Supplementary Material to Trajectory-Based Off-Policy Deep RL

Anonymous Authors[1]

A. Proof of Proposition 1

The weighted importance sampling estimator of the expected cost is given by

$$\hat{J}^{\text{WIS}}(\theta) = \frac{1}{\sum_{i=0}^{M} w(\tau_i, \theta)} \sum_{i=0}^{M} w(\tau_i, \theta) R(\tau_i) \,, \quad (1)$$

as derived in Sec. 3. Talking the derivative with respect to the policy parameters, we obtain the policy gradient formulation from theorem 1 as shown in (7).

B. Experimental Details

In the following section, details about the reference implementations of REINFORCE, TRPO and PPO and their parameter settings are summarized for the benchmark experiments and the ablation study. Information about the benchmark environments is given in Sec. B.2

B.1. Algorithm Configurations

The reference implementations of the benchmark algorithms REINFORCE, TRPO and PPO are from the Garage RL framework (Duan et al., 2016). A hyper-parameter grid search has been conducted for each algorithm and each environment on separate random seeds. The parameter ranges and selected hyper-parameters are indicated in Tab. 1. For the benchmark itself, ten runs have been conducted for each algorithm and each environment on the random seeds (404, 931, 159, 380, 858, 708, 16, 448, 136, 989).

The configuration of the DD-OPG method is summarized in Tab. 1.

B.2. Benchmark Environments

The benchmark environments are cartpole, mountaincar and swimmer from the Garage RL framework. Details about the input and state dimensions, as well as the task horizons are listed in Tab. 2.

Table 1. Algorithm hyper-parameters for the benchmark tasks.

Algorithm	Parameter	Range	Selected
REINFORCE	Batch size	[400, 5000]	5000
	Step size	[0.0001, 0.1]	0.03
TRPO	Batch size	[400, 5000]	5000
	Step size	[0.0001, 0.1]	0.1
PPO	Batch size	[400, 5000]	2000
	Step size	[0.0001, 0.2]	0.2

Algorithm	Parameter	Symbol	Selected
DD-OPG	Temperature	λ	0.1
	Penalty	γ	0.05
	Lengthscale	$\log \Sigma$	$3\mathbf{I}$
	Path buffer	N_{max}	50

Table 2. Information about the benchmark environments.

Environment	Inputs	States	Horizon
Cartpole	1	4	100
Mountaincar	1	2	500
Swimmer	2	13	1000

References

Duan, Y., Chen, X., Houthooft, R., Schulman, J., and Abbeel, P. Benchmarking deep reinforcement learning for continuous control. In *International Conference on Machine Learning (ICML)*, pp. 1329–1338, 2016.

Figure 1. Derivation of the weighted IS policy gradient.

$$
\begin{aligned}
\nabla_\theta \hat{J}^{\mathrm{WIS}}(\theta) &= \nabla_\theta \left(\left(\sum_{i=1}^{N} \frac{p(\tau_i \mid \theta)}{\frac{1}{N}\sum_j p(\tau_i \mid \theta_j)} \right)^{-1} \right) \sum_{i=0}^{N} \frac{p(\tau_i \mid \theta)}{\frac{1}{N}\sum_j p(\tau_i \mid \theta_j)} R(\tau_i) + \\
&\quad \left(\sum_{i=1}^{N} \frac{p(\tau_i \mid \theta)}{\frac{1}{N}\sum_j p(\tau_i \mid \theta_j)} \right)^{-1} \sum_{i=0}^{N} \nabla_\theta \left(\frac{p(\tau_i \mid \theta)}{\frac{1}{N}\sum_j p(\tau_i \mid \theta_j)} R(\tau_i) \right) && (2) \\
&= -\left(\sum_{i=1}^{N} w_i(\theta) \right)^{-2} \left(\sum_{i=1}^{N} \nabla_\theta w_i(\theta) \right) \left(\sum_{i=1}^{N} w_i(\theta) R(\tau_i) \right) + \\
&\quad \left(\sum_{i=1}^{N} w_i(\theta) \right)^{-1} \left(\sum_{i=1}^{N} \nabla_\theta w_i(\theta) R(\tau_i) \right) && (3) \\
&= -\frac{1}{Z^2} \left(\sum_{i=1}^{N} \nabla_\theta w_i(\theta) \right) \left(\sum_{i=1}^{N} w_i(\theta) R(\tau_i) \right) + \frac{1}{Z} \left(\sum_{i=1}^{N} \nabla_\theta w_i(\theta) R(\tau_i) \right) && (4) \\
&= \frac{1}{Z} \left(\sum_{i=1}^{N} \nabla_\theta w_i(\theta) R(\tau_i) - \sum_{i=1}^{N} \nabla_\theta w_i(\theta) \frac{\sum_{i=1}^{N} w_i(\theta) R(\tau_i)}{Z} \right) && (5) \\
&= \frac{1}{Z} \left(\sum_{i=1}^{N} \nabla_\theta w_i(\theta) R(\tau_i) - \sum_{i=1}^{N} \nabla_\theta w_i(\theta) \hat{J}^{\mathrm{WIS}}(\theta) \right) && (6) \\
\nabla_\theta \hat{J}^{\mathrm{WIS}}(\theta) &= \frac{1}{Z} \sum_{i=1}^{N} \nabla_\theta w_i(\theta) \left(R(\tau_i) - \hat{J}^{\mathrm{WIS}}(\theta) \right) && (7)
\end{aligned}
$$