Understanding, Predicting and Better Resolving Q-Value Divergence in Offline-RL

Yang Yue*[1] Rui Lu*[1] Bingyi Kang*[2] Shiji Song[1] Gao Huang†[1]
[1] Department of Automation, BNRist, Tsinghua University [2] ByteDance

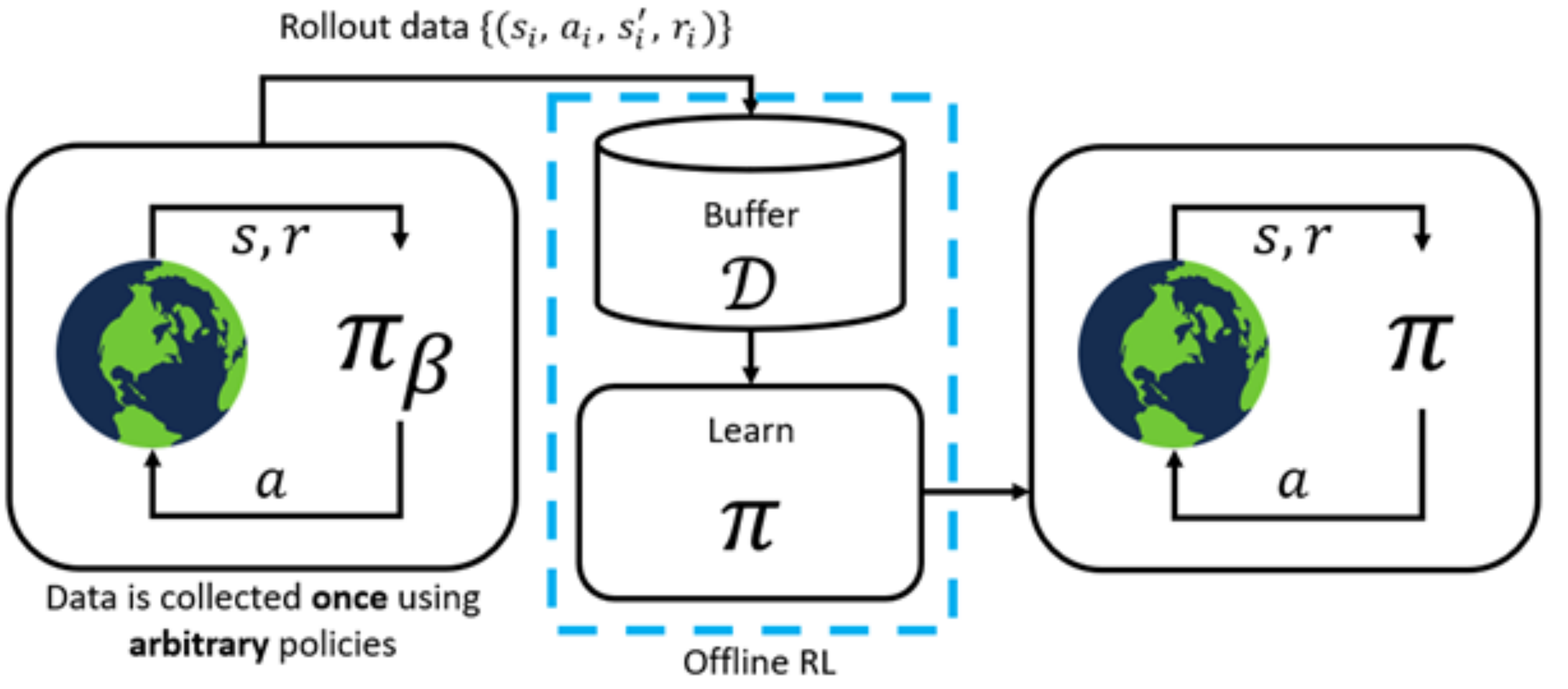
NeurIPS 2023

Presenter: Yang Yue

yueyang22f@gmail.com

Background: Offline RL

Offline RL:



Data is collected once as a dataset. All algorithms are trained offline without collecting data again. (like supervised learning)

Figure credits to Levine et al., 2020.

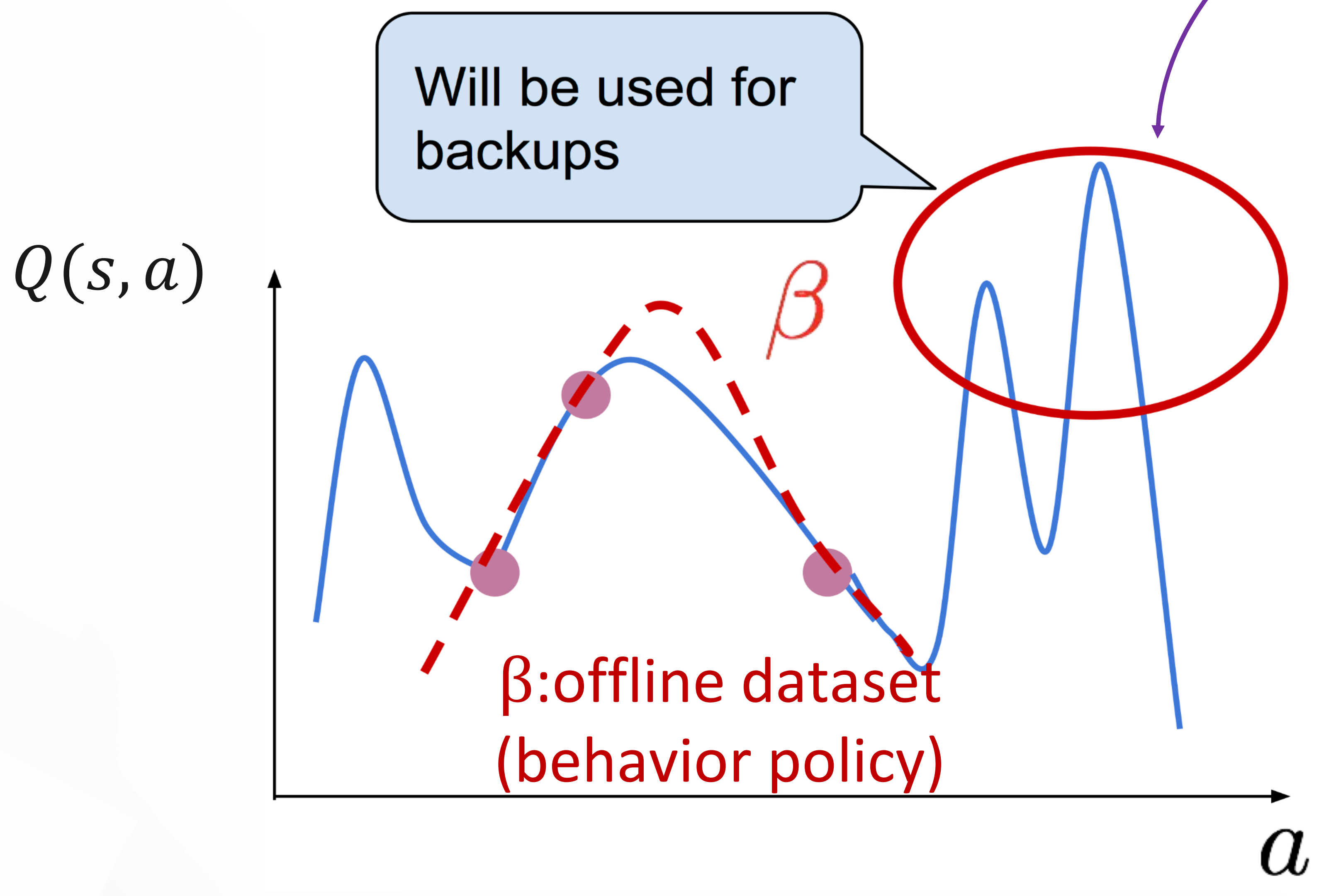
Background: Extrapolation Error

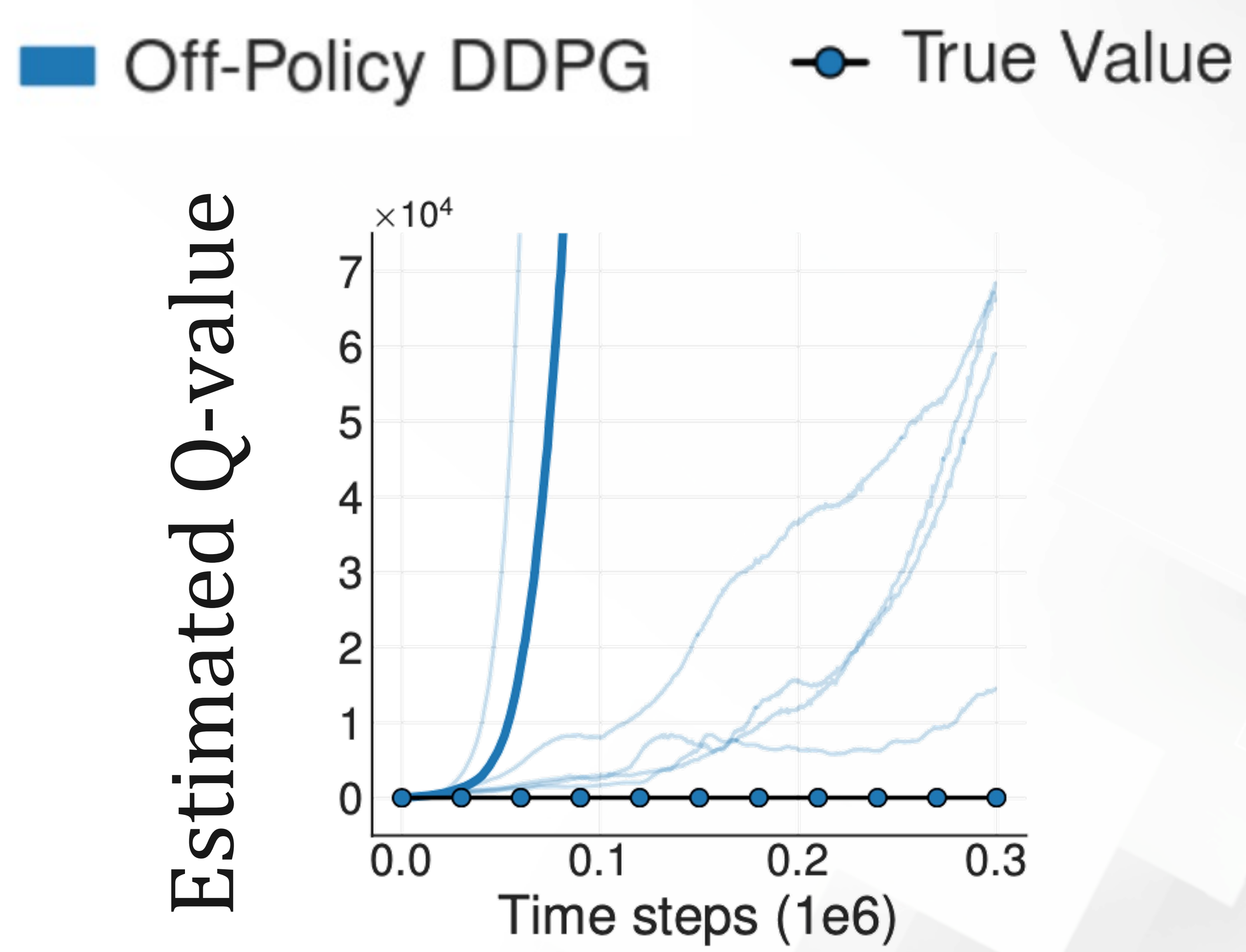
- Fundamental issue in offline RL:

Distributional shift causes cumulative extrapolation errors

$$\bar{Q} \leftarrow r(s,a) + \gamma \max_{a'} \hat{Q}_{\boldsymbol{\theta}}(s', a')$$

Query the value of a’ that are unseen in the dataset





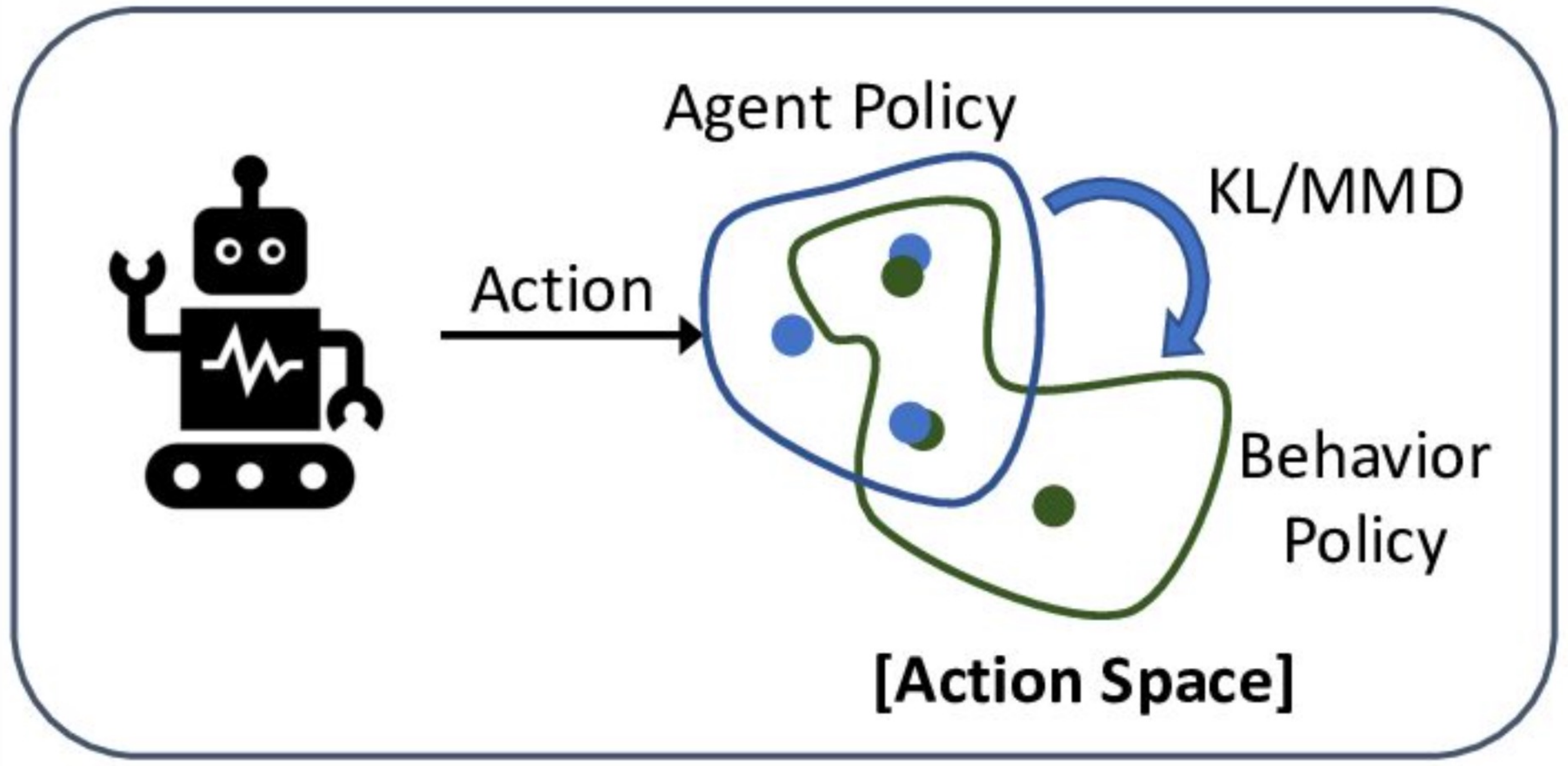
Introduction: Offline RL Methods

- To mitigate extrapolation errors

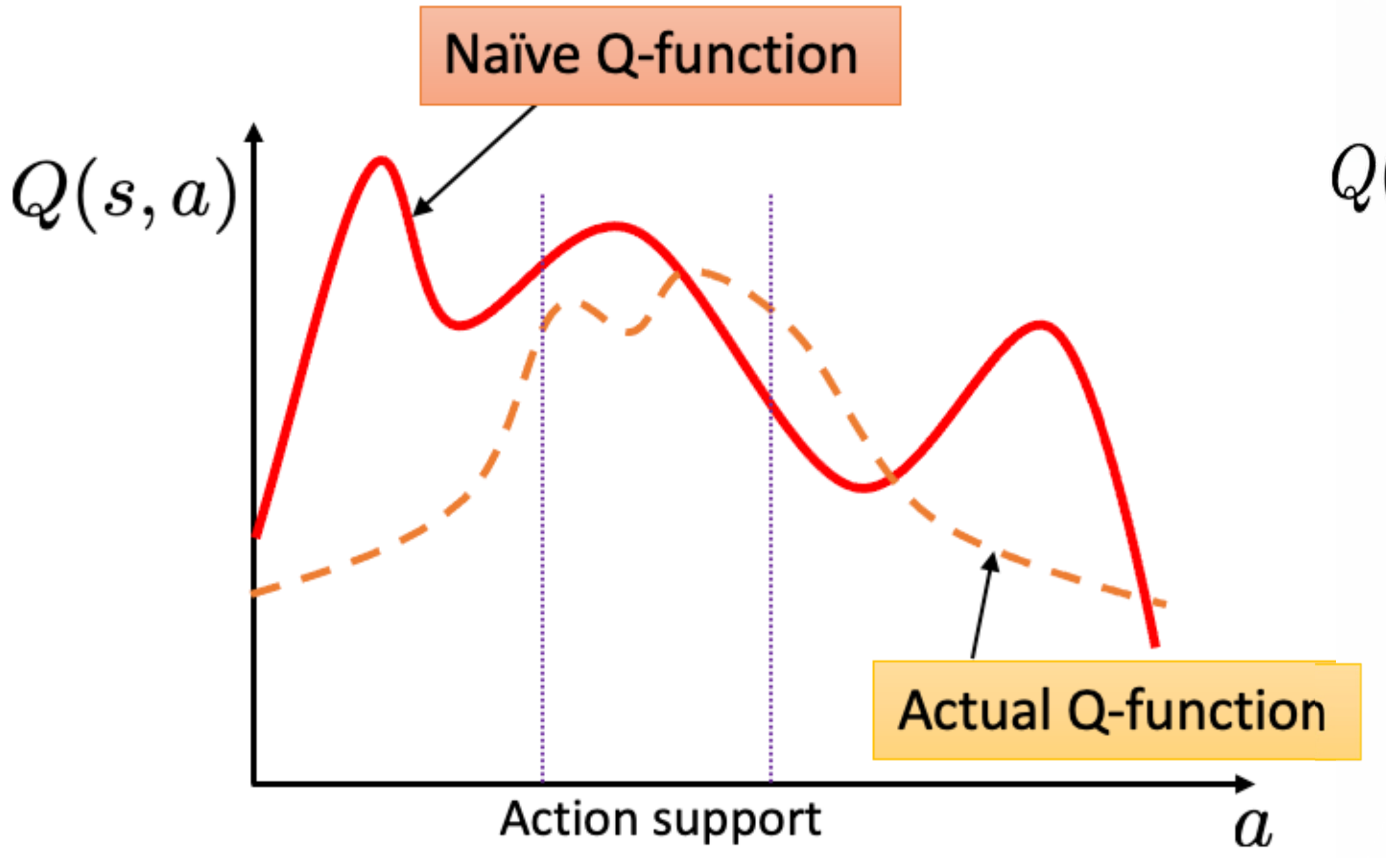
$$\bar{Q} \leftarrow r(s,a) + \gamma \max_{a'} \hat{Q}_{\boldsymbol{\theta}}(s', a')$$

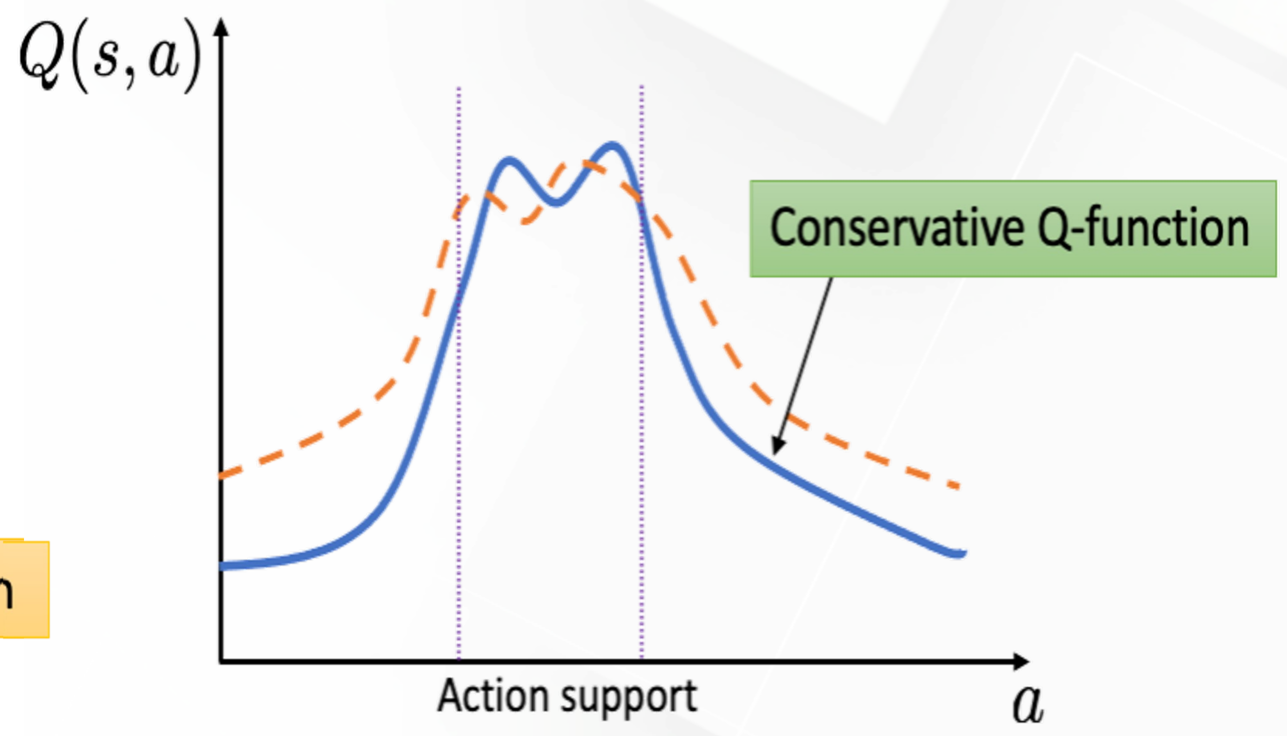
Query the value of a' that are unseen in the dataset

Policy Constraint



Conservative Q-learning



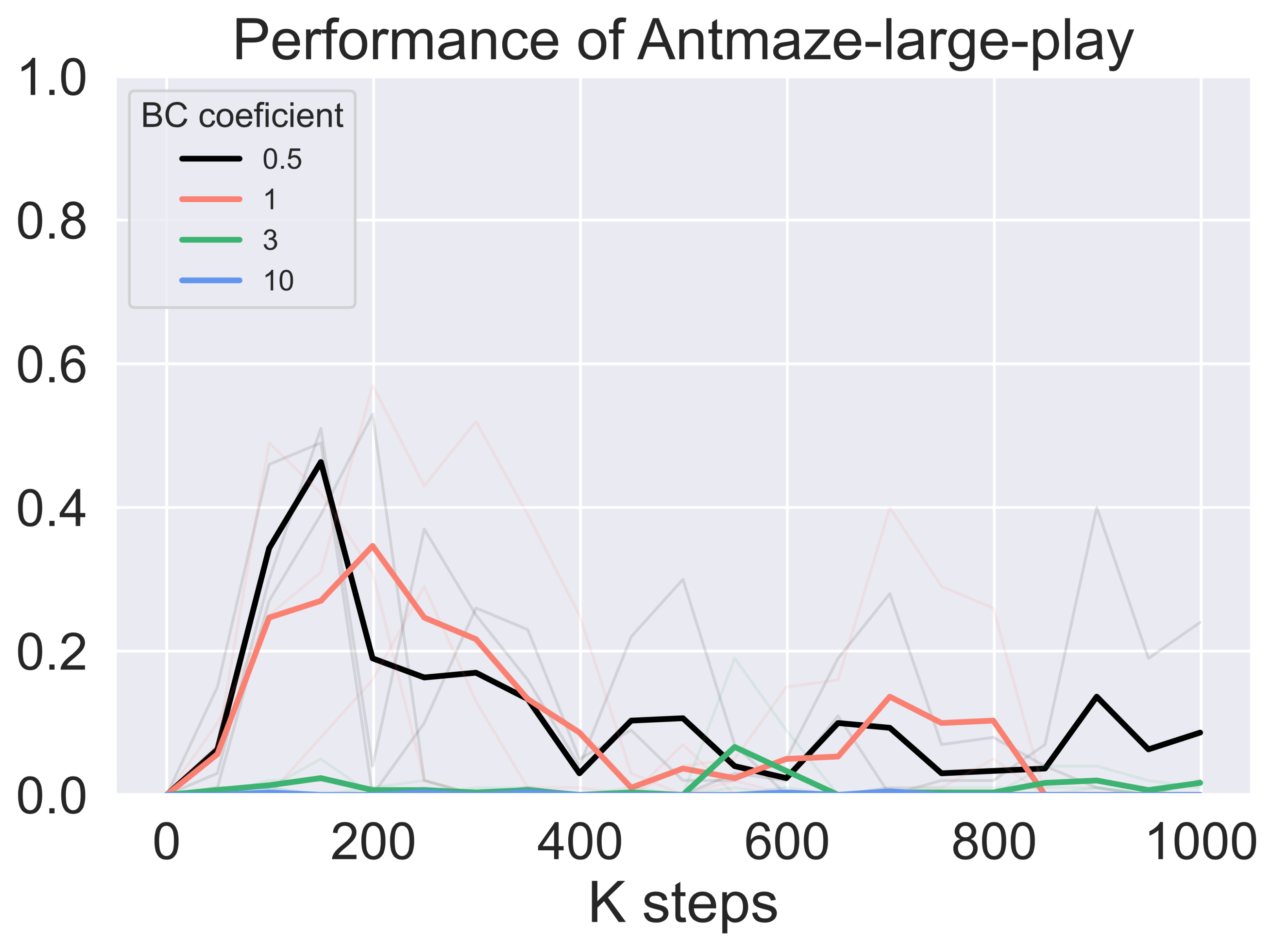


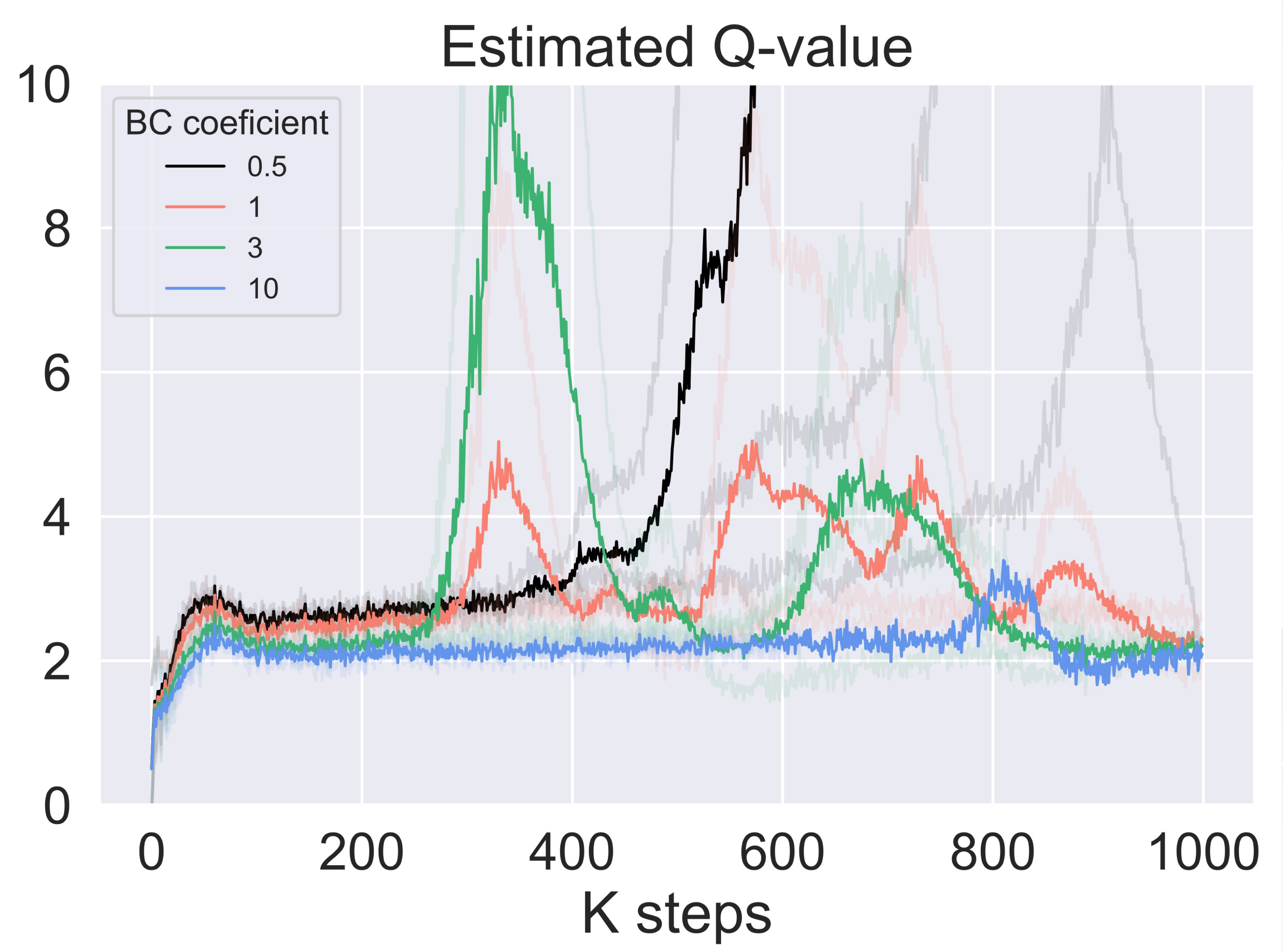
Introduction: Offline RL Methods

- Policy constrain / conservative Q is effective but introduce detriment bias

Introduction: Offline RL Methods

- Policy constrain / conservative Q is effective but introduce detriment bias
  - The trade-off between performance and constraint needs to be balanced manually for each task.



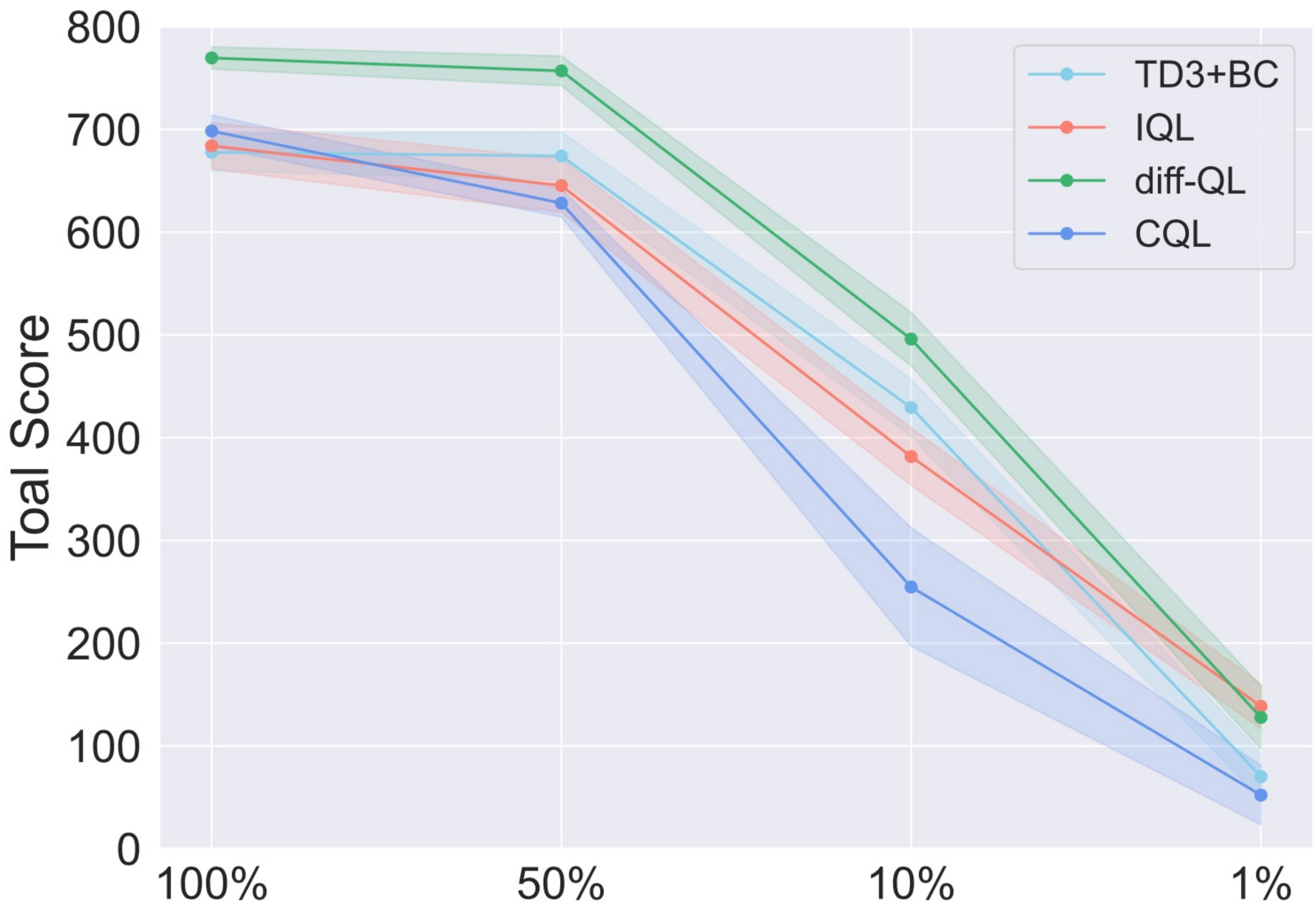


Introduction: Offline RL Methods

清華大學 Tsinghua University

- Policy constrain / conservative Q is effective but introduce detriment bias
  - The trade-off between performance and constraint needs to be balanced manually for each task.
  - Incapable of dealing with divergence in a **data-scarce scenario** where OOD actions are more likely to happen.

The performance of popular offline RL algorithms with the varying X% Mujoco Locomotion dataset



Motivation

- Current methodologies leave several questions unanswered and certain limitations.

  - How does Q-value divergence actually occur?

  - How to avoid detrimental bias?

Self-Excite Eigenvalue Measure

- Identify **self-excitation cycle**

$$\bar{Q} \leftarrow r(s,a) + \gamma \max_{a'} \hat{Q}_{\boldsymbol{\theta}}(s',a')$$

Bellman Update: Update θ by gradient descent

$$\bar{Q}_{\theta}(s,a) \qquad\qquad r(s,a) + \gamma \max_{a'} \hat{Q}_{\boldsymbol{\theta}}(s',a')$$

Neural Network Generalization:
elevate the target Q-value

Self-Excite Eigenvalue Measure

- We develop theoretical tools with **Neural Tangent Kernel (NTK)** to enable three parts
  - Understanding Q-value divergence
  - Predicting Q-value divergence
  - Better resolving Q-value divergence

Understanding Q-value Divergence

- (Theorem 1 and 3 in our paper) Q-value divergence happens when the maximal eigenvalue of the following matrix A_t (namely SEEM) is greater to 0:

$$\boldsymbol{A}_t = (\gamma\phi_{\boldsymbol{\theta}_t}(\boldsymbol{X}_t^*) - \phi_{\boldsymbol{\theta}_t}(\boldsymbol{X}))^\top \phi_{\boldsymbol{\theta}_t}(\boldsymbol{X}) = \gamma \boldsymbol{G}_{\boldsymbol{\theta}_t}(\boldsymbol{X}_t^*, \boldsymbol{X}) - \boldsymbol{G}_{\boldsymbol{\theta}_t}(\boldsymbol{X}, \boldsymbol{X})$$

X is (s, a) points in the dataset

X_t^* is $\left(s', \pi_{\theta_t}(s')\right)$ points, potentially OOD

$G_{\theta_t}(X, X') = \phi_\theta(X)^T \phi_\theta(X')$ is the NTK matrix depicting the strength of the bond between X and X′ due to generalization, where $\phi_\theta(X) := \nabla_\theta Q_\theta(X)$

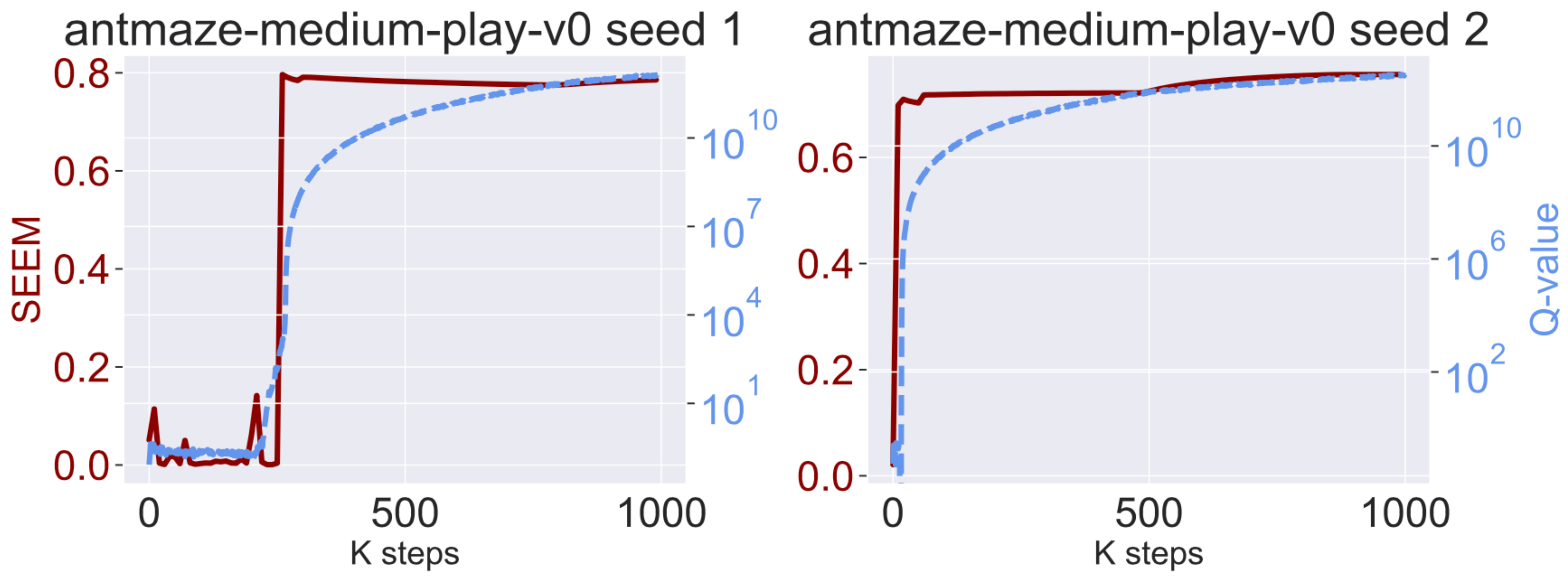
γ is discount factor

Intuition: when the generalization bond between dataset points and OOD points is excessively strong, the divergence happens.

Predicting Q-value Divergence

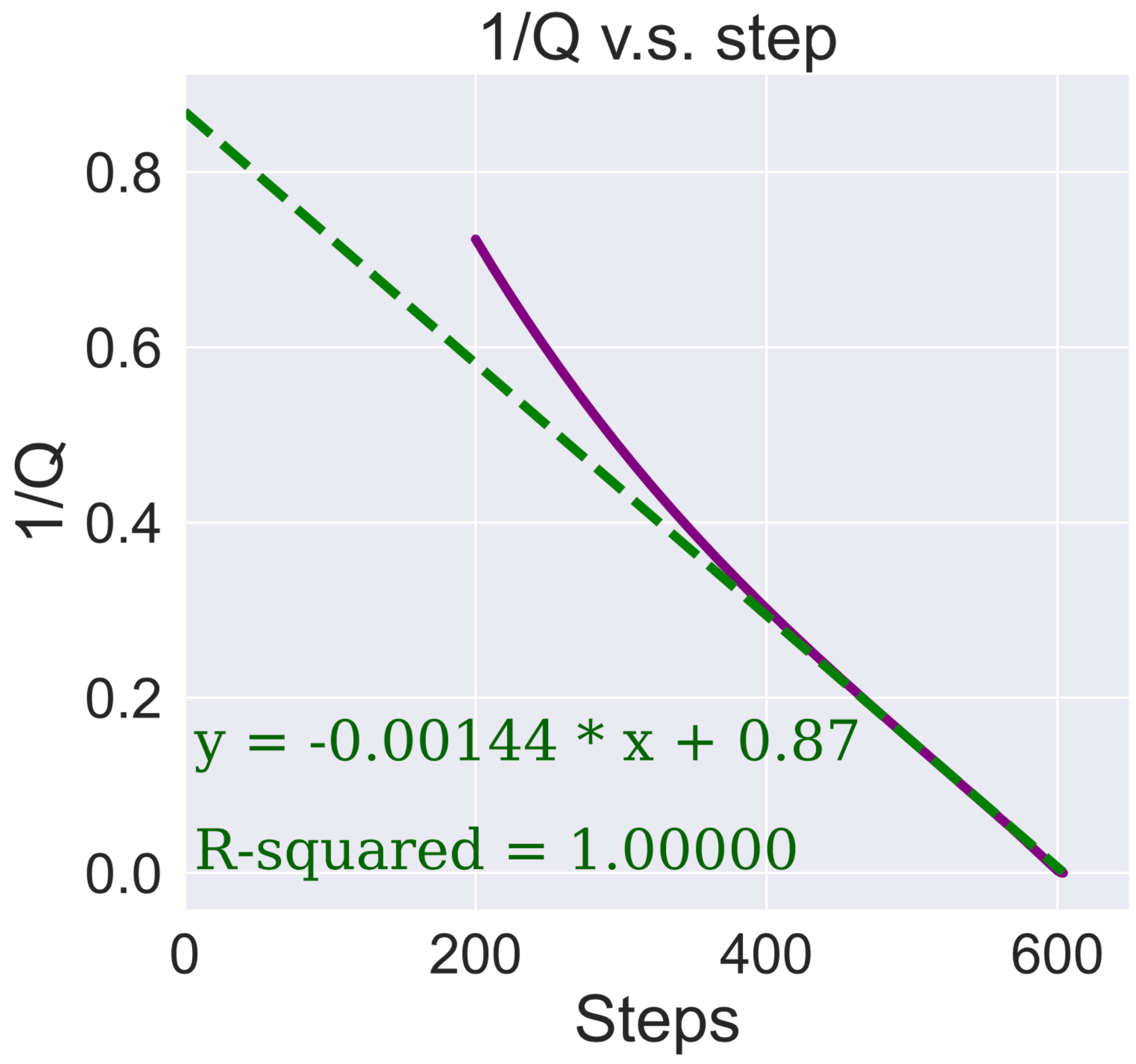
- We can monitor SEEM value to know whether the training will diverge.
- The divergence indication property of SEEM:

the prediction Q-value is stable until the normalized kernel matrix's SEEM rises up to a large positive value



Predicting Q-value Divergence

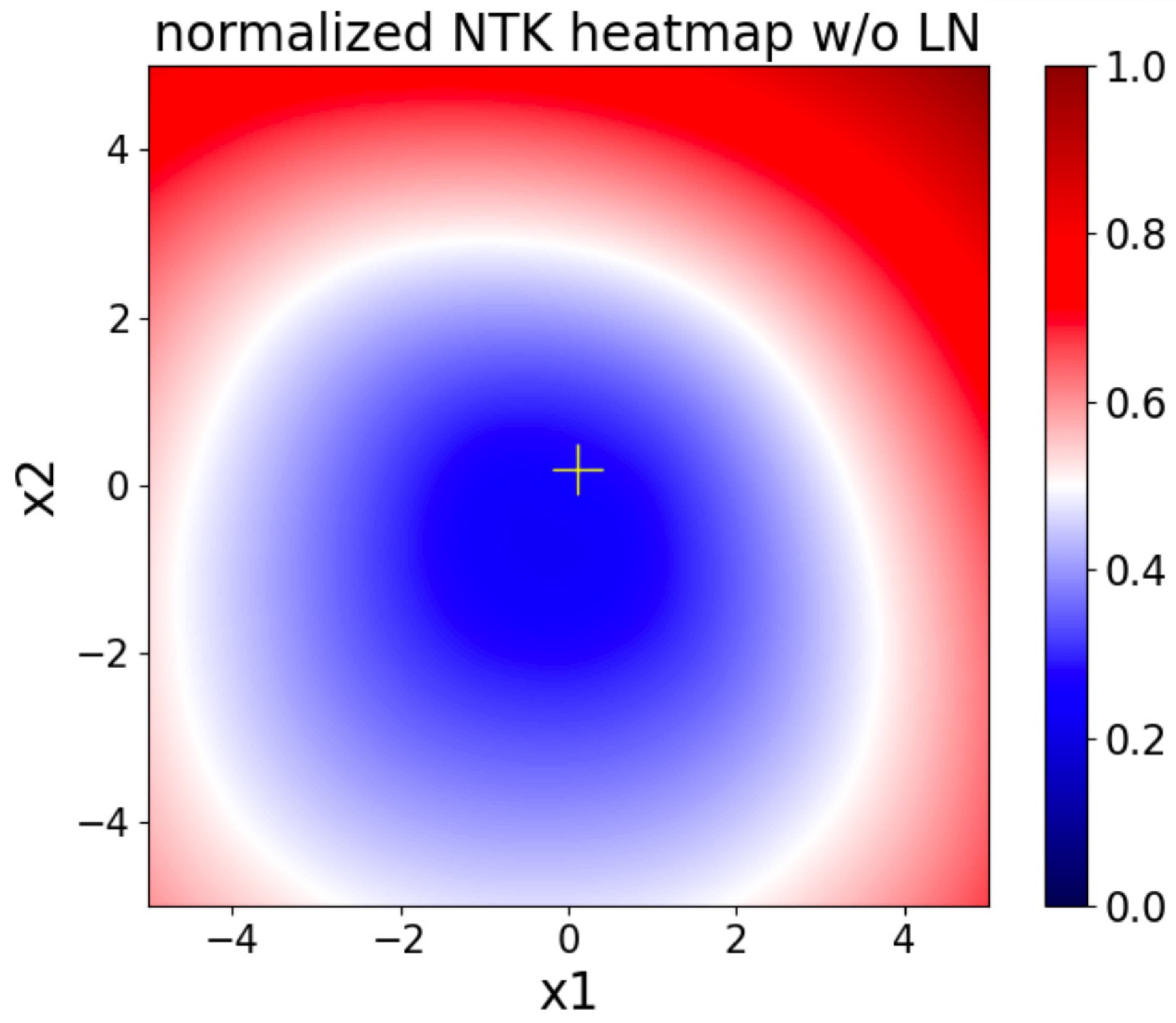
- SEEM is able to predict the order of the growth for the estimated Q-value:
  - With SGD optimizer (Theorem 4): The inverse of Q-value decreases linearly along the timestep.



Better Resolving Q-value Divergence

The abnormal generalization of MLP network:

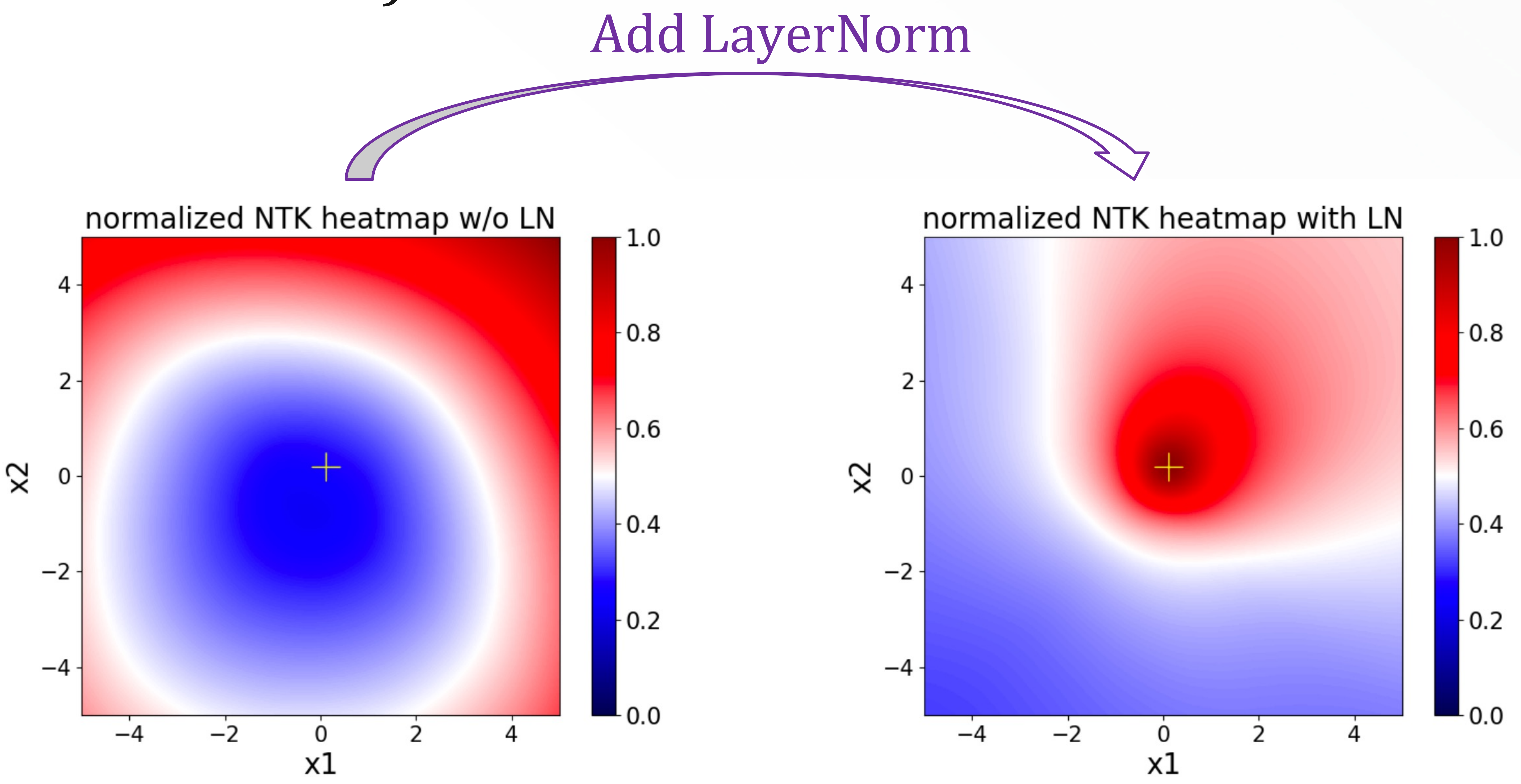
When updating the value of the point x_0 with cross mark, values of points far way x_0 changes more dramatically than near ones.



This indicates an intriguing approach to avoid divergence: **regularizing the model's generalization on out-of-distribution predictions**.

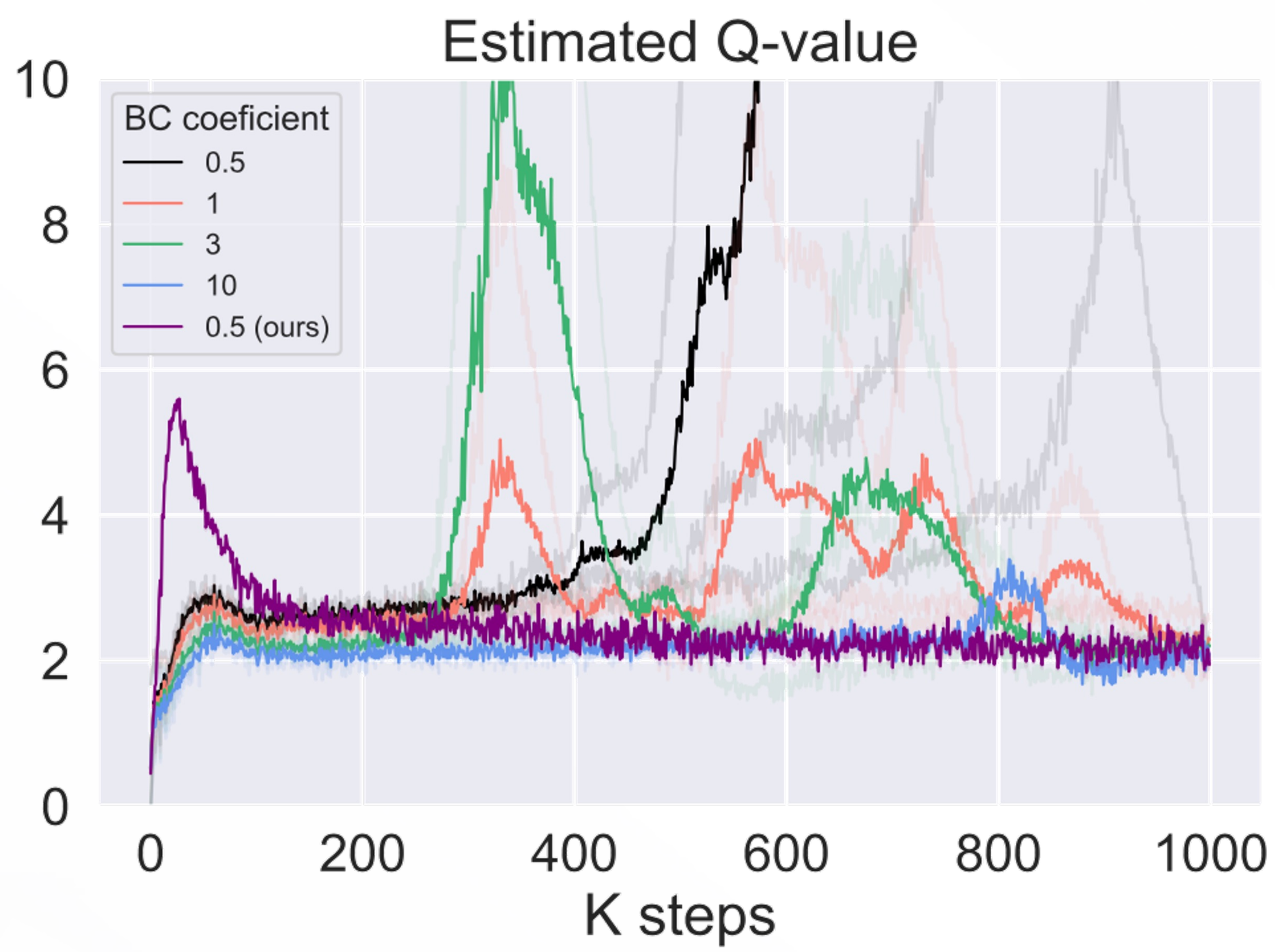
Better Resolving Q-value Divergence

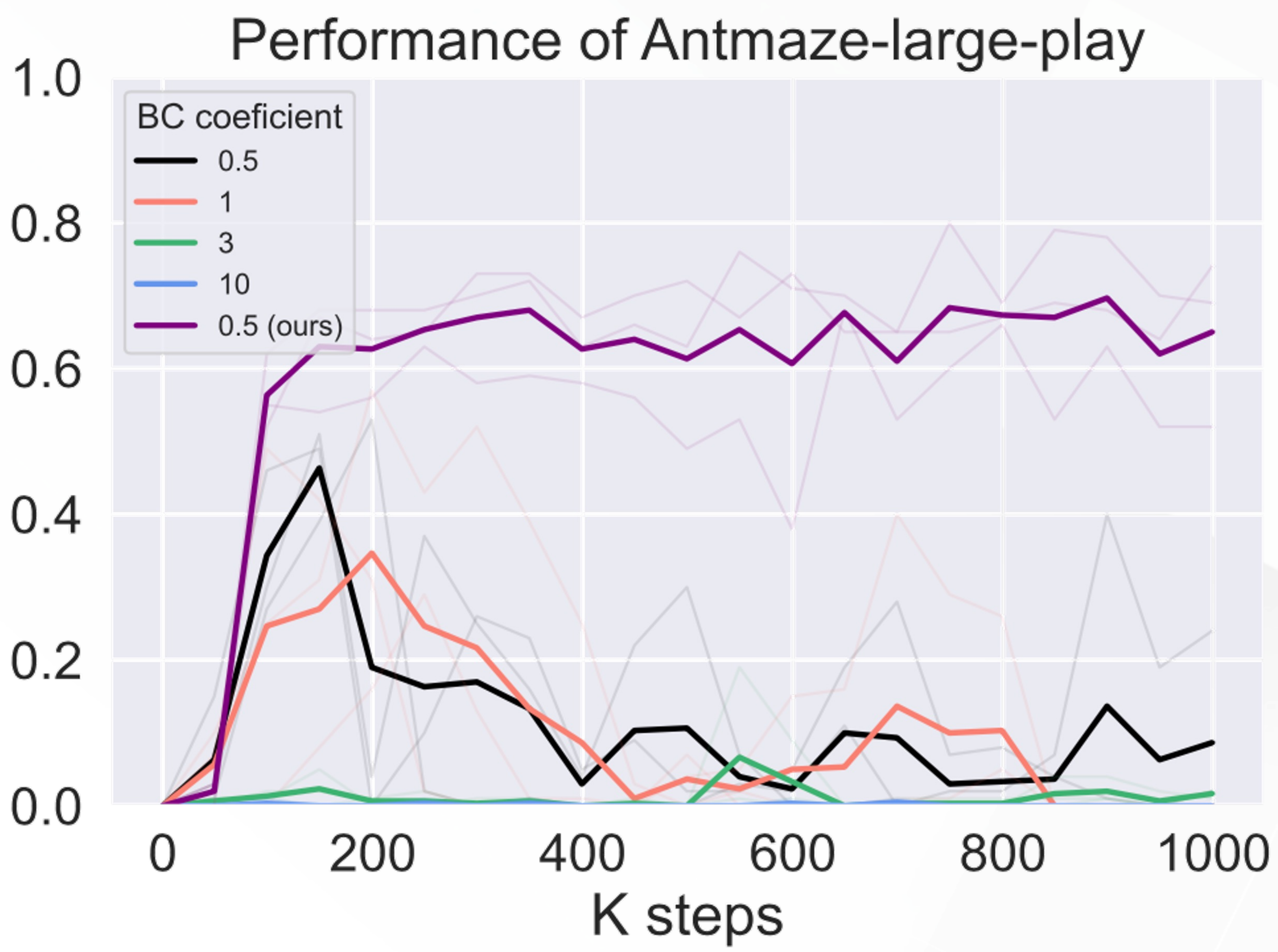
- How to regularize the model's generalization on out-of-distribution predictions
  - Simple Layer Normalization (we theoretically prove LayerNorm bounds SEEM in Proposition 1 and 2)



Better Resolving Q-value Divergence

- Avoid detrimental bias mentioned before





Better Resolving Q-value Divergence

Tsinghua University

- Achieve SOTA results on challenging Antmaze benchmark

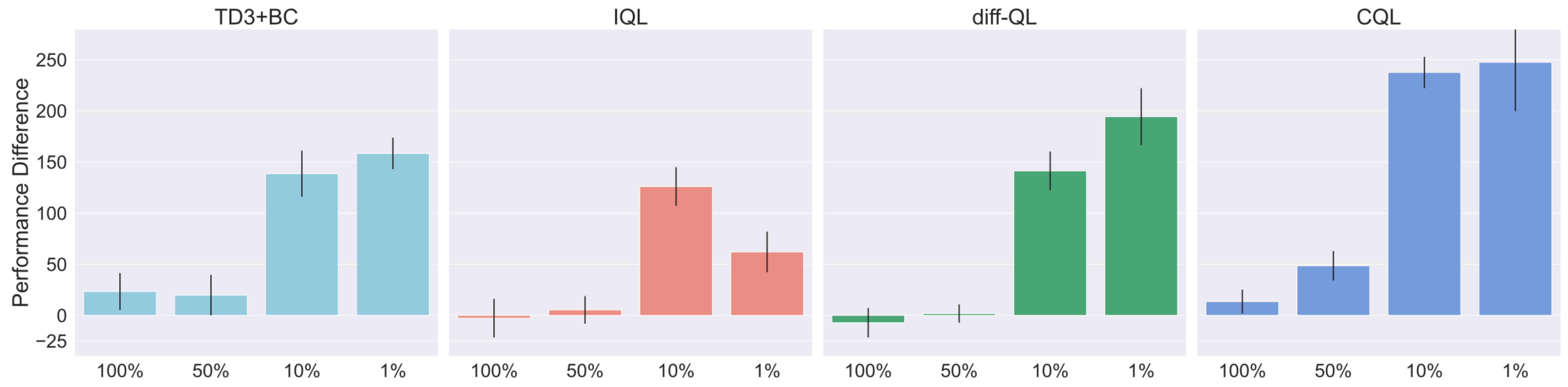
Dataset	TD3+BC	IQL	MSG	sfBC	diff-QL	ours
antmaze-umaze-v0	40.2	87.5	**98.6**	93.3	95.6 (96.0)	94.3 ± 0.5 (97.0)
antmaze-umaze-diverse-v0	58.0	62.2	76.7	86.7	69.5 (84.0)	$\mathbf{88.5 \pm 6.1}$ (95.0)
antmaze-medium-play-v0	0.2	71.2	83.0	**88.3**	0.0 (79.8)	85.6 ± 1.7 (92.0)
antmaze-medium-diverse-v0	0.0	70.0	83.0	**90.0**	6.4 (82.0)	83.9 ± 1.6 (90.7)
antmaze-large-play-v0	0.0	39.6	46.8	63.3	1.6 (49.0)	$\mathbf{65.4 \pm 8.6}$ (74.0)
antmaze-large-diverse-v0	0.0	47.5	58.2	41.7	4.4 (61.7)	$\mathbf{67.1 \pm 1.8}$ (75.7)
average	16.4	63.0	74.4	77.2	29.6 (75.4)	**80.8** (87.4)

Better Resolving Q-value Divergence

Tsinghua University

☐ Effectiveness in data-scarce scenarios

The performance difference between baseline with LayerNorm and without it using the same X% dataset.



Take-home Message

- Q-value divergence arises from the improper neural network generalization.

- SEEM is a framework to accurately depict and predict how improper generalization causes the divergence with NTK tool.

- Regularizing abnormal generalization by LayerNorm
  - Avoid detrimental bias and achieve SOTA
  - Enable algorithms in data-scarce scenarios