We thank the reviewers for their close reading of the paper and helpful feedback. We share the desire for more practical demonstrations of the algorithm, and so we've collected results on additional tasks (Figure 1). We observe that DualDICE continues to provide more accurate and stable results compared to the baselines, especially in continuous-action settings. Applying DualDICE to more complex and real-world problems is an exciting direction we are actively exploring.

We are also excited to apply ideas from DualDICE to the policy improvement problem, as mentioned by the reviewers. We are exploring several potential approaches to this problem. For example, one can use the density ratio estimates provided by DualDICE to modify (importance-weight) the off-policy data distribution before passing it to a policy gradient or Q-learning method. Or, one can use the divergence quantity approximated by the DualDICE objective to perform a form of *conservative policy improvement* in a behavior-agnostic fashion. We are actively working on these directions and plan to include them as part of a future work.

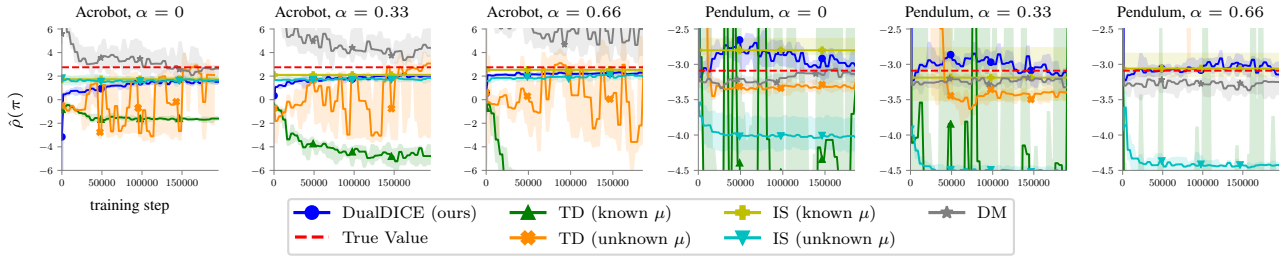

Figure 1: We perform OPE on additional control tasks (Acrobot and Pendulum) using our method compared to a number of baselines. We find that our method continues to perform well against previous OPE methods (baselines not shown perform even worse). Similar to the results in the main paper, we find that the baselines can perform reasonably well on discrete control (Acrobot) but performance degrades when in a continuous control setting (Pendulum). Interestingly some of the baselines perform better on lower α, potentially because this leads to a more diverse off-policy dataset.

Responses to individual reviewer questions are below:

How are these assumptions handled practically, e.g. $d^\pi(s,a) > 0$ implies that $d^D(s,a) > 0$? This assumption is necessary for the theory, but in practice we found it unnecessary to enforce it explicitly. Indeed, in DualDICE we train the density ratio models ν, ζ parameterized by (smooth) neural networks via SGD on batches of data collected from the distribution d^D. Thus, extrapolation would not lead to a "blow-up" in ν-value for unseen s, a ($d^D(s,a) = 0$) . If one so desires, one may explicitly enforce the ratio of occupancy measures to be bounded (see the definition of $\mathcal{C}$ in the main paper), although we found this unnecessary.

What are the x-axes in figure 2 and figure 4? The x-axis is training step, in the context of training neural network function approximators for ν, ζ. We will remedy this in the final draft.

Where did the convex problem from section 3.2 come from? This convex problem is a special case of the variational form of the Ali-Silvey or Csiszár-Morimoto divergences, which include the well-known KL-divergence by using $f(x) = \exp(x-1)$. Our objective in Sec 3.2 using $f(x) = x^2/2$ may be interpreted as a variant of the Pearson χ^2 divergence. We will make the connections more explicit in the paragraph (L206–214).

Why report the median? Are there outliers? This choice was arbitrary. Plotting with the mean plus/minus standard error yields similar plots. We will clarify this in the final draft.

Sutton and Barto (2018) show in section 11.6 (specifically example 11.4) that... two MDPs that generate the same data distribution have different optimal Bellman Errors with different optimal parameters. How do you justify using a Bellman Error-based objective function with function approximation? This is an interesting connection. While our loss function bears a similarity to squared Bellman error, the two are in general not the same, thanks to the dual variables introduced in our objective. Furthermore, the non-learnability in the S&B example, in our opinion, results mostly from the particular form of function approximation that does not include crucial information to distinguish distinct states. We will comment on this with more details in the final version that has more space for elaboration.

The figures are overall too small... In Figure 2 the x axis label is missing. The x-axis is training step. We will increase the size of the figures and improve their overall presentation in the final draft (utilizing the allowed 9^{th} page).

Before turning to the appendix, it was not completely clear to me what were the actual TD and IS algorithms. TD here is based on the COP-TD algorithm (Gelada & Bellemare). IS refers to weighted step-wise importance sampling. We will add these clarifications in the final draft.

It would be good to mention that correcting updates by importance-weighting the entire trajectory does not require the Markov assumption. Thanks for the comment; we will do this in the final version.