We thank all reviewers for their valuable feedback. We are encouraged that they found our algorithm novel (**R1**), our paper well-written (**R1**, **R2**, **R3**, **R4**) with sound claims (**R2**), solid theoretical justifications (**R3**), and clear technical expositions (**R4**). We are honored that **R4** recognizes the potential value of our work to the RL community. We provide detailed responses to their major concerns below:

[**R1**, **R4**]: **1. Evaluation on more complex domains.** We appreciate this valuable suggestion. To better illustrate the performance of our approach, we provide more evaluations on the *Humanoid* task (given the limited time constraint), which is a challenging domain with high state-action dimension ($\mathcal{S} \times \mathcal{A} = \mathbb{R}^{376} \times \mathbb{R}^{17}$). The strength of *OPOLO* is more significant in this domain ((Figure 1), while its counterparts can be prone to sub-optimality (*DAC*) or overfitting (*ValueDICEfO*) (see our response **2**).

[**R2**, **R4**]: **2. Solid comparison with stronger baselines.** Following this informative suggestion, we compare with three more baselines: ① *ValueDICE* as **R4** mentioned. We would like to emphasize that *ValueDICE* is a LfD approach which is ***not directly applicable*** to LfO (see Sec 8.8), as it requires the expert actions at our disposal. For fairer comparisons, we implemented its variant ② *ValueDICEfO* (as suggested by **R2**), which replaces ground-truth expert actions with pseudo ones provided by an inverse model. Thanks to **R2**'s valuable suggestions, we also implemented ③ *DACfO*, a variation of *DAC* that learns the discriminator on (s, s') instead of (s, a); Although *ValueDICEfO* and *DACfO* ***have not been investigated by other prior work***, we still found them quite interesting and relevant to our setting. **Results** in Figure 1 (learning efficiency) and Table 2 (asymptotic performance) shows that: *OPOLO* (blue) in general 1) learns ***faster*** than *DACfO* (red), 2) yields ***higher*** asymptotic performance than *DACfO* and *ValueDICEfO* (green), and 3) is more ***robust*** than other off-policy baselines including *ValueDICE* (orange) which uses expert actions. *OPOLO* is the only approach that consistently achieves competitive performance regarding both sample-efficiency and asymptotic performance across all tasks, and is therefore more stable compared with *ValueDICE*. As for the LfO baseline *ValueDICEfO*, its performance compared with *ValueDICE* can be further deteriorated by potential *action-drifts*, as the inferred actions are not guaranteed to recover expertise (see Sec 3.4 and Sec 8.3).

[**R3**]: **3. Comparison with other choices of f-divergence.** Following this valuable suggestion, we evaluated the effects of different f-functions, where $f(x) = \frac{1}{p}|x|^p$, $f^*(y) = \frac{1}{q}|y|^q$, s.t. $\frac{1}{p} + \frac{1}{q} = 1, p, q > 1$, as adopted by *DualDICE* (Nachum'19). We observed that *OPOLO* yields reasonable performance across different f-functions, although our choice ($q = p = 2$) turns out to be most stable. Results using the *Ant* task is illustrated in Figure 2.

[**R2**]: **4. Conceptual resemblance to prior art: *DICE* and the inverse-action regularization.** We appreciate this insightful comment for drawing a nice connection between *OPOLO* and other prior arts. We would like to highlight that: 1) Our approach is inspired by while different from *DICE* , as it is the first work to extend *DICE* to a more challenging scenario (LfO), which is non-trivial especially when the philosophy of *DICE* is not directly applicable to this setting, for which we have provided theoretical analysis (Sec 8.8). 2) Unlike prior art that empirically validated the effects of an inverse-action model, we provide solid interpretations of its functionality, i.e. a *mode-covering* regularizer, by both theoretical derivations and empirical ablation studies.

[**R4**]: **5. Effects of learning discriminator using fresh data.** We appreciate this insightful suggestion. We had similar ideas before, by training discriminator D using *on-policy* data, which did not bring us much benefit in terms of the learning efficiency. We attribute this phenomenon to a *training distribution drift*, i.e. the *on-policy* dataset seen by D differs from the *off-policy* ones used to train π and Q, and the (potential) overfitting of D may cause it forget on how to distinguish stale (off-policy) samples. We consider it analogous to a *catastrophic forgetting* issue.

Env	HalfCheetah	Hopper	Walker	Swimmer	Ant	Humanoid
opolo(-x)	**7632.80±128.88**	3581.85±19.08	3947.72±97.88	257.38±4.28	**5783.57±651.98**	**4699.68±1245.81**
DAC	6900.00±131.24	3534.42±10.27	**4131.05±174.13**	232.12±2.04	5424.28±594.82	2303.97±379.28
DACfO	7035.63±444.14	3522.95±93.15	3033.02±207.63	185.28±2.67	4920.76±872.66	640.49±233.43
ValueDICE	5696.94±2116.94	**3591.37±8.60**	1641.58±1230.73	**262.73±7.76**	3486.87±1232.25	942.47±730.13
ValueDICEfO	4770.37±644.49	3579.51±10.23	431.00±140.87	**265.05±3.45**	75.08±400.87	198.39±65.46

Table 1: Performance after training with 10^6 interaction steps

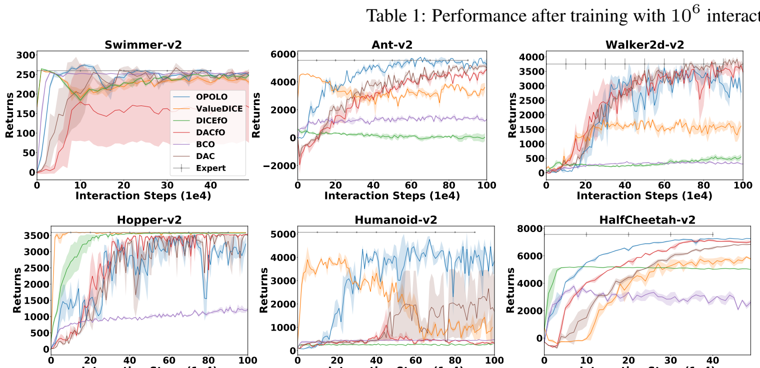

Figure 1: Learning curves averaged over 3 random seeds.

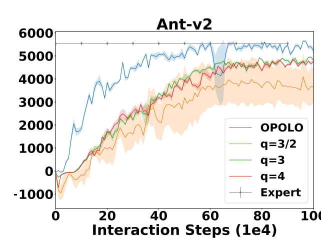

Figure 2: different f-functions.