# Reinforcement Learning from Imperfect Demonstrations under Soft Expert Guidance

Mingxuan Jing[1]*, Xiaojian Ma[12]*, Wenbing Huang[1]*, Fuchun Sun[1], Chao Yang[1], Bin Fang[1], Huaping Liu[1]

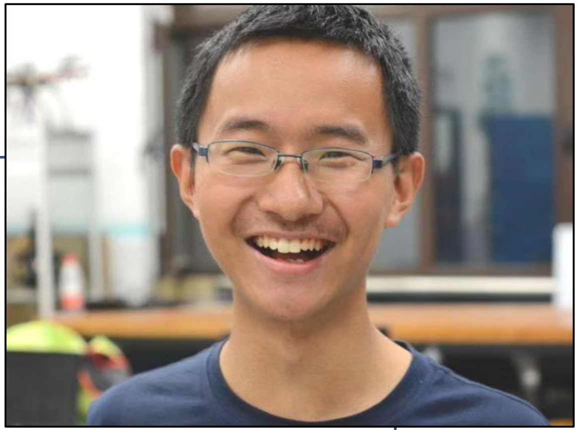
[1]Beijing National Research Center for Information Science and Technology (BNRist), State Key Lab on Intelligent Technology and Systems,
Department of Computer Science and Technology, Tsinghua University,
[2]Center for Vision, Cognition, Learning and Autonomy,Department of Statistics, UCLA
*These authors contributes the same

# • About the student authors


**Mingxuan Jing**

Ph.D. student
Tsinghua University,
Advisor: Prof. Fuchun Sun.

**Research interests:**
- Imitation learning
- Robotics and control
- Robot-human interaction



**Xiaojian Ma**

Master student
UCLA,
Advisor: Prof. Songchun Zhu

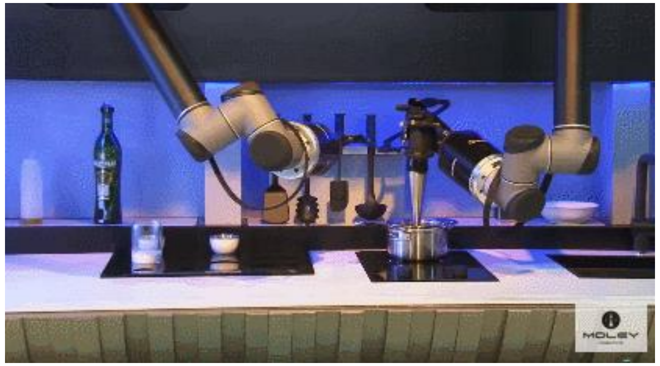
**Research interests:**
- Imitation learning
- Robotics
- Reinforcement Learning

# Content

# • Background & Motivation

- **The goals of robot learning:**

Long-term and complex skills

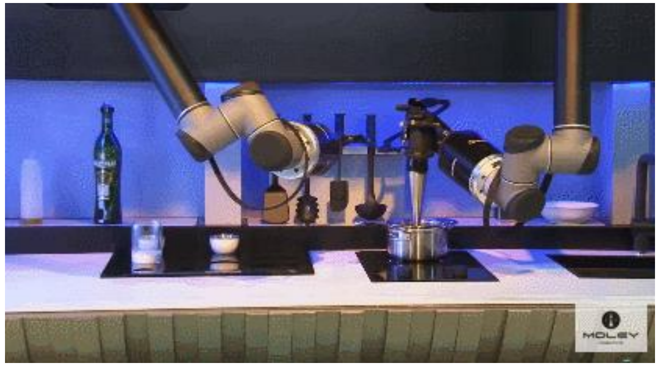


Human-friendly and collaborative

# • Background & Motivation

- **How to achieve these goals:**

Long-term and complex skills




Reinforcement Learning

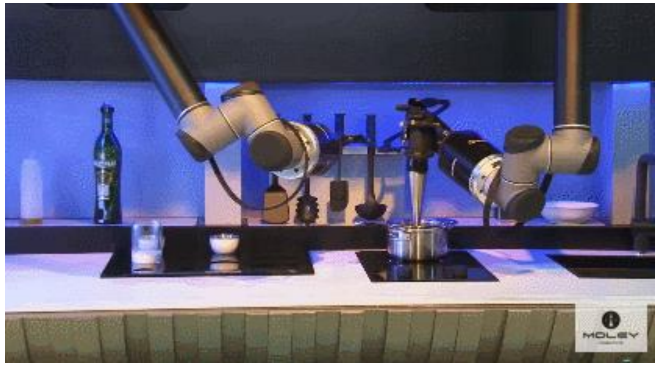
Human-friendly and collaborative

Imitation Learning

## • Background & Motivation

- **The problem we meet:**

Long-term and complex skills

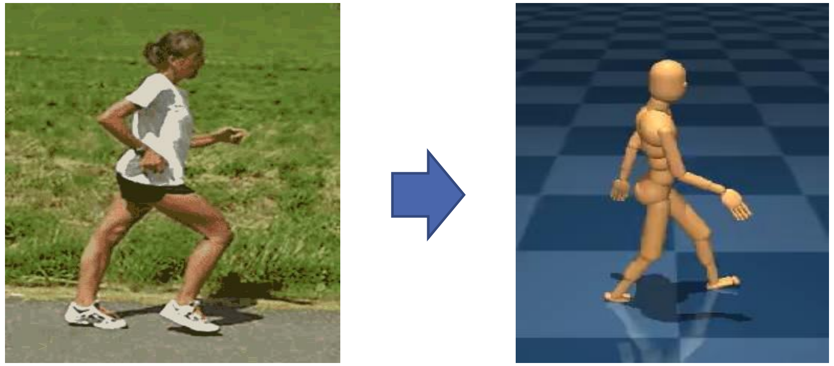


Reinforcement Learning

Reward signal is sparse, exploration is hard

!

Human-friendly and collaborative

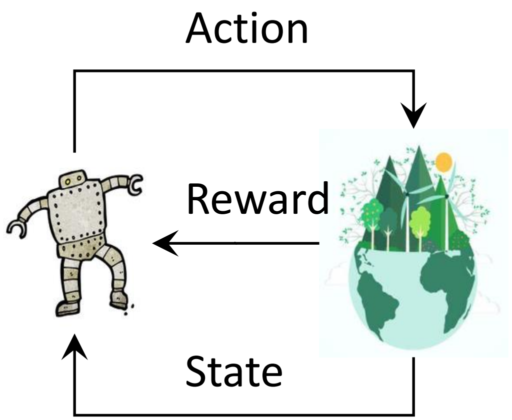


Imitation Learning

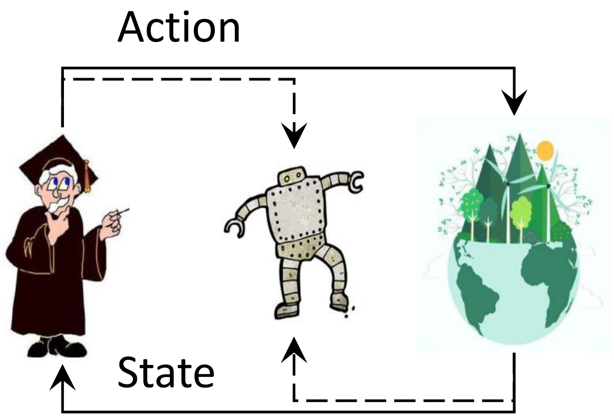
Demonstrations are noisy, sometimes imperfect

# Background & Motivation

- **Reinforcement Learning with Demonstration: Combining of the RL and IL**




Why not make a combination?



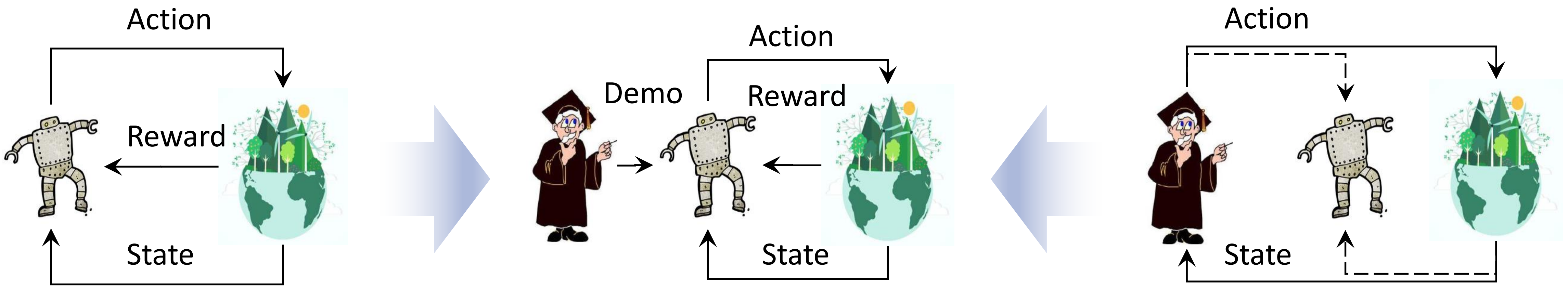

**Reinforcement learning**
When reward signal is sparse, it will be:
- hard to find effective policy with heuristic exploration strategies
- hard to discriminate and optimize policies with similar reward sums.

**Imitation learning**
Without extera reward guidance, it will:
- require high quality demonstrations.
- not be able to tall the different among expert demonstrations.


7

# • Background & Motivation

- **Reinforcement Learning with Demonstration: Combining of the RL and IL**




**Reinforcement learning**

When reward signal is sparse, it will be:

- hard to find effective policy with heuristic exploration strategies
- hard to discriminate and optimize policies with similar reward sums.

**RL with Demonstration**

Use demonstrations and sparse reward simultaneously for better policy learning.

**Imitation learning**

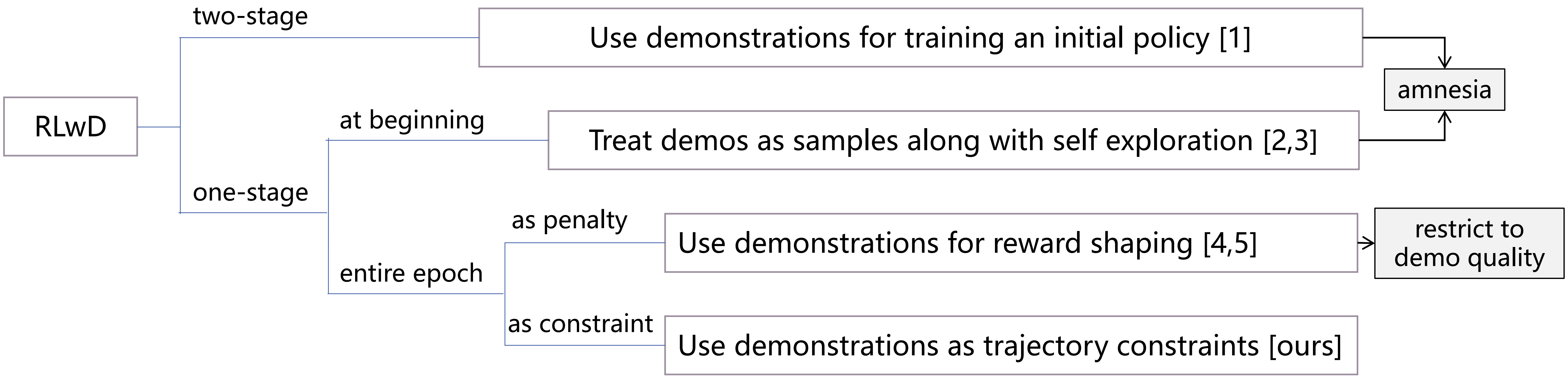
Without extera reward guidance, it will:

- require high quality demonstrations.
- not be able to tall the different among expert demonstrations.


8

# • Background & Motivation

- **Reinforcement Learning with Demonstration: Type of methods**




[1] Silver D, Huang A, Maddison C J, et al. **Mastering the game of Go with deep neural networks and tree search**. nature, 2016
[2] Hester T, Vecerik M, Pietquin O, et al. **Deep q-learning from demonstrations**. AAAI, 2018.
[3] Večerík M, Hester T, Scholz J, et al. **Leveraging demonstrations for deep reinforcement learning on robotics problems with sparse rewards**. arXiv, 2017.
[4] Brys T, Harutyunyan A, Suay H B, et al. **Reinforcement learning from demonstration through shaping**. IJCAI, 2015.
[5] Kang B, Jie Z, Feng J. **Policy optimization with demonstrations**. ICML. 2018: 2474-2483.

# Methodology & Implementation

- **Notations:**
  - State $s \in R^n$
  - Action $a \in R^m$
  - Policy $\pi(s) \rightarrow a$ or $\pi(a|s) \rightarrow p(a|s)$
  - Transition $Tr(s,a) \rightarrow p(s'|s,a)$
  - Reward $r(s,a) \in R$


- **Final optimization goal:**

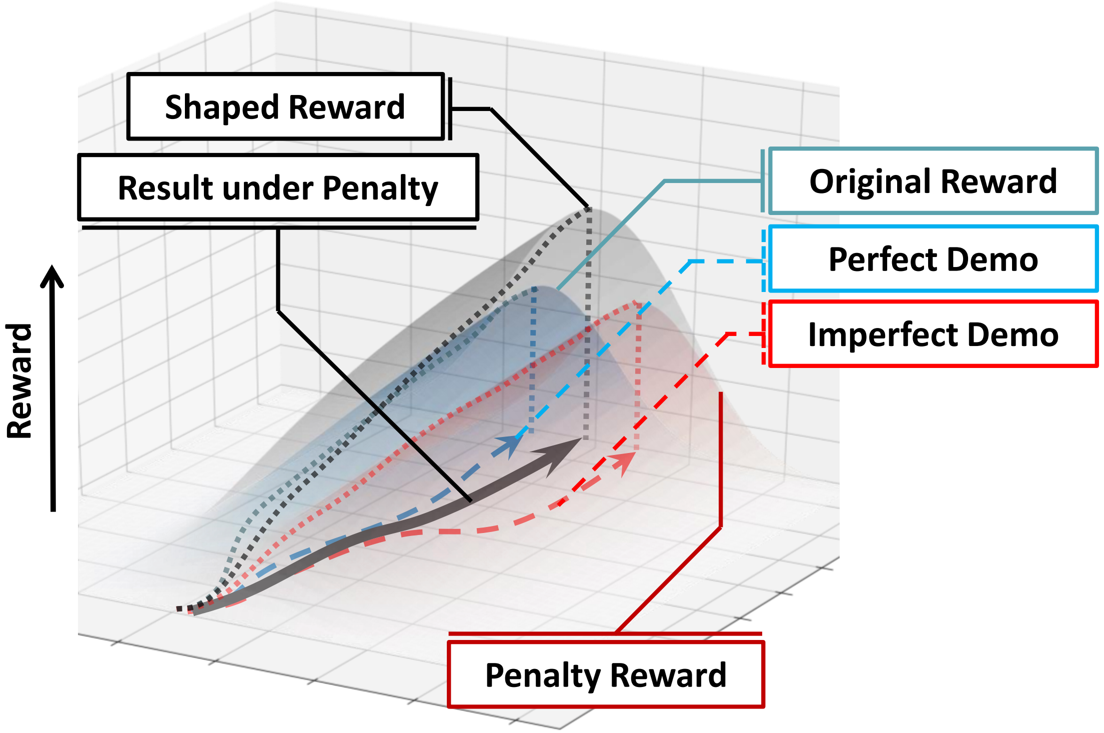
$$\eta(\pi) = E_{s_0 \sim \rho_0(s_0), a_t \sim \pi(a_t|s_t), s_{t+1} \sim P(s_{t+1}|s_t, a_t)}[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)]$$

- **Imperfectness:**

$$\pi_\theta^+ \in \left\{\pi: \arg\max_{\pi} \eta(\pi_\theta) \text{ AND } \frac{\partial \eta(\pi_\theta)}{\partial \theta} = 0\right\}$$
$$\pi_\theta^- \in \left\{\pi: \left\{\eta(\pi_\theta) < \eta(\pi_\theta^+) \text{ AND } \frac{\partial \eta(\pi_\theta)}{\partial \theta} = 0\right\} \text{ OR } \left\{\frac{\partial \eta(\pi_\theta)}{\partial \theta} \neq 0\right\}\right\}$$

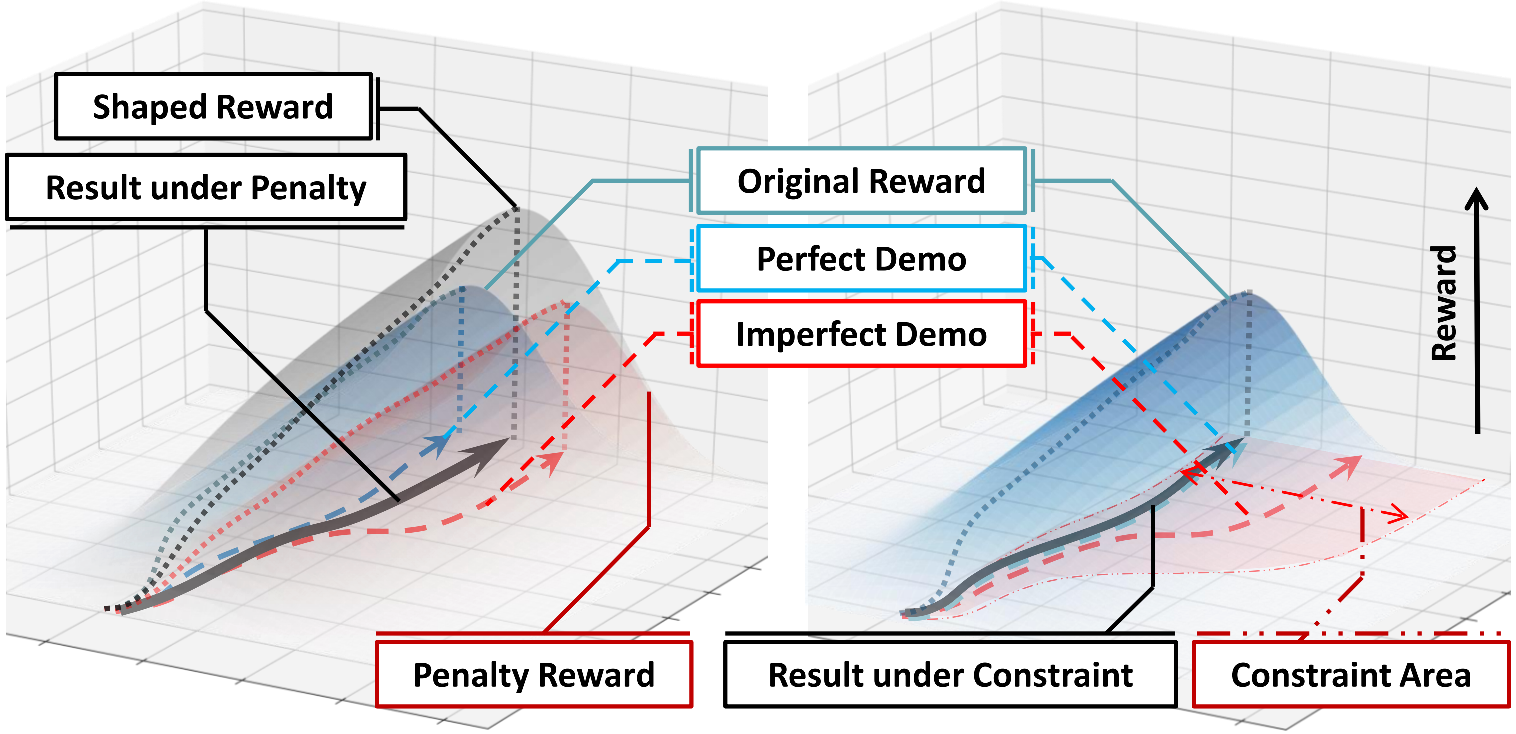
# • Background & Motivation

- **Reinforcement Learning with Demonstration: Impact of imperfectness**



# Background & Motivation

- **Reinforcement Learning with Demonstration: Impact of imperfectness**

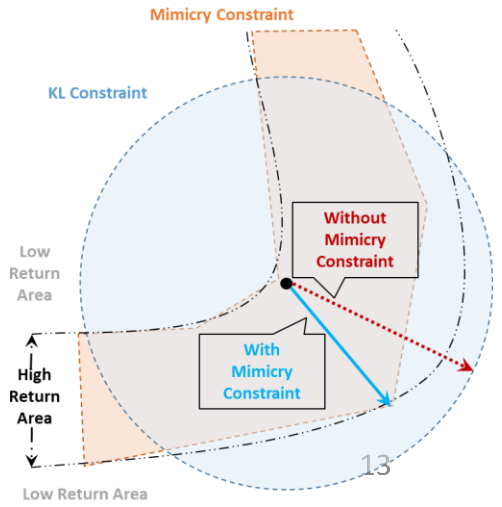


# • Background & Motivation

- **Reinforcement Learning from Imperfect Demonstrations under Soft Expert Guidance:**
  - **Intuition:**

    Combining the benefit of IL and RL by one constrained policy optimization, which only obeys RL update when constraints are invalid, and obey IL update when meet the constraint boundary.

  - **Highlights:**
    - Without expert demo, our method have the same performance as basic RL method
    - With perfect demo, our training efficiency is similar as IL
    - Tolerant to imperfect demos and keep the optimality of the learned policy



# • Methodology & Implementation

- **Modeling & Optimization goal:**
  - The exploring region refer to demo is modeled as a state-wides distribution distance between current policy and demo, and the overall optimization goal can be written as:

$$\theta_{k+1} = \arg\max_{\theta} \quad \eta(\pi_{\theta_k})$$
$$\text{s.t.} \quad \mathbb{D}\left[\rho_{\pi_{\theta_k}}(s,a) \| \rho_{\pi_{\theta^-}}(s,a)\right] \leqslant d_k$$
$$\mathbb{D}_{\mathrm{KL}}\left[\pi_{\theta_k}(a|s) \| \pi_{\theta_{k+1}}(a|s)\right] \leqslant \delta,$$
$$d_{k+1} \leftarrow d_k + d_k \cdot \epsilon,$$

- **Discrepancy choosing:**
  - Use sample-based Maximum Mean Discrepancy(MMD) measurement:

$$\mathrm{MMD}^2(D^\pi, D^E) = \frac{1}{m(m-1)} \sum_{i\neq j}^{m} k\left((s_i^\pi, a_i^\pi), (s_j^\pi, a_j^\pi)\right) + \frac{1}{n(n-1)} \sum_{i\neq j}^{n} k\left((s_i^E, a_i^E), (s_j^E, a_j^E)\right) - \frac{2}{mn} \sum_{i,j=1}^{m} k\left((s_i^E, a_i^E), (s_j^\pi, a_j^\pi)\right)$$

  - Use Mean Square Error (Behavior Cloning-like):

$$L_2(\pi) = \sum_{(s,a)\in D^E} \|a - \pi(s)\|_2^2$$

# • Methodology & Implementation

- **Solving the constraint problem using duality**
  - Linearizing the original optimization goal around current parameter $\theta_k$:

$$\begin{aligned}\theta_{k+1} = \arg\max_{\theta} \ & \eta(\pi_{\theta_k}) \\ \text{s.t.} \ & \mathbb{D}\left[\rho_{\pi_{\theta_k}}(s,a) \| \rho_{\pi_{\theta-}}(s,a)\right] \leqslant d_k \\ & \mathbb{D}_{\mathrm{KL}}\left[\pi_{\theta_k}(a|s) \| \pi_{\theta_{k+1}}(a|s)\right] \leqslant \delta\end{aligned} \Rightarrow \begin{aligned}\theta_{k+1} = \arg\max_{\theta} \ & g^T(\theta-\theta_k) \\ \text{s.t.} \ & b^T(\theta-\theta_k) + d_{\theta_k} \leqslant d_k \\ & \frac{1}{2}(\theta-\theta_k)^T H(\theta-\theta_k) \leqslant \delta\end{aligned}$$

  - Solving the dual form of the linearized optimization goal:

$$\begin{aligned}\theta_{k+1} = \arg\max_{\theta} \ & g^T(\theta-\theta_k) \\ \text{s.t.} \ & b^T(\theta-\theta_k) + d_{\theta_k} \leqslant d_k \\ & \frac{1}{2}(\theta-\theta_k)^T H(\theta-\theta_k) \leqslant \delta\end{aligned} \Rightarrow \begin{aligned}& \max_{\substack{\lambda \geq 0 \\ \nu \succeq 0}} \frac{-1}{2\lambda}\left(g^T H^{-1} g - 2r^T\nu + \nu^T S \nu\right) + \nu^T c - \frac{\lambda\delta}{2}, \\ & \text{where } r \doteq g^T H^{-1} B,\ S \doteq B^T H^{-1} B \\ & \theta^* = \theta_k + \frac{1}{\lambda^*} H^{-1}(g - B\nu^*)\end{aligned}$$

# Methodology & Implementation

- **Overall algorithm block:**

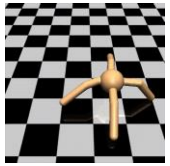
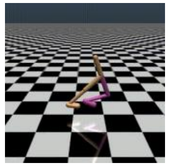
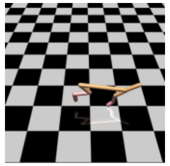
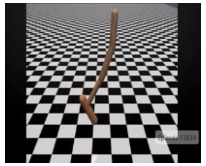
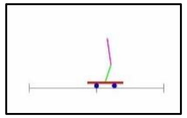
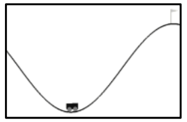
**Algorithm 1** RLfD with a Soft Constraint

**Input:** Imperfect expert demonstrations $\mathcal{D}_E = \{\zeta_i^E\}$, initial policy $\pi_{\theta_0}$, initial constraints tolerance $d_0$, $\delta$, annealing factor $\epsilon$, maximal iterations $N$.
**for** k = 0 to $N$ **do**
  Sample roll-out $\mathcal{D}_\pi$ with $\pi_{\theta_k}$.
  Estimate $\hat{g}$, $\hat{b}$, $\hat{H}$ with samples from $\mathcal{D}_E$ and $\mathcal{D}_\pi$.
  **if** the optimization problem (5) is feasible **then**
    Solve the dual problem (6) to obtain $\lambda^*$, $\nu^*$.
    Compute update step proposal $\Delta\theta$ as (7).
    Update the policy by backtracking line-search along $\Delta\theta$ to ensure the satisfaction of constraints.
  **else**
    Update the policy via the recovery objective (9).
  **end if**
  Annealing the tolerance $d_k$: $d_{k+1} \leftarrow d_k + d_k \cdot \epsilon$.
**end for**

# • Experiment & Result

- **Two goals to be verified:**
  1. Under the same imperfect expert settings, can our method attains better performative results versus the counterparts that do not employ demonstrations as a soft constraint?
  2. How can the different settings of imperfect expert data, i.e. quality and amount, affect the performances of our method and baselines?

- **Sparse reward settings:**
  - TYPE 1: reward = 1 when the agent reaches the terminal state, otherwisely 0.
    - MountainCar
  - TYPE 2: reward = 1 for every time the agent moves forward over a specific number of units.
    - Hopper, HalfCheetah, Walker2d, Ant
  - TYPE 3: reward = 1 when the second pole stays above a specific height
    - InvertedDoublePendulum

# • Experiment & Result

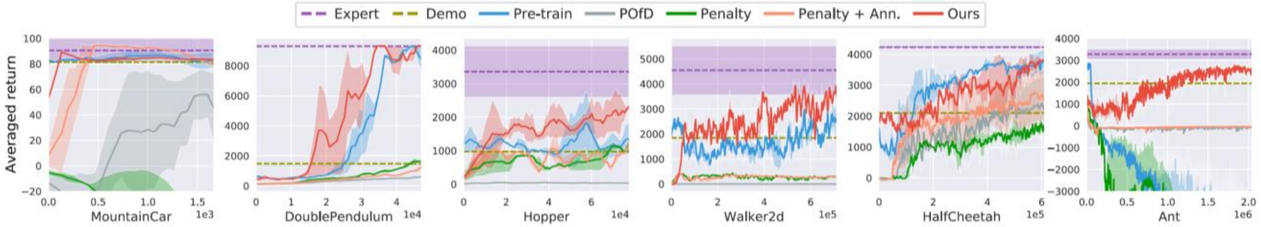
- **Comparison with other methods:**

We evaluate our algorithm with several baselines under imperfect demonstration and sparse reward settings:

- PPO, MMD-IL: pure RL & IL settings, use sparse rewards or imperfect demonstrations separately.
- Pre-training: pre-train the policy supervisely using demonstrations, then do RL using sparse rewards.
- POfD, Penalty, Penalty+Ann: use demonstrations as discrepancy penalty or reward shapping.

The results show that using demonstrations as constraints leads to better performence comparing with other baselines.

| | MountainCar | DoublePendulum | Hopper | Walker2d | HalfCheetah | Ant |
|---|---|---|---|---|---|---|
| $\mathcal{S}$ / $\mathcal{A}$ | $\mathbb{R}^4$ / $\{0,1\}$ | $\mathbb{R}^{11}$ / $\mathbb{R}^1$ | $\mathbb{R}^{11}$ / $\mathbb{R}^3$ | $\mathbb{R}^{17}$ / $\mathbb{R}^6$ | $\mathbb{R}^{17}$ / $\mathbb{R}^6$ | $\mathbb{R}^{111}$ / $\mathbb{R}^8$ |
| Setting / Demo | **S1** / 81.25 | **S3** / 1488.28 | **S2** / 969.71 | **S2** / 1843.75 | **S2** / 2109.80 | **S2** / 1942.05 |
| PPO | -0.74±9.61 | 302.77±37.09 | 17.09±13.54 | 1.54±5.75 | 978.84±665.61 | -2332.95±2193.85 |
| MMD-IL | 82.99±4.57 | 218.43±13.72 | 118.66±0.38 | 8.88±6.07 | 161.74±219.85 | 967.83±0.87 |
| Pre-train | 83.35±6.32 | 8928.79±388.61 | 1356.47±470.43 | 2607.38±301.94 | 3831.96±150.30 | -5377.25±1682.56 |
| POfD | 45.01±28.16 | 628.47±69.36 | 32.13±24.23 | -1.48±0.03 | 2801.59±66.03 | -68.59±19.17 |
| Penalty | -120.29±48.30 | 1902.95±210.41 | 1225.03±296.52 | 286.23±12.46 | 1517.68±35.85 | -3711.12±794.97 |
| Penalty + Ann. | 79.00±1.04 | 1671.78±108.80 | 1220.10±112.74 | 282.00±6.70 | 2592.94±870.04 | -116.89±88.01 |
| Ours | **83.46±1.42** | **9331.40±5.95** | **2329.89±125.85** | **3483.78±269.59** | **4106.69±95.47** | **2645.58±118.55** |



# • Experiment & Result

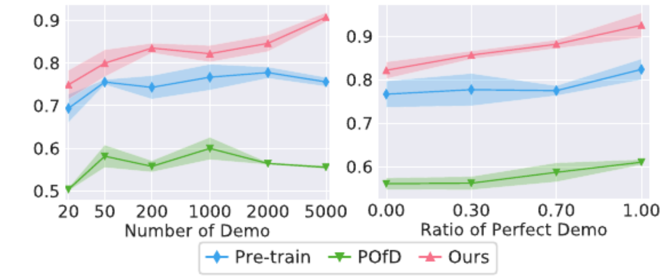



Figure 3: Results on *HalfCheetah* task with different imperfect expert setting. **Left**: Different number of state-action pairs; **Right**: Different level of imperfectness.

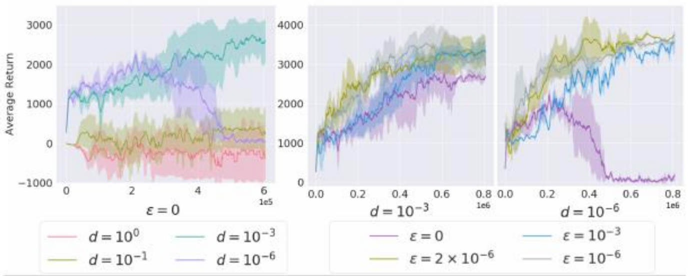



Figure 4: Learning curves over on *HalfCheetah* task. **Left**: ablation study about different tolerance factor $d$; **Right**: sensitivity of choosing fixed or annealing strategy of tolerance.

- **Ablations analysis:**

We evaluated the performance of our method on different settings:

- **Different imperfect settings:** we compare the learning performance of pre-training, POfD and our method under different amount of state-action pairs & demonstrations with different imperfectness.
- **Different tolerances:** we compare the learning performance of our method under different tolerance factors & different annealing factors.

The results show that our method can be more robust and efficient when dealing with fewer imperect demonstrations, and can tolerant to the minor changes of annelling factor $\varepsilon$ under proper tolerance factor $d$.

# Thanks!

MingXuan Jing, XiaoJian Ma, WenBing Huang, FuChun Sun, Chao Yang, Bin Fang, HuaPing Liu,

**Reinforcement Learning from Imperfect Demonstrations under Soft Expert Guidance**, The AAAI Conference on Artificial Intelligence *(AAAI)*, 2020.