}_\pi^\eta[f_i] - \mathbb{E}_{\pi_E}^\eta[f_i]$$

We derive WIEM_η in the following, which is equivalent to $\text{GIRL}_{\delta_{\mathcal{C}_{convex}},H,\eta}$ and a generalisation of worst-case excess IRL algorithms.

The optimal cost function c_π^* (given a previously learned policy π) must satisfy the following equality[5]:

$$\int_{s_0} p_0(s_0)\big[\mu_\pi(s_+, a_+|s_0) - \mu_{\pi_E}(s_+, a_+|s_0)\big]c_\pi^*(s_+, a_+) = \max_i \mathbb{E}_{\mu_\pi}[f_i] - \mathbb{E}_{\mu_{\pi_E}}[f_i] \tag{7}$$

As the objective solution is known, we propose to replace the cost optimisation step in the template procedure we provide in Algorithm 1 with the following simple quadratic loss:

$$\mathcal{L}_{\text{convex}}(w) = \Big(\mathbb{E}_{\mu_\pi}[wf^T] - \mathbb{E}_{\mu_{\pi_E}}[wf^T] - \max_i \mathbb{E}_{\mu_\pi}[f_i] - \mathbb{E}_{\mu_{\pi_E}}[f_i]\Big)^2 \tag{8}$$

[5] c.f. the proof of proposition 8 in Appendix F

Given that the set of feasible cost function is convex, we can update the loss using projected gradient updates. We also use an approximation of this loss in practice:

$$\bar{\mathcal{L}}_{\text{convex}}(w) = \Big(\sum_{S^+,A^+} wf^T(s,a) - \sum_{S_E^+,A_E^+} wf^T(s,a) - \max_i \big[\sum_{S^+,A^+} f_i(s,a) - \sum_{S_E^+,A_E^+} f_i(s,a) \big] \Big)^2 \tag{9}$$

The algorithm we propose is then defined as follows:

Algorithm 4 WIEM

1: **Input:** Expert trajectories $\tau_E \sim \pi_E$, initial policy π_{θ_0} and initial cost function w_0
2: **for** $e \in [1, N]$ **do**
3: Sample trajectories $\tau \sim \pi_{\theta_i}$
4: Sample states randomly $(S_t, A_t) \sim \tau$ and $(S^+, A^+) = (S_{t+k}, A_{t+k})$ where $k \sim \eta$
5: Sample states randomly $(S'_t, A'_t) \sim \tau_E$ and $(S_E^+, A_E^+) = (S'_{t+k}, A'_{t+k})$ where $k \sim \eta$
6: Update the cost weights w_i to minimise $\bar{\mathcal{L}}_{\text{convex}}(w_i)$
7: Project the cost weights on the feasible set $\mathcal{C}_{\text{convex}}$
8: Update θ_i using soft actor critic to minimise $w_{i+1}f^T$
9: **Return:** $(\pi_{\theta_N}, D_{w_N})$

C MULTI-TASK SETTING

Classically, the multi-task setting is defined by considering a task space Θ and for each task $\theta \in \Theta$ the associated Markov decision process $\mathcal{M}_\theta = \{\mathcal{S}, \mathcal{A}, \mathcal{P}, c_\theta, \gamma, p_0\}$. Depending on the context, the objective is then to either solve the RL or the IRL problems for the set of MDPs $(\mathcal{M}_\theta)_{\theta\in\Theta}$ by averaging the losses with respect to a task distribution $\mathcal{F}$. This is equivalent in principle to solving the problem for the MDP $\bar{\mathcal{M}} = \{\mathcal{S} \times \Theta, \bar{\mathcal{A}}, \bar{\mathcal{P}}, \bar{c}, \gamma, \bar{p}_0\}$ where for any states (s, s'), tasks (θ, θ') and action a, the following equalities hold true:

$$\begin{aligned}
\bar{\mathcal{A}}(s,\theta) &= \mathcal{A}(s) \\
\bar{\mathcal{P}}(s',\theta'|s,\theta,a) &= \mathcal{P}(s'|s,a)\delta(\theta' = \theta) \\
\bar{c}(s,\theta,a) &= c_\theta(s,a) \\
\bar{p}_0(s,\theta) &= p_0(s)\mathcal{F}(\theta)
\end{aligned}$$

We adapt the latter formulation here for the sake of coherence with previous sections.

The difficulty in multi-task settings arises from ray-interference: when the cost function encourages conflicting behaviours for different tasks, the learning objective plateaus Schaul et al. (2019). This stagnates the progress of the policy, which in turn complicates the IRL problem as these plateaus are an opportunity for the discriminator to over-fit the replay-buffer. To alleviate this issue, we propose to *augment the data-set* as proposed in Section C.1 (by using MEGAN coupled with the Idle subroutine.

C.1 IDLE : AN ON-POLICY DATA AUGMENTATION ROUTINE

If the state space $\mathcal{S}$ is very large, or even continuous, it becomes quite unlikely to encounter the same state twice in a (finite) trajectory from a given data-set. In particular, this renders quite difficult the estimation of future state distribution $P_\pi^\eta(.|s)$ (where $s \sim \rho_\pi(.|s_0)$).

To circumvent this issue, we propose to use the following on-policy data augmentation scheme, modeled as a game between a discriminator $D : (\mathcal{S} \times \mathcal{A} \times \mathcal{S}) \to [0, 1]$ and a generator $G : \mathcal{S} \to \Delta(\mathcal{S} \times \mathcal{A})$. The objective of the generator is to produce future states similar to the gathered samples while the discriminator D aims to identify true samples from generated ones, with the following score function:

$$V(D, G) = \mathbb{E}\big[\log(D(s_+, a_+|s)) + \log(1 - D(s_g, a_g|s))\big]$$

where the expectation is taken w.r.t the future state distribution $P_\pi^\eta(.|s)$ for (s_+, a_+), the generator distribution $G(.|s)$ for (s_g, a_g), and the marginal over the initial state distribution of the occupancy measure $\rho_\pi(.|s_0)$ for s. Solving this game approximates P_π^η:

Proposition 9 *$(\tilde{D}, \tilde{G}) = (\frac{1}{2}, P_\pi^\eta)$ is a Nash-equilibrium of the following zero-sum game:*

$$D^* : \min_D V(D,G) \quad \textit{and} \quad G^* : \max_G V(D,G) \tag{10}$$

From this proposition, we derive in Algorithm 5, a method to approximate future state distributions that is used as a subroutine in GIRL. This approach is theoretically feasible given an on-policy data set (such as the expert's trajectories). In practice, we noticed that approximating P_π^η using (Idle) produces reliable generators when the variance of η is relatively small, so that future state samples fall within a reduced range with a high probability. Inversely, if η has a high variance, samples from P_π^η would be along extended horizon and the (Idle) discriminator easily picks up on the parts being learned and halts the generator's improvement. This either leads to vanishing gradient updates or a mode collapse.

Algorithm 5 Idle (an on-policy future state generator)

1: **Input:** On-policy trajectories τ, initial discriminator D_{ϕ_0} and initial generator G_{ν_0}
2: **for** $e \in [1, N]$ **do**
3: Sample states randomly $(S_t, A_t) \sim \tau$
4: Sample $(S^+, A^+) = (S_{t+k}, A_{t+k})$ where $k \sim \eta$
5: Sample $(S_G^+, A_G^+) \sim G_{\nu_i}(S)$
6: Update the discriminator parameter ϕ_i to minimise:

$$\sum_{S_t, S^+, A^+} \log(D_{\phi_i}(s_+, a_+|s)) + \sum_{S_t, S_G^+, A_G^+} \log(1 - D_{\phi_i}(s_+, a_+|s))$$

7: Update the generator parameter ν_i to minimise: $\sum_{S_t} \log(D(G(s)|s))$
8: **Return:** (D_{ϕ_N}, G_{ν_N})

D Experiments for the Multi-task setting and the Idle procedure

D.1 Fetch-Reach environment

We consider in this section the *FetchReach*[6] task from the *MuJoCo* based environments Plappert et al. (2018). To evaluate the generalisability of the learned policies, we only generate expert trajectories for a subset of possible tasks (only target positions that are 5-10 cm away from the initial gripper's position[7] to be precise). We evaluate the learned policies in the *learned setting* (same horizon and same tasks) and in a *generalisability setting* (twice the training horizon and the full range of tasks). As in the simple task setting, we asses performances in terms of normalised cumulative costs.

In Figure 4a, we compare the performances of MEGAN (with and without the data augmentation) and GAIL over the training. We observe that both GAIL and MEGAN (without Idle) struggle in solving the problem. However, using the Idle generator reduces the undesirable effect of ray interference and stabilises the training. Performance wise, the learned policy using MEGAN outperforms the expert demonstrations in the training tasks while at the same time providing comparable performances in the remaining set of tasks (as provided in Figure 4b).

It is arguable that the success of MEGAN in the multi-task setting is explained with the Idle procedure. A similar approach on GAIL might be appealing. However, this is not feasible in practice. It is true that the reasoning provided in Section C.1 can be developed for any distribution η, this entails that we can use the same approach to learn a generator that mimics $\rho_\pi(.|s_0)$. Unfortunately, this is not feasible in practice (due to the high variance of ρ_π, as explained in Section C.1). The issue at play here is that

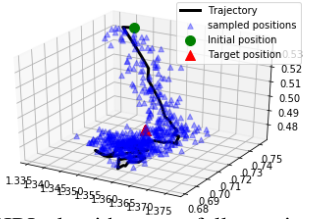
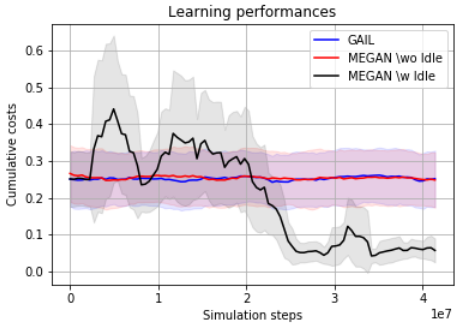


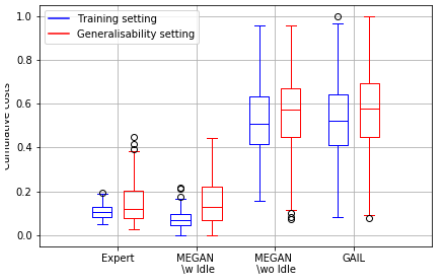
Figure 5: Idly generated samples from expert trajectory

[6] To the extent of our knowledge, this is the first reported performances of IRL algorithms on a fully continuous environment

[7] the maximum range of the arm is about 25 cm



(a) Normalised cumulative costs over training



(b) Normalised cumulative costs (best performance)

Figure 4: Performances in the Fetch Reach setting

the latter distribution (ρ_π) covers all the observations of the trajectory. Indeed, from our early empirical attempts, we noticed that learning such distribution is unstable: we either over-fit a sub-set of the trajectory (notably the stationary distribution, which is hurtful for our purpose) or we do not learn the distribution at all. On the other hand, learning P_π^η conditioned on some intermediate state is a lot easier when we choose η such that it only covers the near future transitions (for example when $\eta = \text{Geom}(0.7)$, the average prediction of $P_\pi^\eta(|s_t)$ is 3 steps in the future).

In Figure 5, we evaluate the learned approximation of $P_{\pi_E}^{\text{Geom}(0.7)}$ on a sample expert trajectory. We plot the evolution of the (true/generated) gripper position in 3D overtime. For each state encountered on the trajectory, the learned generator outputs 10 samples (in blue). Clearly, the future state generator is reliable; this is successful because the distribution η is a short term prediction: the learned generator maps current states to the possible ones in the next few steps.

D.2 Multi-task 2-D navigation

In this section we report the performances on a custom-made multi-task navigation environment. The goal is to navigate from an initial position to a target position while avoiding four lakes. The state space is constructed by concatenating the coordinates of the agent, the coordinates of the target as well as the distance from the centre of each of the four lakes. The action space is the norm 2 ball $\{x \in \mathbb{R}^2 \textbf{ s.t. } \|x\|_2 \leq 1\}$. The transition kernel is a Dirac mass at the sum of the previous position and the action vector. If the sum is within one of the lakes or outside the grid, then the new position is the projection of the previous position on the border according to the action direction. In Figure 6, we render the environment to provide an idea about the task at hand: the lakes are painted in blue, the agent is the red square, the target is the green square, and the grey pixels are the possible positions. These positions are the subset of $[-10, 10]^2$ that excludes the points within the lakes. The goal is to navigate around the lakes in order to reach the target position.

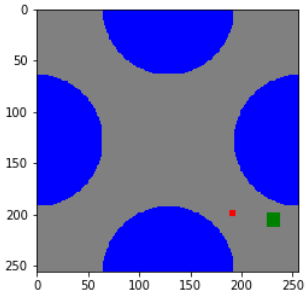


Figure 6: 2-D navigation environment

Learning the Idle generator In this section we analyse the ability to learn P_π^η and ρ_π using Algorithm 5. As discussed in Section C.1, when the target distribution is a short term prediction of future states, the obtained generator is reliable. We consider in what follows $\eta = \text{Geom}(0.7)$ to satisfy this condition. We use the same hyper-parameters to learn the generator in both cases (P_π^η and ρ_π). We use 3-layers deep, 64-neurons wide neural networks for the generator and the discriminator, a batch size of 256 and we iterate the algorithm for 10000 steps. In Figures 7 and 8, we plot the expert trajectories with black lines, the initial position with green dots, the target position with red dots,

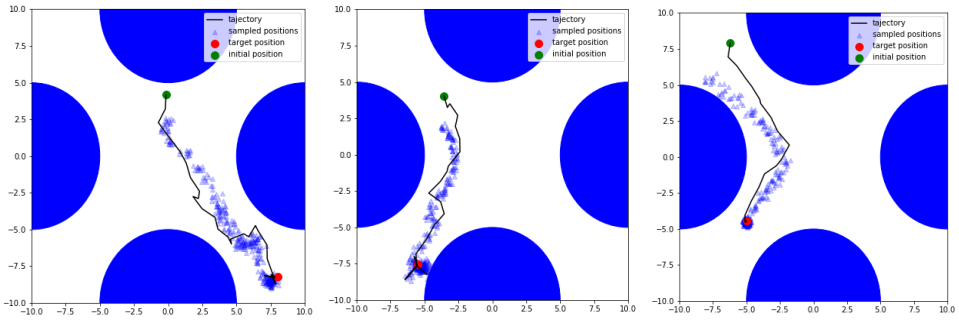


Figure 7: P_π^η learned generator

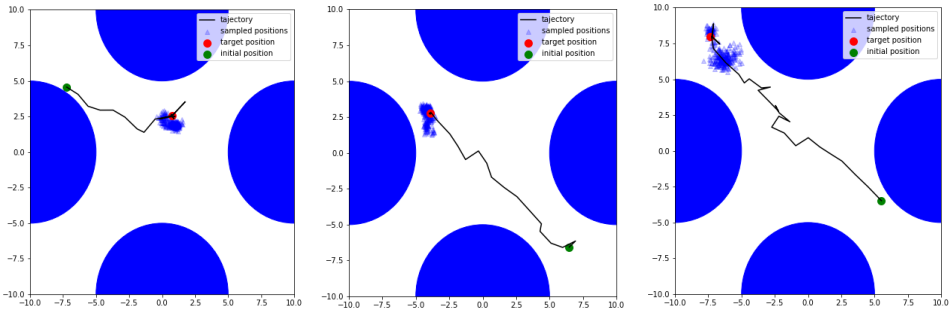


Figure 8: ρ_π learned generator

and the sampled states with blue triangles. On one hand, we observe in Figure 7 that the P_π^η learned generator provides reliable samples (in the sense that they follow the trails of expert trajectories). On the other hand, in Figure 8, we observe that the learned ρ_π generator over-fits the stationary distribution and only samples states around the target position. The mode collapse is essentially explained by the fact that most of the samples from the ρ_π distribution are indeed around the target position.

Learned cost function As discussed in Section 5, we only obtain expert-like performances when using the Idle generator. However, given that the considered state-space in this section is a 2-D plan, we can visualise the learned (state only) cost function with a heat-map. We consider five particular tasks that coincide with reaching the top-left, center, top-right, bottom-left and bottom-right of the map. Figures 9 and 10 coincide with such heat-maps, with darker shades for higher costs and brighter colours for lower ones. In Figure 9, we observe that the GAIL learned costs are particularly low in the vicinity of the target position while they are evenly spreaded elsewhere. This entails from the discriminator over-fitting the replay-buffer as most of the observations are drawn from the stationary distribution. On the other hand, in Figure 10, we observe that the MEGAN learned costs are high outside of the paths that lead to the target, and decrease exponentially as we get closer to the goal position.

E Additional experimental details

E.1 Maximum Mean Discrepancy evaluation

Formally, given a reproducing kernel Hilbert space (RKHS) of real-valued functions $\mathcal{H}$, the MMD between two distributions P and Q is defined as: $\text{MMD}_{\mathcal{H}}(P, Q) = \sup_{f \in \mathcal{H}} \mathbb{E}_{X \sim P}[f(X)] - \mathbb{E}_{Y \sim Q}[f(Y)]$. Recall that the reproducing property of RKHS, implies that there is a one to one correspondence between positive definite kernels k and RKHSs $\mathcal{H}$ such that every function $f \in \mathcal{H}$ verifies $f(x) = \langle f, k(., x) \rangle_{\mathcal{H}}$ (where $\langle \, , \rangle_{\mathcal{H}}$ denotes the RKHS inner product). We propose to evaluate

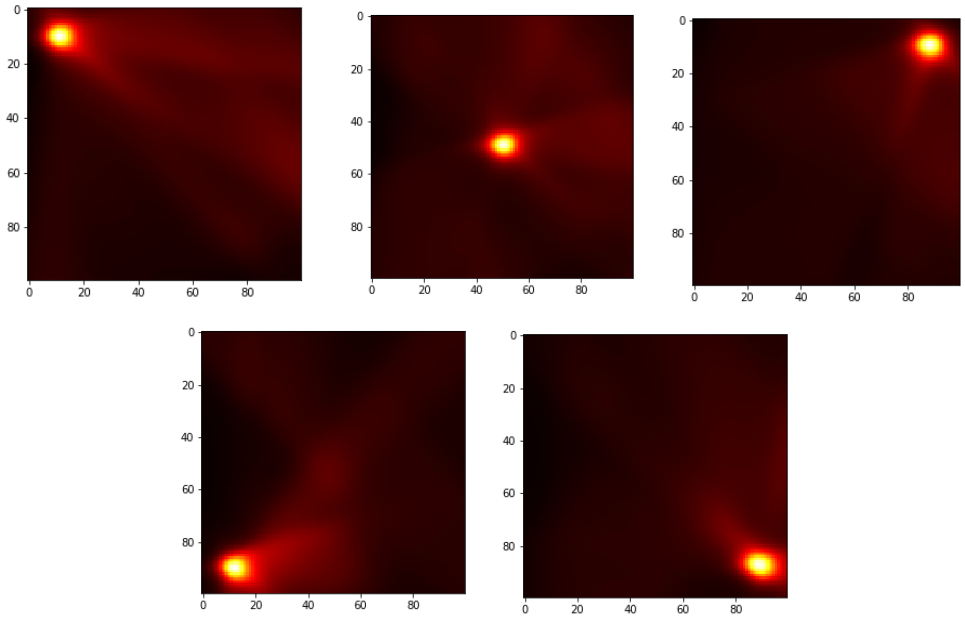


Figure 9: GAIL learned cost heat-map as a function of the target position

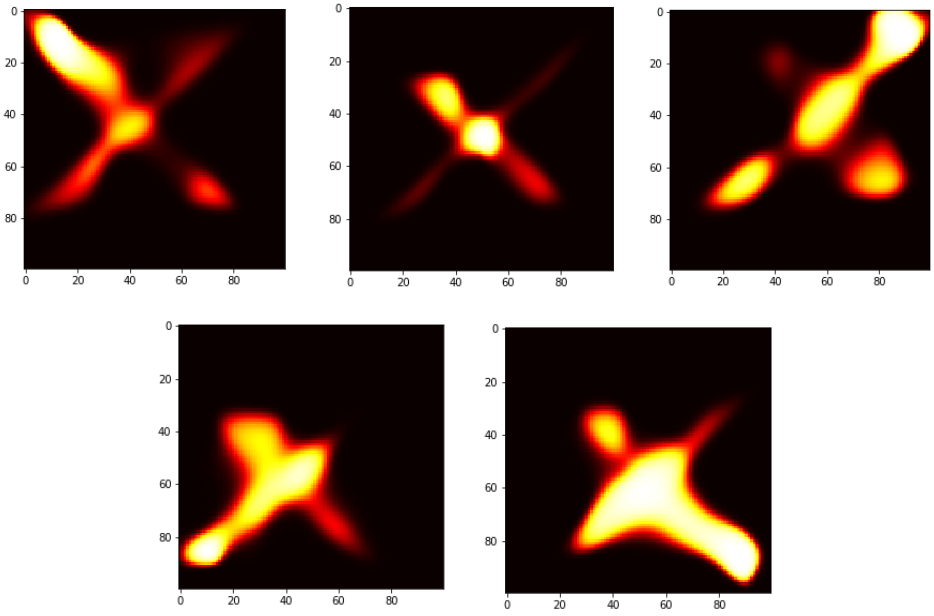


Figure 10: MEGAN learned cost heat-map as a function of the target position

the MMD using a kernel two-sample test with the following unbiased estimator Gretton et al. (2012):

$$\mathrm{MMD}^2_{\mathcal{H}}(P,Q) = \frac{1}{N(N-1)}\sum_{i\neq j} k(x_i,x_j) + \frac{1}{N(N-1)}\sum_{i\neq j} k(y_i,y_j) - \frac{1}{N^2}\sum_{i,j} k(x_i,y_j)$$

where $(x_i)_{i=0}^N$ are sampled according to P and $(y_i)_{i=0}^N$ are sampled according to Q. In the experimental analysis, we only consider the RKHS associated with the radial basis function $k(x,y) = \exp(\|x-y\|^2/d)$ (where d is the dimension of the variables x and y).

E.2 HYPER-PARAMETERS

In this section we provide a detailed description of the used implementation as well as the selected hyper-parameters.

Expert demonstrations of length MAX-LENGTH are stored in a demonstrator replay-buffer. We use two additional replay-buffers (one for the policy and one for the expert), with a maximum capacity of 10^6 transitions that are initially empty. In each cycle, N trajectories from the demonstrator replay-buffer are sampled and added to the expert replay-buffer. The policy generates then N trajectories, that are stored in the policy replay-buffer. In the multi-task setting, the tasks of these trajectories are the same ones in the expert's samples. The policy is updated each cycle using SAC for SAC-EPOCH epochs with a bath size of SAC-BATCH. Every D-UPDATE-RATE cycles, the discriminator is updated for D-EPOCH epochs with a bath size of D-BATCH. The algorithm runs until the policy generates MAX-TRANSITIONS transitions in total.

The policy, as well as the underlying value functions, are approximated using an N-LAYER-P deep, HIDDEN-P wide neural networks. The discriminator is approximated using an N-LAYER-D deep, HIDDEN-D wide neural network.

ENV-ID	*Hopper*	*Half-Cheetah*	*Ant*	*FetchReach*	*2-D Maze*
N	90	90	90	90	90
MAX-LENGTH	500	500	500	100	50
SAC-EPOCH	500	500	500	300	150
SAC-BATCH	256	256	256	1024	128
D-UPDATE-RATE	1	1	1	5	1
D-EPOCH	50	50	50	500	300
D-BATCH	512	512	512	512	128
N-LAYER-P	3	3	3	4	4
HIDDEN-P	64	64	64	64	64
N-LAYER-D	1	1	1	3	3
HIDDEN-D	16	32	32	16	16
MAX-TRANSITIONS	10^7	10^7	10^7	10^6	5×10^5

F PROOF OF TECHNICAL RESULTS

We provide in this section proofs for all stated technical results. To find a particular one, please refer to the following:

Section F.1: Useful intermediate results as well as their proof.

Section F.2: Proofs for the theoretical claims stated in Section 2.3 and Appendix A.

Section F.3: Proofs for the theoretical claims stated in Section 2.4.

Section F.4: Proof for the theoretical claims stated in Section C.1.

Section F.5: Proof for the theoretical claims stated in Section 4.

F.1 USEFUL INTERMEDIATE RESULTS

For the sake of conciseness, we start by providing important intermediate results that will be used in the proofs of propositions 6, 2, and 3. The first one (Proposition 10) transforms η-weighted γ discounted functional averaged over π-generated trajectory into expectations with respect to ρ_π and

P_π^η or, equivalently into expectations with respect to μ_π. The second one (Proposition 11) guarantees a one on one mapping between occupancy measures and policies.

Proposition 10 *For any distribution η, and for any mapping $f : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, the following identity holds:*

$$\mathbb{E}^\eta_{p_0,\pi}[\sum_t \gamma^t f(s_t, a_t)] = \int_{s_0,s,a,s_+,a_+} p_0(s_0)\rho_\pi(s, a|s_0)P^\eta_\pi(s_+, a_+|s, a)f(s_+, a_+) \tag{11}$$

$$= \int_{s_0,s_+,a_+} p_0(s_0)\mu_\pi(s_+, a_+|s_0)f(s_+, a_+) \tag{12}$$

Proposition 11 *Let $(\phi_t)_{t=0}^\infty$ be a strictly positive real valued convergent series (i.e. $\sum_t \phi_t < \infty$ and $\phi_t > 0$), and let $\Phi_\pi(s, a|s_0)$ be the ϕ-weighted occupancy measure associated to the policy π:*

$$\Phi_\pi(s, a|s_0) := \sum_t \phi_t \mathbb{P}_\pi(s_t = s, a_t = a|s_0)$$

Then for a given ϕ-weighted occupancy measure $\Phi \in \{\Phi_\pi | \pi : \mathcal{S} \to \Delta(\mathcal{A})\}$:

1- *Φ is the ϕ-weighted occupancy measure of $\pi_\Phi := \frac{\Phi(s,a|s_0)}{\int_{a'} \Phi(s,a'|s_0)}$*

2- *π_Φ is the only policy whose ϕ-weighted occupancy measure is Φ*

PROOF OF PROPOSITION 10:

The proof of the first equality relies on some algebraic manipulations and the law of total expectation.

$$\mathbb{E}^\eta_{p_0,\pi}[\sum_t \gamma^t f(s_t, a_t)] := \int_{s_0,s,a} p_0(s_0)P^\eta_\pi(s, a|s_0)\mathbb{E}_{\pi,\delta_{(s,a)}}[\sum_t \gamma^t f(s_t, a_t)] \tag{13}$$

$$= \int_{s_0,s,a} p_0(s_0) \sum_k \eta(k)\mathbb{P}_\pi(s_k = s, a_k = a|s_0)\mathbb{E}_\pi[\sum_t \gamma^t f(s_{t+k}, a_{t+k})|s_k = s, a_k = a] \tag{14}$$

$$= \int_{s_0,s_k,a_k} p_0(s_0) \sum_{k,t} \gamma^t \eta(k)\mathbb{E}_\pi\Big[\mathbb{E}_\pi\big[f(s_{t+k}, a_{t+k})|s_k, a_k\big]\Big|s_0\Big] \tag{15}$$

where $\mathbb{E}_{\pi,\delta_{(s,a)}}$ designate the expectation over trajectories initialised at the state action couple (s, a). Using the law of total expectation we can assert that:

$$\mathbb{E}_\pi\Big[\mathbb{E}_\pi\big[f(s_{t+k}, a_{t+k})|s_k, a_k\big]\Big|s_0\Big] = \mathbb{E}_\pi[f(s_{t+k}, a_{t+k})|s_0] \tag{16}$$

$$= \mathbb{E}_\pi\Big[\mathbb{E}_\pi\big[f(s_{t+k}, a_{t+k})|s_t, a_t\big]\Big|s_0\Big] \tag{17}$$

From this relationship, it follows that:

$$\mathbb{E}^\eta_{p_0,\pi}[\sum_t \gamma^t f(s_t, a_t)] = \int_{s_0,s_t,a_t} p_0(s_0) \sum_{k,t} \gamma^t \eta(k)\mathbb{E}_\pi\Big[\mathbb{E}_\pi\big[f(s_{t+k}, a_{t+k})|s_t, a_t\big]\Big|s_0\Big] \tag{18}$$

$$= \int_{s_0,s,a} p_0(s_0) \sum_t \gamma^t \mathbb{P}_\pi(s_t = s, a_t = a|s_0)\mathbb{E}_{\pi,\delta_{(s,a)}}[\sum_k \eta(k)f(s_k, a_k)] \tag{19}$$

$$= \int_{s_0,s,a} p_0(s_0)\rho_\pi(s, a|s_0)\mathbb{E}_{\pi,\delta_{(s,a)}}[\sum_k \eta(k)f(s_k, a_k)] \tag{20}$$

$$= \int_{s_0,s,a} p_0(s_0)\rho_\pi(s, a|s_0) \sum_k \eta(k) \int_{s_+,a_+} \mathbb{P}_\pi(s_{t+k} = s_+, a_{t+k} = a_+|s_t = s, a_t = a)f(s_+, a_+) \tag{21}$$

$$= \int_{s_0,s,a,s_+,a_+} p_0(s_0)\rho_\pi(s, a|s_0)P^\eta_\pi(s_+, a_+|s, a)f(s_+, a_+) \tag{22}$$

This concludes the proof of the first equality in Proposition 10.

The proof of the second equality relies on the Markov property of the environment and some algebraic manipulations.

$$\mathbb{E}^{\eta}_{p_0,\pi}[\sum_t \gamma^t f(s_t,a_t)] = \int_{s_0,s,a,s_+,a_+} p_0(s_0)\rho_\pi(s,a|s_0)P^{\eta}_{\pi}(s_+,a_+|s,a)f(s_+,a_+) \tag{23}$$

$$= \int_{s_0,s,a,s_+,a_+} p_0(s_0)\sum_t \gamma^t \mathbb{P}_\pi(s_t=s,a_t=a|s_0)\sum_k \eta(k)\mathbb{P}_\pi(s_{t+k}=s_+,a_{t+k}=a|s_t=s,a_t=a)f(s_+,a_+) \tag{24}$$

$$= \int_{s_0,s_+,a_+} p_0(s_0)\sum_{t,k}\gamma^t\eta(k)\int_{s,a}\Big(\mathbb{P}_\pi(s_t=s,a_t=a|s_0)\mathbb{P}_\pi(s_{t+k}=s_+,a_{t+k}=a|s_t=s,a_t=a)\Big)f(s_+,a_+) \tag{25}$$

$$= \int_{s_0,s_+,a_+} p_0(s_0)\sum_{t,k}\gamma^t\eta(k)\mathbb{P}_\pi(s_{t+k}=s_+,a_{t+k}=a|s_0)f(s_+,a_+) \tag{26}$$

$$= \int_{s_0,s_+,a_+} p_0(s_0)\mu_\pi(s_+,a_+|s_0)f(s_+,a_+) \tag{27}$$

This concludes the proof of Proposition 10.

PROOF OF PROPOSITION 11:

For the first assertion of the proposition, recall that:

$$\Phi_\pi(s,a|s_0) := \sum_t \phi_t \mathbb{P}_\pi(s_t=s,a_t=a|s_0) \tag{28}$$

$$= \sum_t \phi_t \mathbb{P}_\pi(s_t=s|s_0)\pi(a|s) := \Phi_\pi(s|s_0)\pi(a|s) \tag{29}$$

This implies that:

$$\frac{\Phi(s,a|s_0)}{\int_{a'}\Phi(s,a'|s_0)} = \frac{\pi(a|s)\Phi_\pi(s|s_0)}{\int_{a'}\pi(a'|s)\Phi_\pi(s|s_0)} = \pi(a|s) \tag{30}$$

For the second assertion of the proposition, consider two policies π_1 and π_2 such that $\Phi_{\pi_1} = \Phi_{\pi_2}$. Notice that:

$$\forall s, s_0 \in \mathcal{S} \qquad \Phi_{\pi_1}(s|s_0) := \sum_t \phi_t \mathbb{P}_{\pi_1}(s_t=s|s_0) = \int_a \Phi_{\pi_1}(s,a|s_0) \tag{31}$$

$$= \int_a \Phi_{\pi_2}(s,a|s_0) = \Phi_{\pi_2}(s|s_0) \tag{32}$$

This can further yield:

$$\forall s \in \mathcal{S}, a \in \mathcal{A} \qquad \Phi_{\pi_1}(s,a|s_0) = \Phi_{\pi_2}(s,a|s_0) \tag{33}$$

$$\Rightarrow \forall s \in \mathcal{S}, a \in \mathcal{A} \qquad \Phi_{\pi_1}(s|s_0)\pi_1(a|s) = \Phi_{\pi_2}(s|s_0)\pi_2(a|s) \tag{34}$$

$$\Rightarrow \forall s \in \mathcal{S}, a \in \mathcal{A} \qquad \pi_1(a|s) = \pi_2(a|s) \tag{35}$$

This concludes the proof of Proposition 11.

F.2 GENERALISED REINFORCEMENT LEARNING

In this section we address the claims stated in section 2.3 as well as those stated in Appendix A. Proposition 1 is recalled for the sake of comprehensiveness, a detailed proof is provided in Geist et al. (2019).

PROOF OF PROPOSITION 5:

In order to obtain the desired result, we exploit both the classical policy gradient theorem and the product derivative rule. Using elementary calculus, we obtain the following:

$$\nabla_\theta \mathbb{E}^\eta_{p_0,\pi_\theta}[Q^c_{\pi_\theta}] = \nabla_\theta \int_{s_0,s_+,a_+} p_0(s_0) P^\eta_{\pi_\theta}(s_+, a_+|s_0) Q^c_{\pi_\theta}(s_+, a_+) \tag{36}$$

$$= \nabla_\theta \int_{s_0,s_+,a_+} p_0(s_0) P^\eta_{\pi_\theta}(s_+|s_0)\pi_\theta(a_+|s_+) Q^c_{\pi_\theta}(s_+, a_+) \tag{37}$$

$$= \nabla_\theta \int_{s_0,s_+} p_0(s_0) P^\eta_{\pi_\theta}(s_+|s_0) v^c_{\pi_\theta}(s_+) \tag{38}$$

$$= \int_{s_0,s_+} p_0(s_0) \Big[v^c_{\pi_\theta}(s_+) \nabla_\theta P^\eta_{\pi_\theta}(s_+|s_0) + P^\eta_{\pi_\theta}(s_+|s_0) \nabla_\theta v^c_{\pi_\theta}(s_+) \Big] \tag{39}$$

$$= \int_{s_0,s_+} p_0(s_0) v^c_{\pi_\theta}(s_+) \nabla_\theta P^\eta_{\pi_\theta}(s_+|s_0) + \underbrace{\int_{s_0,s_+} p_0(s_0) P^\eta_{\pi_\theta}(s_+|s_0) \nabla_\theta v^c_{\pi_\theta}(s_+)}_{A} \tag{40}$$

Recall that the policy gradient theorem can be written simply as:

$$\forall s_0 \in \mathcal{S} \qquad \nabla_\theta \mathbb{E}_{\pi_\theta}[\sum_t \gamma^t c(s_t, a_t)|s_0] = \nabla_\theta \int_a \pi_\theta(a|s_0) Q^c_{\pi_\theta}(s_0, a) \tag{41}$$

$$= \nabla_\theta v^c_{\pi_\theta}(s_0) = \int_s \rho_\pi(s|s_0) \mathbb{E}_{a\sim\pi_\theta}[Q^c_{\pi_\theta}(s, a) \nabla_\theta \log \pi_\theta(a|s)] \tag{42}$$

This concludes our proof as:

$$A = \int_{s_0,s_+} p_0(s_0) P^\eta_{\pi_\theta}(s_+|s_0) \nabla_\theta v^c_{\pi_\theta}(s_+) = \int_{s_0,s_+,s} p_0(s_0) P^\eta_{\pi_\theta}(s_+|s_0) \rho_\pi(s|s_+) \mathbb{E}_{a\sim\pi_\theta}[Q^c_{\pi_\theta}(s, a) \nabla_\theta \log \pi_\theta(a|s)] \tag{43}$$

PROOF OF PROPOSITION 6:

The proof relies on the definition of the advantage function to obtain the first equality and on proposition 10 to obtain the second one. Recall that:

$$\mathbb{E}^\eta_{p_0,\pi_n}\Big[\sum_t \gamma^t A_{\pi_o}(s_t, a_t)\Big] = \mathbb{E}^\eta_{p_0,\pi_n}\Big[\sum_t \gamma^t (c(s_t, a_t) + \gamma v^c_{\pi_o}(s_{t+1}) - v^c_{\pi_o}(s_t))\Big] \tag{44}$$

$$= \mathbb{E}^\eta_{p_0,\pi_n}[-v^c_{\pi_o}(s_0) + \sum_t \gamma^t c(s_t, a_t)] \tag{45}$$

$$= -\underbrace{\int_{s_0} p_0(s_0) v^c_{\pi_o}(s_0)}_{\mathcal{L}^\eta_0(\pi_o, c)} + \underbrace{\mathbb{E}^\eta_{p_0,\pi_n}[\sum_t \gamma^t c(s_t, a_t)]}_{\mathcal{L}^\eta_0(\pi_n, c)} \tag{46}$$

$$\iff \mathcal{L}^\eta_0(\pi_n, c) = \mathcal{L}^\eta_0(\pi_o, c) + \mathbb{E}^\eta_{p_0,\pi_n}\Big[\sum_t \gamma^t A_{\pi_o}(s_t, a_t)\Big] \tag{47}$$

This concludes the proof of the first equality. In addition, by observing that A_π is a mapping from $\mathcal{S} \times \mathcal{A}$ to $\mathbb{R}$, we can apply proposition 10 to further simplify the expectation term:

$$\mathbb{E}^\eta_{p_0,\pi_n}\Big[\sum_t \gamma^t A_{\pi_o}(s_t, a_t)\Big] = \int_{s_0,s,a,s_+,a_+} p_0(s_0) \rho_\pi(s, a|s_0) P^\eta_\pi(s_+, a_+|s, a) A_{\pi_o}(s_+, a_+) \tag{48}$$

This concludes the proof as we have:

$$\mathcal{L}^\eta_0(\pi_n, c) = \mathcal{L}^\eta_0(\pi_o, c) + \int_{s_0,s,a,s_+,a_+} p_0(s_0) \rho_\pi(s, a|s_0) P^\eta_\pi(s_+, a_+|s, a) A_{\pi_o}(s_+, a_+) \tag{49}$$

F.3 GENERALISED INVERSE REINFORCEMENT LEARNING

In this section, we address the claims stated in section 2.4.

PROOF OF PROPOSITION 2:

The proof relies on the properties of saddle point Hiriart-Urruty & Lemaréchal (2013). Let $\tilde{c}$, $\tilde{\pi}$ and $\hat{\pi}$ be respectively defined as:

$$\begin{aligned}
&\tilde{c} \in \mathrm{IRL}^{\eta}_{\psi,\Omega}(\pi_E) \\
&\tilde{\pi} \in \mathrm{RL}^{\eta}_{\Omega}(\tilde{c}) = \mathrm{RL}^{\eta}_{\Omega} \circ \mathrm{IRL}^{\eta}_{\psi,\Omega}(\pi_E) = \arg\min_{\pi} \mathcal{L}^{\eta}_{\Omega}(\pi, \tilde{c}) \\
&\hat{\pi} \in \arg\min_{\pi} \max_{c} L(\pi, c)
\end{aligned}$$

Our goal is to prove that $\tilde{\pi} = \hat{\pi}$. Equivalently (due to proposition 11) this boils down to proving that $\mu_{\tilde{\pi}} = \mu_{\hat{\pi}}$. Using Proposition 10 and 11, we can re-write:

$$\mathcal{L}^{\eta}_{\Omega}(\pi, c) := \mathbb{E}^{\eta}_{p_0,\pi}\big[\sum_t \gamma^t c(s_t, a_t)\big] - \Omega(\pi) \tag{50}$$

$$= \int_{s_0,s,a,s_+,a_+} p_0(s_0)\rho_\pi(s, a|s_0)P^{\eta}_{\pi}(s_+, a_+|s, a)c(s_+, a_+) - \Omega(\pi) \tag{51}$$

$$= \bar{\mathcal{L}}^{\eta}_{\Omega}(\mu_\pi, c) = \int_{s_0,s_+,a_+} p_0(s_0)\mu_\pi(s_+, a_+|s_0)c(s_+, a_+) - \Omega(\mu_\pi) \tag{52}$$

This implies that :

$$\mu_{\tilde{\pi}} \in \arg\min_{\mu\in\mathcal{D}} \bar{\mathcal{L}}^{\eta}_{\Omega}(\mu, \tilde{c}) = \arg\min_{\mu\in\mathcal{D}} \bar{L}(\mu, \tilde{c}) \tag{53}$$

where:

$$\bar{L} : \mathcal{D} \times \mathbb{R}^{\mathcal{S}\times\mathcal{A}} \to \mathbb{R} \tag{54}$$

$$\mathcal{D} = \Big\{\mu_\pi : \mu_\pi(s, a|s_0) = \sum_{t,k} \gamma^t \eta(k)\mathbb{P}_\pi(s_{t+k} = s, a_{t+k} = a|s_0)\Big|\pi : \mathcal{S} \to \Delta(\mathcal{A})\Big\} \tag{55}$$

$$\bar{L}(\mu_\pi, c) = L(\pi, c) = -\Omega(\pi) - \psi(c) + \int_{s_0} p_0(s_0)d_c(\pi\|\pi_E)(s_0) \tag{56}$$

$$= -\Omega(\mu_\pi) - \psi(c) + \int_{s_0,s_+,a_+} p_0(s_0)c(s_+, a_+)\Big[\mu_\pi(s_+, a_+|s_0) - \mu_{\pi_E}(s_+, a_+|s_0)\Big] \tag{57}$$

In addition, for the same reasons, we have that:

$$\begin{aligned}
&\tilde{c} \in \arg\max_{c} \min_{\pi} \mathcal{L}^{\eta}_{\Omega}(\pi, c) - \mathcal{L}^{\eta}_{\Omega}(\pi^*_E, c) - \psi(c) = \arg\max_{c} \min_{\mu\in\mathcal{D}} \bar{L}(\mu, c) \\
&\mu_{\hat{\pi}} \in \arg\min_{\mu\in\mathcal{D}} \max_{c} \bar{L}(\mu, c)
\end{aligned} \tag{58}$$

Notice that $\mathbb{R}^{\mathcal{S}\times\mathcal{A}}$ is convex, $\bar{L}(\mu, c)$ is convex w.r.t μ and concave w.r.t c (due to convexity of ψ and $-\Omega$). Therefore, the minmax duality property holds as soon as $\mathcal{D}$ is compact and convex.

Convexity and compacity of $\mathcal{D}$: We prove this under the assumption that η is a geometric distribution (i.e. $\eta = \mathrm{Geom}(\gamma_\eta)$). To establish the convexity, we prove that $\mathcal{D} = \{f : \mathcal{S} \to \Delta(\mathcal{S} \times \mathcal{A})\}$ where f is a solution of the following equations:

$$\forall s, s_0 \in \mathcal{S}, a \in \mathcal{A} \quad \begin{aligned}
&\int_a f(s, a|s_0) = \int_a g(s, a|s_0) + \int_{s',a'} \gamma f(s', a'|s_0)\mathcal{P}(s|s', a') \\
&\int_a f(s, a|s_0) = \int_a h(s, a|s_0) + \int_{s',a'} \gamma_\eta f(s', a'|s_0)\mathcal{P}(s|s', a') \\
&\int_a g(s, a|s_0) = 1 + \int_{s',a'} \gamma_\eta \mathcal{P}(s|s', a')g(s', a'|s_0) \\
&\int_a h(s, a|s_0) = 1 + \int_{s',a'} \gamma \mathcal{P}(s|s', a')h(s', a'|s_0) \\
&f(s, a|s_0) \geq 0, \quad g(s, a|s_0) \geq 0, \quad h(s, a|s_0) \geq 0
\end{aligned} \tag{59}$$

To this end, notice that for any policy π, we can verify that $(f = \mu_\pi, g = P^\eta_\pi, h = \rho_\pi)$ is a solution to Equation 59. We focus now on the converse statement. Let (f, g, h) a solution to Equation 59. Given the third equality:

$$\int_a g(s, a|s_0) = 1 + \int_{s',a'} \gamma_\eta \mathcal{P}(s|s', a')g(s', a'|s_0), \tag{60}$$

we exploit a classical result from the MDP literature [Puterman (2014), Section 6.9.2], to derive the existence of a policy π_g such that:

$$\begin{aligned} g(s, a|s_0) &= \pi_g(a|s)\Big[1 + \int_{s',a'} \gamma_\eta \mathcal{P}(s|s', a')g(s', a'|s_0)\Big] \\ \textbf{where } \forall s_0 \in \mathcal{S} \quad \pi_g(a|s) &= \frac{g(s, a|s_0)}{\int_{a'} g(s, a'|s_0)} \end{aligned} \tag{61}$$

From Equation 61, we conclude that g is the unique fixed point of a γ_η-contraction. We can verify that $g = P^\eta_{\pi_g}$ is the unique solution of Equation 61. Using a similar reasoning with respect to the fourth equality in Equation 59, we conclude that $h = \rho_{\pi_h}$ where for any state $s_0 \in \mathcal{S}$, we have $\pi_h(a|s) = \frac{h(s,a|s_0)}{\int_{a'} h(s,a'|s_0)}$. From this, we conclude that for any solution (f, g, h) to Equation 59, there exist two policies π_h and π_g such that:

$$\begin{aligned} \forall s, s_0 \in \mathcal{S}, a \in \mathcal{A} \qquad & \int_a f(s, a|s_0) = P^\eta_{\pi_g}(s|s_0) + \int_{s',a'} \gamma f(s', a'|s_0)\mathcal{P}(s|s', a') \\ & \int_a f(s, a|s_0) = \rho_{\pi_h}(s|s_0) + \int_{s',a'} \gamma_\eta f(s', a'|s_0)\mathcal{P}(s|s', a') \cdot \\ & f(s, a|s_0) \geq 0 \end{aligned} \tag{62}$$

which is equivalent to [Puterman (2014), Section 6.9.2]:

$$\begin{aligned} \forall s, s_0 \in \mathcal{S}, a \in \mathcal{A} \qquad & f(s, a|s_0) = \pi_f(a|s)\Big[P^\eta_{\pi_g}(s|s_0) + \int_{s',a'} \gamma f(s', a'|s_0)\mathcal{P}(s|s', a')\Big] \\ & f(s, a|s_0) = \pi_f(a|s)\Big[\rho_{\pi_h}(s|s_0) + \int_{s',a'} \gamma_\eta f(s', a'|s_0)\mathcal{P}(s|s', a')\Big]. \\ & \pi_f(a|s) = \frac{f(s, a|s_0)}{\int_{a'} f(s, a'|s_0)}, \quad f(s, a|s_0) \geq 0 \end{aligned} \tag{63}$$

Notice that the first equality in Equation 63, implies that f is the unique fixed point of a γ-contraction. We also notice that:

$$f^0_{\pi_g,\pi_f}(s_+, a_+|s_0) := \sum_{k,t} \gamma^t \gamma^k_\eta \int_{s,a} \mathbb{P}^t_{\pi_f}(s_+, a_+|s)\mathbb{P}^k_{\pi_g}(s, a|s_0) \tag{64}$$

$$= \sum_{k,t>0} \gamma^t \gamma^k_\eta \int_{s,a} \mathbb{P}^t_{\pi_f}(s_+, a_+|s)\mathbb{P}^k_{\pi_g}(s, a|s_0) + \sum_k \gamma^k_\eta \int_{s,a} \mathbb{P}^0_{\pi_f}(s_+, a_+|s)\mathbb{P}^k_{\pi_g}(s, a|s_0) \tag{65}$$

$$= \gamma\pi_f(a_+|s_+) \int_{s',a'} f^0_{\pi_g,\pi_f}(s', a'|s_0)\mathcal{P}(s|s', a') + \sum_k \gamma^k_\eta \mathbb{P}^k_{\pi_g}(s_+|s_0)\pi_f(a_+|s_+) \tag{66}$$

$$= \pi_f(a_+|s_+)\Big[\int_{s',a'} \gamma f^0_{\pi_g,\pi_f}(s', a'|s_0)\mathcal{P}(s_+|s', a') + P^\eta_{\pi_g}(s_+|s_0)\Big] \tag{67}$$

where $\mathbb{P}^n_\pi(s, a|s^0) = \mathbb{P}_\pi(s_n = s, a_n = a|s_0 = s^0)$. Thus, we conclude $f^0_{\pi_g,\pi_f}$ is the unique solution to the first equality. Similarly, we notice that the second equality is a γ_η-contraction, whose unique

fixed point is:

$$f^1_{\pi_h,\pi_f}(s_+, a_+|s_0) := \sum_{k,t} \gamma^t \gamma^k_\eta \int_{s,a} \mathbb{P}^k_{\pi_f}(s_+, a_+|s) \mathbb{P}^t_{\pi_h}(s, a|s_0) \tag{68}$$

$$= \sum_{k>0,t} \gamma^t \gamma^k_\eta \int_{s,a} \mathbb{P}^k_{\pi_f}(s_+, a_+|s) \mathbb{P}^t_{\pi_h}(s, a|s_0) + \sum_t \gamma^t \int_{s,a} \mathbb{P}^0_{\pi_f}(s_+, a_+|s) \mathbb{P}^t_{\pi_h}(s, a|s_0) \tag{69}$$

$$= \pi_f(a_+|s_+) \Big[\int_{s',a'} \gamma_\eta f^1_{\pi_h,\pi_f}(s', a'|s_0) \mathcal{P}(s_+|s', a') + \rho_{\pi_h}(s_+|s_0) \Big] \tag{70}$$

We derive from the previously discussed statement, that if (f, g, h) is a solution to Equation 59, then there exist three policies (π_f, π_g, π_h) such that:

$$\begin{aligned} f &= f^0_{\pi_g,\pi_f} = f^1_{\pi_h,\pi_f} \\ g &= P^\eta_{\pi_g} \\ h &= \rho_{\pi_h} \end{aligned} \tag{71}$$

However, not any random choice of policies (π_f, π_g, π_h) can satisfy Equation 71. By varying γ and γ_η, we notice that in order for the first equality to hold, the following equality must be satisfied for any integers (k, t), and for any states (s_+, a_+, s_0):

$$\int_{s,a} \mathbb{P}^t_{\pi_f}(s_+, a_+|s) \mathbb{P}^k_{\pi_g}(s, a|s_0) = \int_{s,a} \mathbb{P}^k_{\pi_f}(s_+, a_+|s) \mathbb{P}^t_{\pi_h}(s, a|s_0) \tag{72}$$

by fixing k at zero and varying t and by fixing t at zero and varying k, we obtain the following constraints:

$$\begin{aligned} P^\eta_{\pi_f} &= P^\eta_{\pi_g} \\ \rho_{\pi_f} &= \rho_{\pi_h} \end{aligned} \tag{73}$$

Using Proposition 11 and Equation 73, it follows that Equation 71 admits a solution if and only if $\pi_f = \pi_g = \pi_h$. This means that if (f, g, h) is a solution to Equation 59, then there exists a policy π such that:

$$\begin{aligned} f &= f^0_{\pi,\pi} = f^1_{\pi,\pi} = \mu_\pi \\ g &= P^\eta_\pi \\ h &= \rho_\pi \end{aligned} \tag{74}$$

This concludes the converse statement, proving that $\mathcal{D}$ is a set of occupancy measures satisfying the set of affine constraints from Equation 59. Consequently, $\mathcal{D}$ is a convex set. In addition, the limit of any sequence of elements from $\mathcal{D}$ will also satisfy Equation 62. From this we establish that $\mathcal{D}$ is closed which implies that it is also compact.

From this we derive that minmax duality holds and that:

$$\min_{\mu \in \mathcal{D}} \max_c \bar{L}(\mu, c) = \max_c \min_{\mu \in \mathcal{D}} \bar{L}(\mu, c) \tag{75}$$

From Equation 58, it follows that $(\mu_{\hat{\pi}}, \tilde{c})$ is a saddle point for the function $\bar{L}$. This implies from Equation 53 that:

$$\mu_{\hat{\pi}}, \mu_{\tilde{\pi}} \in \arg\min_{\mu \in \mathcal{D}} \bar{L}(\mu, \tilde{c}) \tag{76}$$

In addition, due to the strict convexity of $\bar{L}$ w.r.t μ (due to assumed strict convexity of Ω) we have that:

$$\mu_{\hat{\pi}} = \mu_{\tilde{\pi}} \tag{77}$$

which concludes our proof.

PROOF OF COROLLARY 2.1:

The proof entails directly from the duality of $\bar{L}$ and that $(\tilde{c}, \mu_{\tilde{\pi}})$ is a saddle point of $\bar{L}$.

PROOF OF PROPOSITION 3:

The proof relies on re-writing the η-weighted entropy reguliser using Proposition 10, and then verifying its convexity with respect to μ_π using the log-sum inequality. In fact, notice that:

$$H^\eta_{p_0}(\pi) = \mathbb{E}^\eta_{p_0,\pi}\Big[\sum_t -\gamma^t \log\big[\pi(a_t|s_t)\big]\Big] \tag{78}$$

$$= \int_{s_0,s_+,a_+} p_0(s_0)\mu_\pi(s_+, a_+|s_0)\log\big[\pi(a_+|s_+)\big] = \bar{H}^\eta_{p_0}(\mu_\pi) \tag{79}$$

Consider two η-weighted occupancy measures μ_1, μ_2, and let π_1, π_2 their respective policies. Let $\lambda \in]0, 1[$:

$$\bar{H}(\lambda\mu_1 + (1-\lambda)\mu_2) = \int_{s_0,s_+,a_+} -p_0(s_0)\big[\lambda\mu_1 + (1-\lambda)\mu_2\big](s_+, a_+|s_0)\log\Big[\frac{\big[\lambda\mu_1 + (1-\lambda)\mu_2\big](s_+, a_+|s_0)}{\int_a \big[\lambda\mu_1 + (1-\lambda)\mu_2\big](s_+, a|s_0)}\Big] \tag{80}$$

Du to the log-sum inequality we have:

$$\big[\lambda\mu_1 + (1-\lambda)\mu_2\big](s_+, a_+|s_0)\log\Big[\frac{\big[\lambda\mu_1 + (1-\lambda)\mu_2\big](s_+, a_+|s_0)}{\int_a \big[\lambda\mu_1 + (1-\lambda)\mu_2\big](s_+, a|s_0)}\Big] \tag{81}$$

$$= \big[\lambda\mu_1 + (1-\lambda)\mu_2\big](s_+, a_+|s_0)\log\Big[\frac{\lambda\mu_1(s_+, a_+|s_0) + (1-\lambda)\mu_2(s_+, a_+|s_0)}{\lambda\int_a \mu_1(s_+, a|s_0) + (1-\lambda)\int_a \mu_2(s_+, a|s_0)}\Big] \tag{82}$$

$$\leq \lambda\mu_1\log\Big[\frac{\lambda\mu_1(s_+, a_+|s_0)}{\lambda\int_a \mu_1(s_+, a|s_0)}\Big] + (1-\lambda)\mu_2\log\Big[\frac{(1-\lambda)\mu_2(s_+, a_+|s_0)}{(1-\lambda)\int_a \mu_2(s_+, a|s_0)}\Big] \tag{83}$$

$$= \lambda\mu_1\log\Big[\frac{\mu_1(s_+, a_+|s_0)}{\int_a \mu_1(s_+, a|s_0)}\Big] + (1-\lambda)\mu_2\log\Big[\frac{\mu_2(s_+, a_+|s_0)}{\int_a \mu_2(s_+, a|s_0)}\Big] \tag{84}$$

This implies that:

$$\bar{H}(\lambda\mu_1 + (1-\lambda)\mu_2) \geq \lambda\bar{H}(\mu_1) + (1-\lambda)\bar{H}(\mu_2) \tag{85}$$

with equality if and only if $\pi_1(a|s) := \frac{\mu_1(s,a|s_0)}{\int_{a'}\mu_1(s,a'|s_0)} = \frac{\mu_2(s,a|s_0)}{\int_{a'}\mu_2(s,a'|s_0)} := \pi_2(a|s)$. This concludes the proof of the η-weighted strict concavity w.r.t the set of measures μ.

F.4 DATA AUGMENTATION

In this section, we provide the proof of Proposition 9. We start by noticing that $V(D, G)$ is the loss function used by conditional generative adversarial neural networks Mirza & Osindero (2014), which minimum w.r.t the discriminator is achieved for the optimal Bayes classifier Goodfellow et al. (2014):

$$D^*(s, a|s_0) = \frac{P^\eta_\pi(s, a|s_0)}{P^\eta_\pi(s, a|s_0) + G(s, a|s_0)} \tag{86}$$

where $G(s, a|s_0)$ is the probability of generating (s, a) using the generator G. From this, we can re-write the generator's loss against an infinite capacity (optimal) discriminator as:

$$V(D^*, G) = D_{KL}(P^\eta_\pi(s, a|s_0)\|\frac{P^\eta_\pi(s, a|s_0)}{P^\eta_\pi(s, a|s_0) + G(s, a|s_0)}) + D_{KL}(G(s, a|s_0)\|\frac{P^\eta_\pi(s, a|s_0)}{P^\eta_\pi(s, a|s_0) + G(s, a|s_0)}) - \log(4) \tag{87}$$

$$= 2D_{JSC}(G(s, a|s_0)\|P^\eta_\pi(s, a|s_0)) - \log(4) \tag{88}$$

where D_{KL} is the KL divergence, and D_{JSC} is the Jenson-Shannon divergence. A global minimum is achieved when G^*:

$$G^*(s, a|s_0) = P^\eta_\pi(s, a|s_0). \tag{89}$$

This concludes the proof as it implies that $(\tilde{D} = \frac{1}{2}, \tilde{G} = P^\eta_\pi)$ is a Nash-equilibrium

F.5 PARTICULAR SETTINGS OF INTEREST

In this section we address the claims stated in section 4.

PROOF OF PROPOSITION 7:

Notice that in this setting:

$$\max_c L(\pi, c) = \max_{c \in \mathcal{C}_{linear}} \int_{s_0} p_0(s_0) d_c(\pi \| \pi_E)(s_0) \tag{90}$$

$$= \max_{c \in \mathcal{C}_{linear}} \int_{s_0} p_0(s_0) \big[\mu_\pi(s_+, a_+|s_0) - \mu_{\pi_E}(s_+, a_+|s_0)\big] c(s_+, a_+) \tag{91}$$

$$= \max_{w \text{ with } \|w\|_2 \leq 1} \int_{s_0} p_0(s_0) \big[\mu_\pi(s_+, a_+|s_0) - \mu_{\pi_E}(s_+, a_+|s_0)\big] \sum_i w_i f_i(s_+, a_+) \tag{92}$$

$$= \max_{w \text{ with } \|w\|_2 \leq 1} \sum_i w_i \int_{s_0} p_0(s_0) \big[\mu_\pi(s_+, a_+|s_0) - \mu_{\pi_E}(s_+, a_+|s_0)\big] f_i(s_+, a_+) \tag{93}$$

$$= \Big\| \int_{s_0} p_0(s_0) \big[\mu_\pi(s_+, a_+|s_0) - \mu_{\pi_E}(s_+, a_+|s_0)\big] f_i(s_+, a_+) \Big\|_2 \tag{94}$$

$$= \Big\| \mathbb{E}_{\mu_\pi}[f] - \mathbb{E}_{\mu_{\pi_E}}[f] \Big\|_2 \tag{95}$$

This concludes the proof.

PROOF OF PROPOSITION 8:

In this case, we notice the following:

$$\max_c L(\pi, c) = \max_{c \in \mathcal{C}_{convex}} \int_{s_0} p_0(s_0) d_c(\pi \| \pi_E)(s_0) \tag{96}$$

$$= \max_{c \in \mathcal{C}_{convex}} \int_{s_0} p_0(s_0) \big[\mu_\pi(s_+, a_+|s_0) - \mu_{\pi_E}(s_+, a_+|s_0)\big] c(s_+, a_+) \tag{97}$$

$$= \max_{w_i > 0 \text{ with } \sum w_i = 1} \int_{s_0} p_0(s_0) \big[\mu_\pi(s_+, a_+|s_0) - \mu_{\pi_E}(s_+, a_+|s_0)\big] \sum_i w_i f_i(s_+, a_+) \tag{98}$$

$$= \max_{w_i > 0 \text{ with } \sum w_i = 1} \sum_i w_i \int_{s_0} p_0(s_0) \big[\mu_\pi(s_+, a_+|s_0) - \mu_{\pi_E}(s_+, a_+|s_0)\big] f_i(s_+, a_+) \tag{99}$$

$$= \max_i \int_{s_0} p_0(s_0) \big[\mu_\pi(s_+, a_+|s_0) - \mu_{\pi_E}(s_+, a_+|s_0)\big] f_i(s_+, a_+) \tag{100}$$

$$= \max_i \mathbb{E}_{\mu_\pi}[f_i] - \mathbb{E}_{\mu_{\pi_E}}[f_i] \tag{101}$$

This concludes the proof.

PROOF OF PROPOSITION 4:

We start by re-writing the cost function as an expectation with respect to the occupancy measure μ_π using Proposition 10:

$$\psi_{GAN}(c) = \begin{cases} \int_{s,a,s_0} p_0(s_0) \mu_{\pi_E}(s, a|s_0) g(c(s, a)) & \text{if } c < 0 \\ +\infty & \text{otherwise} \end{cases} \tag{102}$$

With this, we can rewrite the objective function $L(\pi, c)$ in this setting as follows:

$$L(\pi, c) = -\Omega(\pi) - \psi(c) + \int_{s_0} p_0(s_0) d_c(\pi \| \pi_E)(s_0) \tag{103}$$

$$= -\Omega(\pi) + \int_{s_0} p_0(s_0) \int_{s,a} \Big[\mu_{\pi_E}(s, a|s_0) g(c(s, a)) + \big(\mu_\pi(s, a|s_0) - \mu_{\pi_E}(s, a|s_0)\big) c(s, a) \Big] \tag{104}$$

Notice that this is the same objective as the one used in GAIL [Ho & Ermon (2016) Appendix A.2], where we compute expectations with respect to μ_π while the do it with respect to ρ_π. Using the same change of variable, we can obtain the folowing:

$$\max_c L(\pi, c) = \max_{D \in (0,1)^{\mathcal{S} \times \mathcal{A}}} -\Omega(\pi) + \int_{s_0} p_0(s_0) \int_{s,a} \mu_\pi(s, a|s_0) \log(D(s, a) - \mu_{\pi_E}(s, a|s_0) \log(1 - D(s, a)) \tag{105}$$

This concludes the proof of Proposition 4, as we can state using Proposition 10:

$$\mathrm{RL}^\eta_\Omega \circ \mathrm{IRL}^\eta_\psi(\pi_E) = \arg\min_\pi \max_c L(\pi, c) \tag{106}$$

$$= \arg\min_\pi -\Omega(\pi) + \max_{D \in (0,1)^{\mathcal{S} \times \mathcal{A}}} \mathbb{E}^\eta_{p_0, \pi}[\log D] - \mathbb{E}^\eta_{p_0, \pi_E}[\log(1 - D))] \tag{107}$$