Striving for simplicity and performance in off-policy DRL: Output Normalization and Non-Uniform Sampling

Che Wang*[1], Yanqiu Wu*[1], Quan Vuong[2], Keith Ross[1]
*Equal Contribution, [1]New York University, [2]University of California San Diego

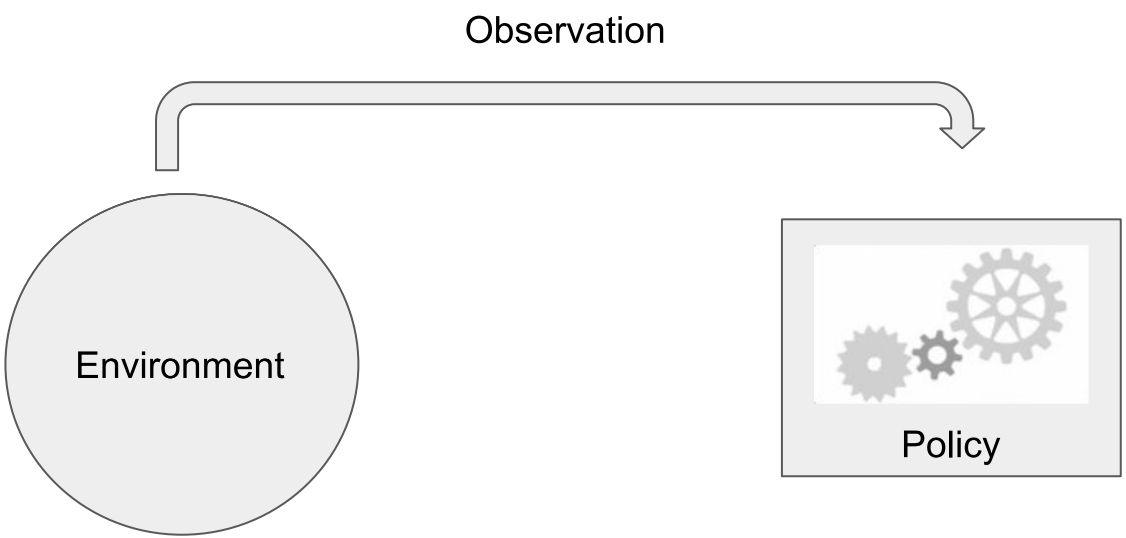
Outline

- Summary of contributions
- Preliminaries
- Entropy maximization
- Squashing exploration problem
- Output normalization and Streamlined Off-Policy (SOP)
- Non-uniform sampling
- Conclusions

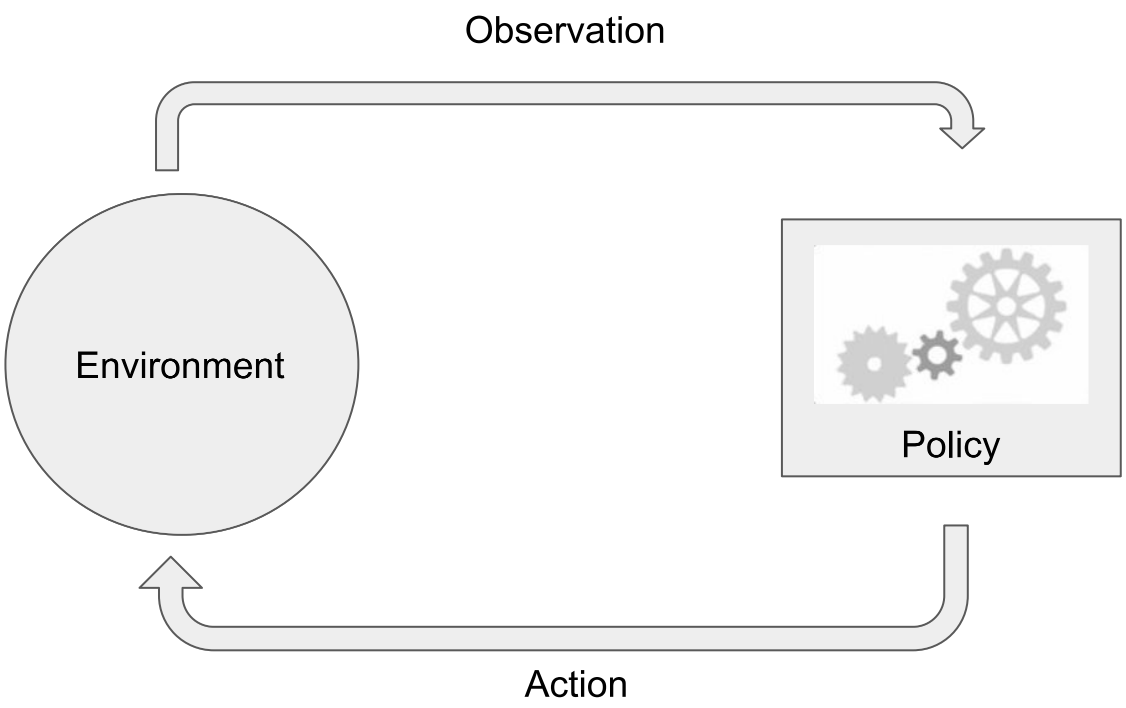
Contributions of this paper:

- Uncover the primary contribution of entropy in maximum entropy RL for MuJoCo
- A streamlined algorithm (SOP), without entropy maximization, matching the sampling efficiency and robust performance of SAC.
- A simple non-uniform sampling scheme to reach SOTA performance

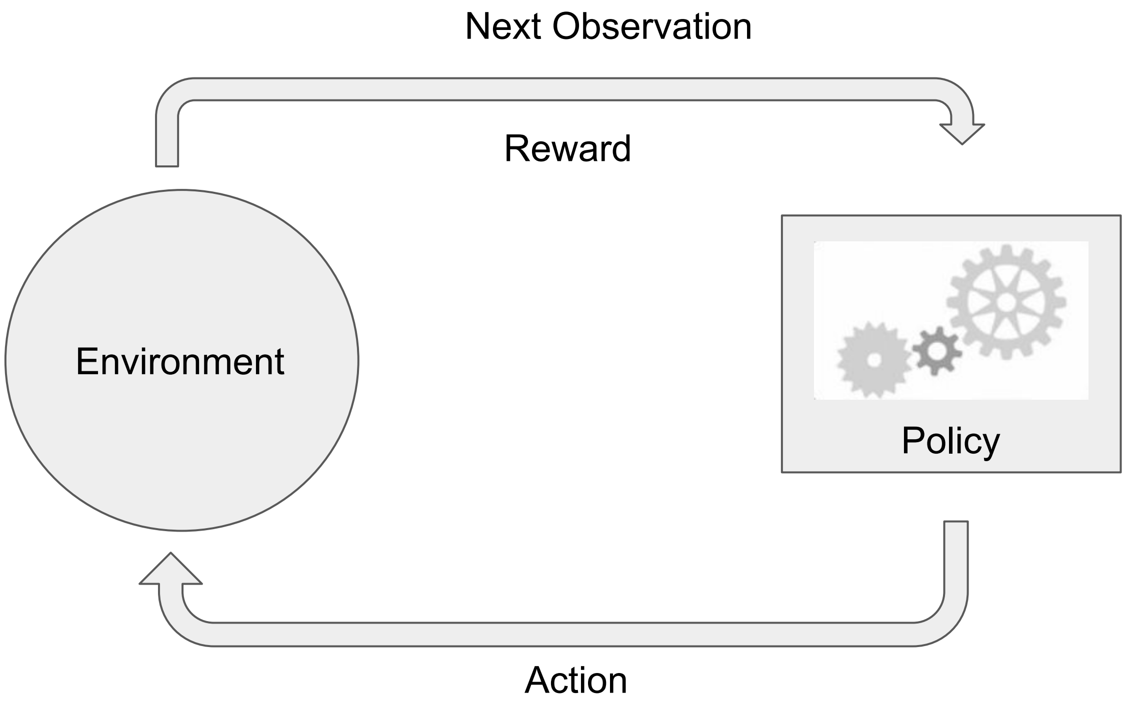
Reinforcement Learning



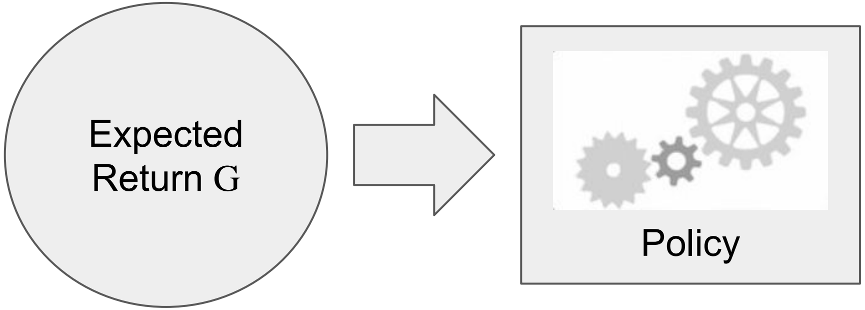
Reinforcement Learning



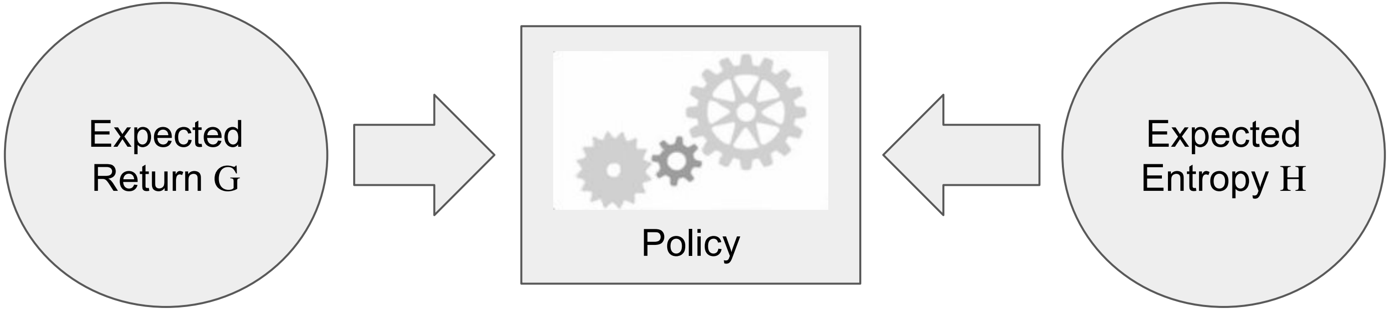
Reinforcement Learning



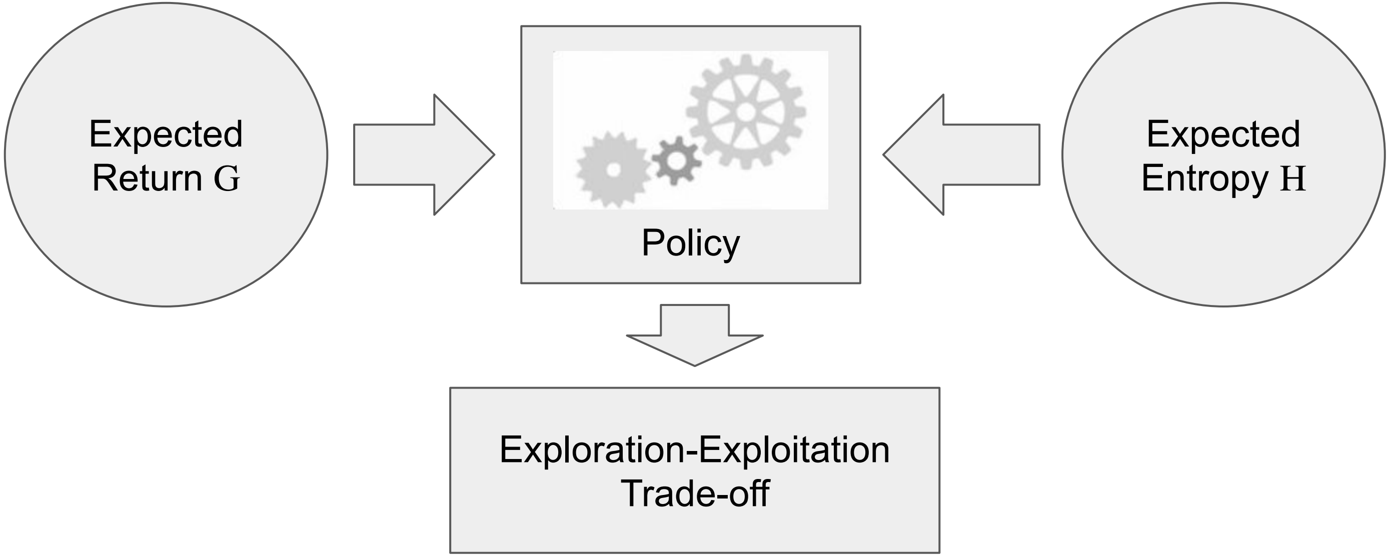
Why Entropy Maximization



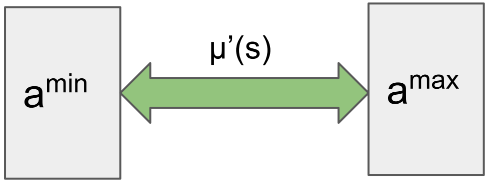
Why Entropy Maximization



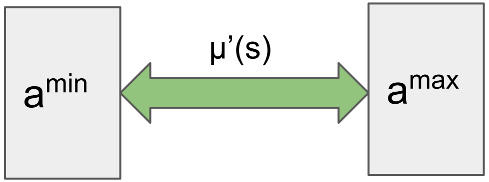
Why Entropy Maximization



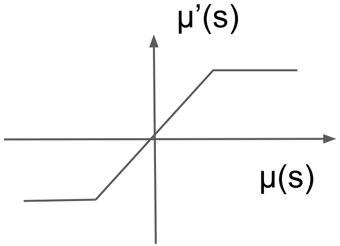
Squashing Exploration Problem



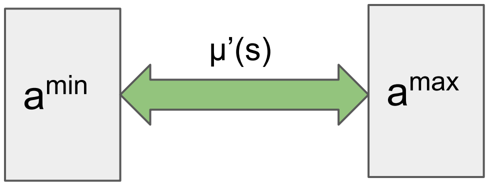
Squashing Exploration Problem



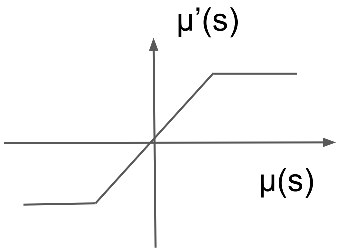
Hard Clip



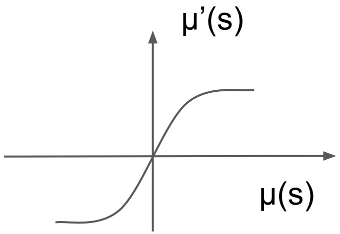
Squashing Exploration Problem



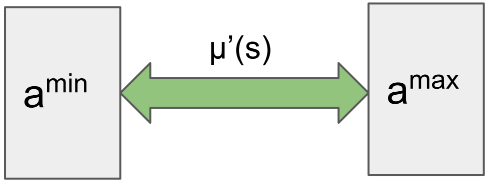
Hard Clip



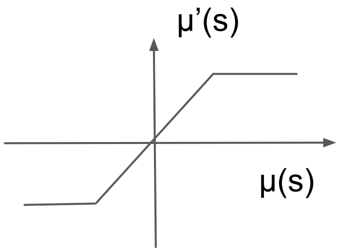
Smooth Squashing (tanh)



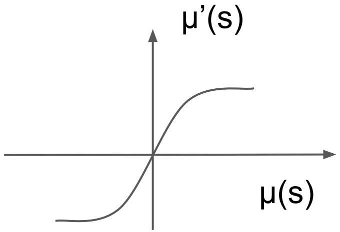
Squashing Exploration Problem



Hard Clip

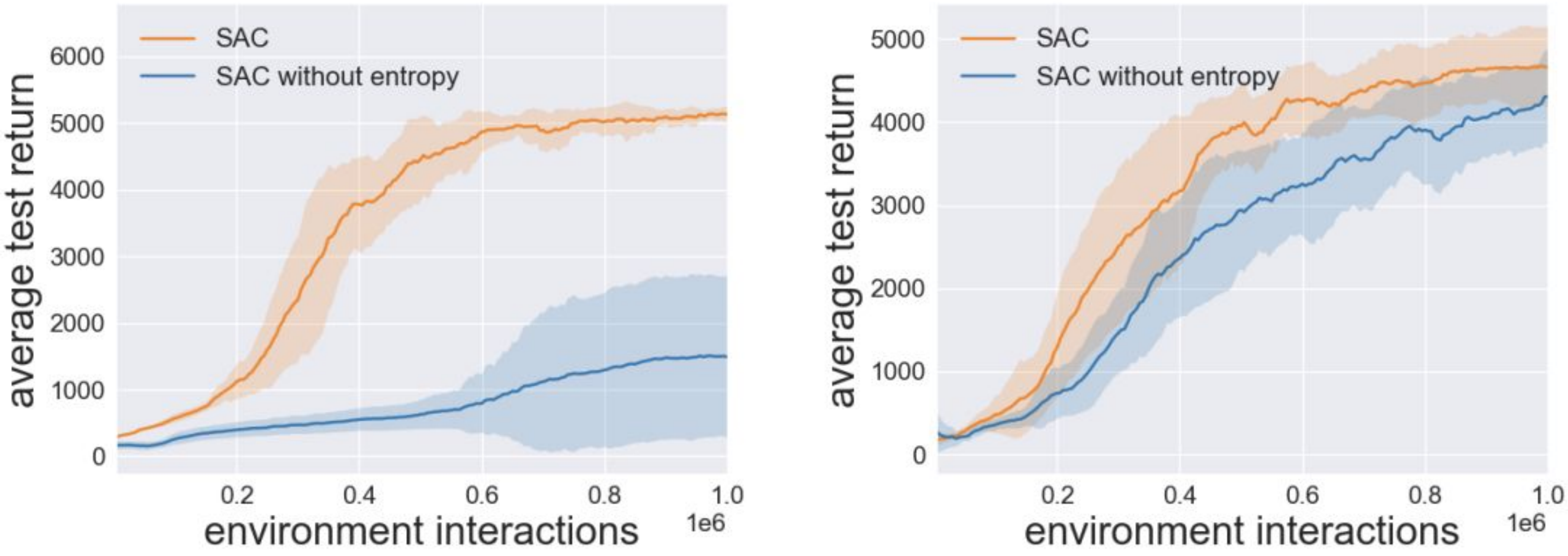


Smooth Squashing



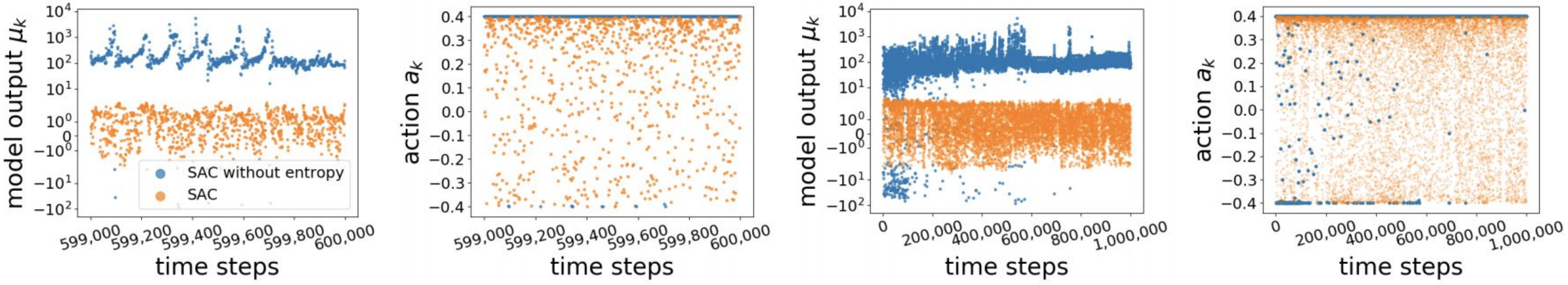
What if | μ(s) | >> 1 over many consecutive states

How Entropy Maximization Helps

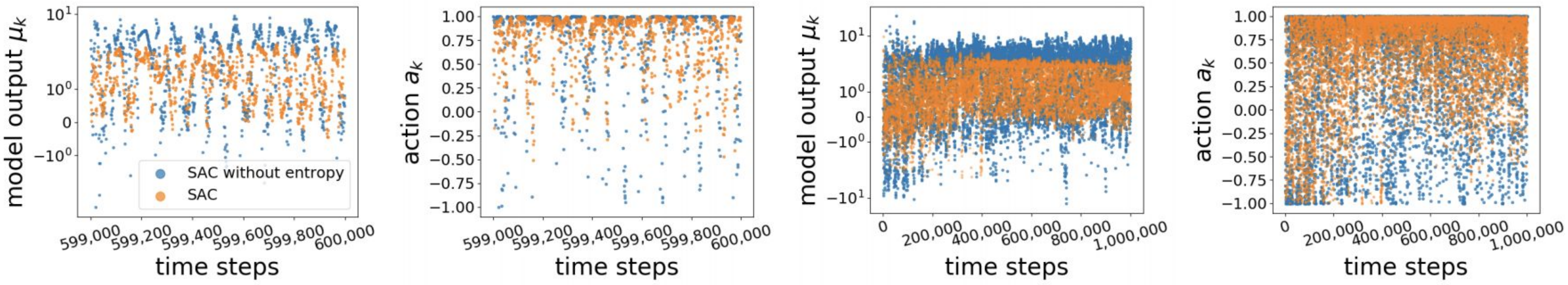


(a) Humanoid-v2 (b) Walker2d-v2

(Haarnoja et al., 2018b)



(a) Humanoid-v2



(b) Walker2d-v2

Inverting Gradients

∇a is the gradient of the policy loss w.r.t to a.

If ∇a suggests increasing a : $\nabla a = \nabla a \cdot \dfrac{a^{max} - a}{a^{max} - a^{min}}$

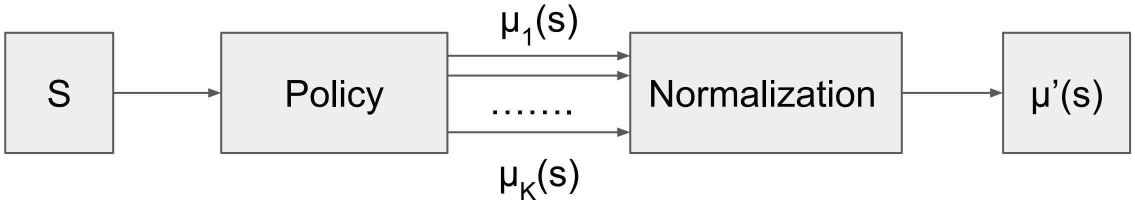
Otherwise : $\nabla a = \nabla a \cdot \dfrac{a - a^{min}}{a^{max} - a^{min}}$

(Hausknecht & Stone, 2015)

We can do something even **SIMPLER**

Output Normalization

Replace entropy maximization → Streamlined Off-Policy (SOP)

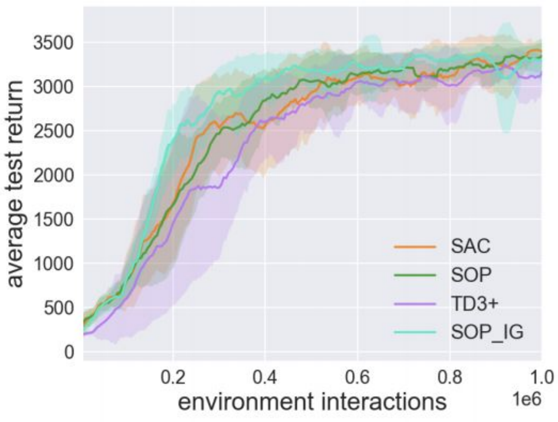


$\mu(s) = (\mu_1(s), \mu_2(s), \ldots, \mu_K(s))$;

$G = \|\, \mu(s) \,\|_1 / K$

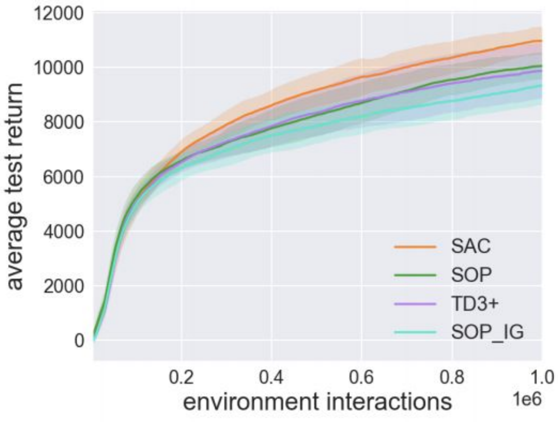
If $G > 1$, $\mu'_k(s) \leftarrow \mu_k(s) / G$; for all $k = 1, \ldots, K$

	DDPG	TD3	SAC	SOP
Target Q Network	✓	✓	✓	✓
Target Policy Network	✓	✓		
Double Q-Learning		✓	✓	✓
Target Policy Smoothing		✓	✓	✓
Delayed Policy Update		✓		
Entropy Maximization			✓	
Normalization				✓

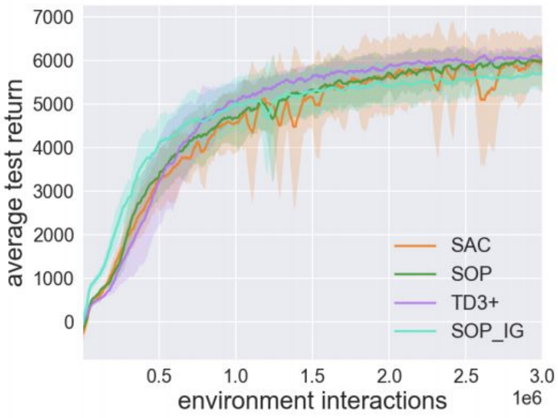


(a) Hopper-v2

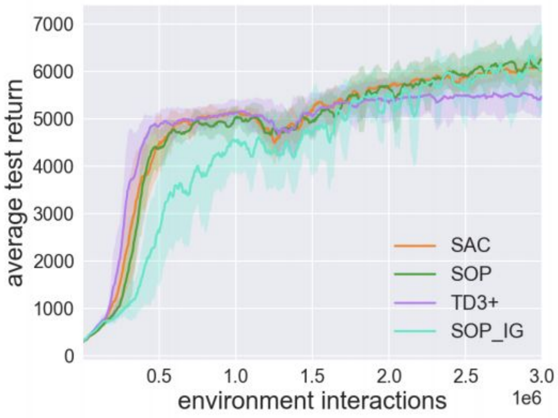
(b) Walker2d-v2



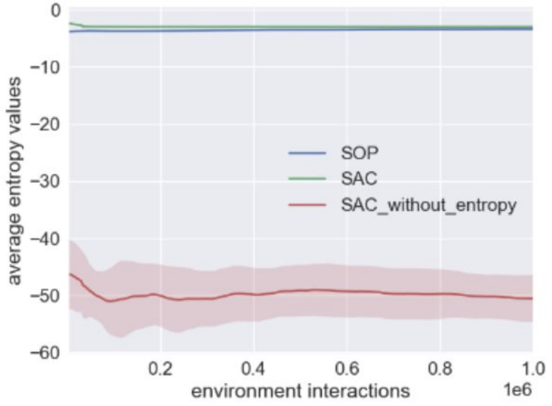
(c) HalfCheetah-v2



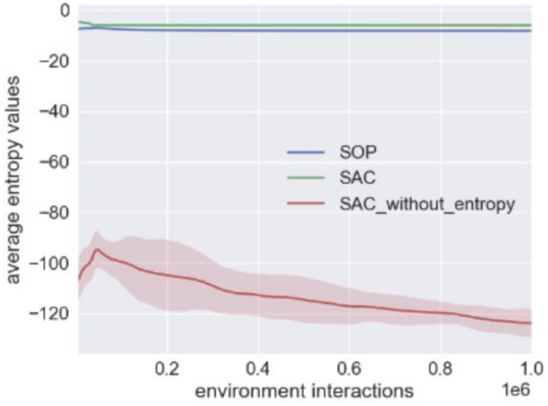
(d) Ant-v2



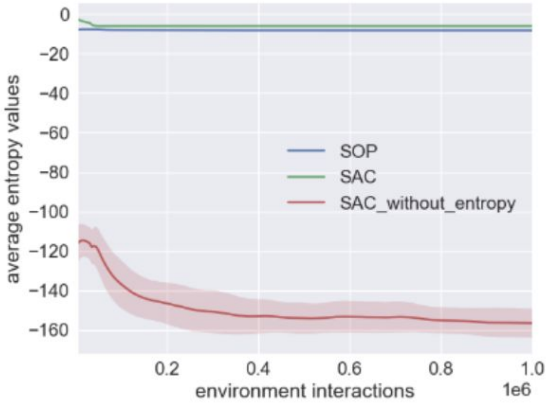
(e) Humanoid-v2



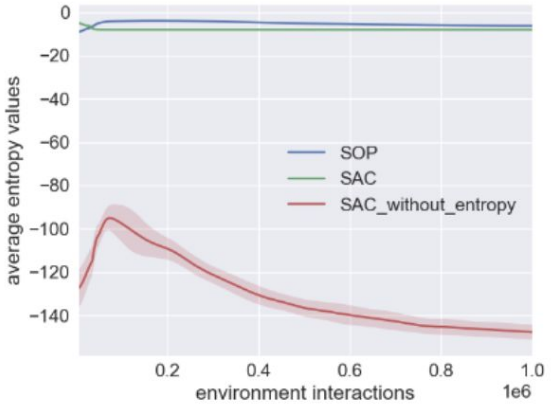
(a) Hopper-v2



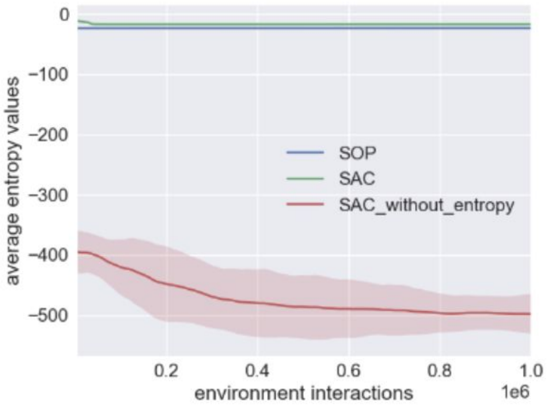
(b) Walker2d-v2



(c) HalfCheetah-v2

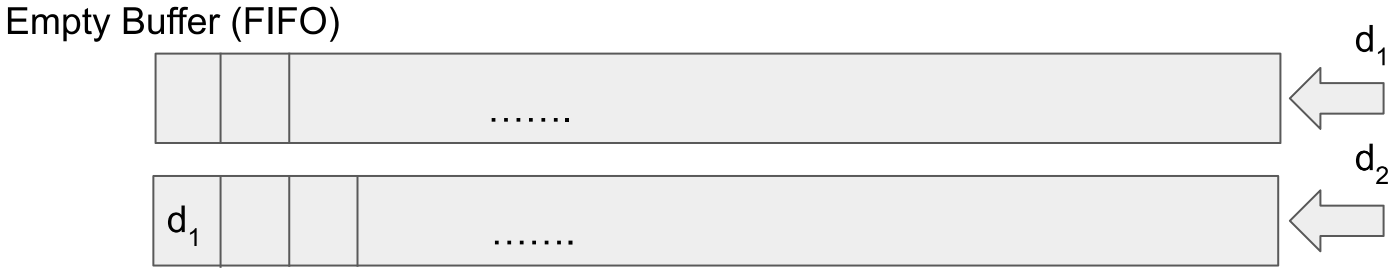


(d) Ant-v2



(e) Humanoid-v2

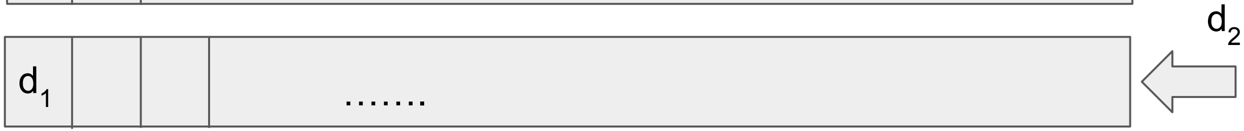
Why Non-uniform Sampling



Uniform Sampling: expected number of times being sampled $E(d_1) = 1$

Why Non-uniform Sampling

Empty Buffer



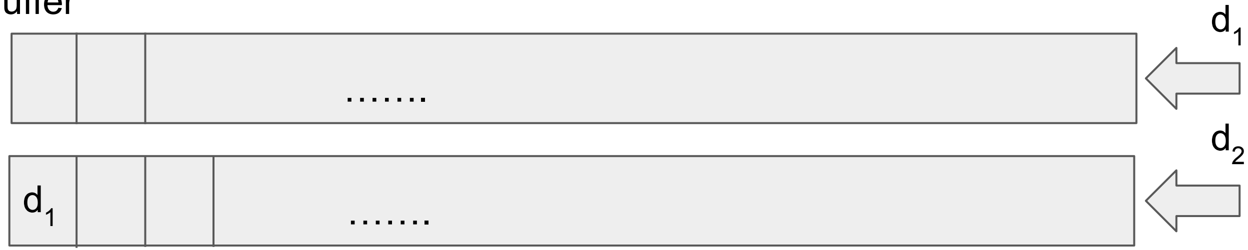
Uniform Sampling: expected number of times being sampled $E(d_1) = 1$

Uniform Sampling: expected number of times being sampled $E(d_1) = 1 + ½$; $E(d_2) = ½$

....

Why Non-uniform Sampling

Empty Buffer



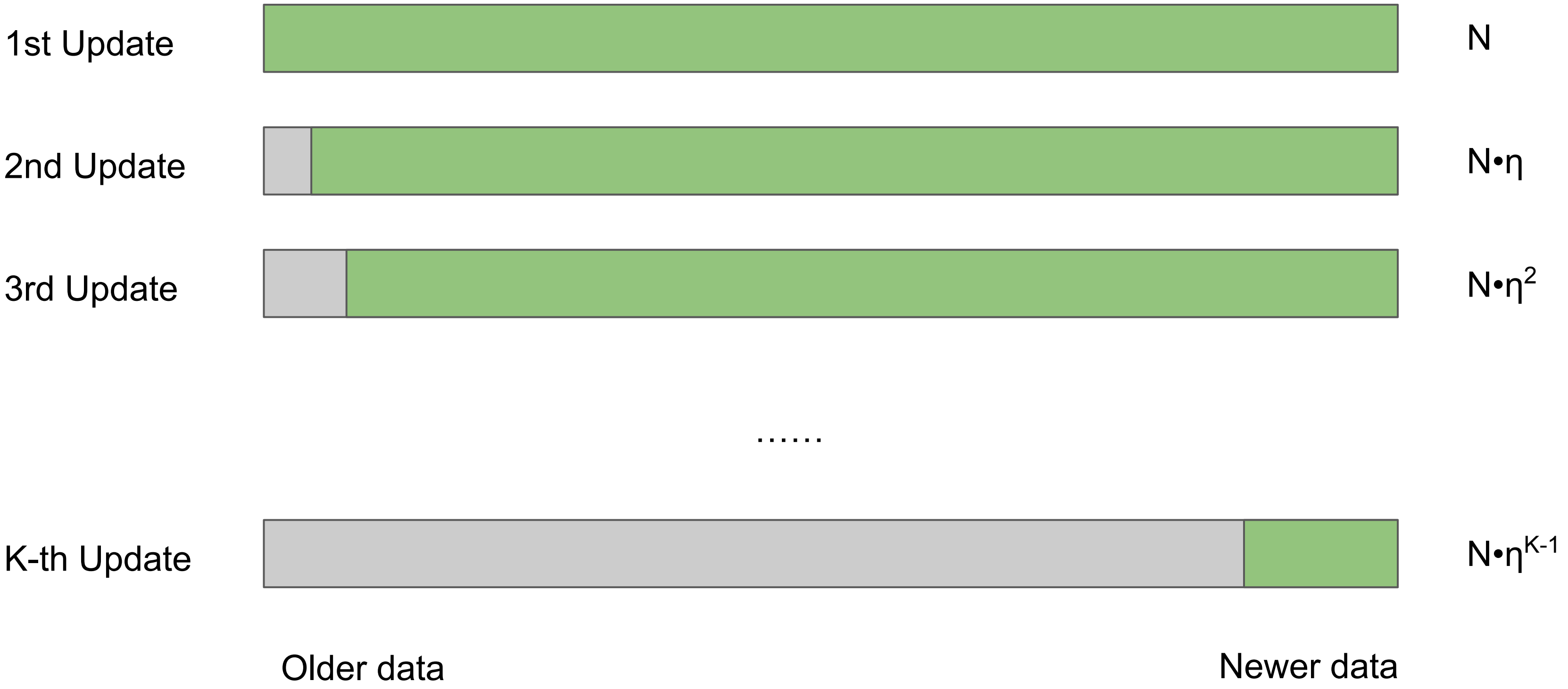
Uniform Sampling: expected number of times being sampled $E(d_1) = 1$

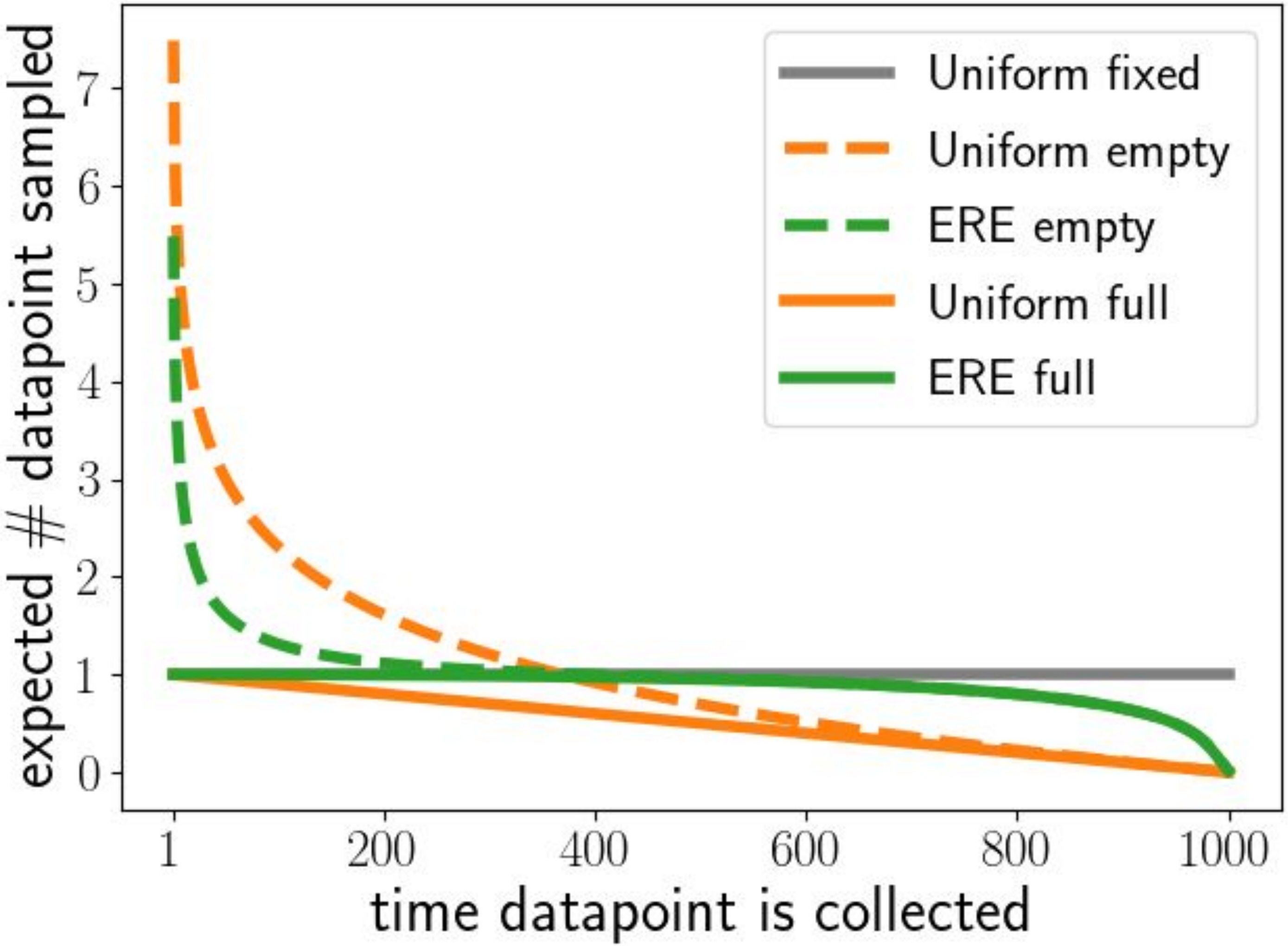
Uniform Sampling: expected number of times being sampled $E(d_1) = 1 + ½$; $E(d_2) = ½$

....

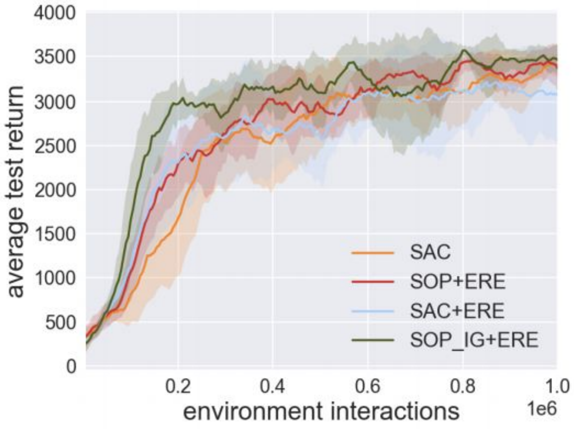
Uniform Sampling: expected number of times being sampled $E(d_t) = \sum_{k = t....T} 1/k$

Emphasizing Recent Experience (ERE)

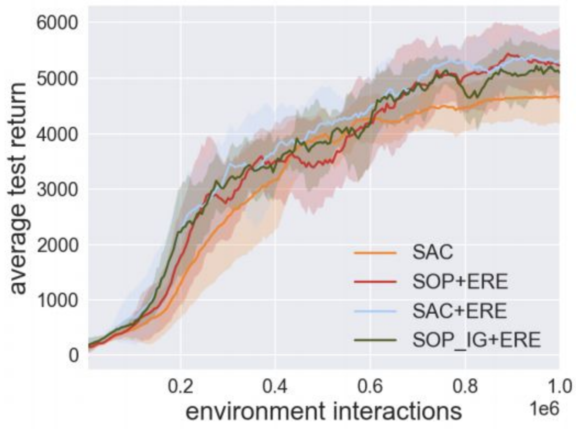




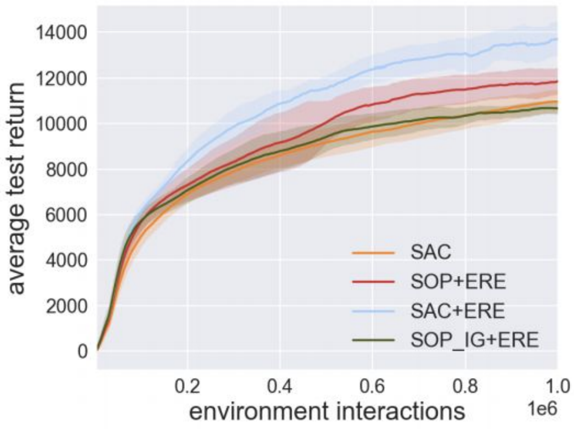
Uniform fixed
Uniform empty
ERE empty
Uniform full
ERE full
expected # datapoint sampled
0
1
2
3
4
5
6
7
1
200
400
600
800
1000
time datapoint is collected



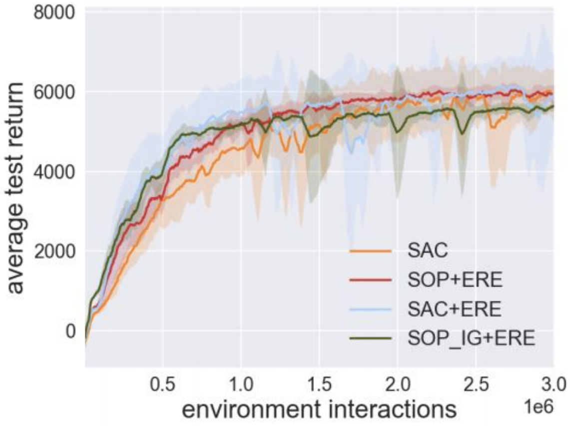
(a) Hopper-v2



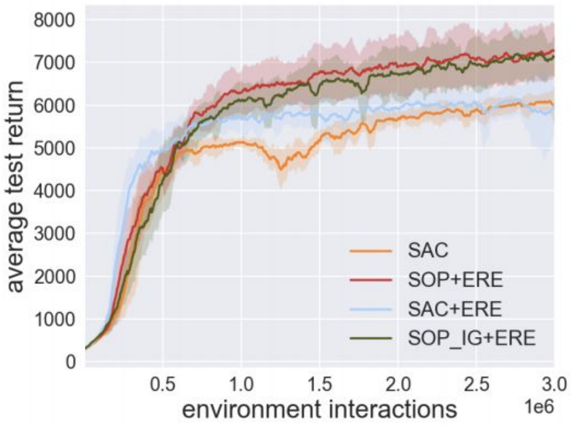
(b) Walker2d-v2



(c) HalfCheetah-v2



(d) Ant-v2



(e) Humanoid-v2

Conclusion

Showed that the primary role of maximum entropy RL for the MuJoCo benchmark is to maintain satisfactory exploration in the presence of bounded action spaces.

A new streamlined algorithm which does not employ entropy maximization but nevertheless matches the sampling efficiency and robust performance of SAC for the MuJoCo benchmarks.

Combined our streamlined algorithm with a simple non-uniform sampling scheme to create a simple algorithm that achieves state-of-the art performance for the MuJoCo benchmark.

Thank you so much for listening!