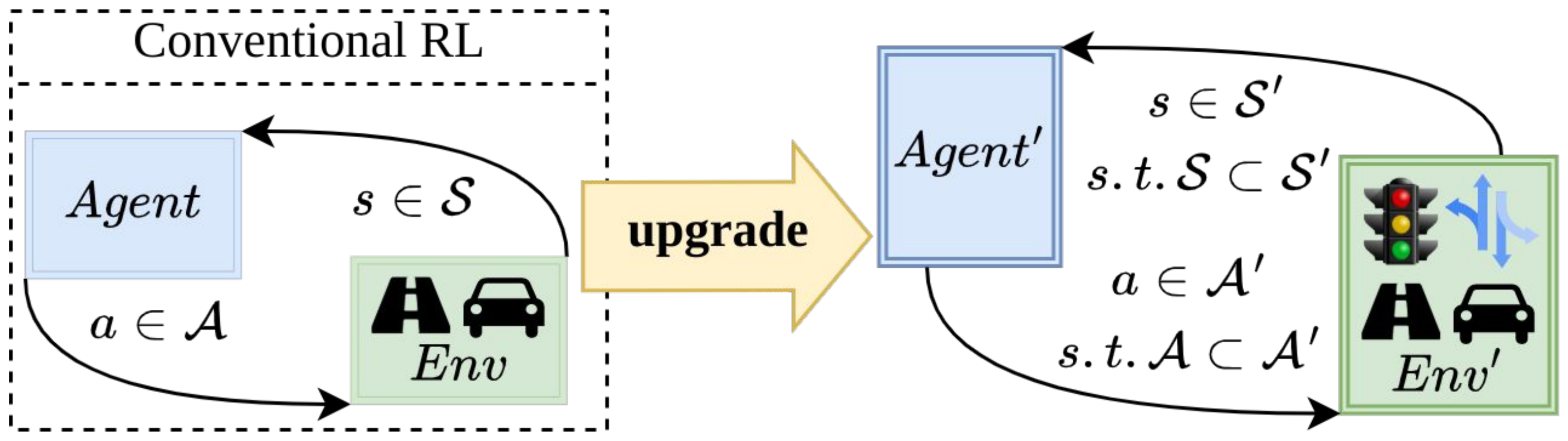
Incremental Reinforcement Learning with Dual-Adaptive ε-greedy Exploration

Wei Ding*, Siyang Jiang*, Hsi-Wen Chen*, Ming-Syan Chen
Graduate Institute of Electrical Engineering, National Taiwan University, Taiwan
{wding, syjiang, hwchen}@arbor.ee.ntu.edu.tw, mschen@ntu.edu.tw

1. Introduction



- Most reinforcement learning frameworks oversimplify the problem by assuming a fixed-yet-known environment and often have difficulty being generalized to real-world scenarios.
- We address a new challenge with a more realistic setting, **Incremental Reinforcement Learning**, where the search space of the Markov Decision Process continually expands.
- While previous methods usually suffer from the lack of efficiency in exploring the unseen transitions, especially with increasing search space, we present a new exploration framework named **Dual-Adaptive ε-greedy Exploration (DAE)** to address the challenge of Incremental RL.
- Specifically, DAE employs a **Meta Policy** and an **Explorer** to avoid redundant computation on those sufficiently learned samples.
- Furthermore, we release a **new testbed** based on a synthetic environment and the Atari benchmark to validate the effectiveness of any exploration algorithms under Incremental RL.
- Experimental results demonstrate that the proposed framework can efficiently learn the unseen transitions in new environments, leading to notable performance improvement, i.e., an average of more than **80%**.

2. Problem Formulation

Markov Decision Process & Q-learning

- tuple M = (S, A, T, R)
- S: state space
- A: action space
- T : S×A → P(S), transition function
- R : S × A → r, predefined reward function
- $\mathcal{V}_\pi(s) = \max_{a\sim\mathcal{A}} \mathcal{Q}_\pi(s,a) = \mathbb{E}_\pi[\sum_{t=0}^{\infty} \gamma^t r_t | s_0 = s]$ (1)
- $\mathcal{Q}_\pi(s_t,a_t) = \mathcal{R}(s_t,a_t) + \gamma \max_{a_{t+1}\sim\mathcal{A}} \mathcal{Q}_\pi(s_{t+1},a_{t+1})$ (2)

Incremental Reinforcement Learning

- $\mathcal{M}' = (\mathcal{S}', \mathcal{A}', \mathcal{T}', \mathcal{R}')$
- $\mathcal{S} \subset \mathcal{S}', \mathcal{A} \subset \mathcal{A}', \mathcal{T} \subset \mathcal{T}', \mathcal{R} \subset \mathcal{R}'$
- Finetune the previous policy for $\mathcal{M}'$ based on and against default trajectory
- Hard exploration problem (could be seen as initialization bias)

3. Dual-Adaptive ε-greedy Exploration

1. **Meta Policy Ψ**
   Adaptively make a trade-off between exploitation and exploration:
   $$\varepsilon_t = \Psi(s_t), s.t.\, 0 \le \Psi(s_t) \le 1, \forall s_t \in \mathcal{S} \;(3)$$
   The meta policy ψ is a deep learning model with one output neuron and sigmoid function.
   This behavior is fashioned into a binary classification problem with pseudo label y defined as:
   $$y = \begin{cases} 1, if\ TD - Error\ rate > \tau \\ 0, otherwise \end{cases}$$

2. **Explorer Φ**
   Adaptively **select least-tried action to explore**:
   $$\Phi(a|s_t) \sim RF(a|s_t), s.t., \sum \Phi(a|s_t) = 1, \Phi(a|s_t) \ge 0, a \in \mathcal{A}$$
   , where we refer to the underlying occurrence of each action as RF (relative frequency).
   The explorer is a deep model with softmax activation function.
   RF of taken action is raised by gradient ascend with loss function defined as the log probability of that action.

4. Expanding World

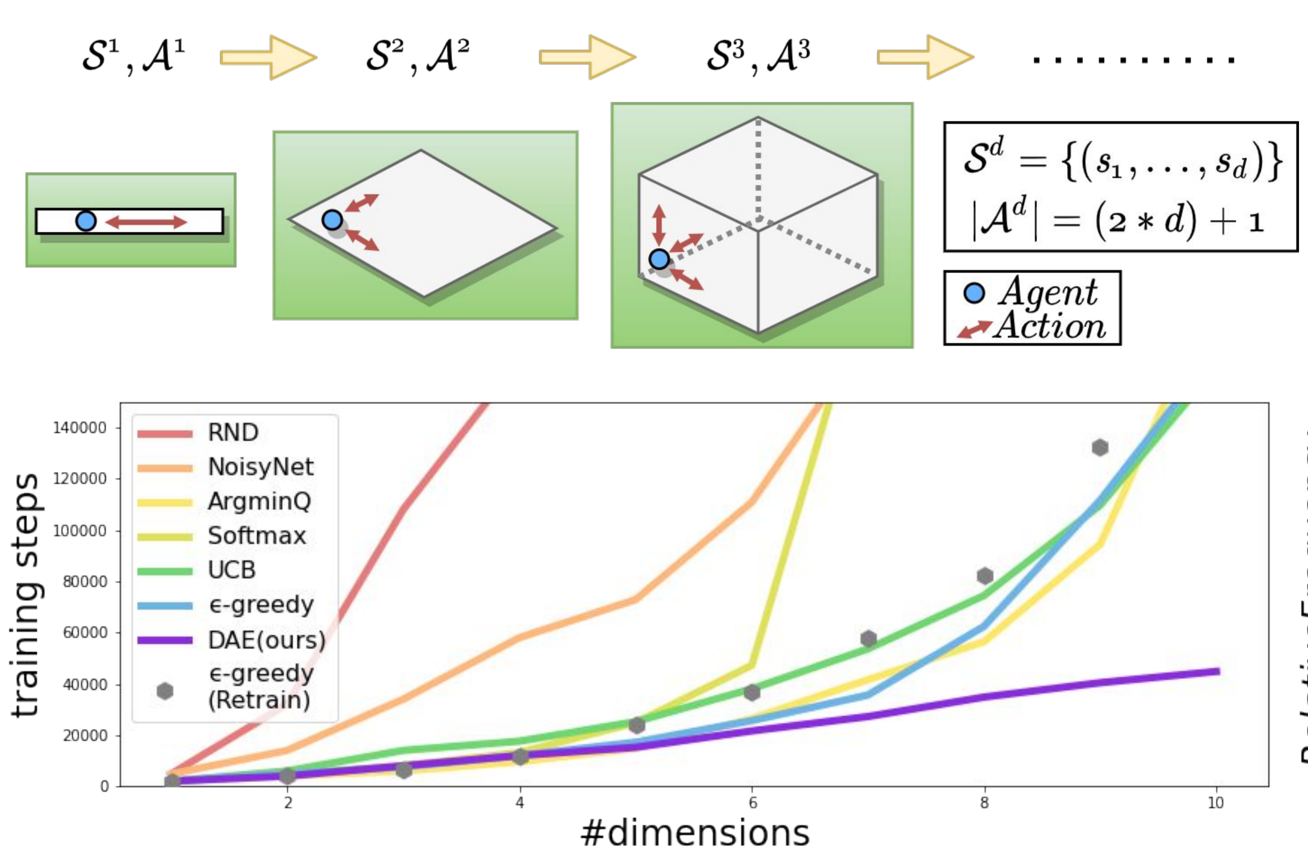
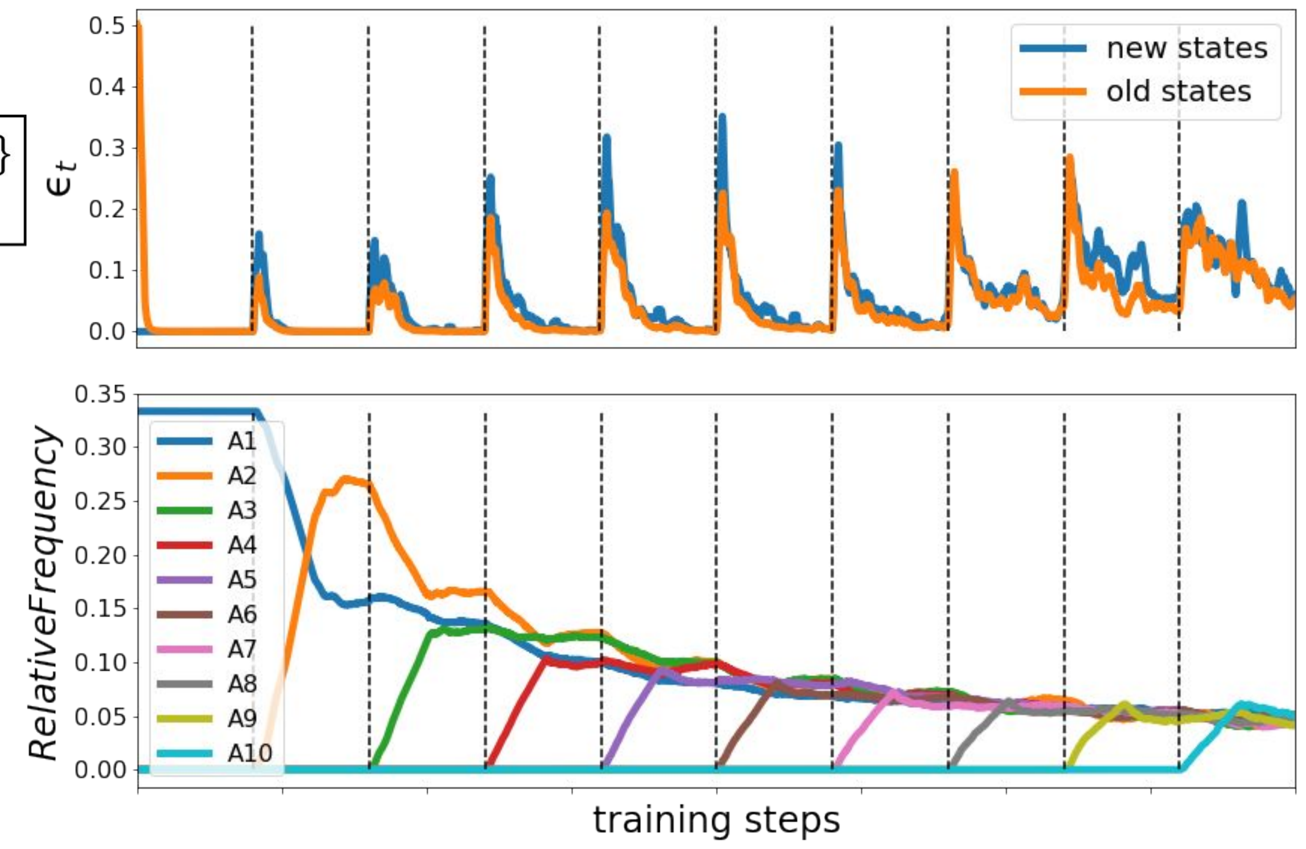


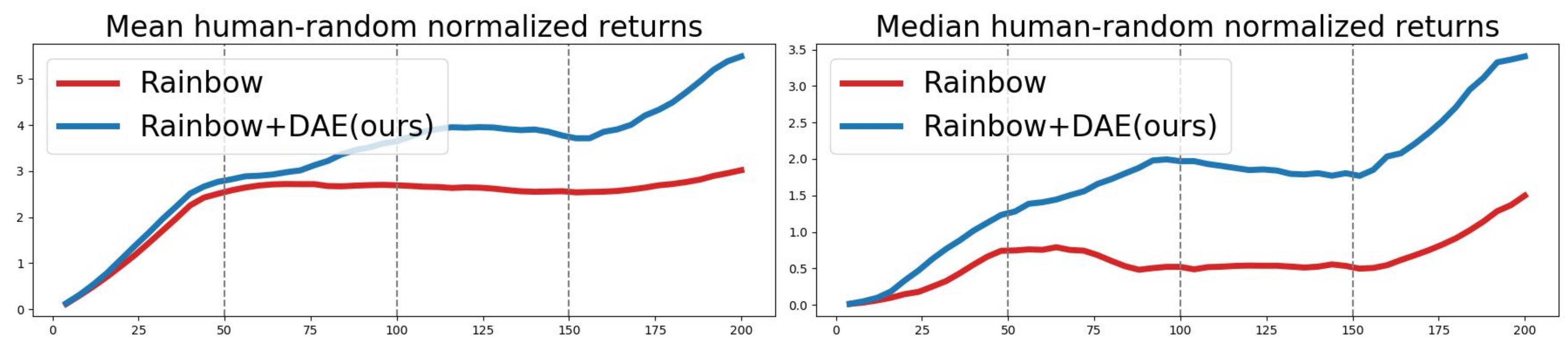
Illustration of Expanding World and the training overhead.



The change of ε_t and the relative frequency.

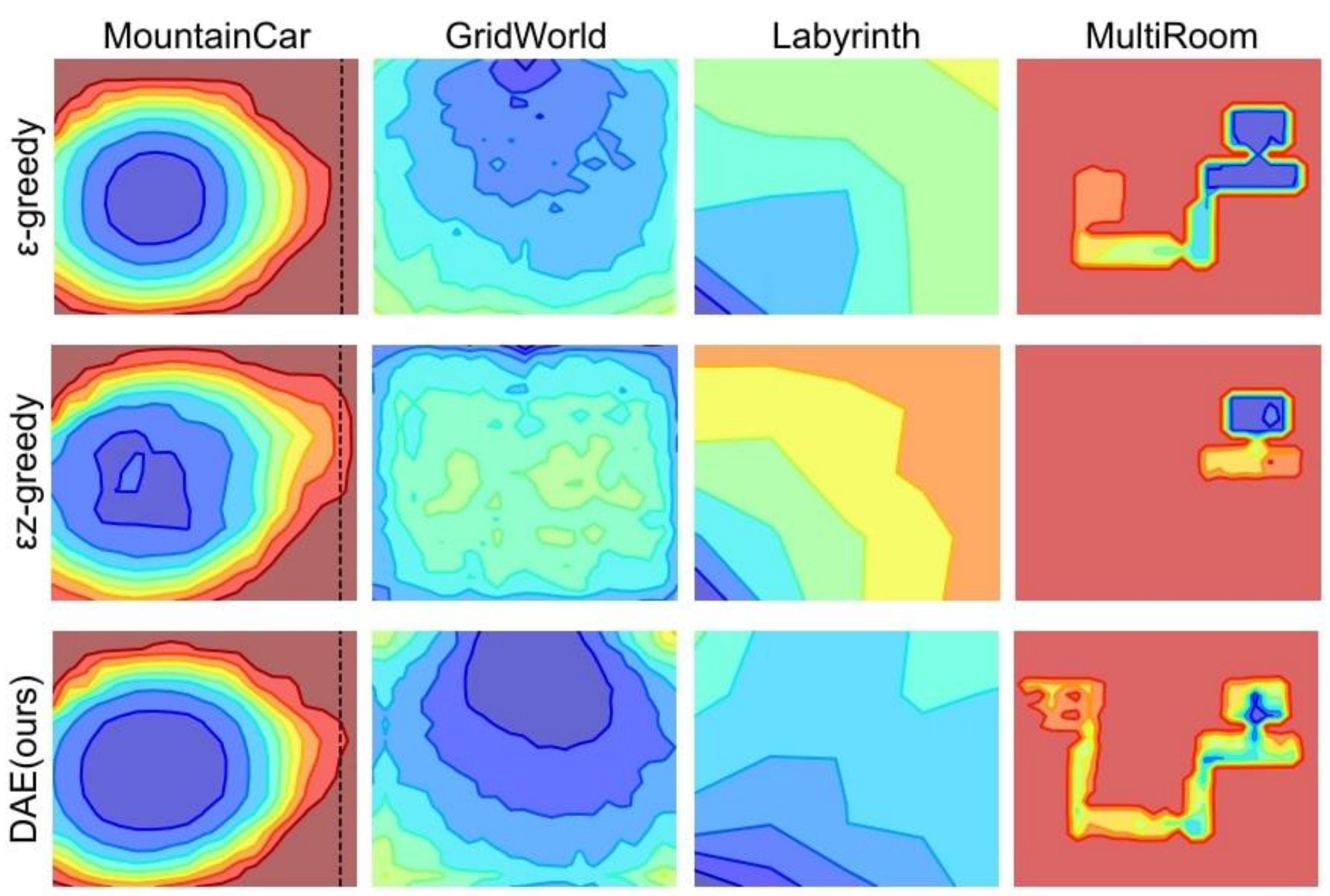
5. Incremental Atari

	Method	Mean		Median	
		best	final	best	final
RL	Rainbow	5.57	5.02	3.42	2.46
Incremental RL	Rainbow	3.23	3.23	2.11	2.11
	DAE	**6.11**	**6.11**	**3.97**	**3.97**



- Arcade Learning Environment
- We carefully select 14 games with different levels of difficulty, each of which has 18 meaningful actions.
- Only six primitive actions are initially available to enable the agent to play the games.
- The rest 12 advanced actions are randomly divided into three groups and added into the environment sequentially.
- We report the mean and median episodic reward.

6. First-Visit Visualization



- We further evaluate the exploration efficiency of DAE for general RL via conducting the First-Visit Visualization.
- These tasks show the state coverage of an exploration algorithm and how quickly it can discover all of the states.
- Specifically, the number of steps the agent takes to discover, i.e., first visit, each state are recorded and visualized into heat maps.
- Blue and green areas take fewer steps to be reached, whereas yellow and red areas take more times.