DROID: Learning from Offline Heterogeneous Demonstrations via Reward-Policy Distillation

Anonymous Author(s)
Affiliation
Address
email

Abstract:

Offline Learning from Demonstrations (OLfD) is valuable in domains where trial-and-error learning is infeasible or specifying a cost function is difficult, such as robotic surgery, autonomous driving, and path-finding for NASA's Mars rovers. However, two key problems remain challenging in OLfD: 1) *heterogeneity*: demonstration data can be generated with diverse preferences and strategies, and 2) *generalizability*: the learned policy and reward must perform well beyond a limited training regime in unseen test settings. To overcome these challenges, we propose Dual Reward and policy Offline Inverse Distillation (DROID), where the key idea is to leverage diversity to improve generalization performance by decomposing common-task and individual-specific strategies and distilling knowledge in both the reward and policy spaces. We ground DROID in a novel and uniquely challenging Mars rover path-planning problem for NASA's Mars Curiosity Rover. We also curate a novel dataset along 163 Sols (Martian days) and conduct a novel, empirical investigation to characterize heterogeneity in the dataset. We find DROID outperforms prior SOTA OLfD techniques, leading to a 26% improvement in modeling expert behaviors and 92% closer to the task objective of reaching the final destination. We also benchmark DROID on the OpenAI Gym Cartpole environment and find DROID achieves 55% (significantly) better performance modeling heterogeneous demonstrations.

Keywords: Learning from Heterogeneous Demonstration, Network Distillation, Offline Imitation Learning

1 Introduction

Deep Reinforcement Learning (Deep RL) has achieved great success in generating high-performance continuous control behaviors. However, Deep RL requires a high-fidelity simulator or reward-annotated dataset [1, 2, 3, 4, 5, 6, 7, 8]. For example, consider the Mars Path-Planning (MPP) problem where one must construct a path through a series of waypoints for the Mars rover to traverse towards a scientific objective. Not only is this challenging due to the chaotic terrain but also due to factors such as physical limitations of the rover's capabilities and unobservable terrain information [9, 10]. Expert human Rover Planners (RPs) at NASA design paths under time and safety constraints based on their expertise crafted over years of experience – knowledge that has yet to be codified and otherwise difficult to capture [9]. Efforts have been made to automate the process with symbolic and connectionist (e.g., Deep RL) approaches [11, 12, 13]. However, these methods do not match the human RPs' success because it is difficult to codify experts' knowledge into a cost function [13, 14] and require a large homogeneous dataset to learn robust policies [9]. Such challenges not only exist in the MPP problem but are prevalent in other robotic applications such as surgery, search and rescue, self-driving, and elderly care [15, 16, 17, 18, 19, 20, 21].

Submitted to the 7th Conference on Robot Learning (CoRL 2023). Do not distribute.

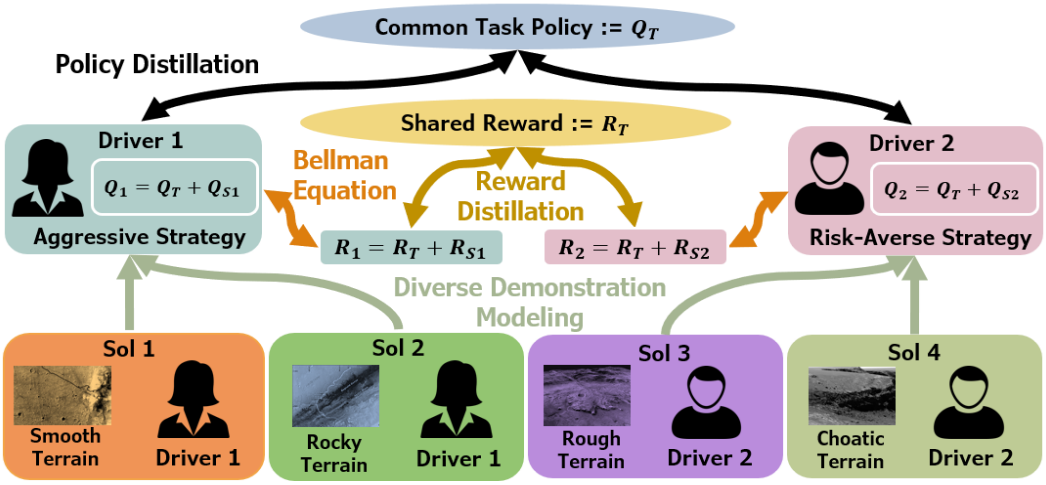

Figure 1: This figure shows an illustration of the proposed algorithm, DROID, in the MPP problem, which performs knowledge distillation across heterogeneous domain settings and expert strategies to a common task policy and a shared task reward.

Learning from Demonstration (LfD) is a promising paradigm to address this challenge for robotics: LfD methods learn by having users demonstrate the desired behavior on the robot, removing the need for cost function specification [22]. However, most LfD approaches (e.g., Inverse Reinforcement Learning, IRL [23, 24]), are limited by the need for many environment interactions [25]. Particularly with robotic applications, the exploratory environment interaction could be costly (e.g., damaged or lost rovers), unethical (e.g., in surgery), or unsafe. Appropriately, Offline LfD (OLfD) has been proposed as framework which allows for training a robot policy solely from pre-recorded demonstrations with no assumption about a viable simulator [26].

OLfD relaxes the requirement for a reward function and a simulator, it however faces several algorithmic challenges that limit its full potential [26, 27]. First, a key challenge for OLfD is heterogeneity within an offline demonstration set. Each expert has individual preferences (stemming from varying cognitive biases [28] or different latent goals [29]) for accomplishing a given task [30, 31]. If the LfD algorithm assumes homogeneity about heterogeneous data, the robot may fail to infer the expert intention [32, 28] and the learned policy may perform arbitrarily poorly [28, 33]. On the other hand, modeling heterogeneous behaviors separately is data-inefficient and prone to overfitting. Thus, leveraging a collection of experts alongside modeling individual-specific preferences can achieve personalization [31, 34] while avoiding the curse of dimensionality [35]. The second critical challenge is learning from a limited dataset [36], which is further complicated by the presence of heterogeneity. The limited data makes it difficult for the learned policy and reward functions to capture the user's latent intentions and to generalize beyond the demonstrated setting [15, 37, 38, 39].

In this paper, we propose a novel OLfD approach, **D**ual **R**eward and Policy **O**ffline **I**nverse **D**istillation (DROID), that simultaneously distills a common *task policy* and *reward* from diverse demonstrators, while modeling individual preferences along both *strategy-specific policies* and *rewards*. This approach allows us to extract an unbiased task objective while understanding various styles of accomplishing the task (Figure 1). Our contributions are three-fold:

1. We curate a novel dataset with RP-designed Mars Curiosity Rover paths for 163 Sols (Martian days) covering various terrains on Mars. We conduct a novel, empirical investigation to characterize heterogeneity in the dataset, motivating the need for a OLfD approach robust to heterogeneity.
2. We propose DROID, which simultaneously distills knowledge through the learned policy and reward. We also introduce two improvements (Augmented Regularization & Reward Maximization) to the underlying IRL algorithm [15] to improve generalization performance.
3. We show DROID achieves 55% better modeling performance (measuring the distance between expert demonstration and generated trajectory) in Cartpole than previous SOTA. On the MPP problem, we also find DROID outperforms SOTA and gets 92% closer to the goal point (an important objective in the Mars path planning domain).

2 Related Work

In this section, we discuss related works in offline learning from heterogeneous demonstration, reward and policy distillation, and the Mars Path Planning problem. There has been extensive work in LfD in robotics and learning from heterogenous demonstrations [29, 31, 33, 34, 35, 40, 41, 42, 43] but our focus is offline learning from diverse demonstrations (for safety-constrained or limited data regimes), an unexplored problem setting.

Offline LfD Despite the abundance of OLfD approaches, few previous methods model heterogeneity within demonstrations [44, 45, 46, 47, 48], which could cause the learned reward function and policy to fail at generalizing beyond the original demonstrated setting [37, 38] and capturing the personalized intention of the expert [39]. Compared with AVRIL [15], DROID explicitly models heterogeneity with leverages additional enhancements to achieve better OLfD performance.

Reward and Policy Distillation Several frameworks consider commonalities between reward functions across heterogeneous demonstrations [31, 34, 41]. However, these methods rely on online interactions with the environment, which is infeasible in many robotic domains. Policy distillation has been studied to improve policy transfer performance [49, 50]. However, DROID is the first to study simultaneous reward and policy distillation, particularly in the challenging setting of OLfD.

Mars Path Planning For NASA, it requires a large amount of human effort for planning and validating commands to the Mars Curiosity Rover [9]. Current approaches such as blind driving, Visual Odometry, and Autonomous Navigation (AutoNav) modes [51] do not consider all hazards that humans deem dangerous to the rover [52]. Hedrick et al. [53] proposes efficient Martian path planning and Rover-IRL [54] learns a cost function from demonstration but both fail to plan under missing/occluded terrain maps which is a key problem in the Mars domain [9].

3 Preliminaries

In this section, we introduce preliminaries on Markov Decision Processes (MDP), Offline Learning from Demonstration (OLfD), and Multi-Strategy Reward Distillation (MSRD).

Markov Decision Process – A MDP, M, is a 6-tuple, $\langle \mathbb{S}, \mathbb{A}, R, T, \gamma, \rho_0 \rangle$. $\mathbb{S}$ and $\mathbb{A}$ correspond to the state / action space, $R(s,a)$ the reward, and $T(s'|s,a)$ the transition probability for state s' after performing action a in state s. $\gamma \in (0,1)$ is the discount factor and ρ_0 denotes the initial state distribution. The policy $\pi(a|s)$ represents the probability of choosing action a in state s. The Q-value is defined as $Q^{\pi}_{R}(s,a) = \mathbb{E}_{\pi,T}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t,a_t) | s_0 = s, a_0 = a\right]$, denoting the expected discounted cumulative return following π in the MDP under reward R.

Offline Learning from Demonstration – IRL considers an MDP sans reward function (MDP\R) and infers the reward function R based on a set of demonstration trajectories $\mathcal{D} = \{\tau_1, \tau_2, \cdots, \tau_N\}$, where N is the number of demonstrations. Our method leverages Approximate Variational Reward Imitation Learning (AVRIL) [55] as the underlying IRL approach. AVRIL considers a distribution over the reward function and approximates the posterior, $p(R)$, with a variational distribution, $q_\phi(R)$. AVRIL introduces a second variational approximation for $Q^{\pi_E}_{R}(s,a)$ with $Q_\theta(s,a)$ and ensures the variational reward distribution, $q_\phi(R)$, is consistent with the variational Q function, $Q_\theta(s,a)$, by Bellman equation. We show the final loss function of AVRIL in Equation 1.

$$L_{\text{AVRIL}}(\theta,\phi) = \sum_{(s,a,s',a')\in\mathcal{D}} \log\left(\frac{\exp(\beta Q_\theta(s,a))}{\sum_{b\in A}\exp(\beta Q_\theta(s,b))}\right) - D_{\text{KL}}(q_\phi(R(s,a))\|p(R(s,a))) + \lambda \log q_\phi(Q_\theta(s,a) - \gamma Q_\theta(s',a')) \tag{1}$$

Multi-Strategy Reward Distillation – We propose DROID's reward distillation approach off a previous reward distillation framework, MSRD [34]. MSRD decomposes the per-strategy reward, R_i, for strategy i, as a linear combination of a common task reward and a strategy-only reward with neural network parameters ϕ_{Task} and $\phi_{\text{S}-i}$: $R_i = R_{\phi_{\text{Task}}} + R_{\phi_{\text{S}-i}}$. MSRD leverages a regularization loss to distill common knowledge into ϕ_{Task} and retains personalized information in $\phi_{\text{S}-i}$.

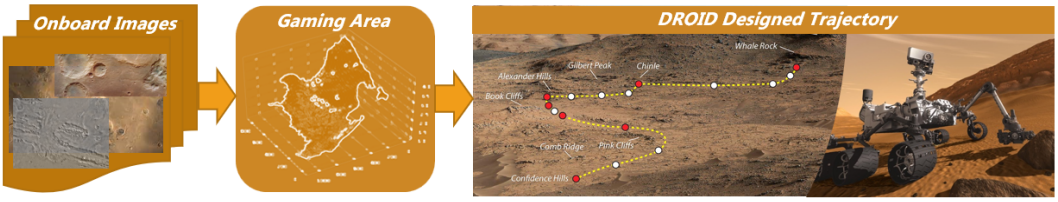

Figure 2: This figure shows the dataset curation process for the MPP problem. We unify the height maps created by onboard cameras into a single "gaming area" (middle figure) and then plan the driving path based on features calculated on the gaming area height map.

4 Mars Curiosity Rover Path Planning Problem

In this section, we introduce the Mars Path Planning (MPP) problem and how we 1) curate the dataset, 2) construct an MDP for OLfD algorithms, 3) analyze heterogeneity present across RPs.

Dataset Curation The raw data consists of height maps created by photos captured by the Mars Rover across multiple Sols (Martian days). The multi-resolution height maps are processed into a single 64x64 "gaming area" by interpolation of overlapping height maps and scaling along each axis (Figure 2). The processed gaming area is then used to calculate nine features identified by RPs: (1) distance to goal point, (2) unknown data percentage, (3) average roughness, (4) maximum roughness, (5) average pitch, (6) average roll, (7) maximum pitch, (8) maximum roll, and (9) turning angle. More details for the features are available in Supplementary.

MDP Problem Setup We create a novel formulation to convert the MPP problem into an MDP: a state contains the terrain information of the Sol and the current and target locations for the path planning. The action space, A, consists of all possible next waypoints in the gaming area. The reward function is constructed as a function of features associated with the path specified by a on the terrain s: $R(s,a) = f(\psi(s,a))$ where $\psi : (S \times A) \to \mathbb{R}^9$ is the path feature mapping.

Analysis of Heterogeneity We perform a PERMANOVA test with $\alpha = 0.05$ and the Holm method for the correction of multi-tests with the 37 different RPs to answer whether different strategies exist among drivers. The test shows significant differences along seven pairs of RPs, particularly along the paths designed by RP 26 with respect to 5 other RPs, as shown in Figure 3 with TSNE [56] dimension reduction to reduce features into two dimensions (Further explanation provided in the supplementary). This result shows heterogeneity in the RP-generated paths from differences in domain settings (Sols) and the expert RP strategies, motivating the need for offline learning from *heterogeneous* demonstration approaches.

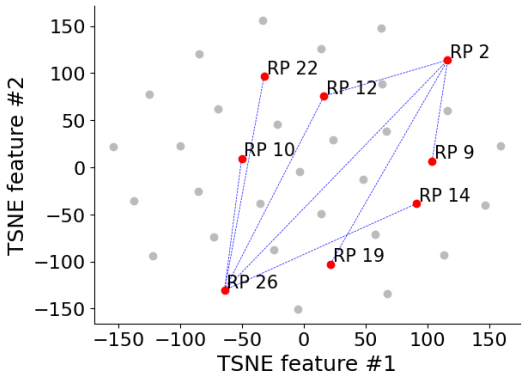

Figure 3: This figure shows heterogeneity in the Mars dataset by visualizing significant differences ($p < .05$ represented by connecting lines) in RP metrics. RP IDs are labeled and marked red.

5 Methods

The challenges of heterogeneity and limited data are prevalent for OLfD in many robotic applications, particularly in the MPP problem as shown in Section 4. To overcome these challenges, we propose our algorithm, **DROID**, for learning high-performing policies by distilling common information across heterogeneous demonstrations in both the policy space and the reward function space.

5.1 Reward Distillation

We propose a novel reward distillation approach tailored for OLfD. We propose to model the reward distributions as mean-field Gaussian distributions partitioned on each state-action pair and let the reward neural networks output the mean and standard deviation of the Gaussian distributions. The advantage of Gaussian-distribution models of task reward ($\mathfrak{R}_{\text{Task}}(s,a) \sim$

$q_{\phi_{\text{Task}}}(R) = \mathcal{N}(\mu_{\text{Task}}(s,a), \sigma^2_{\text{Task}}(s,a)))$ and strategy rewards ($\mathfrak{R}_{\text{S}-i}(s,a) \sim q_{\phi_{\text{S}-i}}(R) = \mathcal{N}(\mu_{\text{S}-i}(s,a), \sigma^2_{\text{S}-i}(s,a))$) is that the summation of two Gaussian distributions is still Gaussian distribution, as shown in Equation 2, where $\mathfrak{R}_{\text{i}}$ denotes the random variable for the reward distribution of strategy i.

$$\mathfrak{R}_i(s,a) = \mathfrak{R}_{\text{Task}}(s,a) + \mathfrak{R}_{\text{S}-i}(s,a) \sim \mathcal{N}(\mu_{\text{Task}}(s,a) + \mu_{\text{S}-i}(s,a), \sigma^2_{\text{Task}}(s,a) + \sigma^2_{\text{S}-i}(s,a)) \quad (2)$$

Through this parameterization, we can perform reward distillation on the strategy reward, as shown in Equation 3, where M is the number of strategies and c_τ is the strategy for τ.

$$L_{\text{RD}}(\{\phi_{\text{S}-i}\}_{i=1}^M; \mathcal{D}) = \mathbb{E}_{(\tau,c_\tau)\in D}\left[||\mu_{\text{S}-c_\tau}(s,a)||\right] \quad (3)$$

Intuitively, L_{RD} pushes the strategy reward to output 0 and therefore encourages any common knowledge to flow to the shared task reward and each individual strategy reward only capture preferences.

5.2 Policy Distillation

We propose DROID leverages commonalities in both reward and policy spaces among heterogeneous demonstrations, to better distill common information in a limited dataset. As policies are implicitly defined via the Q function in AVRIL by $\pi(a|s) = \arg\max_{a\in A} Q(s,a)$, we construct the Q function for each strategy as a combination of task Q function and strategy Q function: $Q_i = Q_{\theta_{\text{Task}}} + Q_{\theta_{\text{S}-i}}$. As such, we propose to regularize the output of the strategy Q-value, $Q_{\text{S}-i}$, as in Equation 4, to encourage common information to be distilled into the task Q-value, Q_{Task}.

$$L_{\text{PD}}(\{\theta_{\text{S}-i}\}_{i=1}^M; \mathcal{D}) = \mathbb{E}_{(\tau,c_\tau)\in D}\left[||Q_{\text{S}-c_\tau}(s,a)||\right] \quad (4)$$

DROID's explicit knowledge distillation across diverse policies aids in improving generalization performance as the shared policy benefits by modeling all demonstrations in the offline dataset.

5.3 Enhancing DROID for Offline LfD

We present two enhancements to construct the inductive bias useful for learning more accurate rewards and better-performing policies in OLfD.

Improvement 1: Augmenting Dataset for Regularization. We introduce an augmented dataset where $\mathcal{D}' = \{(s,b)|\forall s \in \mathcal{D}, b \in \mathbb{A}\}$, for regularizing $q_\phi(R(s,a))$ to a prior $p(R(s,a))$. By extending the operation of D_{KL} to be on the entire action space, we are encouraging a more conservative estimate of the reward for any action that is not taken by the demonstrator, following the pessimistic principle in OLfD [57].

$$L^+_{\text{KL}}(\phi) = \sum_{s\in\mathcal{D}, a\in\mathbb{A}} D_{\text{KL}}(q_\phi(R(s,a))||p(R(s,a))) \quad (5)$$

Intuitively, Equation 5 could be viewed as a data augmentation technique to regularize reward learning across a larger action space. In contrast, AVRIL's variational lower-bound regularizes $L_{\text{KL}}(\phi) = \sum_{(s,a)\in\mathcal{D}} D_{\text{KL}}(q_\phi(R(s,a))||p(R(s,a)))$, on (s,a) samples from $\mathcal{D}$. We provide a lemma to describe the effect in the supplementary.

Improvement 2: Reward Maximization. The second improvement we propose is to maximize the reward given to the demonstrated action, as shown in Equation 6.

$$L_{\text{max-action}}(\phi; \mathcal{D}) = -\sum_{(s,a)\in\mathcal{D}} r(s,a) \quad \text{where } r(s,a) \sim q_\phi(R(s,a)) \quad (6)$$

$L_{\text{max-action}}$ makes the reward learning signal propagate faster instead of relying on the two-stage process of 1) Q function learning (first-term of L_{AVRIL}) and then 2) reward learning by compatibility (third-term of L_{AVRIL}). The two improvements also create a contrastive objective where demonstrated action's reward is maximized while the un-demonstrated action's reward is regularized to be close to the prior distribution. We summarize the loss function for DROID in Equation 7.

$$\begin{aligned} L = &\sum_{(s,a)\in\mathcal{D}} \log \frac{\exp \beta Q_\theta(s,a)}{\sum_{b\in A}\exp(\beta Q_\theta(s,b))} - \sum_{s\in\mathcal{D}, a\in\mathbb{A}} D_{\text{KL}}(q_\phi(R(s,a))||p(R(s,a))) + L_{\text{max-action}}(\phi;\mathcal{D}) \\ &+ \sum_{(s,a,s',a')\in\mathcal{D}} \lambda \log q_\phi(Q_\theta(s,a) - \gamma Q_\theta(s',a')) + L_{\text{RD}}(\{\phi_{\text{S}-i}\}_{i=1}^M;\mathcal{D}) + L_{\text{PD}}(\{\theta_{\text{S}-i}\}_{i=1}^M;\mathcal{D}) \end{aligned} \quad (7)$$

6 Results

In this section, we show DROID achieves strong performance in both the CartPole environment [58] and the more difficult Mars Path Planning problem. We test DROID against the baseline techniques of: a) AVRIL Batch (which assumes homogeneity by training a single LfD model on all expert data), b) AVRIL Single (which handles heterogeneity without knowledge sharing by training separate IRL models for each expert), and c) MSRD (which performs reward distillation). Given the key challenges of *heterogeneity* and *generalization (along both the learned policy and reward)*, we focus our analysis around three questions:

Q1: Diverse Demonstration Modeling – How well does DROID perform at modeling latent expert preferences from heterogeneous demonstrations in offline LfD?

Q2: Policy Performance – How well can the learned policies perform in an unseen holdout test dataset (e.g., modeling unseen demonstrations in CartPole, unseen terrains in MPP)?

Q3: Reward Generalizability – How successful are the learned rewards in encoding experts' latent objectives and inducing high-performing downstream policies?

6.1 Cartpole

We evaluate DROID's trajectories against baselines on four metrics: Frechet Distance [59], KL Divergence [60], Undirected Hausdorff Distance [61], and Average Log Likelihood of expert demonstration under the learned policy. We train each technique on a heterogeneous dataset of 60 trajectories from 20 distinct strategies generated by jointly optimizing an environment reward and a diversity reward from DIAYN [62]. The diverse strategies include swinging to different ends of the track and oscillating at varying periodicity. More experiment details and videos of demonstrations and learned policies are provided in the supplementary.

Q1: Diverse Demonstration Modeling. Table 1 (left) summarizes the results of modeling and imitation on the training demonstrations. We find DROID performs significantly ($p < .05$) better on the KL Divergence (19%) and Frechet Distance (32%) metrics compared to the best baselines, showing that DROID models heterogeneous behaviors better and minimizes deviation between the learned policy's behavior and the expert demonstrations.

Q2: Policy Performance. We study how well each method's policy performs on unseen demonstrations of a given strategy. Results in Table 1 section "Policy Performance" show DROID significantly outperforms ($p < .05$) baselines along KL Divergence (30%) and Undirected Hausdorff (55%). Especially, DROID's performance gain over MSRD being larger on generalization than imitation shows the policy distillation in DROID is essential to learn a generalizable policy.

Table 1: This table shows performance comparisons and significance of DROID and baselines in Cartpole (left) and MPP (right). Bold denotes the best-performing models for the metric. * denotes significance of $p < .05$ against the second-best approach.

CartPole

Benchmark Method	KL Divergence	Frechet Distance	Undirected Hausdorff	Log Likelihood
Diverse Demonstration Modeling ($n = 40$)				
AVRIL Batch	7.608	0.933	0.729	-48.113
AVRIL Single	10.051	1.294	0.895	-48.910
MSRD	7.479	0.621	0.476	-40.453
DROID (ours)	**6.047***	**0.425***	**0.261**	**-37.948**
Policy Performance ($n = 20$)				
AVRIL Batch	8.367	1.004	0.786	-52.843
AVRIL Single	7.960	1.006	0.717	-54.023
MSRD	7.582	0.584	0.458	-44.173
DROID (ours)	**5.271***	**0.412**	**0.207***	**-38.057**
Reward Generalizability ($n = 20$)				
AVRIL Batch	8.923	**1.197**	**1.152**	-180.305
AVRIL Single	9.017	1.418	1.280	-178.544
MSRD	8.368	1.403	1.274	-179.694
DROID (ours)	**8.048**	1.441	1.336	**-175.528***

Mars Path Planning

Benchmark Methods	Undirected Hausdorff	Distance from Waypoint	Final Distance	Log Likelihood
Diverse Demonstration Modeling ($n = 114$)				
AVRIL Batch	8.825	10.517	1.428	-9.129
AVRIL Single	10.099	8.445	7.356	-15.662
MSRD	8.162	7.580	4.476	-27.667
DROID (ours)	**6.780***	**4.592**	**0.070***	**-7.261***
Policy Performance ($n = 49$)				
AVRIL Batch	8.387	10.020	3.878	-22.623
AVRIL Single	8.336	8.888	5.951	**-17.357**
MSRD	9.732	8.886	7.462	-148.964
DROID (ours)	**6.144***	**6.407**	**0.277***	-18.483
Reward Generalizability ($n = 49$)				
AVRIL Batch	9.385	9.501	1.823	-15.501
AVRIL Single	9.396	9.502	1.925	-15.653
MSRD	8.799	8.867	1.268	**-14.515**
DROID (ours)	**8.240***	**8.764**	**0.433***	-15.050

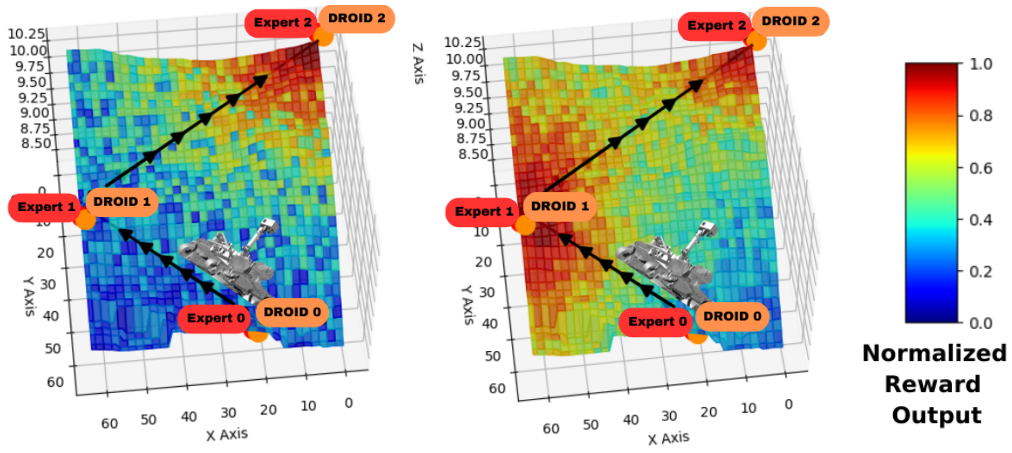

Figure 4: This figure visualizes actual 3D Mars terrain map with DROID's mean estimate of the learned task (left) and strategy (right) reward for Sol 2163. X-axis and Y-axis correspond to the surface coordinates, Z-axis corresponds to elevation, and heatmap coloring is the normalized reward output. The black line with arrows interlayed represents the path of the rover. The orange labels are DROID's found waypoints and the red labels are ground-truth demonstration waypoints (Point 0 is the starting point, Point 1 is the intermediate waypoint, and Point 2 is the final waypoint selected).

Q3: Reward Generalizability. As a further analysis of generalization performance, we study how successful the learned reward functions are at inducing high-performing policies downstream. We train offline RL policies with CQL [63] and compare performances on the holdout test set. The results in Table 1 section "Reward Generalizability" show DROID achieves significantly better ($p < .05$) performance on log-likelihood (underperforms on Frechet Distance and Undirected Hausdorff), showing DROID's reward function can induce a similarly well-performing policy with a stronger performance in expert behaviors modeling.

6.2 Mars Path Planning

With the success of DROID on Cartpole, we further test it against benchmarks on the more challenging MPP problem, where RPs optimize for a complex objective considering goal locations, strategies, and safety constraints. Since there is no clear ground truth reward in the MPP problem, we study the performance along four metrics: Distance from Waypoint (distance away from expert waypoints), Final Distance (distance from desired goal point), Undirected Hausdorff Distance (how closely the generated path and expert path align), and Average Log Likelihood (how well the learned models expert's trajectory). We train each technique on a dataset of 163 distinct sols (each with three waypoints: start point, midpoint, and goal point) from 37 RPs. Note data is extremely limited as each RP is associated with only 2-5 Sols and we treat each RP as a unique expert strategy for each technique. We retain a holdout test dataset representing 20% of Sols which is used to evaluate the generalization performance. More experiment details are in the supplementary.

Q1: Diverse Demonstration Modeling. We show in Table 1 (right) that DROID is more successful at the imitation objective with respect to baseline approaches. DROID outperforms ($p < .05$) baselines by 95% on reaching the goal point (Final Distance) along with 20% better modeling on the strategic preference (Log Likelihood). Policy distillation ensures DROID accomplishes the task goal while per-strategy decomposition allows it to model expert waypoint preferences well.

Q2: Policy Performance. On a holdout set of unseen Sols, Table 1 shows that DROID achieves better ($p < .05$) performance on the Undirected Hausdorff (26%) along with Final distance (92%) compared to best baselines. Despite limited and heterogeneous data, DROID's learned policy generalizes well to achieve task performance and model expert preferences closely compared to baselines.

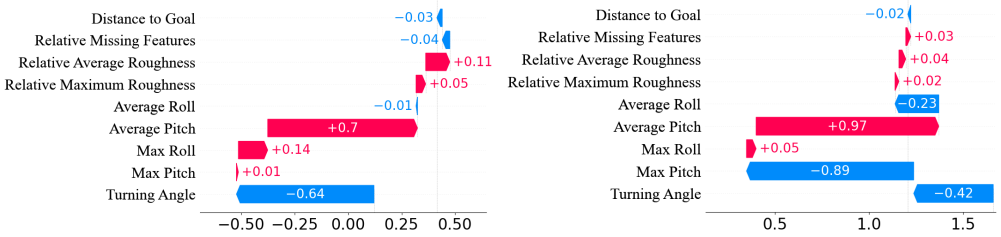

Figure 5: This figure shows Shapley values (the contribution of each of the composite features to the model's reward estimate) for the learned rewards of RP 1 (left) and 5 (right) evaluated on Sol 2030's demonstrated path.

Q3: Reward Generalizability. Similar to the Cartpole experiment, we evaluate how successful the learned reward functions are at inducing high-performing policies downstream. We train CQL policies. The results in Table 1 section "Reward Generalizability" show DROID's learned reward successfully induces policies with significantly better ($p < .05$) performance along the Undirected Hausdorff metric (12%) and the Final Distance metric (65%).

6.2.1 Qualitative Analysis

In this section, we present qualitative analysis of DROID, especially in understanding RPs' preferences in the domain of MPP. We visualize the task reward and strategy reward learned by DROID for a randomly selected Sol (Sol 2163) in Figure 4 left and right, respectively. We observe that the task reward encodes the common goal of converging at the goal point by giving high rewards to the goal point area. In contrast, the strategy reward correctly identifies the midpoint preference. DROID can successfully decompose the shared goal along with modeling a given RP's latent strategy on unseen domains by highlighting their RP-specific cost function on a terrain map.

We further perform a Shapley value analysis [64] on each RPs' learned reward function. The analysis measures how adding a feature would change the prediction output and is helpful in comparing the relative importance different RPs place on features, even on Sols they have not explicitly planned on. As shown in Figure 5, we evaluated two randomly selected RPs (1 and 5) on Sol 2030. We observe that all drivers value that the path has a low pitch. However, RP 1 prefer to have a smaller turning angle, while RP 5 values a lower pitch more. This demonstrates how DROID can model heterogeneous strategies and quantify the influence of specific features on the modeled objective function. We can identify why certain RPs like or dislike a given path and understand which features contribute to that assessment.

7 Conclusion, Limitations, and Future Work

In this paper, we introduce an OLfD technique, DROID, that expands the applicability of OLfD with heterogeneous and limited data by a novel decomposition of the policy and reward models. Our results on both simulated and real-world data demonstrate that DROID outperforms SOTA methods, particularly in capturing difficult-to-articulate knowledge from rover path planners at NASA.

There are several limitations with DROID. Firstly, DROID assumes experts have stationary preferences across demonstrations, which may be violated if experts gain more knowledge about the domain and adapt their strategies. Secondly, DROID requires retraining to adapt to new data and cannot improve its model in an online fashion. Thirdly, in the MPP domain, we extract nine features from height maps, which may not encompass all features an RP considers when making plans. In future work, we plan to explore modeling nonstationary strategies for demonstrators along with continual learning in the DROID model. In the MPP domain, we plan to test DROID's utility with RP's daily workflow, explore automatic feature extraction for the Mars terrain information, and leverage DROID's interpretability to gain insights during new-RP training procedures.

References

[1] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, and S. Levine. Soft actor-critic algorithms and applications. *CoRR*, abs/1812.05905, 2018.

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

[3] R. Paleja, Y. Niu, A. Silva, C. Ritchie, S. Choi, and M. Gombolay. Learning interpretable, high-performing policies for continuous control problems. *arXiv preprint arXiv:2202.02352*, 2022.

[4] E. Seraj, Z. Wang, R. Paleja, D. Martin, M. Sklar, A. Patel, and M. Gombolay. Learning efficient diverse communication for cooperative heterogeneous teaming. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, pages 1173–1182, 2022.

[5] A. Silva, N. Moorman, W. Silva, Z. Zaidi, N. Gopalan, and M. Gombolay. Lancon-learn: Learning with language to enable generalization in multi-task manipulation. *IEEE Robotics and Automation Letters*, 7(2):1635–1642, 2022. doi:10.1109/LRA.2021.3139667.

[6] E. Seraj, L. Chen, and M. C. Gombolay. A hierarchical coordination framework for joint perception-action tasks in composite robot teams. *IEEE Transactions on Robotics*, 38(1):139–158, 2021.

[7] S. Konan, E. Seraj, and M. Gombolay. Iterated reasoning with mutual information in cooperative and byzantine decentralized teaming. *arXiv preprint arXiv:2201.08484*, 2022.

[8] S. G. Konan, E. Seraj, and M. Gombolay. Contrastive decision transformers. In *6th Annual Conference on Robot Learning*, 2022.

[9] D. M. Gaines, R. C. Anderson, G. B. Doran, W. Huffman, H. Justice, R. M. Mackey, G. R. Rabideau, A. R. Vasavada, V. Verma, T. A. Estlin, L. M. Fesq, M. D. Ingham, M. W. Maimone, and I. A. D. Nesnas. Productivity challenges for mars rover operations, 2016.

[10] A. R. Vasavada. Mission Overview and Scientific Contributions from the Mars Science Laboratory Curiosity Rover After Eight Years of Surface Operations. *Space Sci Rev*, 218(3):14, 2022.

[11] D. I. Koutras, A. C. Kapoutsis, A. A. Amanatiadis, and E. B. Kosmatopoulos. Marsexplorer: Exploration of unknown terrains via deep reinforcement learning and procedurally generated environments. *Electronics*, 10(22), 2021. ISSN 2079-9292. doi:10.3390/electronics10222751. URL https://www.mdpi.com/2079-9292/10/22/2751.

[12] R. Hu and Y. Zhang. Fast path planning for long-range planetary roving based on a hierarchical framework and deep reinforcement learning. *Aerospace*, 9(2), 2022. ISSN 2226-4310. doi:10.3390/aerospace9020101. URL https://www.mdpi.com/2226-4310/9/2/101.

[13] J. Zhang, Y. Xia, and G. Shen. A novel deep neural network architecture for mars visual navigation. *CoRR*, abs/1808.08395, 2018. URL http://arxiv.org/abs/1808.08395.

[14] T.-H. Cheng, C.-P. Wei, and V. S. Tseng. Feature selection for medical data mining: Comparisons of expert judgment and automatic approaches. In *19th IEEE symposium on computer-based medical systems (CBMS'06)*, pages 165–170. IEEE, 2006.

[15] A. J. Chan and M. van der Schaar. Scalable bayesian inverse reinforcement learning, 2021.

[16] F. Jarboui and V. Perchet. Offline inverse reinforcement learning. *CoRR*, abs/2106.05068, 2021. URL https://arxiv.org/abs/2106.05068.

[17] S. A. Murphy, M. J. van der Laan, and J. M. Robins. Marginal Mean Models for Dynamic Regimes. *J Am Stat Assoc*, 96(456):1410–1423, Dec 2001.

[18] A. Kumar, A. Singh, S. Tian, C. Finn, and S. Levine. A workflow for offline model-free robotic reinforcement learning. *CoRR*, abs/2109.10813, 2021. URL https://arxiv.org/abs/2109.10813.

[19] X. Fang, Q. Zhang, Y. Gao, and D. Zhao. Offline reinforcement learning for autonomous driving with real world driving data. In *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, pages 3417–3422, 2022. doi:10.1109/ITSC55140.2022.9922100.

[20] R. F. Prudencio, M. R. O. A. Maximo, and E. L. Colombini. A survey on offline reinforcement learning: Taxonomy, review, and open problems. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–0, 2023. doi:10.1109/tnnls.2023.3250269. URL https://doi.org/10.1109%2Ftnnls.2023.3250269.

[21] M. Fatemi, M. Wu, J. Petch, W. Nelson, S. J. Connolly, A. Benz, A. Carnicelli, and M. Ghassemi. Semi-markov offline reinforcement learning for healthcare, 2022.

[22] S. Schaal. Learning from demonstration. In M. C. Mozer, M. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9. MIT Press, 1997. URL https://proceedings.neurips.cc/paper/1996/file/68d13cf26c4b4f4f932e3eff990093ba-Paper.pdf.

[23] J. Fu, K. Luo, and S. Levine. Learning robust rewards with adverserial inverse reinforcement learning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[24] A. Hussein, M. M. Gaber, E. Elyan, and C. Jayne. Imitation learning: A survey of learning methods. *ACM Computing Surveys (CSUR)*, 50(2):1–35, 2017.

[25] I. Kostrikov, K. K. Agrawal, D. Dwibedi, S. Levine, and J. Tompson. Discriminator-actor-critic: Addressing sample inefficiency and reward bias in adversarial imitation learning. *arXiv preprint arXiv:1809.02925*, 2018.

[26] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. *CoRR*, abs/2108.03298, 2021. URL https://arxiv.org/abs/2108.03298.

[27] G. Tucker. Tackling open challenges in offline reinforcement learning, Aug 2020. URL https://ai.googleblog.com/2020/08/tackling-open-challenges-in-offline.html.

[28] E. F. Morales and C. Sammut. Learning to fly by combining reinforcement learning with behavioural cloning. In *Proceedings of the International Conference on Machine Learning (ICML)*, page 76, 2004.

[29] S. Nikolaidis, R. Ramakrishnan, K. Gu, and J. Shah. Efficient model learning from joint-action demonstrations for human-robot collaborative tasks. In *2015 10th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 189–196. IEEE, 2015.

[30] L. Chen, R. R. Paleja, M. Ghuy, and M. C. Gombolay. Joint goal and strategy inference across heterogeneous demonstrators via reward network distillation. *CoRR*, abs/2001.00503, 2020. URL http://arxiv.org/abs/2001.00503.

[31] L. Chen, S. Jayanthi, R. Paleja, D. Martin, V. Zakharov, and M. Gombolay. Fast lifelong adaptive inverse reinforcement learning from demonstrations, 2023.

[32] S. Amershi, M. Cakmak, W. B. Knox, and T. Kulesza. Power to the people: The role of humans in interactive machine learning. *AI Magazine*, 35(4):105–120, Dec. 2014. doi:10.1609/aimag.v35i4.2513. URL https://ojs.aaai.org/index.php/aimagazine/article/view/2513.

[33] R. Paleja, A. Silva, L. Chen, and M. Gombolay. Interpretable and personalized apprenticeship scheduling: Learning interpretable scheduling policies from heterogeneous user demonstrations. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6417–6428. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/477bdb55b231264bb53a7942fd84254d-Paper.pdf.

[34] L. Chen, R. R. Paleja, M. Ghuy, and M. C. Gombolay. Joint goal and strategy inference across heterogeneous demonstrators via reward network distillation. In *Proceedings of the International Conference on Human-Robot Interaction (HRI)*, 2020.

[35] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard. Recent advances in robot learning from demonstration. *Annual Review of Control, Robotics, and Autonomous Systems*, 3, 2020.

[36] X. Chen, A. Ghadirzadeh, T. Yu, J. Wang, A. Y. Gao, W. Li, L. Bin, C. Finn, and C. Zhang. Lapo: Latent-variable advantage-weighted policy optimization for offline reinforcement learning. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 36902–36913. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/efb2072a358cefb75886a315a6fcf880-Paper-Conference.pdf.

[37] A. Szot, A. Zhang, D. Batra, Z. Kira, and F. Meier. BC-IRL: Learning generalizable reward functions from demonstrations. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Ovnwe_sDQW.

[38] S. Yue, G. Wang, W. Shao, Z. Zhang, S. Lin, J. Ren, and J. Zhang. Clare: Conservative model-based reward learning for offline inverse reinforcement learning, 2023.

[39] J. Maghakian, P. Mineiro, K. Panaganti, M. Rucker, A. Saran, and C. Tan. Personalized reward learning with interaction-grounded learning (igl), 2023.

[40] A. Correia and L. A. Alexandre. A survey of demonstration learning, 2023.

[41] Y. Li, J. Song, and S. Ermon. Infogail: Interpretable imitation learning from visual demonstrations. *Advances in Neural Information Processing Systems*, 30, 2017.

[42] R. R. Paleja, A. Silva, L. Chen, and M. Gombolay. Interpretable and personalized apprenticeship scheduling: Learning interpretable scheduling policies from heterogeneous user demonstrations. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2020.

[43] L. Chen, R. Paleja, and M. Gombolay. Learning from suboptimal demonstration via self-supervised reward regression. In *Proceedings of Conference on Robot Learning (CoRL)*, 2020.

[44] S. Ross, G. J. Gordon, and J. A. Bagnell. No-regret reductions for imitation learning and structured prediction. *CoRR*, abs/1011.0686, 2010. URL http://arxiv.org/abs/1011.0686.

[45] M. Bain and C. Sammut. A framework for behavioural cloning. *Machine Intelligence 15, Intelligent Agents*, 15, 03 2000.

[46] I. Kostrikov, O. Nachum, and J. Tompson. Imitation learning via off-policy distribution matching. *CoRR*, abs/1912.05032, 2019. URL http://arxiv.org/abs/1912.05032.

[47] I. Kostrikov, K. K. Agrawal, S. Levine, and J. Tompson. Addressing sample inefficiency and reward bias in inverse reinforcement learning. *CoRR*, abs/1809.02925, 2018. URL http://arxiv.org/abs/1809.02925.

[48] D. Jarrett, I. Bica, and M. van der Schaar. Strictly batch imitation learning by energy-based distribution matching, 2021.

[49] A. A. Rusu, S. G. Colmenarejo, C. Gulcehre, G. Desjardins, J. Kirkpatrick, R. Pascanu, V. Mnih, K. Kavukcuoglu, and R. Hadsell. Policy distillation, 2016.

[50] J. Xing, T. Nagata, X. Zou, E. O. Neftci, and J. L. Krichmar. Policy distillation with selective input gradient regularization for efficient interpretability. *ArXiv*, abs/2205.08685, 2022.

[51] S. Daftry, N. Abcouwer, T. Del Sesto, S. Venkatraman, J. Song, L. Igel, A. Byon, U. Rosolia, Y. Yue, and M. Ono. Mlnav: Learning to safely navigate on martian terrains. *IEEE Robotics and Automation Letters*, 7(2):5461–5468, 2022.

[52] E. Hilgemann. How to drive a mars rover, Dec 2020. URL https://medium.com/predict/how-to-drive-a-mars-rover-6f0870b0c8e1.

[53] G. Hedrick, N. Ohi, and Y. Gu. Terrain-aware path planning and map update for mars sample return mission. *IEEE Robotics and Automation Letters*, 5(4):5181–5188, 2020. doi:10.1109/LRA.2020.3005123.

[54] M. Pflueger, A. Agha, and G. S. Sukhatme. Rover-irl: Inverse reinforcement learning with soft value iteration networks for planetary rover path planning. *IEEE Robotics and Automation Letters*, 4(2):1387–1394, 2019. doi:10.1109/LRA.2019.2895892.

[55] A. J. Chan and M. van der Schaar. Scalable bayesian inverse reinforcement learning. *arXiv preprint arXiv:2102.06483*, 2021.

[56] L. Van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[57] L. Shi, G. Li, Y. Wei, Y. Chen, and Y. Chi. Pessimistic q-learning for offline reinforcement learning: Towards optimal sample complexity, 2022.

[58] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[59] K. Toohey and M. Duckham. Trajectory similarity measures. *SIGSPATIAL Special*, 7:43–50, 05 2015. doi:10.1145/2782759.2782767.

[60] S. Kullback and R. A. Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79 – 86, 1951.

[61] F. Hausdorff. *Grundzüge der Mengenlehre*. Chelsea, 1914. https://en.wikipedia.org/wiki/Grundz

[62] B. Eysenbach, A. Gupta, J. Ibarz, and S. Levine. Diversity is all you need: Learning skills without a reward function. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=SJx63jRqFm.

[63] A. Kumar, A. Zhou, G. Tucker, and S. Levine. Conservative q-learning for offline reinforcement learning. *CoRR*, abs/2006.04779, 2020. URL https://arxiv.org/abs/2006.04779.

[64] L. S. Shapley. A value for n-person games. In H. W. Kuhn and A. W. Tucker, editors, *Contributions to the Theory of Games II*, pages 307–317. Princeton University Press, Princeton, 1953.