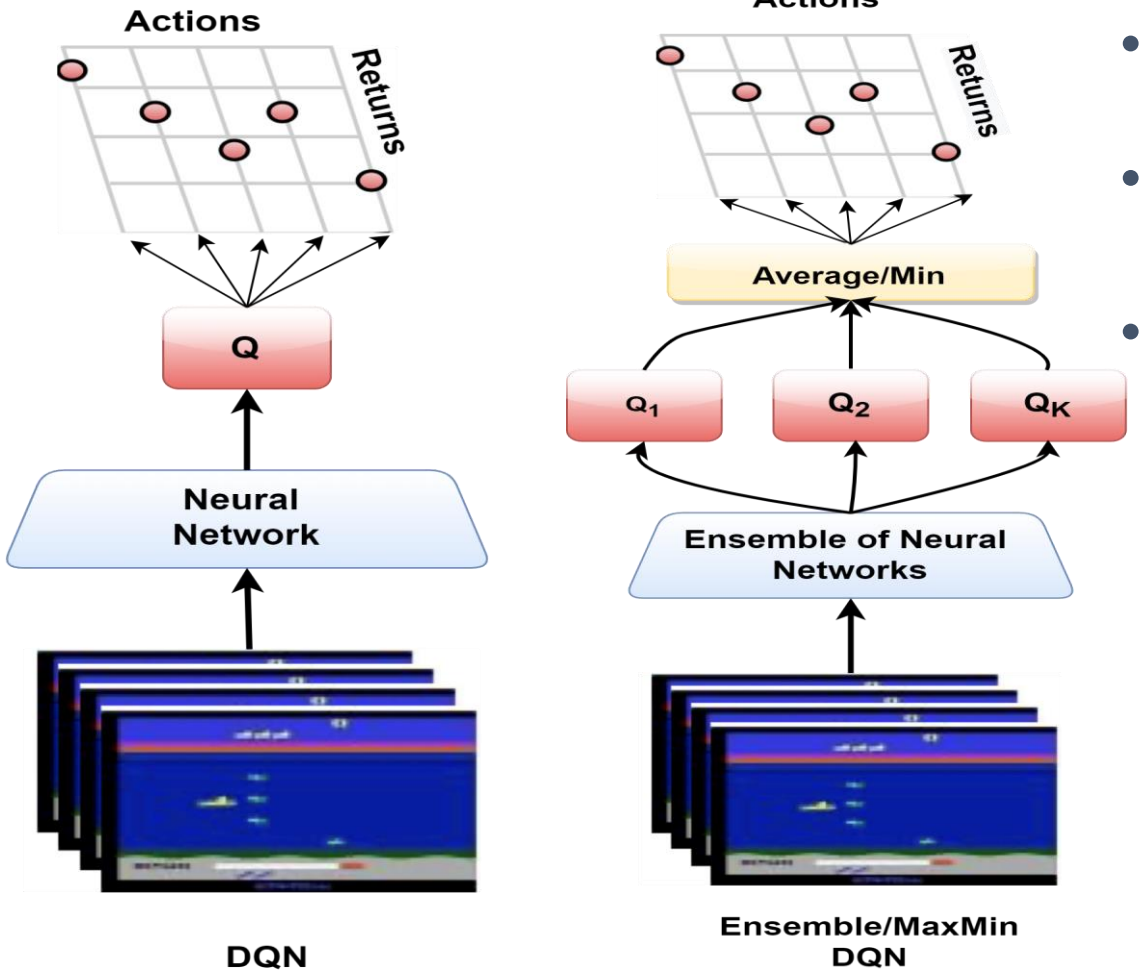
DNS: Determinantal Point Process Based Neural Network Sampling

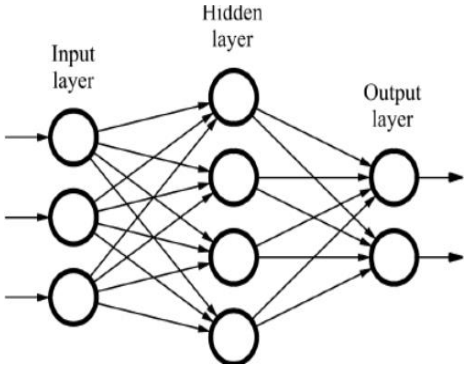
Hassam Sheikh, Kizza Frisbee, Mariano Phielipp

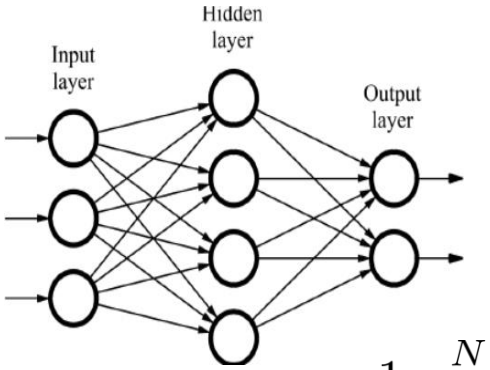
Ensembles in Reinforcement Learning



- Address Overestimation Bias
- Sample Efficiency
- Exploration and Exploitation

Ensembles for Overestimation Bias





-5	18	7	1
25	6	7	9

=

10	12	7	4

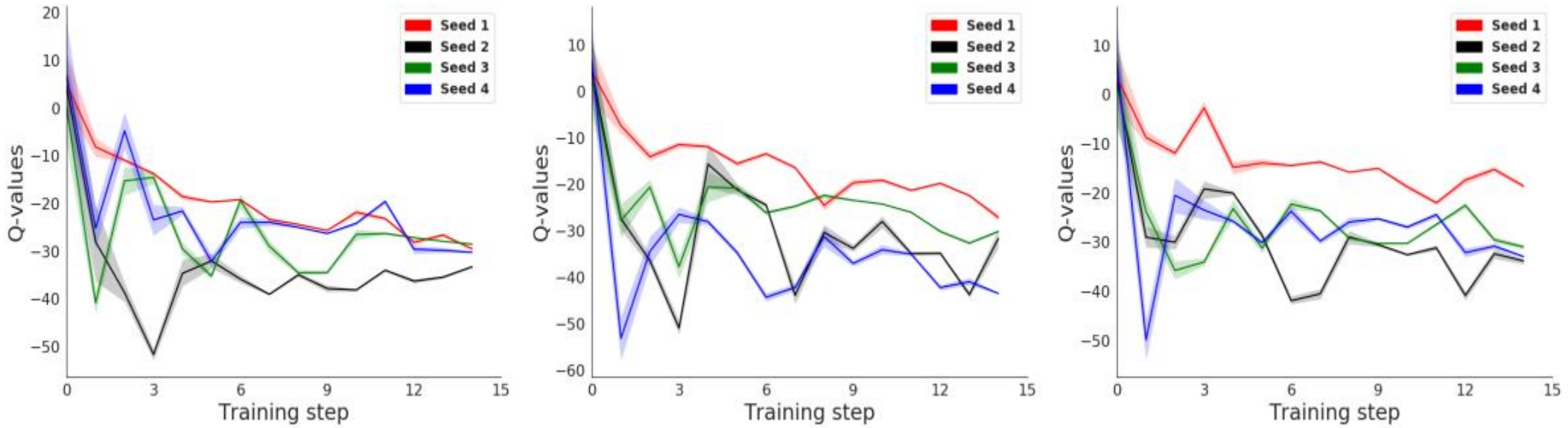
$$Q_E(\cdot) = \frac{1}{N}\sum_{i=1}^{N} Q_i(\cdot)$$

$$Y_t^E = R_t + \max_{a' \in \mathcal{A}} Q_E(s_{t+1}, a')$$

Averaged-DQN: Variance Reduction and Stabilization for Deep Learning (Oron Anschel et al) Reinforcement

DNS: Determinantal Point Process Based Neural Network Sampling

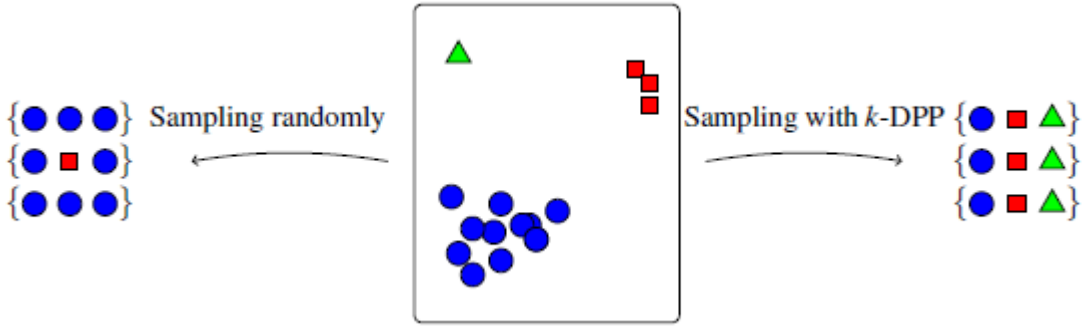
- Long training time and high computation requirements can make ensemble RL infeasible for wide scale use.
- We can exploit the collapse for critics to speed up training time



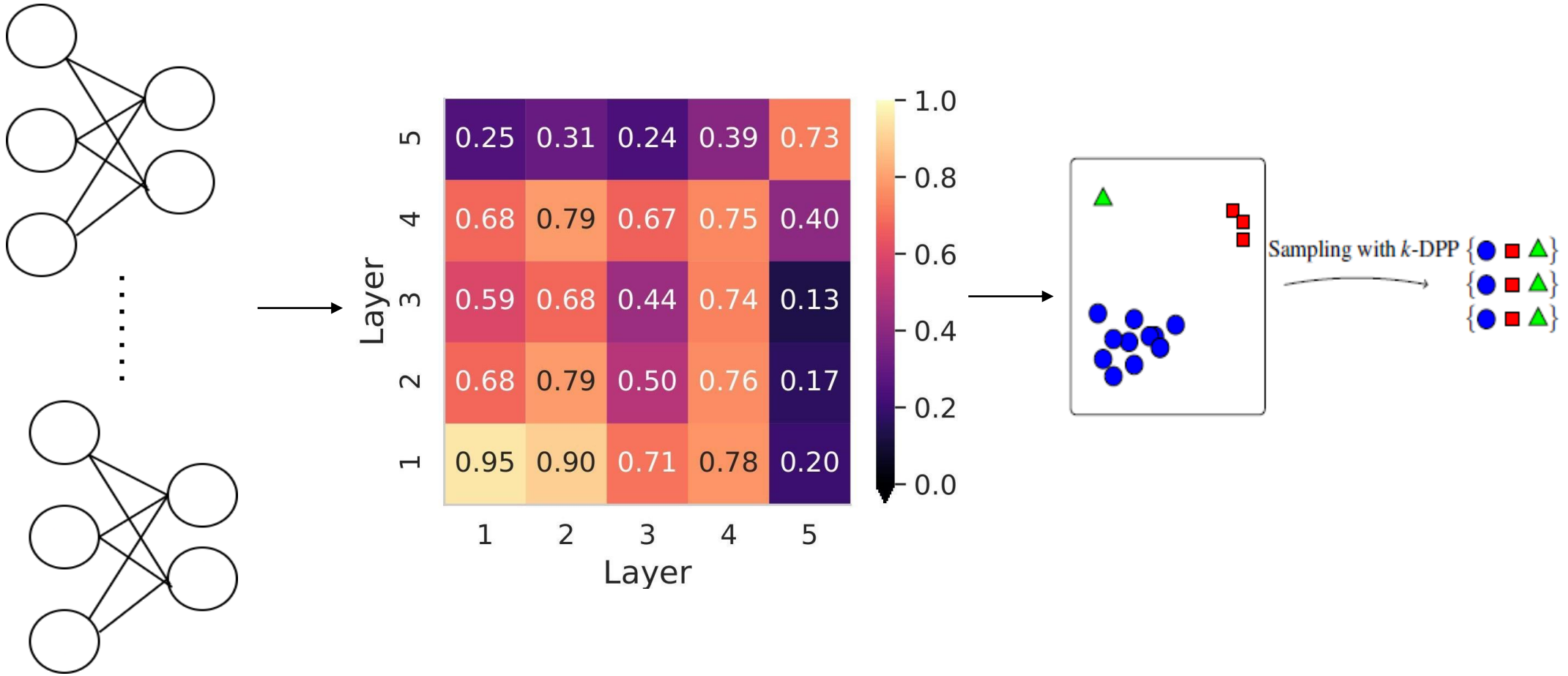
(a) Ant-v2 (b) HalfCheetah-v2 (c) Walker-v2

DPP: Determinantal Point Process

- A Determinantal point process (DPP) is a random point process useful for the combinatorial problem of selecting a diverse sample from a set.
- A DPP for a given finite set defines a probability distribution over subsets, where subsets containing diverse items have high probability and are thus more likely to be selected.



Determinantal Point Process Based Neural Network Sampling

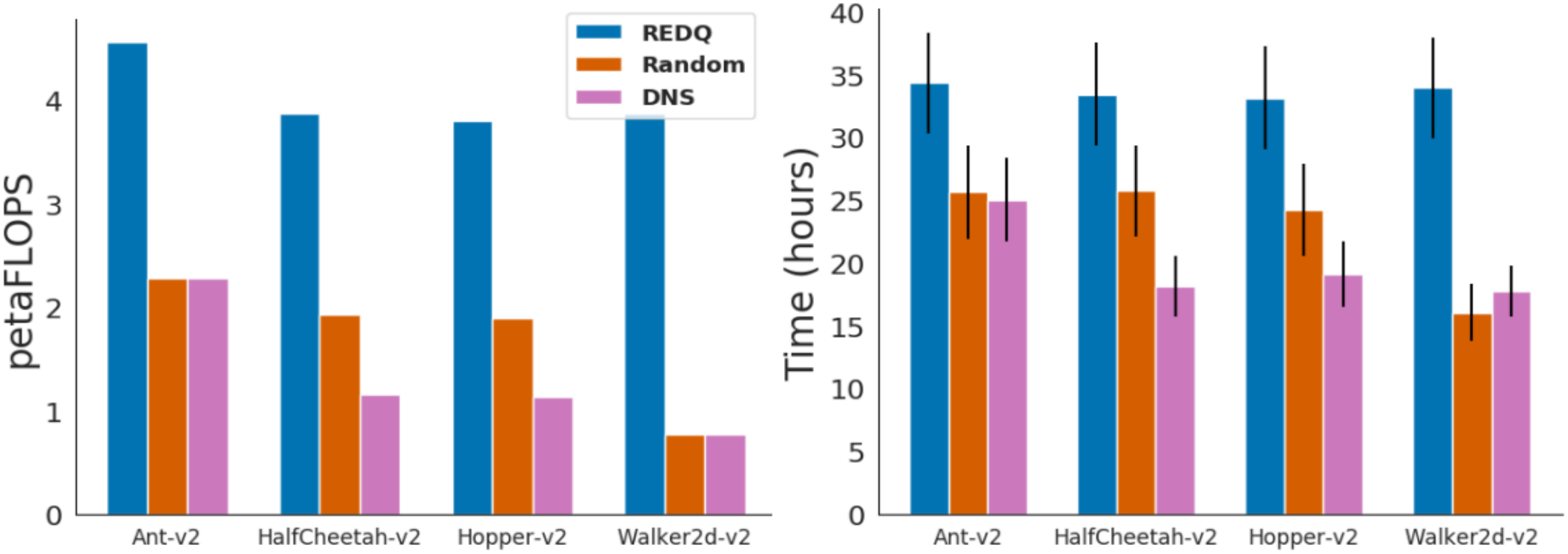


Results

Table 1: Max average return for 10 runs of 300K time steps. Maximum value for each task is bolded. ± corresponds to a single standard deviation over runs

Environment	Baseline	Random	DNS
Ant-v2	2543.1 ± 2595.7	2666.8 ± 2262.6	**3167.2 ± 2484.7**
HalfCheetah-v2	9818.8 ± 1445.2	9474.3 ± 991.1	**9931.0 ± 819.1**
Hopper-v2	2544.2 ± 1468.21	2374.9 ± 1405.8	**2967.8 ± 1128.9**
Walker2d-v2	2414.4 ± 1580.0	1946.4 ± 1287.9	**2802.3 ± 1272.1**

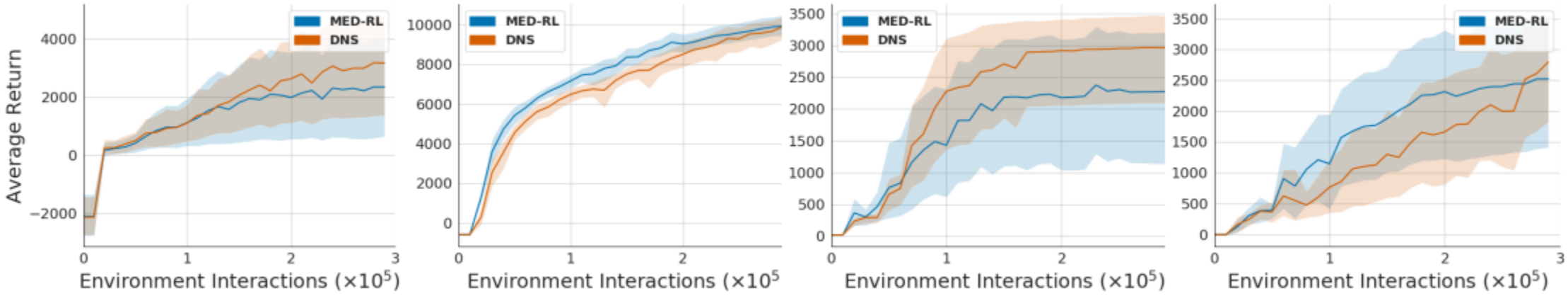
Computation Cost



(a) Computation cost of the backpropagation method in terms of petaFLOPS

(b) Average wall-clock training time in hours

DNS vs MED-RL



(a) Ant-v2 (b) HalfCheetah-v2 (c) Hopper-v2 (d) Walker2d-v2

Thank You