DeepMind

Active Offline Policy Selection

Yutian Chen*, Ksenia Konyushkova*,
Tom Le Paine, Caglar Gulcehre, Cosmin Paduraru,
Daniel J Mankowitz, Misha Denil, Nando de Freitas
(*: equal contributions)

NeurIPS 2021

Offline Reinforcement Learning & Policy Selection

- Challenge for RL application: environment interactions are often expensive

Offline Reinforcement Learning & Policy Selection

- Challenge for RL application: environment interactions are often expensive
- Offline RL: training policies on **logged data** without additional interactions

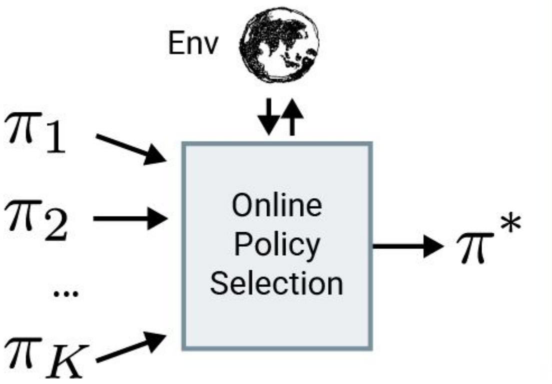
Offline Reinforcement Learning & Policy Selection

- Challenge for RL application: environment interactions are often expensive
- Offline RL: training policies on **logged data** without additional interactions
- How do we choose the best policy for deployment?

Offline Reinforcement Learning & Policy Selection

- Challenge for RL application: environment interactions are often expensive
- Offline RL: training policies on **logged data** without additional interactions
- How do we choose the best policy for deployment?

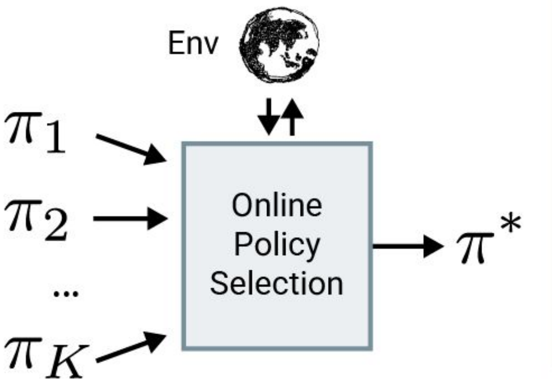
Online Policy Selection



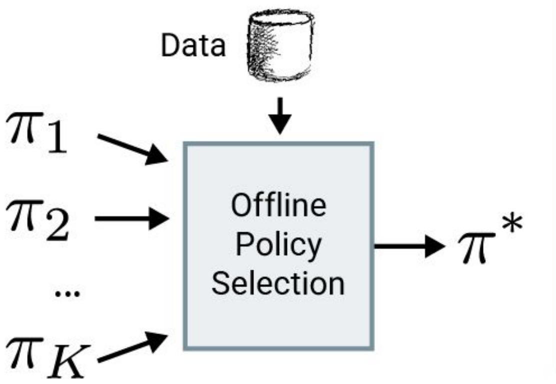
Offline Reinforcement Learning & Policy Selection

- Challenge for RL application: environment interactions are often expensive
- Offline RL: training policies on **logged data** without additional interactions
- How do we choose the best policy for deployment?

Online Policy Selection



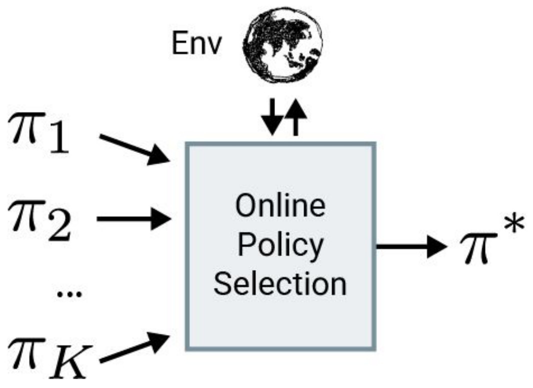
Offline Policy Selection



Offline Reinforcement Learning & Policy Selection

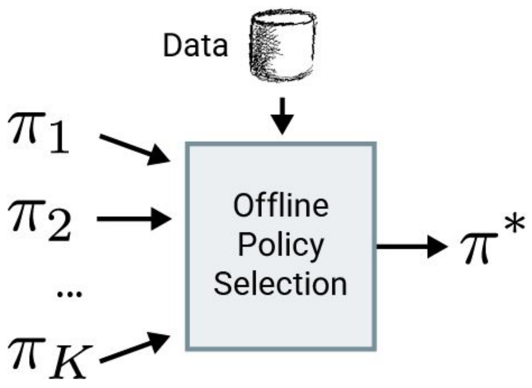
- Challenge for RL application: environment interactions are often expensive
- Offline RL: training policies on **logged data** without additional interactions
- How do we choose the best policy for deployment?

Online Policy Selection



+

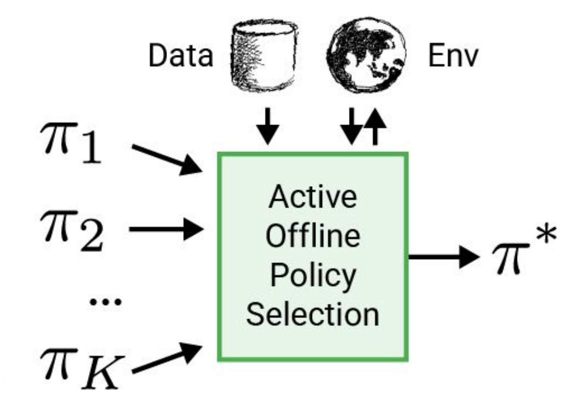
Offline Policy Selection



Offline Reinforcement Learning & Policy Selection

- Challenge for RL application: environment interactions are often expensive
- Offline RL: training policies on **logged data** without additional interactions
- How do we choose the best policy for deployment?

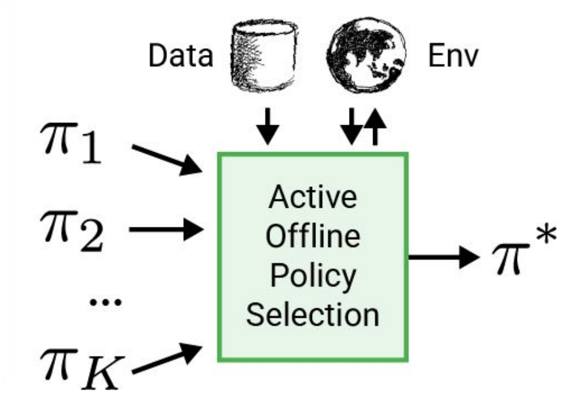
Problem setting: Active Offline Policy Selection (active ops)



Offline Reinforcement Learning & Policy Selection

- Challenge for RL application: environment interactions are often expensive
- Offline RL: training policies on **logged data** without additional interactions
- How do we choose the best policy for deployment?

Problem setting: Active Offline Policy Selection (active ops)



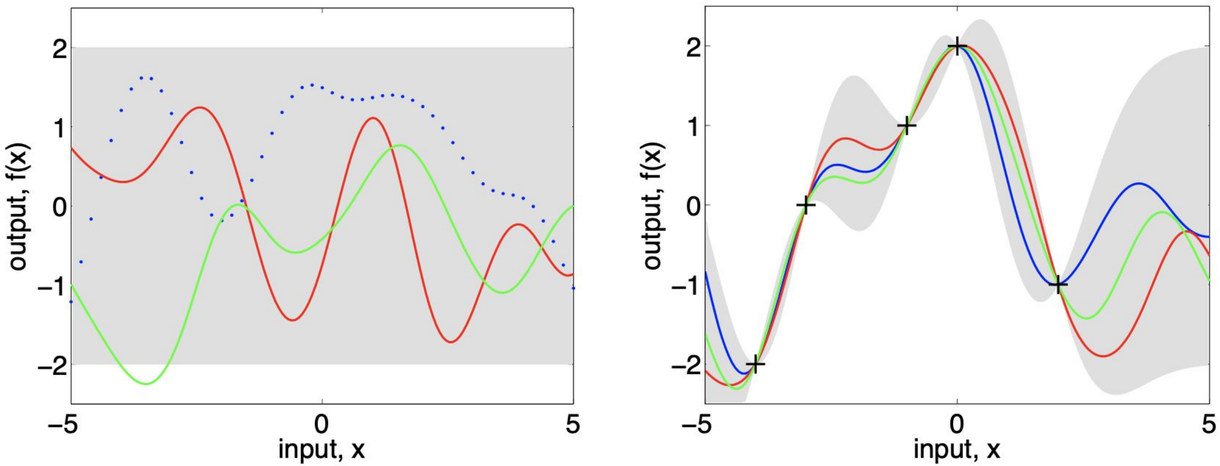
Which policy to evaluate to find a good policy for deployment?

Bayesian Optimization in one slide

- Goal: maximizing an expensive-to-query black-box function

$$\arg\max_{x \in \mathcal{X}} f(x)$$

- Probabilistic model for $f(x)$: Gaussian process



(a), prior (b), posterior

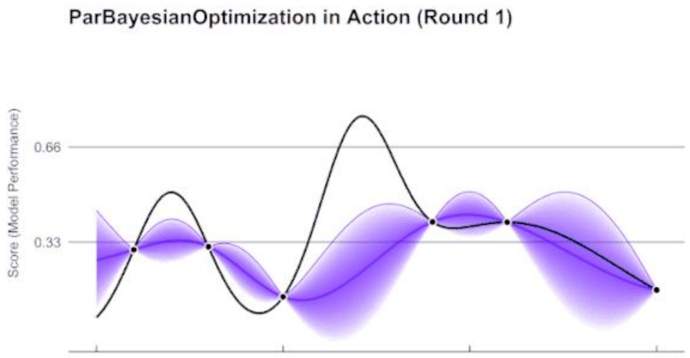
Bayesian Optimization in one slide

- Goal: maximizing an expensive-to-query black-box function

$$\arg\max_{x \in \mathcal{X}} f(x)$$

- Probabilistic model for $f(x)$: Gaussian process

- Iteratively finds the next query point with both high posterior mean and high posterior variance (optimism in the face of uncertainty)



Active ops as Bayesian optimization

Problem: $\arg\max_{1 \leq k \leq K} \mu(\pi_k)$

policy π

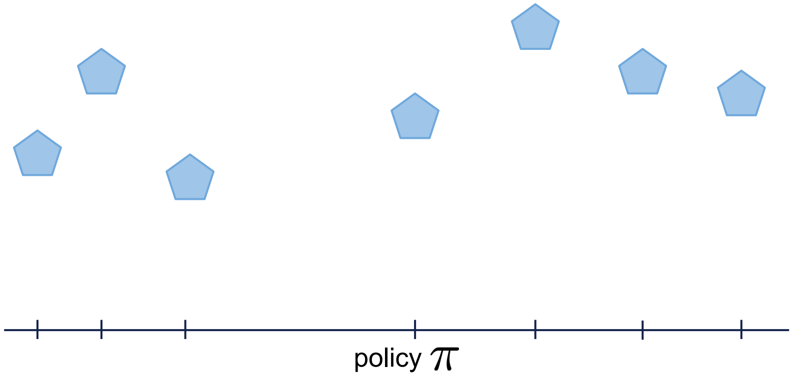
Active ops as Bayesian optimization

Problem: $\underset{1 \le k \le K}{\arg\max}\ \mu(\pi_k)$

1. : Off-policy evaluation (OPE)

   Precomputed a priori

   Comes from 



Active ops as Bayesian optimization

Problem: $\underset{1 \leq k \leq K}{\arg\max}\ \mu(\pi_k)$

1. : Off-policy evaluation (OPE)

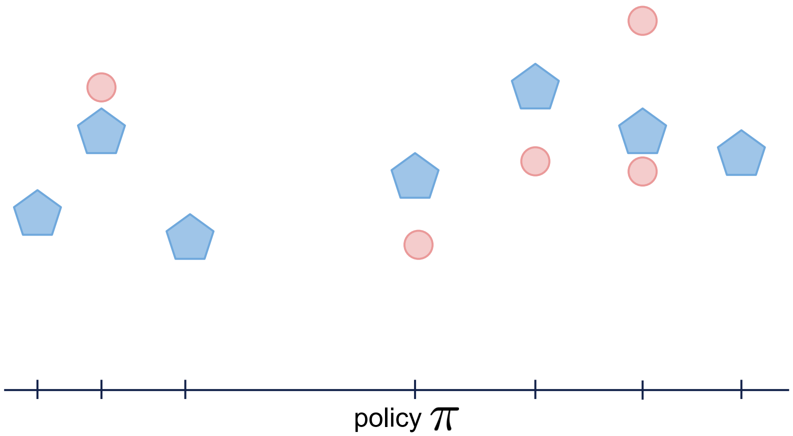
   Precomputed a priori

   Comes from

2. : Episodic return

   Expensive to sample (use active learning)

   Comes from



Active ops as Bayesian optimization

Problem: $\arg\max_{1 \le k \le K} \mu(\pi_k)$

1.  : Off-policy evaluation (OPE)

   Precomputed a priori

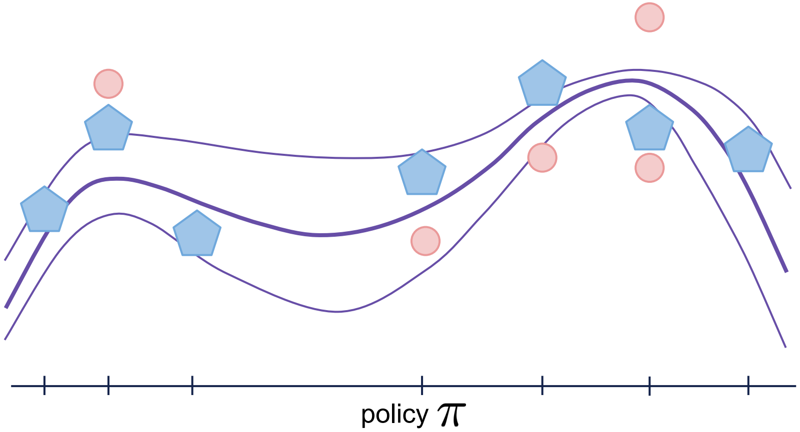
   Comes from 

2.  : Episodic return

   Expensive to sample (use active learning)

   Comes from 

Then, Gaussian Processes to model correlation between policies



Policy Kernel

- Insight: similar actions -> similar performance
- Measure the similarity between policies by their **actions** on a set of states

Policy Kernel

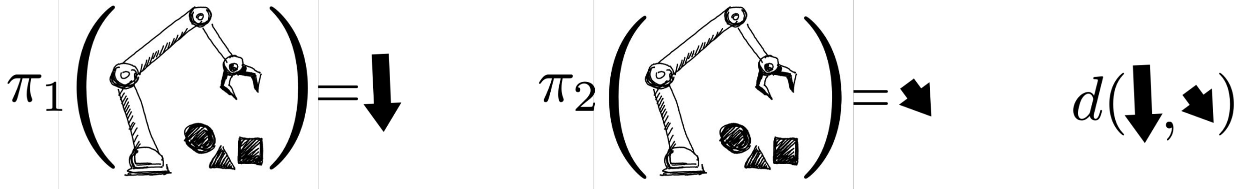
- Insight: similar actions -> similar performance
- Measure the similarity between policies by their **actions** on a set of states

Policy 1

Policy 2

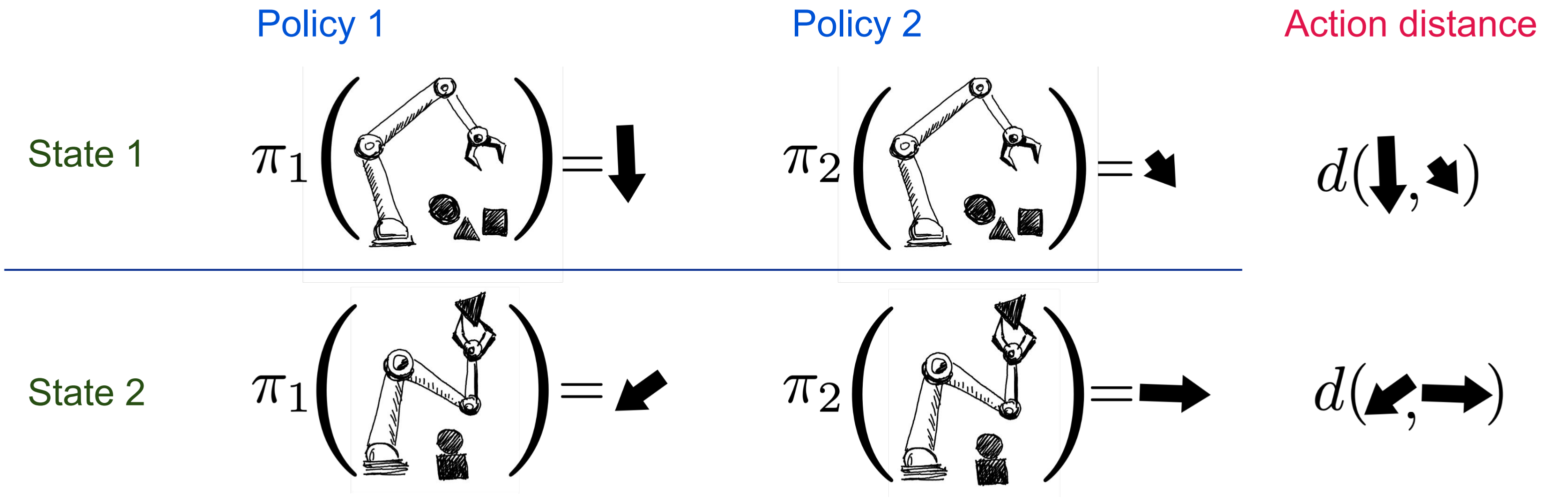
Action distance

State 1



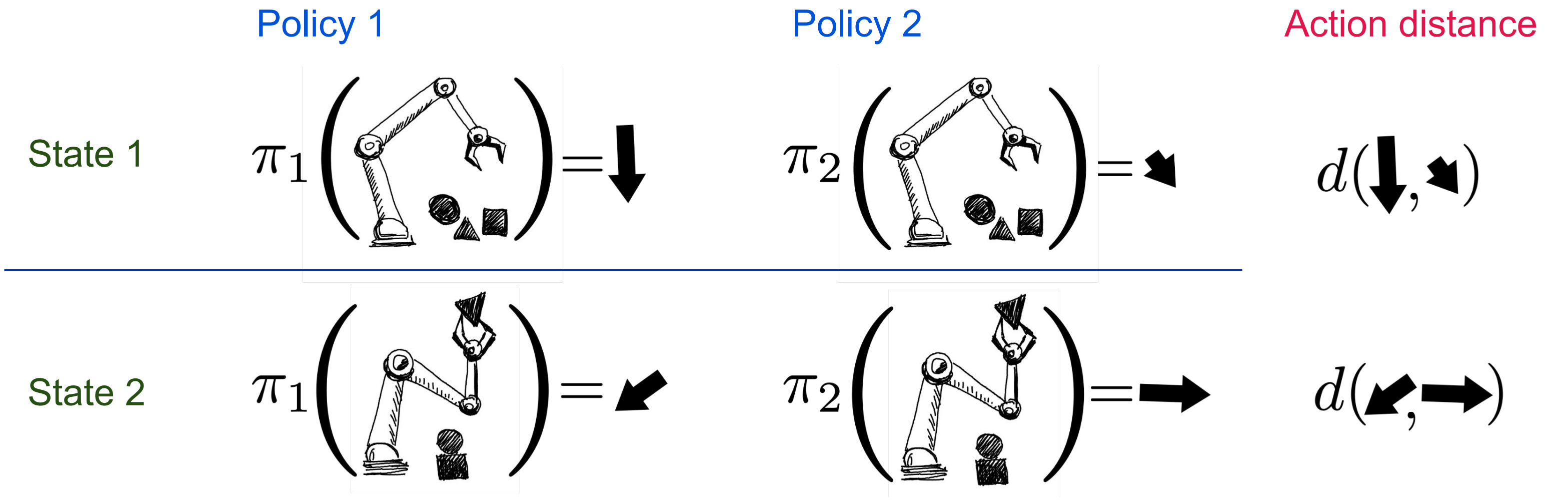
Policy Kernel

- Insight: similar actions -> similar performance
- Measure the similarity between policies by their **actions** on a set of states



Policy Kernel

- Insight: similar actions -> similar performance
- Measure the similarity between policies by their **actions** on a set of states



Average over states: $d(\pi_1, \pi_2) = [d(\downarrow, \searrow) + d(\swarrow, \rightarrow)]/2$

Experiments: domains

Three sets of control task suites (2 continuous and 1 discrete action space)

- DM Control Suite (9 environments)
- Manipulation Playground (4 tasks)
- Atari (3 games)

Experiments: domains

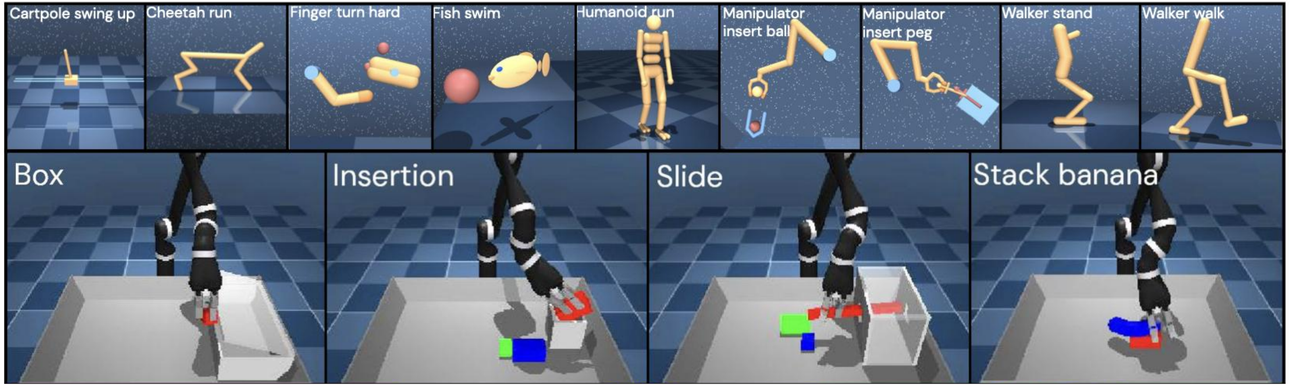
Three sets of control task suites (2 continuous and 1 discrete action space)

- DM Control Suite (9 environments)
- Manipulation Playground (4 tasks)
- Atari (3 games)

Experiments: domains

Three sets of control task suites (2 continuous and 1 discrete action space)

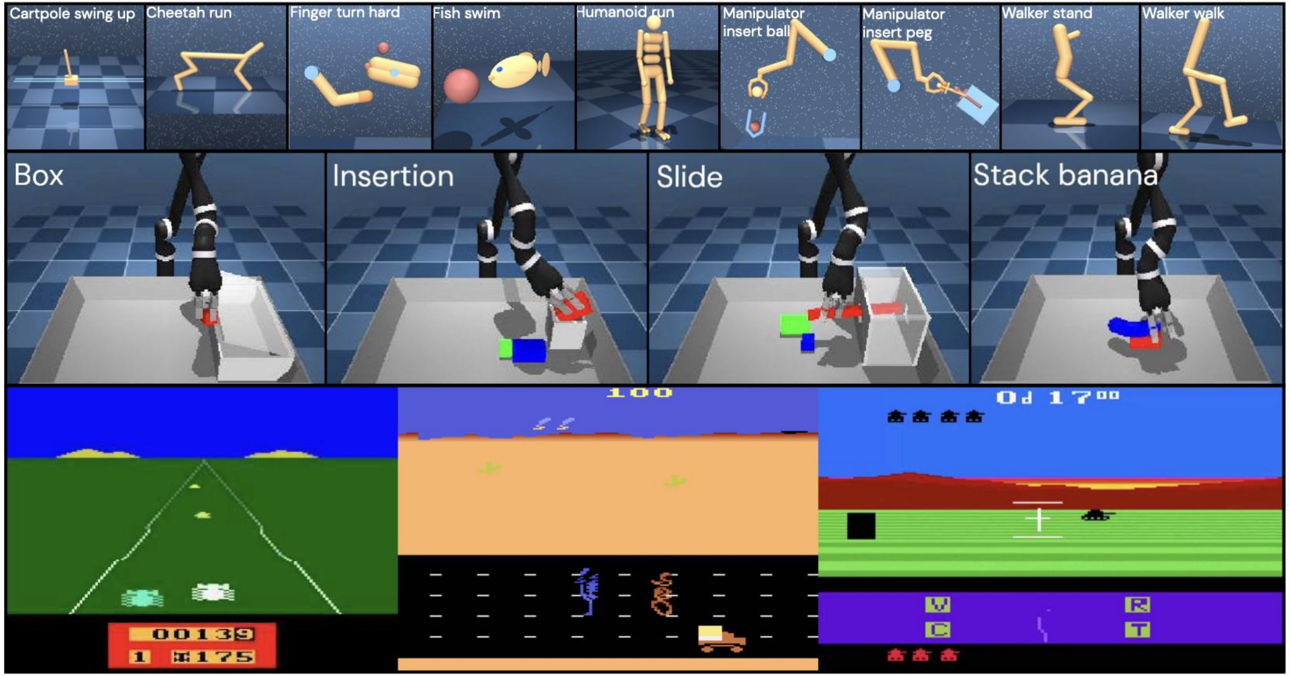
- DM Control Suite (9 environments)
- Manipulation Playground (4 tasks)
- Atari (3 games)



Experiments: domains

Three sets of control task suites (2 continuous and 1 discrete action space)

- DM Control Suite (9 environments)
- Manipulation Playground (4 tasks)
- Atari (3 games)

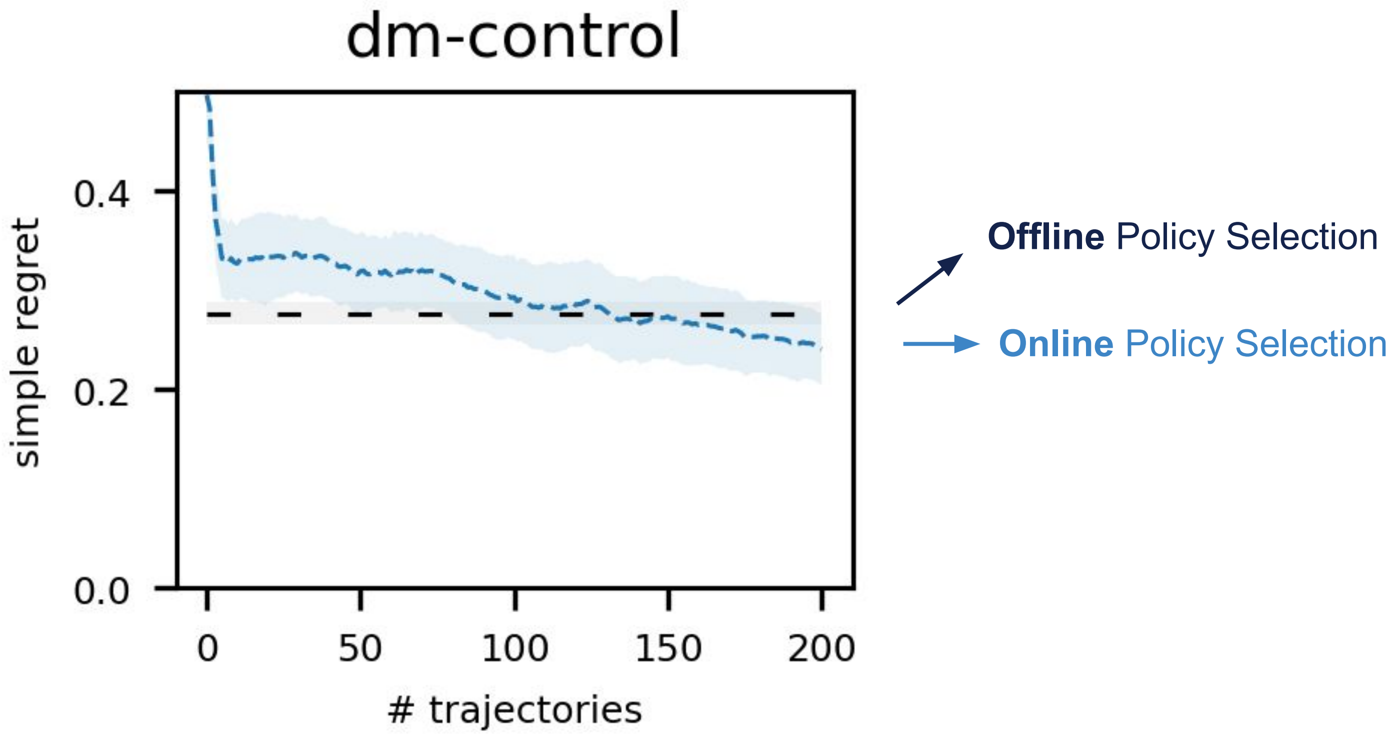


Experiments: quantitative results

Simple regret averaged over 100 experiments with 50 policies each in each of 9 environments of dm-control.

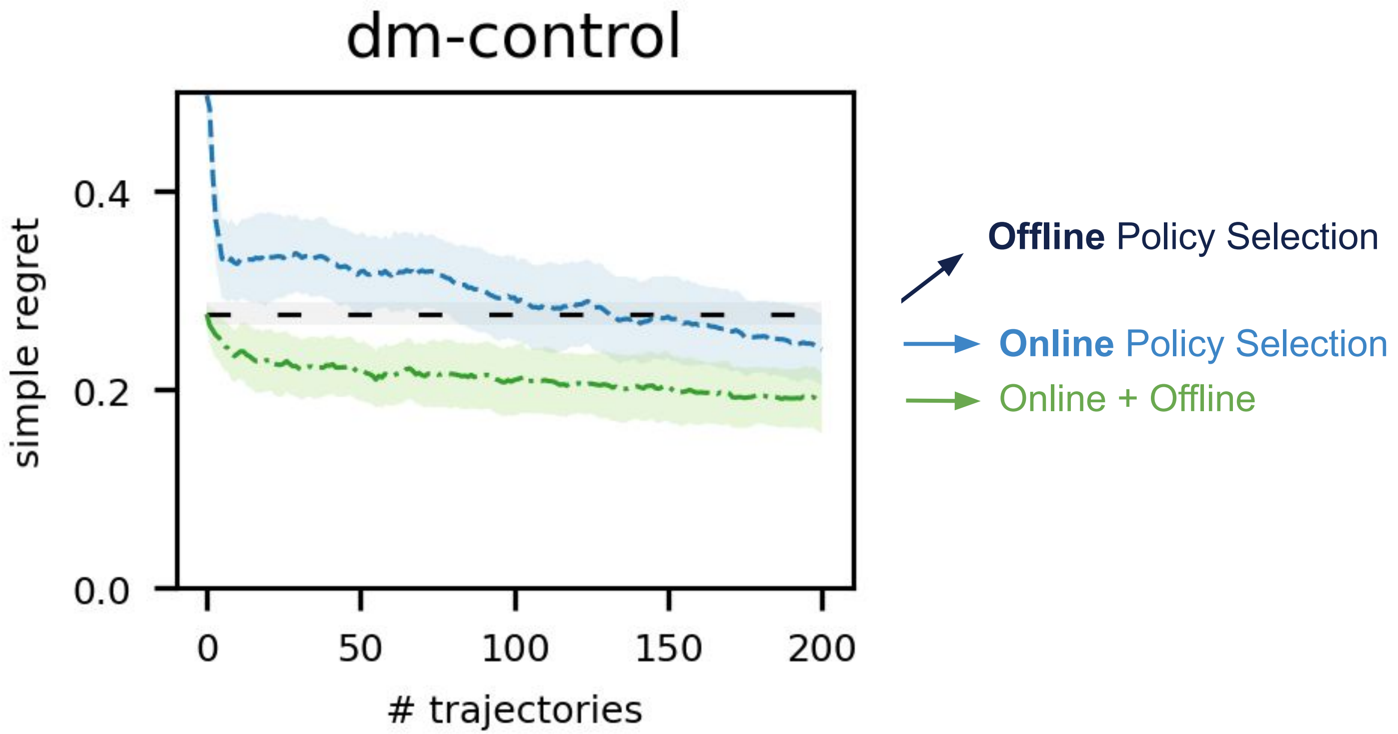
Experiments: quantitative results

Simple regret averaged over 100 experiments with 50 policies each in each of 9 environments of dm-control.



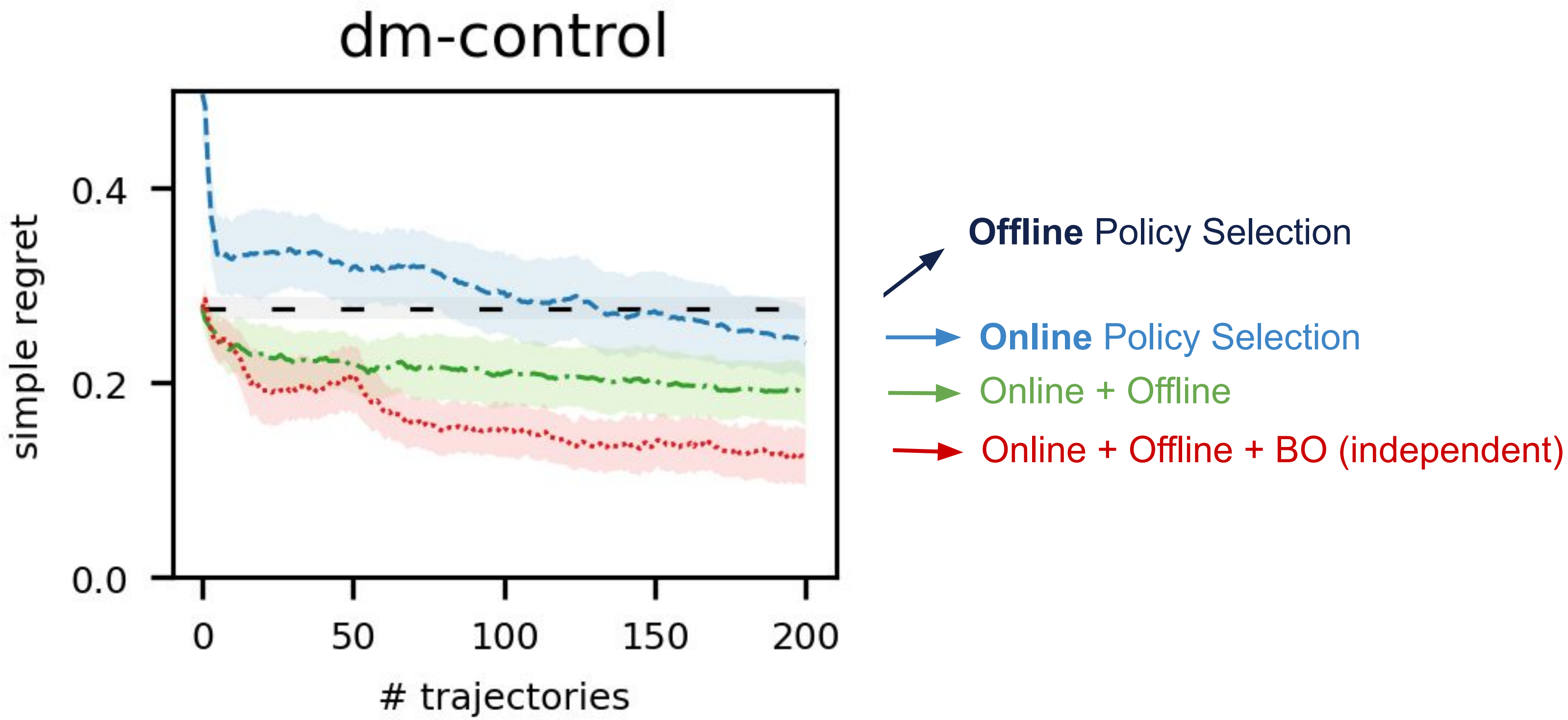
Experiments: quantitative results

Simple regret averaged over 100 experiments with 50 policies each in each of 9 environments of dm-control.



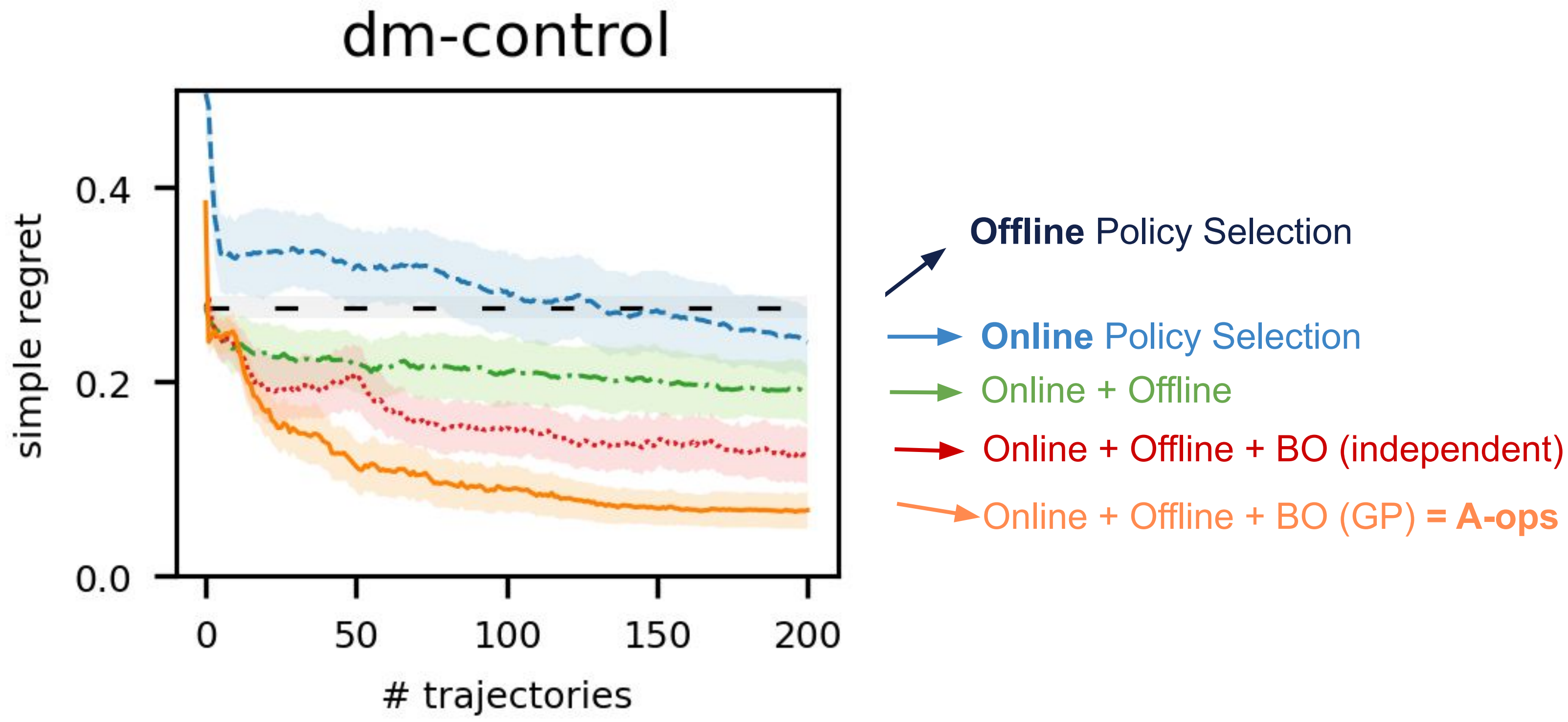
Experiments: quantitative results

Simple regret averaged over 100 experiments with 50 policies each in each of 9 environments of dm-control.



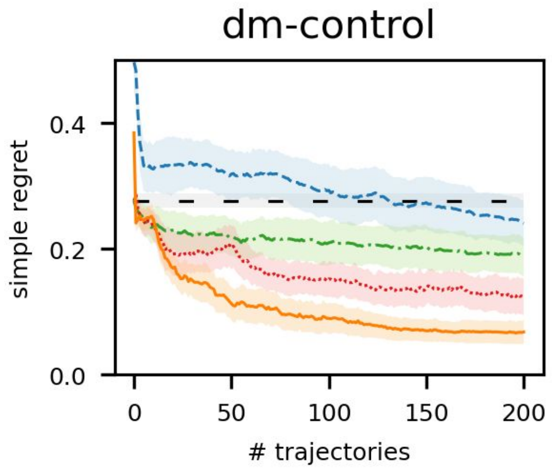
Experiments: quantitative results

Simple regret averaged over 100 experiments with 50 policies each in each of 9 environments of dm-control.



Experiments: quantitative results

The same results hold in MPG and Atari domains with **200 policies.**



Offline Policy Selection

Online Policy Selection

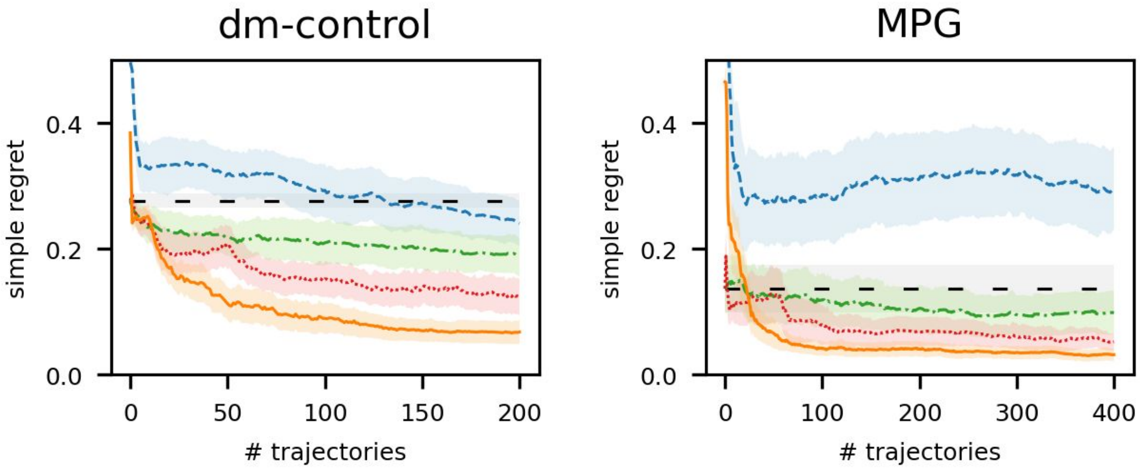
Online + Offline

Online + Offline + BO (independent)

Online + Offline + BO (GP) **= A-ops**

Experiments: quantitative results

The same results hold in MPG and Atari domains with **200 policies.**



Offline Policy Selection

Online Policy Selection

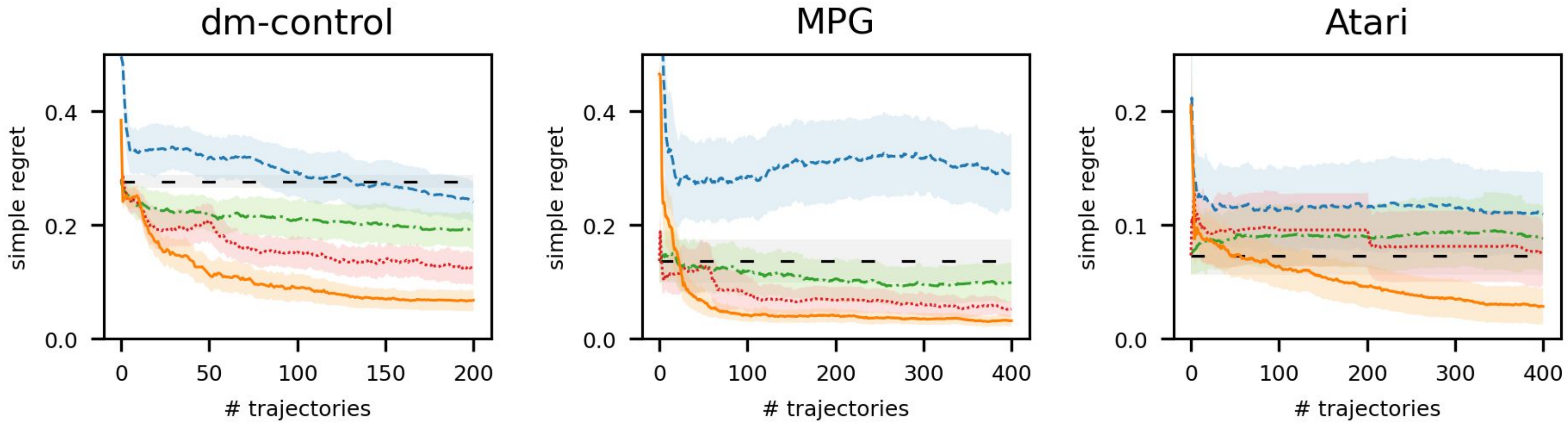
Online + Offline

Online + Offline + BO (independent)

Online + Offline + BO (GP) **= A-ops**

Experiments: quantitative results

The same results hold in MPG and Atari domains with **200 policies.**



Offline Policy Selection

Online Policy Selection

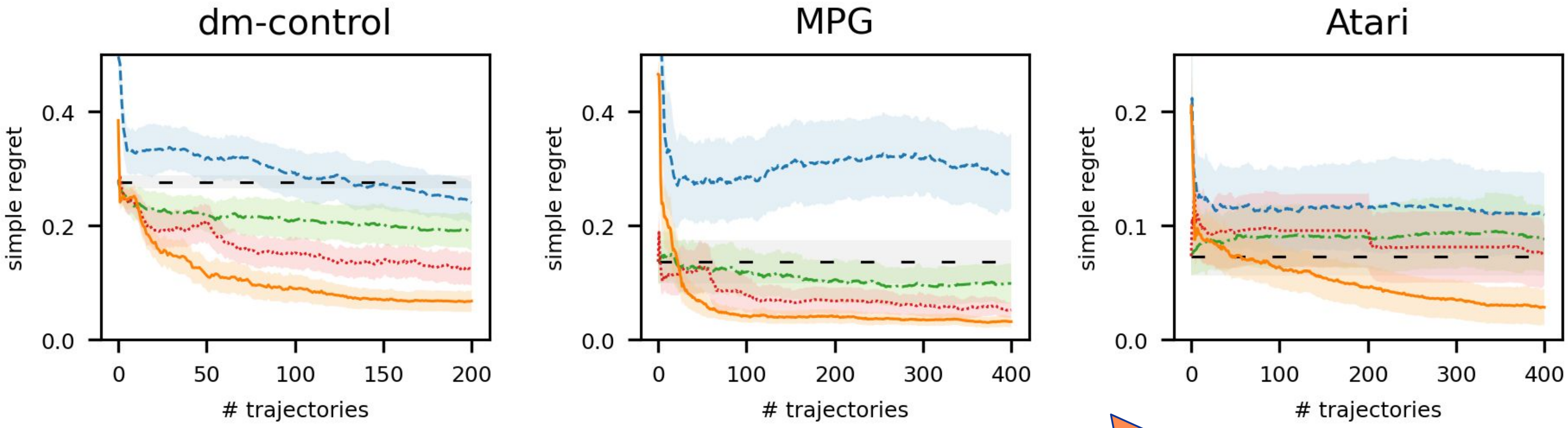
Online + Offline

Online + Offline + BO (independent)

Online + Offline + BO (GP) **= A-ops**

Experiments: quantitative results

The same results hold in MPG and Atari domains with **200 policies.**



Offline Policy Selection

Online Policy Selection

Online + Offline

Online + Offline + BO (independent)

Online + Offline + BO (GP) **= A-ops**

Does not depend on the difficulty of policy training.

Experiments: qualitative results

Experiments: qualitative results

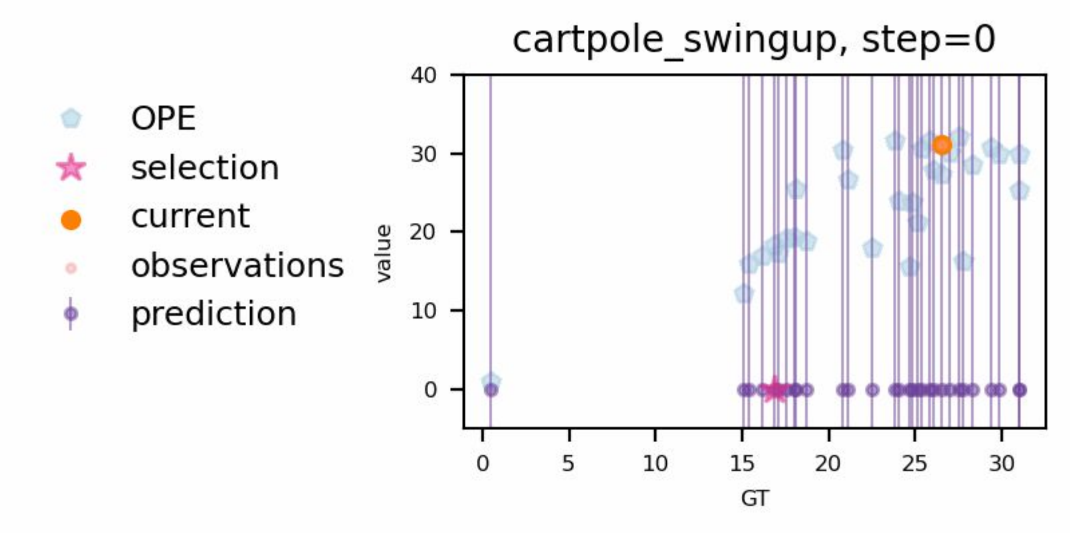
Online Policy Selection

A-ops

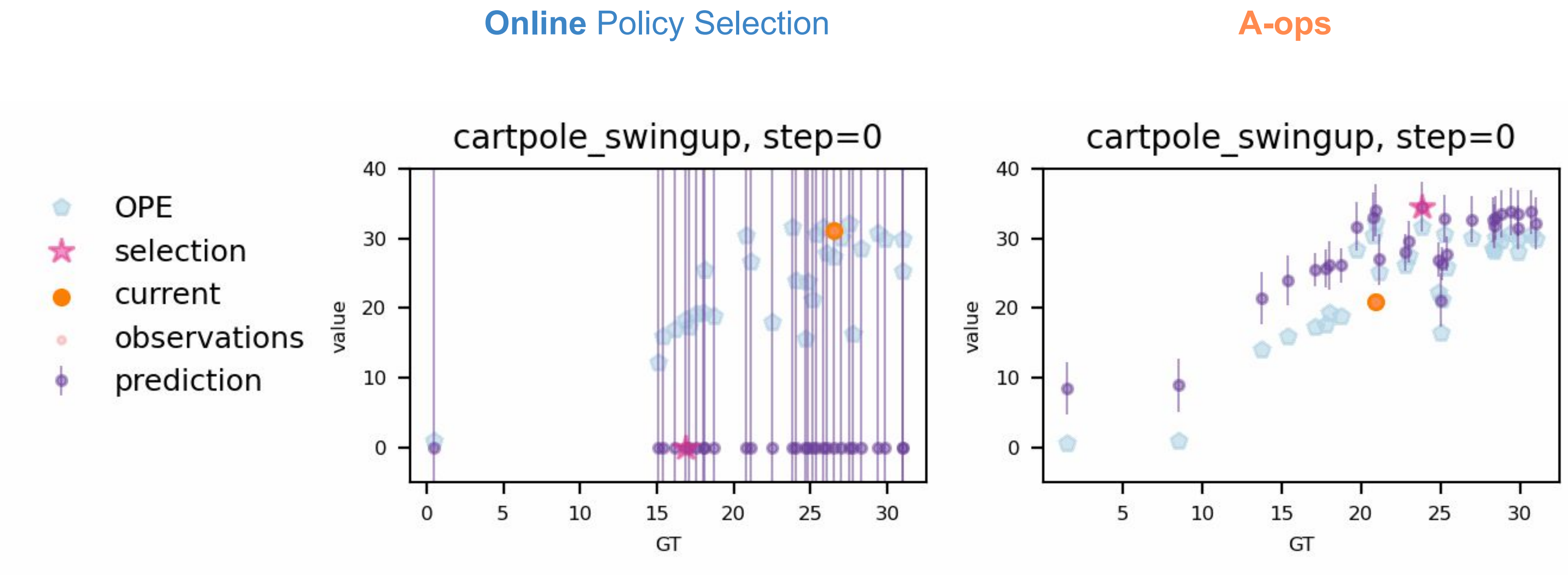
Experiments: qualitative results

Online Policy Selection

A-ops



Experiments: qualitative results



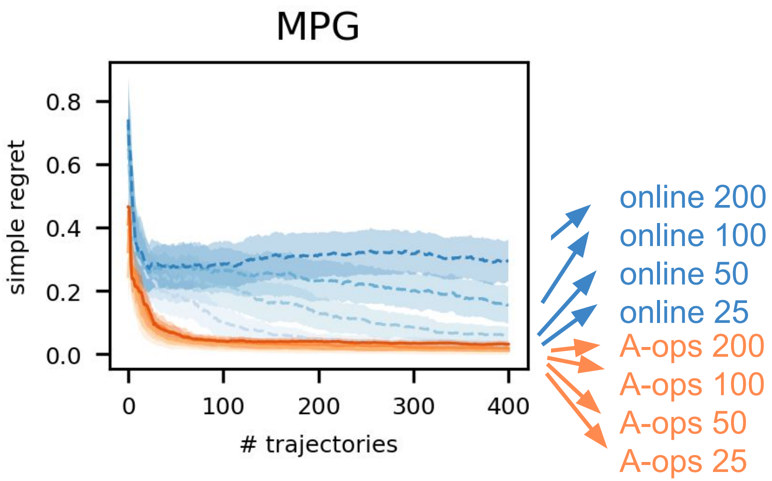
Experiments: ablations

Experiments: ablations

Scaling with number of policies

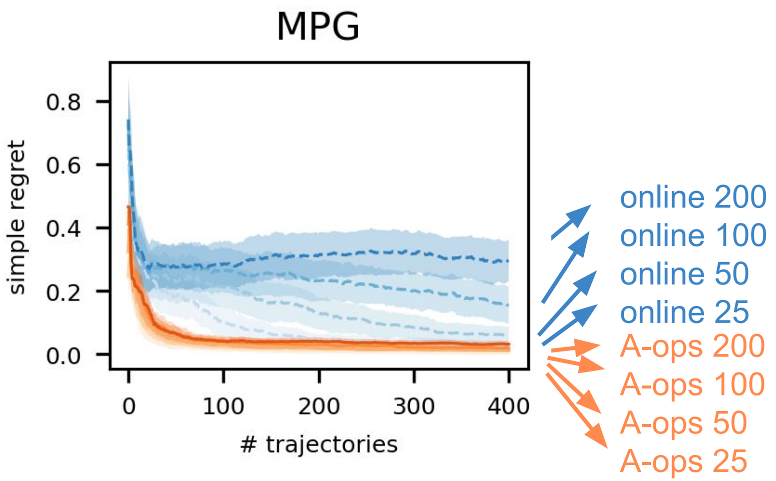
Experiments: ablations

Scaling with number of policies



Experiments: ablations

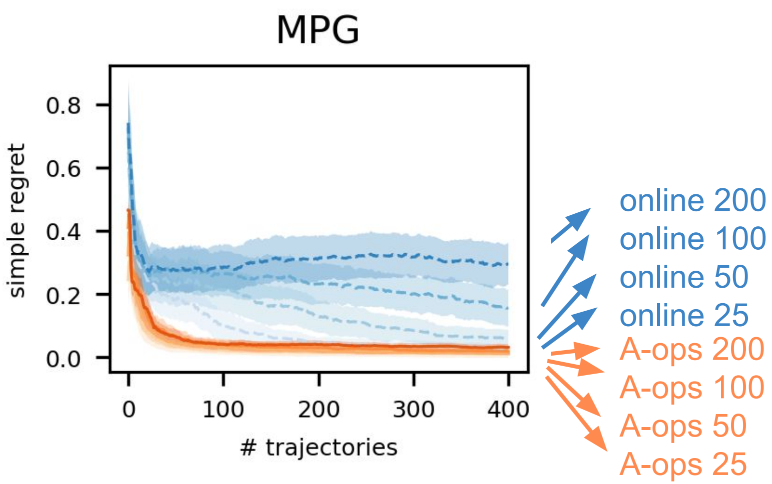
Scaling with number of policies



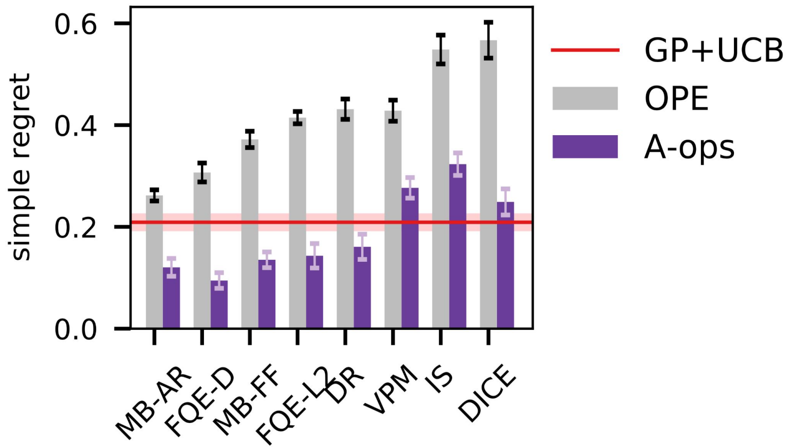
Different OPE methods

Experiments: ablations

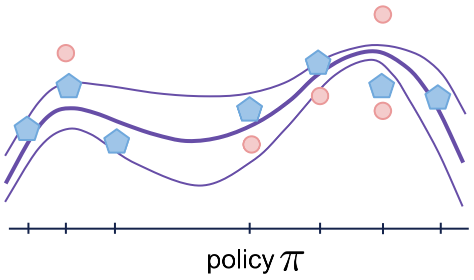
Scaling with number of policies



Different OPE methods

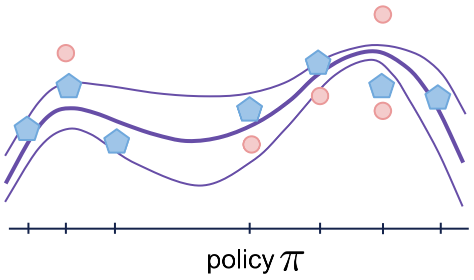


Conclusions, limitations and future work

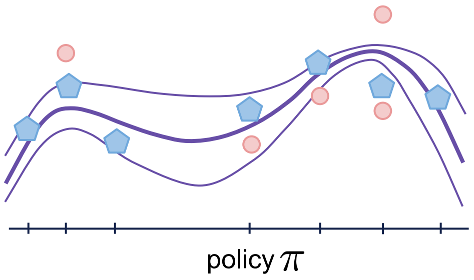


Conclusions, limitations and future work

A-ops has shown promising results in a number of environments.



Conclusions, limitations and future work



A-ops has shown promising results in a number of environments.

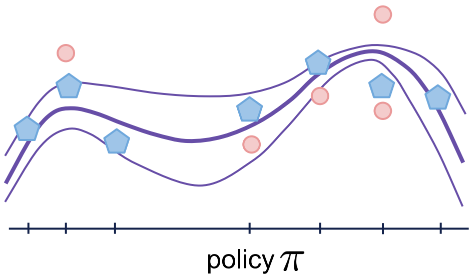
Future work:

Conclusions, limitations and future work

A-ops has shown promising results in a number of environments.

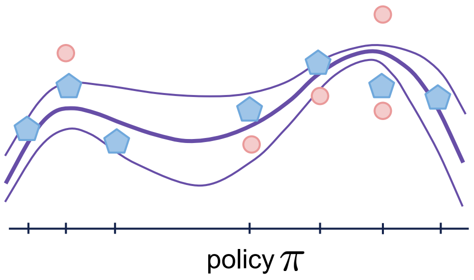
Future work:

- Use of trajectories:
  Help to improve offline policy evaluation, or policies themselves



Conclusions, limitations and future work

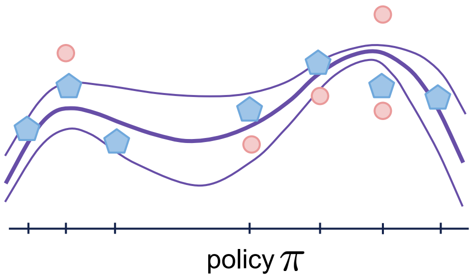
A-ops has shown promising results in a number of environments.



Future work:

- Use of trajectories:
  Help to improve offline policy evaluation, or policies themselves
- States for kernel
  Some states are more informative than others

Conclusions, limitations and future work

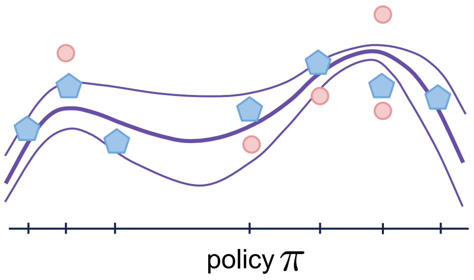


A-ops has shown promising results in a number of environments.

Future work:

- Use of trajectories:
  Help to improve offline policy evaluation, or policies themselves
- States for kernel
  Some states are more informative than others
- Safety risks
  BO methods + safe exploration

Conclusions, limitations and future work

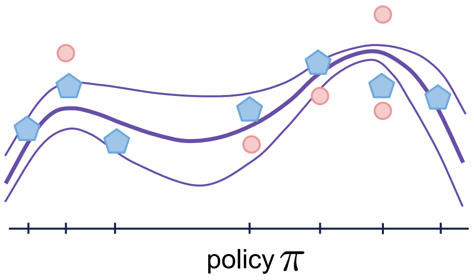


A-ops has shown promising results in a number of environments.

Future work:

- Use of trajectories:
  Help to improve offline policy evaluation, or policies themselves
- States for kernel
  Some states are more informative than others
- Safety risks
  BO methods + safe exploration
- Hope that our findings could accelerate progress in solving real world problems with offline RL!

Conclusions, limitations and future work



A-ops has shown promising results in a number of environments.

Future work:

- Use of trajectories:
  Help to improve offline policy evaluation, or policies themselves

- States for kernel
  Some states are more informative than others

- Safety risks
  BO methods + safe exploration

- Hope that our findings could accelerate progress in solving real world problems with offline RL!

Thank you for your attention!