Query-Policy Misalignment in Preference-Based Reinforcement Learning

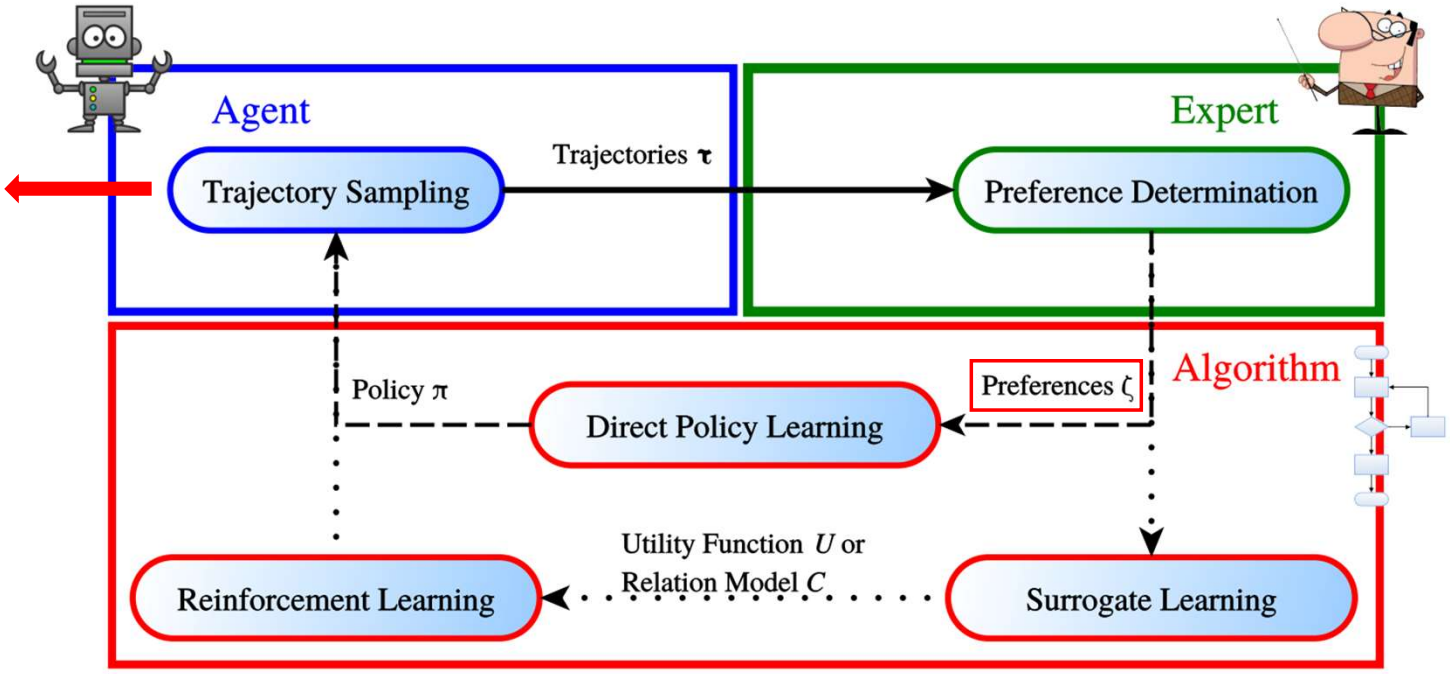
Xiao Hu[*1], **Jianxiong Li**[*1], **Xianyuan Zhan**[†1], **Qing-Shan Jia**[†1], **Ya-Qin Zhang**[†1]
[1] Tsinghua University, Beijing, China
{hu-x21,li-jx21}@mails.tsinghua.edu.cn,
zhanxianyuan@air.tsinghua.edu.cn,jiaqs@tsinghua.edu.cn

ICLR 2024 Spotlight

Background

Preference-Based Reinforcement Learning (PbRL)

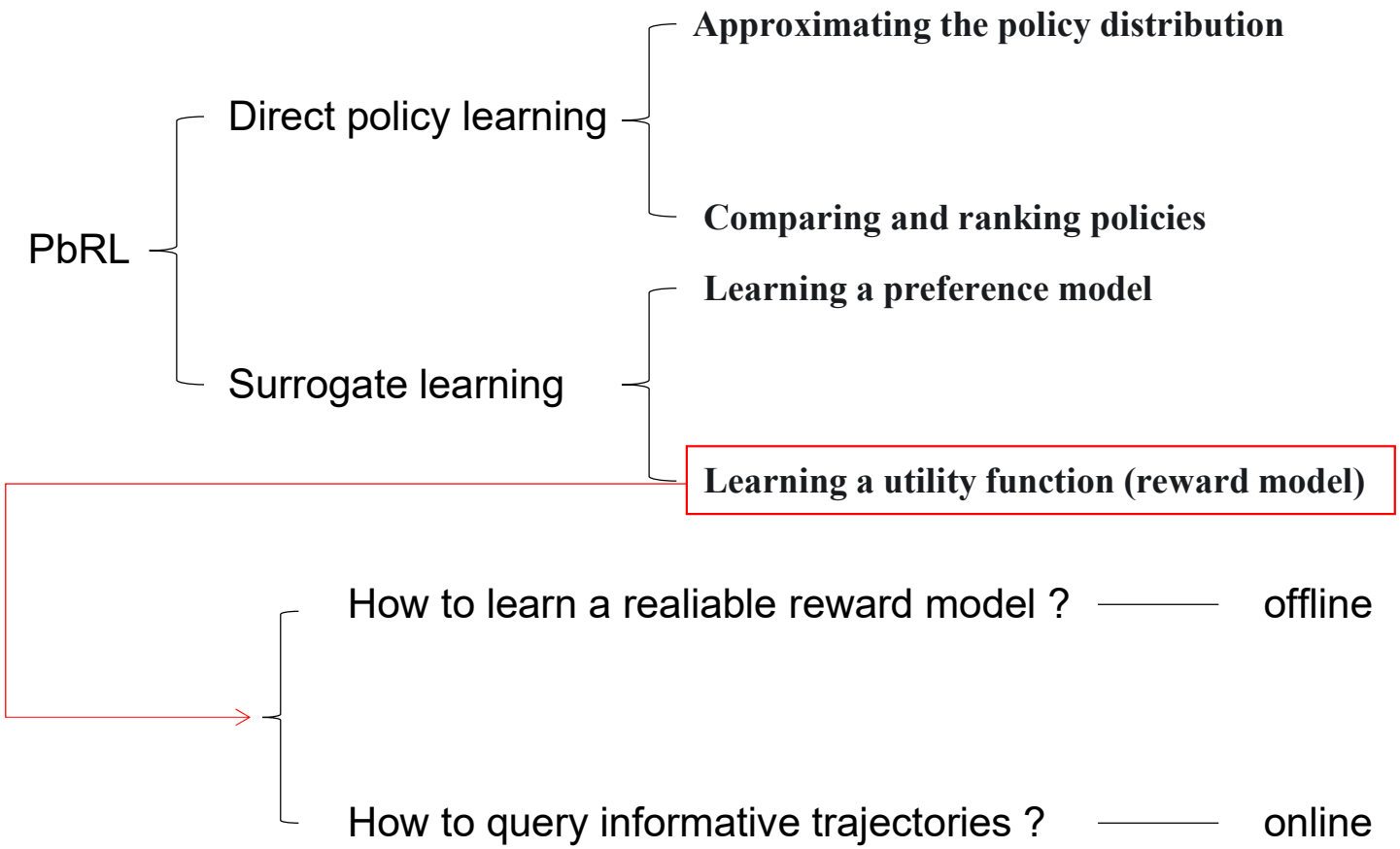
interact with the environment



- **realizable**
- **effortless**
- **scalable**
- **economical**

The agent does not receive a reward after an interacting step, but selects some historical trajectories to query the preferences of the expert.

Background


PbRL
Direct policy learning
Approximating the policy distribution
Comparing and ranking policies
Surrogate learning
Learning a preference model
Learning a utility function (reward model)
How to learn a realiable reward model ?
offline
How to query informative trajectories ?
online

Background

How to learn a realiable reward model ?

trajectory $\sigma : \{s_k, a_k, \cdots, s_{k+L-1}, a_{k+L-1}\}$

preference predictor $P_\psi\left[\sigma^1 \succ \sigma^0\right] = \dfrac{\exp \sum_t \widehat{r}_\psi\left(s_t^1, a_t^1\right)}{\sum_{i\in\{0,1\}} \exp \sum_t \widehat{r}_\psi\left(s_t^i, a_t^i\right)}$

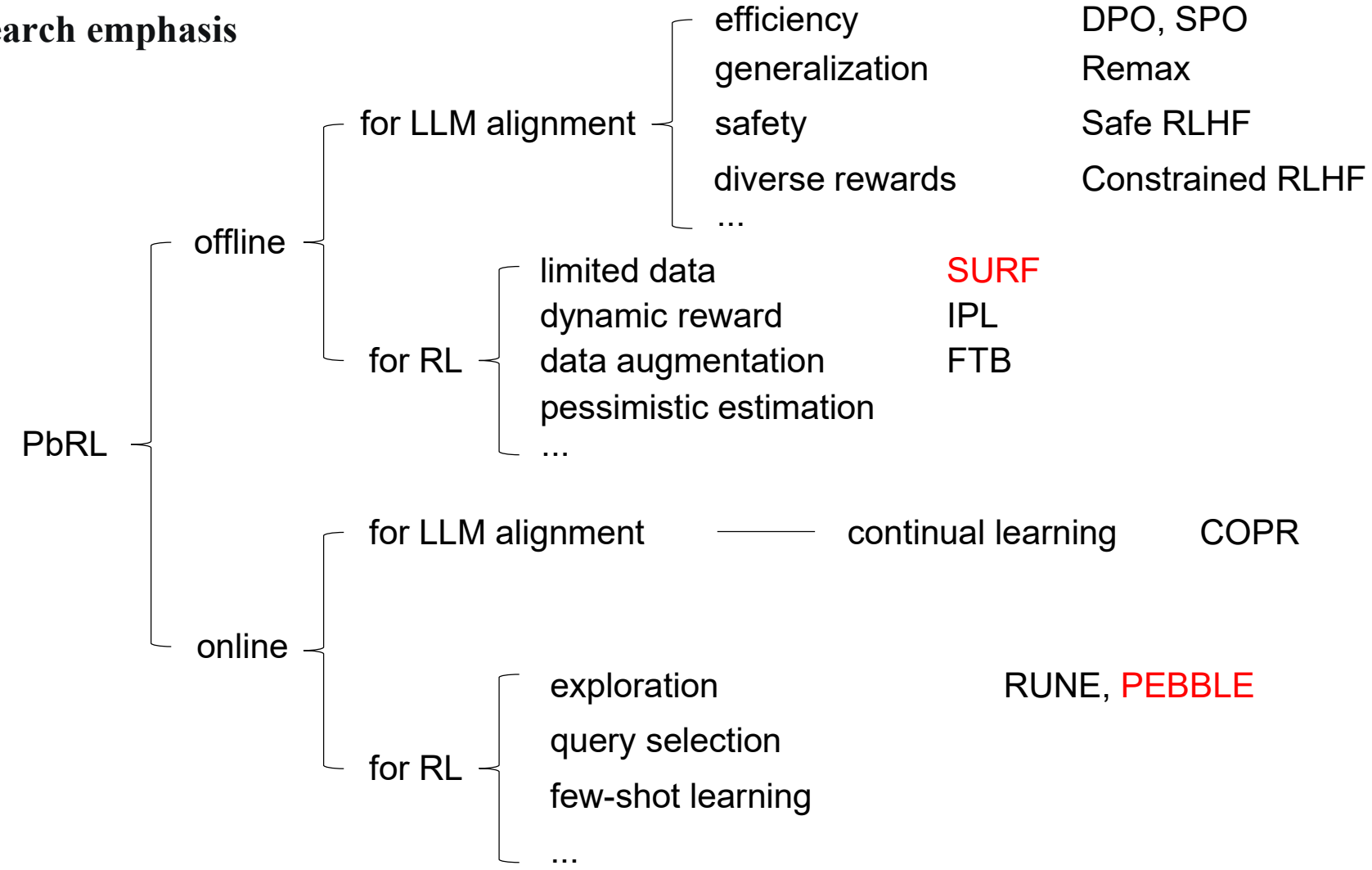
loss $\mathcal{L}^{\text{reward}} = -\underset{(\sigma^0,\sigma^1,y)\sim\mathcal{D}^\sigma}{\mathbb{E}}\left[(1-y)\log P_\psi\left[\sigma^0 \succ \sigma^1\right] + y \log P_\psi\left[\sigma^1 \succ \sigma^0\right]\right]$

How to query informative trajectories ?

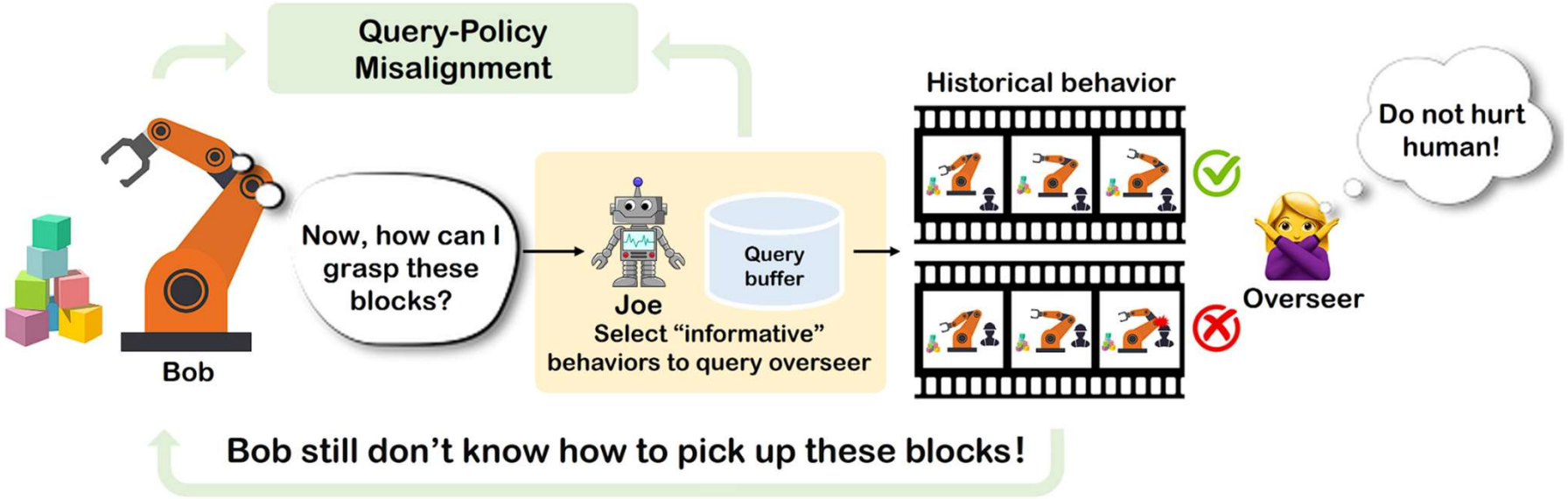
- random query
- entropy-based query
- disagreement-based query

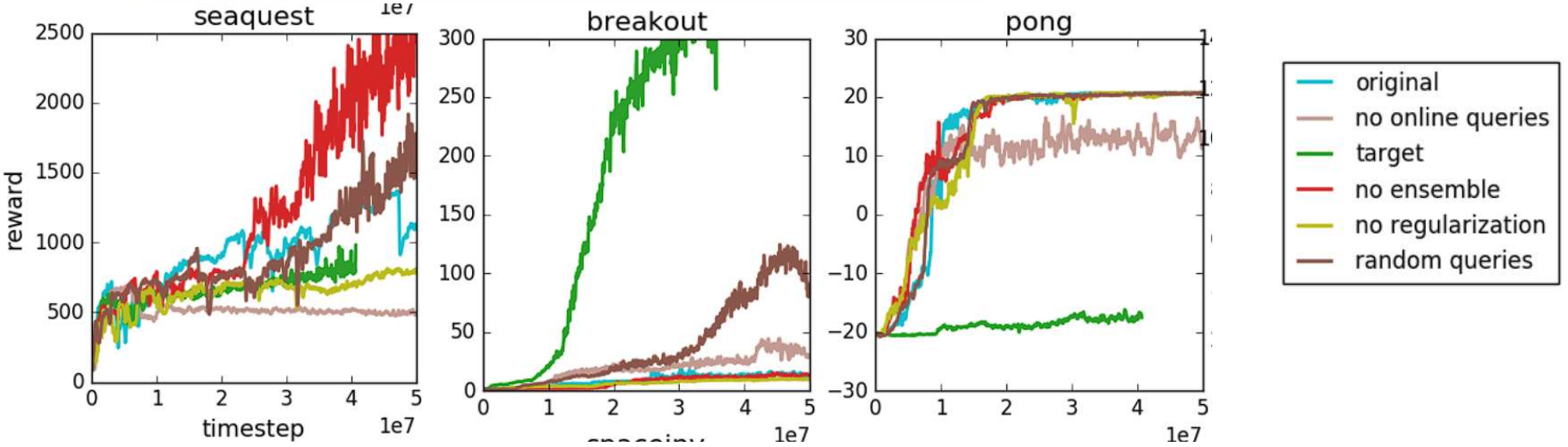
Background

Research emphasis

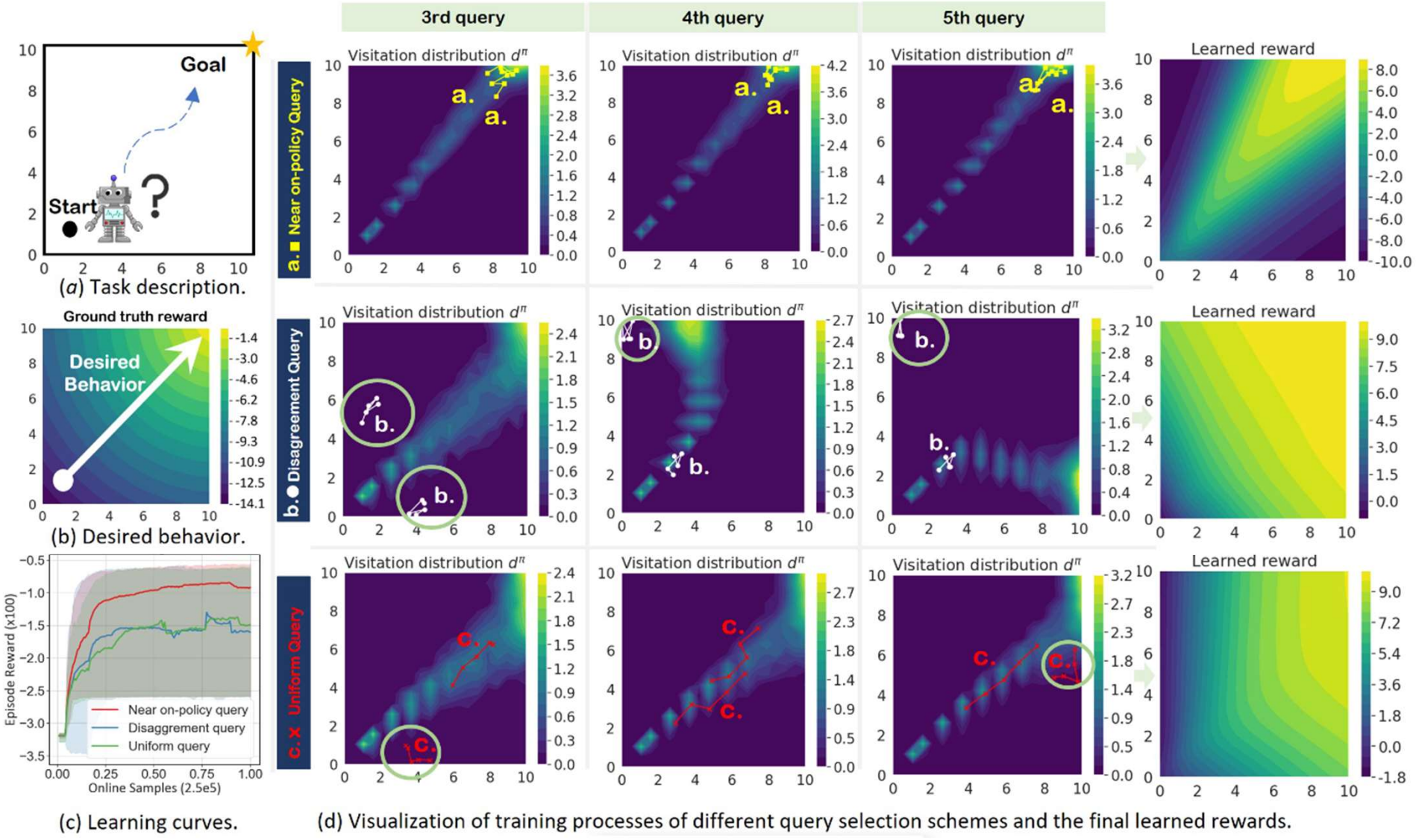


Motivation

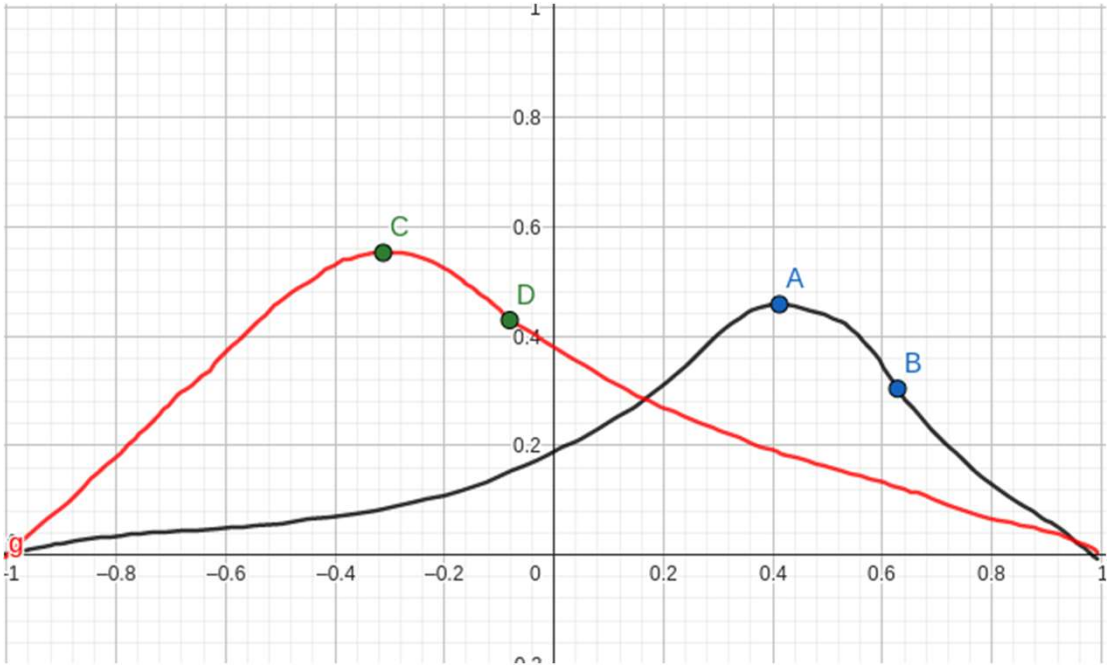

Query-Policy Misalignment
Historical behavior
Do not hurt human!
Now, how can I grasp these blocks?
Query buffer
Joe
Select "informative" behaviors to query overseer
Overseer
Bob
Bob still don't know how to pick up these blocks!


seaquest
breakout
pong
reward
timestep
1e7
original
no online queries
target
no ensemble
no regularization
random queries

Motivation


Goal
Start
?
(a) Task description.
Ground truth reward
Desired Behavior
(b) Desired behavior.
Episode Reward (x100)
Online Samples (2.5e5)
Near on-policy query
Disaggrement query
Uniform query
(c) Learning curves.
3rd query
4th query
5th query
Learned reward
a. Near on-policy Query
b. Disagreement Query
c. Uniform Query
(d) Visualization of training processes of different query selection schemes and the final learned rewards.

Motivation



Theorem 1. *Given the two conditions* $\|\widehat{r}_\psi - r\|_{d^\pi} \leq \epsilon$ *and* $\|Q^\pi_{\widehat{r}_\psi} - \hat{Q}^\pi_{\widehat{r}_\psi}\|_{d^\pi} \leq \alpha$, *the value approximation error* $\|Q^\pi_r - \hat{Q}^\pi_{\widehat{r}_\psi}\|_{d^\pi}$ *is upper bounded as:*

$$\|Q^\pi_r - \hat{Q}^\pi_{\widehat{r}_\psi}\|_{d^\pi} \leq \frac{\epsilon}{1-\gamma} + \alpha \tag{5}$$

near on-policy selection can boost the performance

Method

Algorithm 1: QPA

Input : Frequency of overseer feedback K, number of queries per feedback session M
Near on-policy buffer size N, data augmentation ratio τ

Initialize : Initialize replay buffer $\mathcal{D}$, query buffer $\mathcal{D}^{\sigma}$, near on-policy buffer $\mathcal{D}^{\text{on}}$ with size N

(Option) Unsupervised pretraining [17]

for *each iteration* **do**

  Collect and store new experience $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s, a, r, s')\}$, $\mathcal{D}^{\text{on}} \leftarrow \mathcal{D}^{\text{on}} \cup \{(s, a, r, s')\}$

  if *iteration* $\%K == 0$ **then**

    /* Near on-policy query selection (see Section 5.1) */

    $\{(\sigma^0, \sigma^1)\}_{i=1}^{M} \sim \mathcal{D}^{\text{on}}$

    Query for preferences $\{y\}_{i=1}^{M}$, and store preference $\mathcal{D}^{\sigma} \leftarrow \mathcal{D}^{\sigma} \cup \{(\sigma^0, \sigma^1, y)\}_{i=1}^{M}$

    for *each gradient step* **do**

      Sample a minibatch preferences $\mathcal{B} \leftarrow \{(\sigma^0, \sigma^1, y)\}_{i=1}^{h} \sim \mathcal{D}^{\sigma}$

      /* Data augmentation for reward learning (see Section 5.3) */

      Generate augmented preferences $\hat{\mathcal{B}} \leftarrow \{(\hat{\sigma}^0, \hat{\sigma}^1, y)\}_{i=1}^{h \times \tau}$ based on $\mathcal{B}$

      Optimize $\mathcal{L}^{\text{reward}}$ in Eq. (4) *w.r.t.* $\widehat{r}_{\psi}$ using $\hat{\mathcal{B}}$

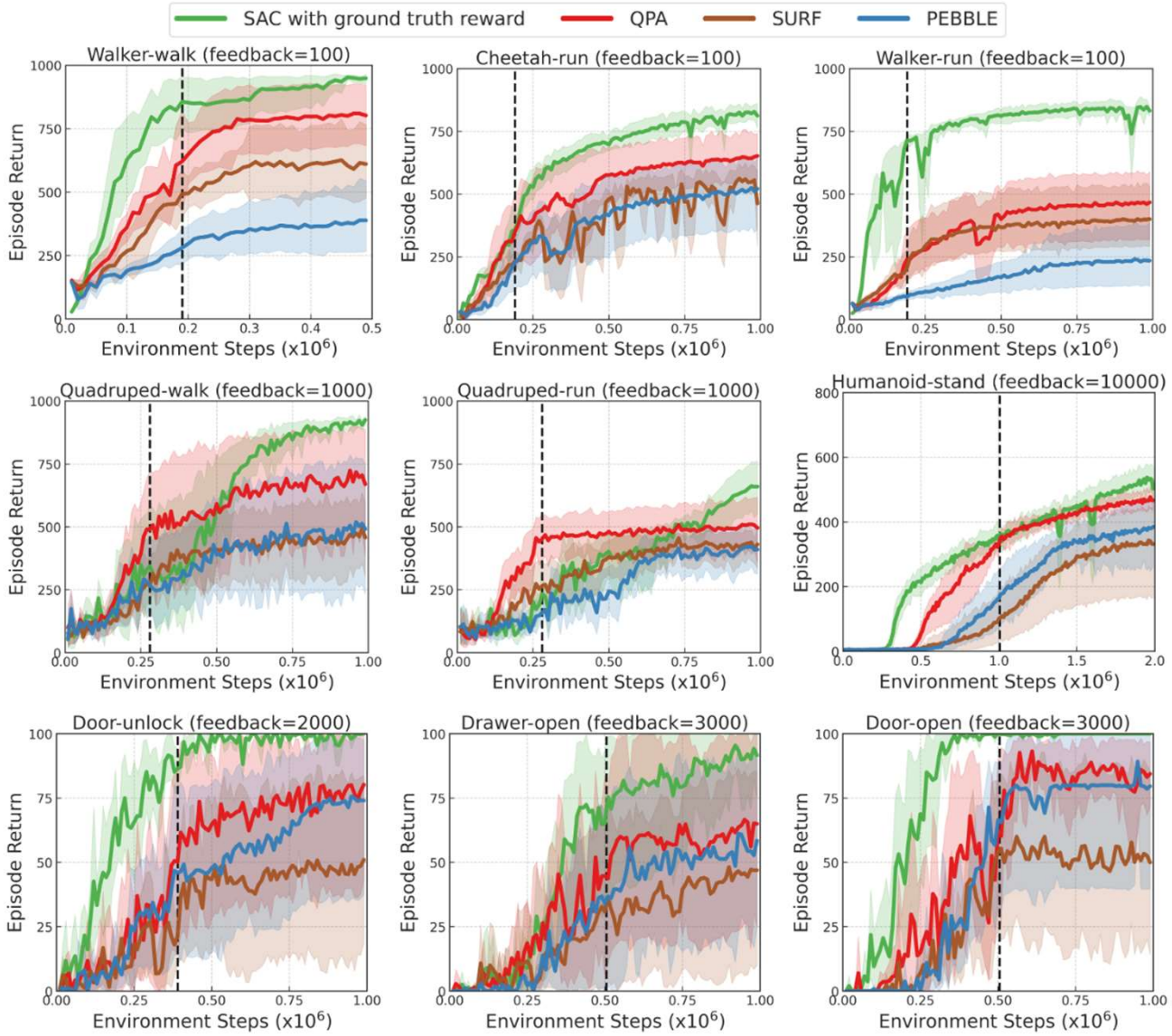
  for *each gradient step* **do**

    /* Hybrid experience replay (see Section 5.2) */

    Sample minibatch $\mathcal{D}_{\text{mini}} \leftarrow \{(s, a, r, s')\}_{i=1}^{\frac{n}{2}} \sim \mathcal{D}$, $\mathcal{D}_{\text{mini}}^{\text{on}} \leftarrow \{(s, a, r, s')\}_{i=1}^{\frac{n}{2}} \sim \mathcal{D}^{\text{on}}$

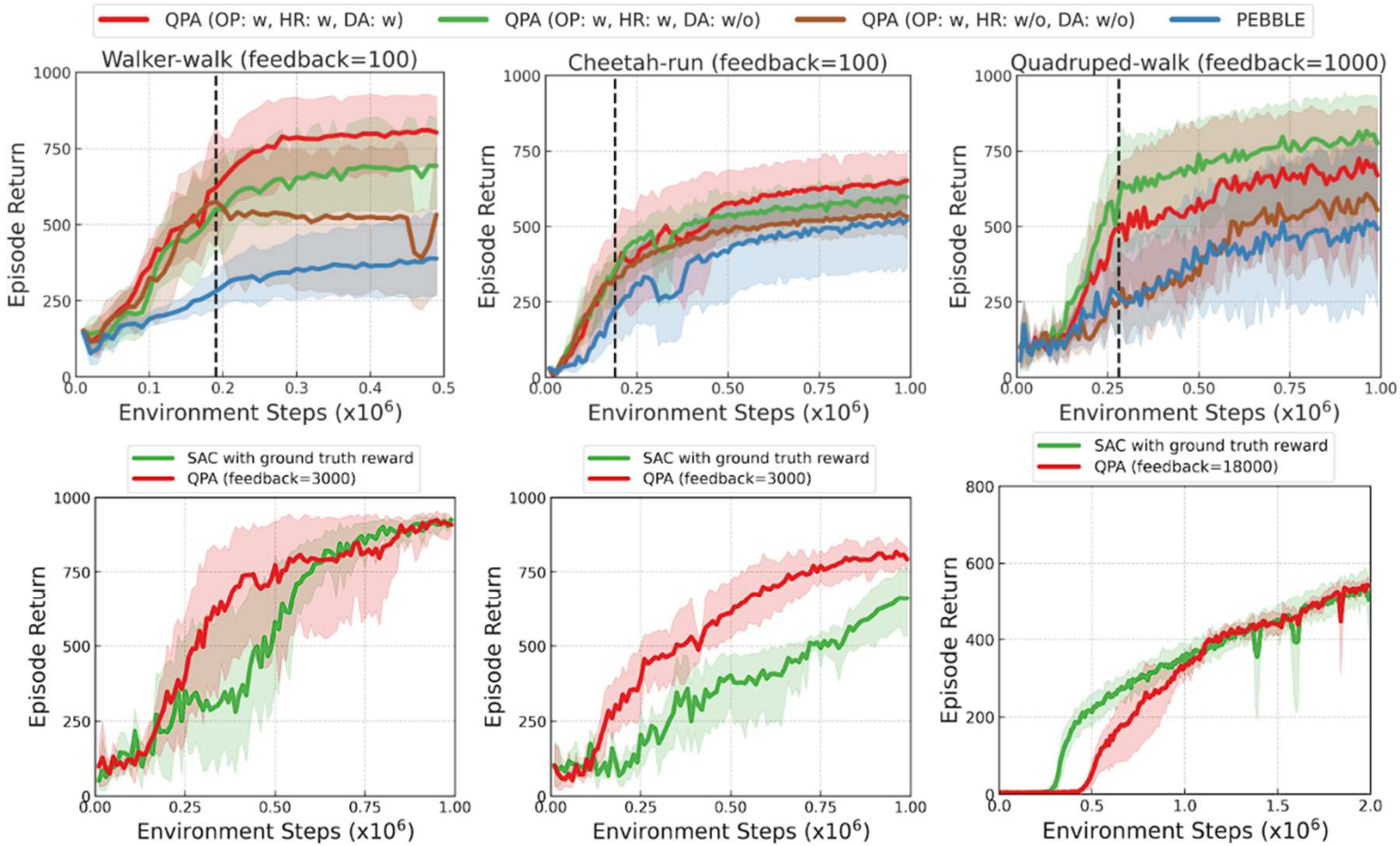
    Optimize SAC agent using $\mathcal{D}_{\text{mini}} \cup \mathcal{D}_{\text{mini}}^{\text{on}}$

Experiment


SAC with ground truth reward
QPA
SURF
PEBBLE
Walker-walk (feedback=100)
Cheetah-run (feedback=100)
Walker-run (feedback=100)
Quadruped-walk (feedback=1000)
Quadruped-run (feedback=1000)
Humanoid-stand (feedback=10000)
Door-unlock (feedback=2000)
Drawer-open (feedback=3000)
Door-open (feedback=3000)
Episode Return
Environment Steps (x10^6)

Experiment

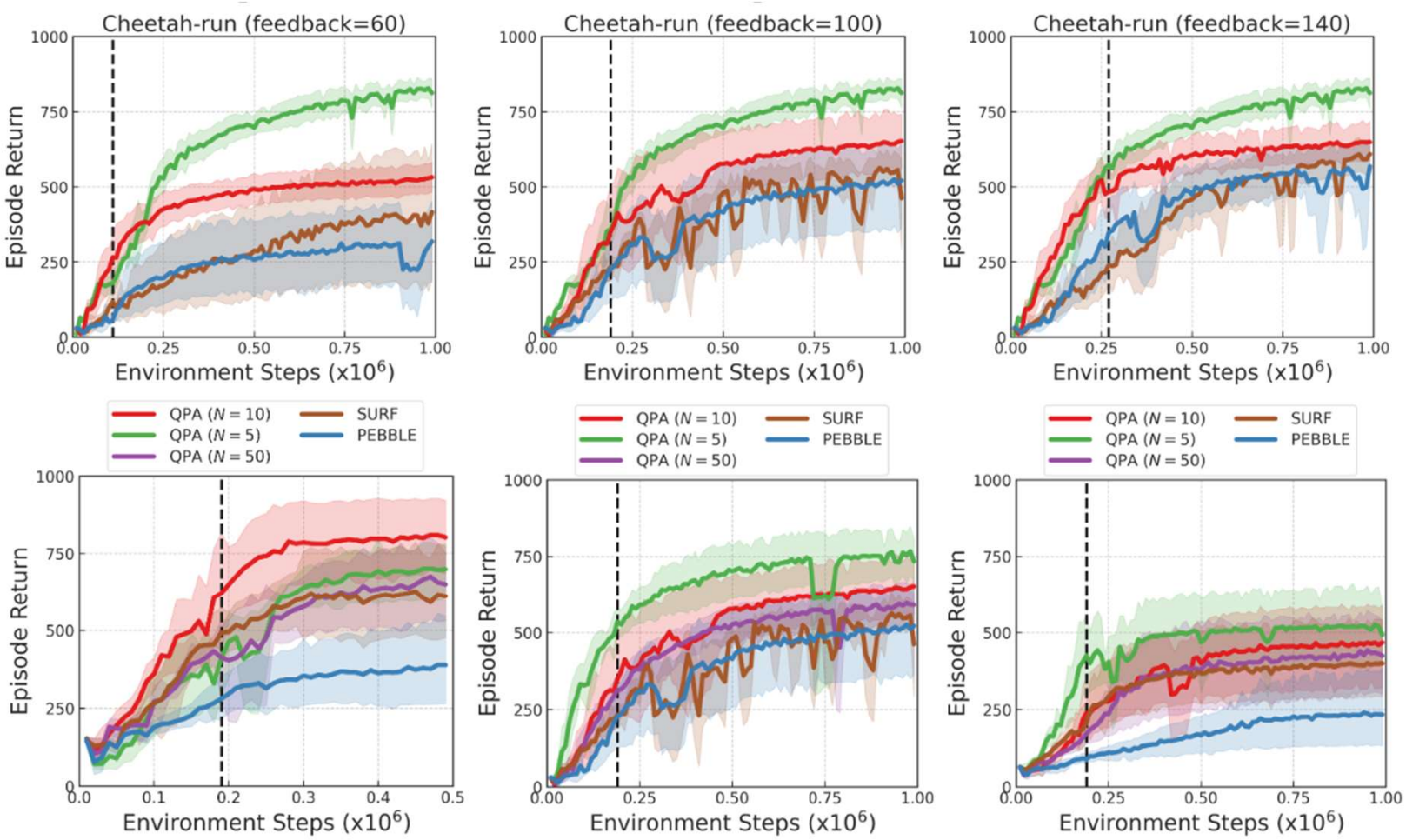
Ablation Study



(a) Quadruped_walk (b) Quadruped_run (c) Humanoid_stand

Experiment

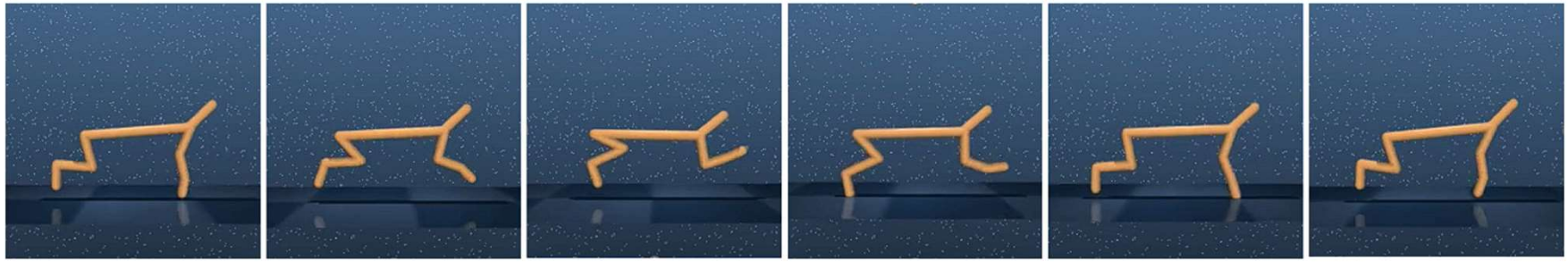
Ablation Study



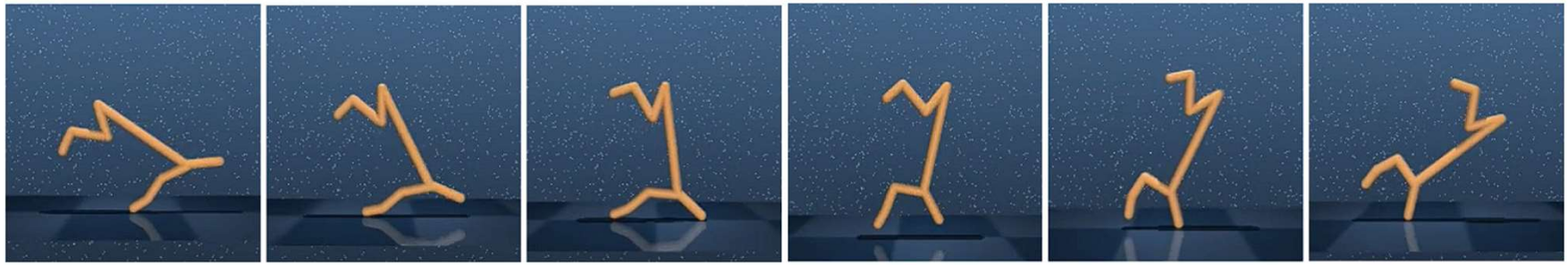
(a) Walker_walk (b) Cheetah_run (c) Walker_run

Experiment

Human Experiments



(a) Agent trained with human preferences



(b) Agent trained with hand-engineered preferences

Discussion

PbRL	Weakly Supervised Learning
limited preferences in offline datasets	semi-supervised learning
active query schemes	active learning
noisy preferences	learning with noisy labels

Difficulties in connecting weakly supervised learning and PbRL

- pair-wise query / relative reward
- distribution shift
- sequence-level preference

Thanks