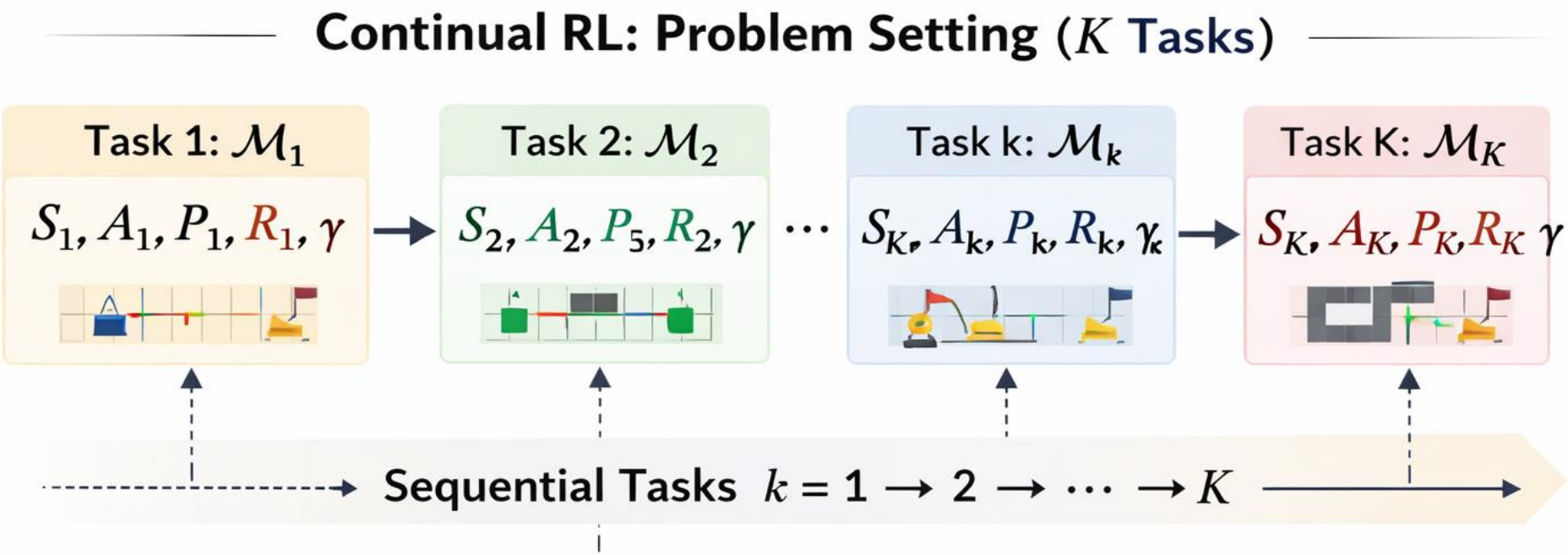
Principled Fast and Meta Knowledge Learners for Continual Reinforcement Learning

Ke Sun*, Hongming Zhang*, Jun Jin, Chao Gao, Xi Chen, Wulong Liu, Linglong Kong

What is Continual RL?

Problem Setting: learning in a sequence of MDP tasks



Key Challenges in Continual (Reinforcement) Learning

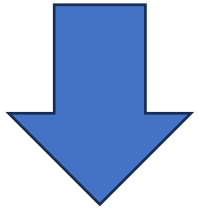
- **Loss of plasticity / Insufficient Forward Transfer:** The agent gradually loses the capability to adapt to the **new environment**.

- **Loss of stability / Catastrophic Forgetting:** when learning multiple sequential tasks, model performance on **previous tasks** significantly **deteriorates** upon training with new data.

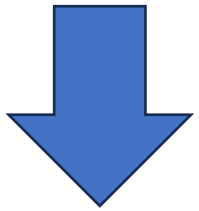
How to trade-off between plasticity and stability?

Main Contributions of Our Work

Foundations of Continual RL



Principled Objectives Function



Efficient and Practical Dual-learner Algorithm

I. Foundations in Continual RL

Foundation 1: measure of task similarity/difference between two tasks (MDP)

- Value-based MDP distance: $d_Q(Q_1^*, Q_2^*)$
- Policy-based MDP distance: $d_\pi(\pi_1^*, \pi_2^*)$

Foundation 2: Definition of Catastrophic Forgetting (weighted distance)

Consider the value function and policy adaption from (k-1)-th task to k-th task

$$CF(Q_{k-1}, Q_k) = \sum_{s,a} \mu_{k-1}^{Q_{k-1}}(s)\pi^{Q_{k-1}}(a|s)d_Q\left(Q_{k-1}(s,a), Q_k(s,a)\right),$$

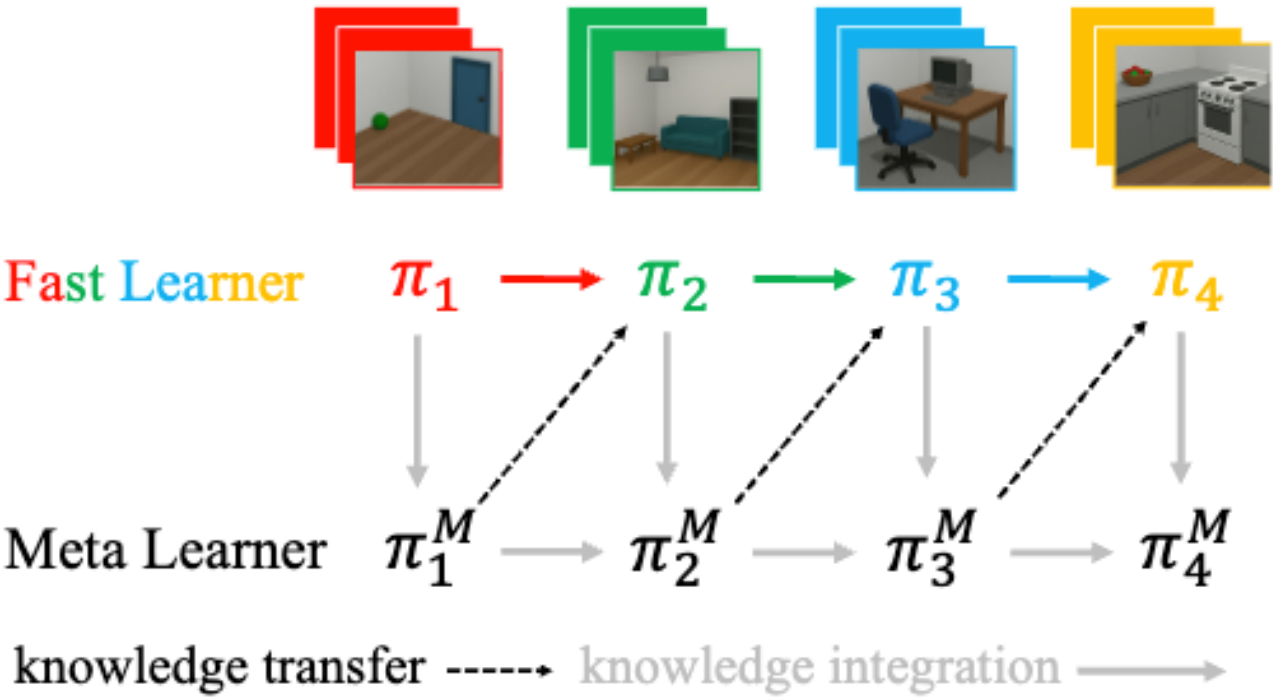
steady state distribution to form the weights

$$CF(\pi_{k-1}, \pi_k) = \sum_{s} \mu_{k-1}^{\pi_{k-1}}(s)d_\pi\left(\pi_k(\cdot|s), \pi_{k-1}(\cdot|s)\right).$$

II. FAME: Principled FAst And MEta Knowledge Learners for Continual RL

Trade-off between **plasticity** (forward transfer) vs **stability** (without forgetting)

hippocampus–cortex analogy



Dual-learner Framework:

- **Fast learner**: knowledge adaptation

- **Meta learner**: knowledge integration

Knowledge Integration: Catastrophic Forgetting Minimization

Value-based Objective Function

$$Q_k^M = \arg\min_{\widetilde{Q}_k^M} \sum_{i=1}^{k-1} \mathbb{E}_{w_i^Q} \left[\log \frac{\pi_{k-1}^M}{\widetilde{\pi}_k^M} \right] + \mathbb{E}_{w_k^Q} \left[\log \frac{\pi^{Q_k}}{\widetilde{\pi}_k^M} \right] = \arg\max_{\widetilde{Q}_k^M} \sum_{i=1}^{k} \mathbb{E}_{w_i^Q} \left[\log \widetilde{\pi}_k^M \right]$$

Policy-based Objective Function

Method 1 (FAME-KL): Policy Distillation under Forward KL Divergence.

$$\pi_k^M = \arg\max_{\widetilde{\pi}_k^M} \sum_{i=1}^{k} \mathbb{E}_{w_i} \left[\log \widetilde{\pi}_k^M \right]$$

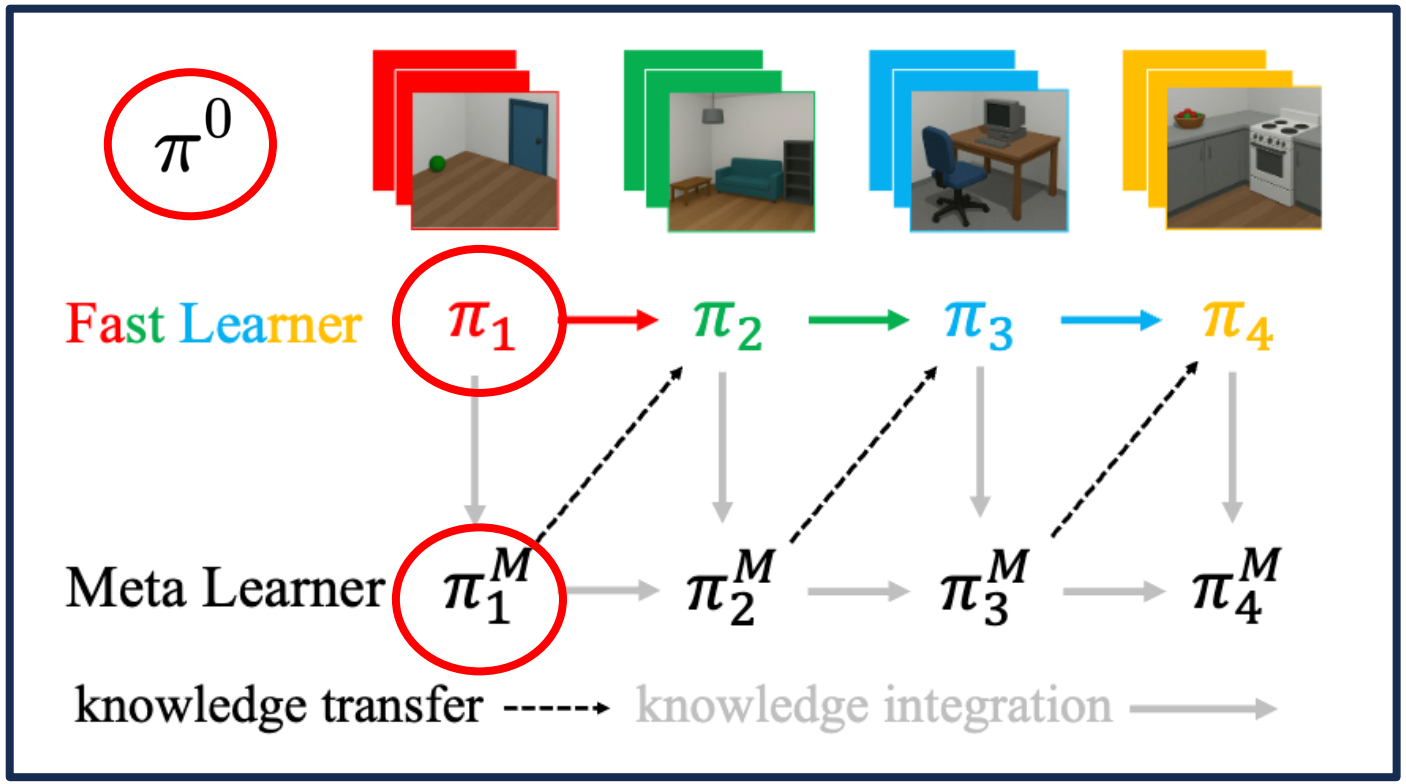
Method 2 (FAME-WD): Wasserstein Distance (WD)-based Knowledge Integration.

$$\pi_k^M = \arg\min_{\widetilde{\pi}_k^M} \left\{ \sum_{i=1}^{k-1} \sum_s \mu_i^{\pi_i}(s) W_2^2 \left(\widetilde{\pi}_k^M(\cdot|s), \pi_{k-1}^M(\cdot|s) \right) + \sum_s \mu_k^{\pi_k}(s) W_2^2 \left(\widetilde{\pi}_k^M(\cdot|s), \pi_k(\cdot|s) \right) \right\}$$

Insight: incremental multi-task learning on a mixture of state-action distributions

Knowledge Transfer: Adaptive Meta Warm-up

- **Fast learner**: adaptation through a selective warm-up among
  - Random Learner/Initialization π^0
  - Previous Fast Learner π_{k-1}
  - Meta learner π_{k-1}^M

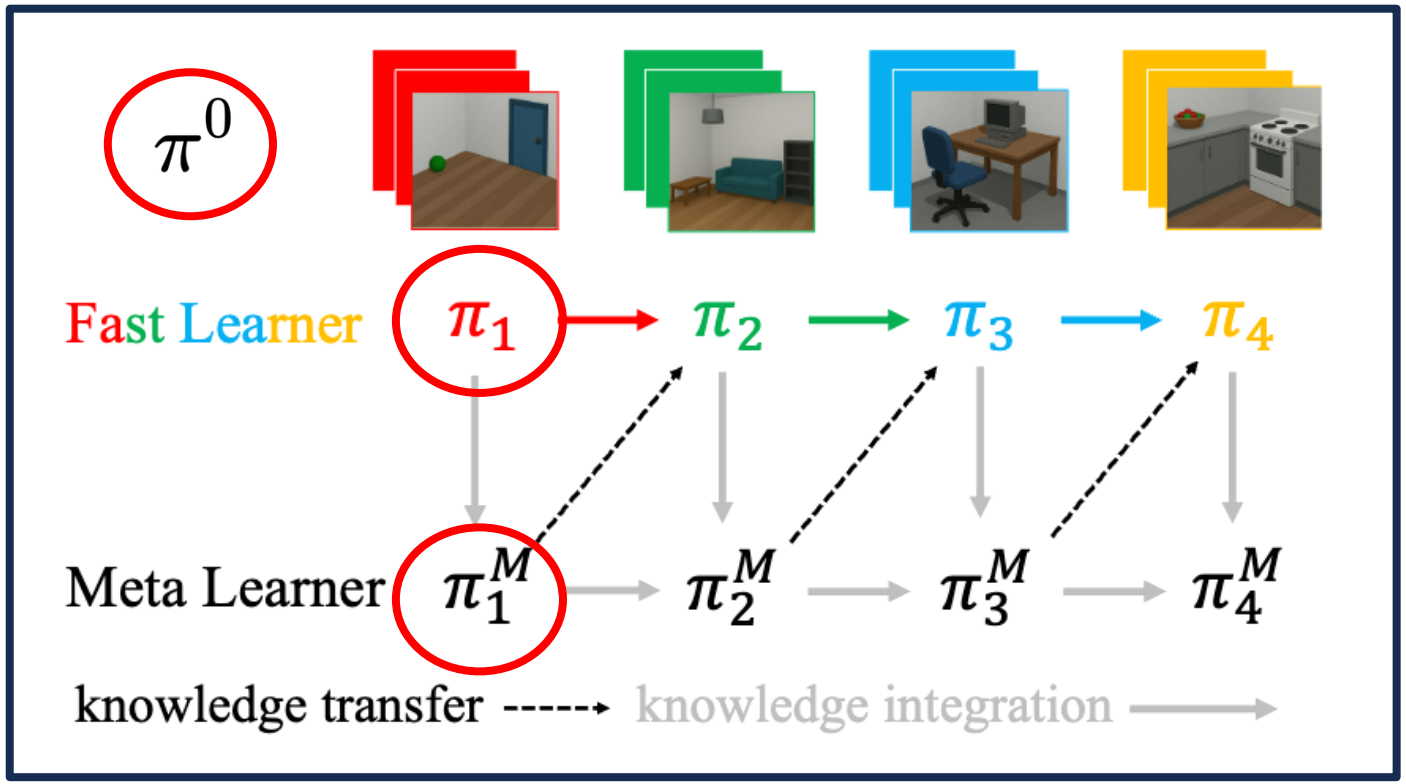


Knowledge Transfer via Adaptive Meta Warm-up

- **Fast learner**: policy evaluation and one-vs-all hypothesis test

$$V_k^f = \mathbb{E}_{\pi_{k-1}}[R], V_k^M = \mathbb{E}_{\pi_{k-1}^M}[R], \text{ and } V_k^r = \mathbb{E}_{\pi^0}[R]$$

$$H_0 : V_k^M \leq \max\left\{V_k^f, V_k^r\right\} \quad \text{vs.} \quad H_1 : V_k^M > \max\left\{V_k^f, V_k^r\right\}.$$



Experiments

Evaluation Metrics

- **Average Performance**: evaluation of the current policy over all past tasks
- **Forward Transfer**: normalized area of the learning curves, similar to AUC
- **Forgetting**: performance difference from the policy at the end of each task

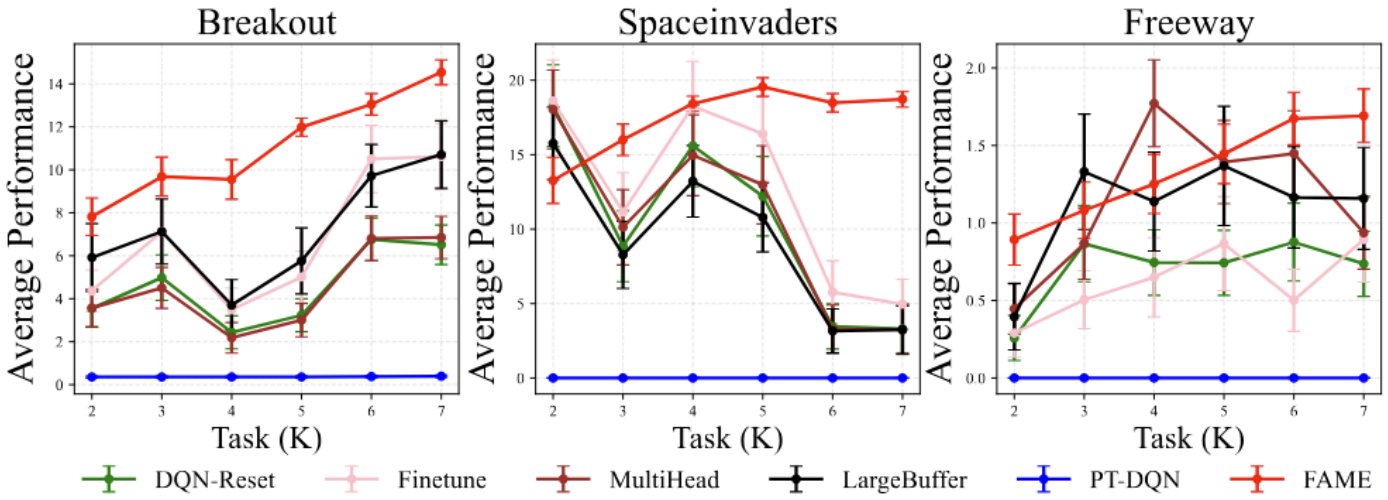
Environments and Algorithms

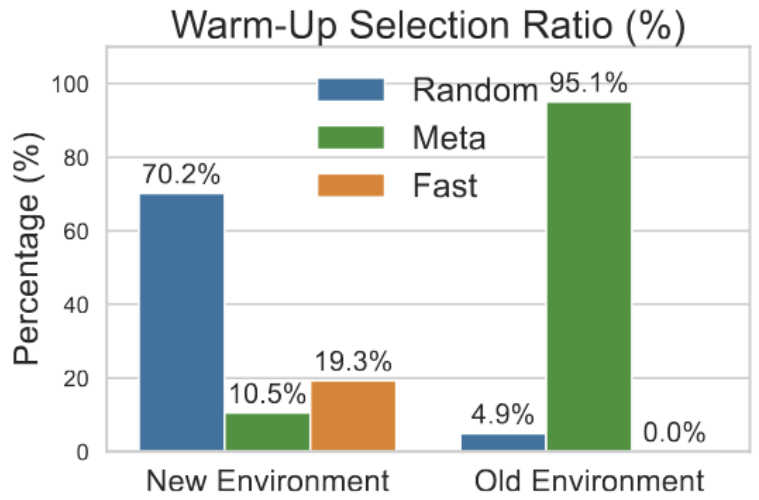
- **MinAtar and Atari Games (DQN, PPO):** pixel-based discrete action tasks
- **MetaWorld (SAC):** robotics arm manipulation tasks

Experiments: MinAtar and Atari Games

Table 1: Main results on **MinAtar** on Average Performance (*Avg. Perf*), Forward Transfer (*FT*), and Forgetting. Results (Mean ± SE) are averaged over 10 sequences, each with 3 seeds. ↑ denotes a positive metric (more is better), while ↓ is a negative one (less is better). `Reset` is the baseline for evaluating FT. Forgetting is normalized by the standard deviation in each task.

Method	**Ave. Perf ↑** Breakout	Spaceinvader	Freeway	**FT ↑**	**Forgetting ↓**
Reset	6.51 ± 1.67	3.29 ± 3.09	0.74 ± 0.38	0.00 ± 0.00	1.31 ± 0.23
Finetune	10.62 ± 2.75	4.95 ± 2.92	0.89 ± 0.49	0.13 ± 0.03	1.26 ± 0.32
MultiHead	6.85 ± 1.76	3.26 ± 2.99	0.94 ± 0.42	-0.01 ± 0.00	1.25 ± 0.22
LargeBuffer	10.71 ± 2.84	3.24 ± 2.91	1.16 ± 0.59	**0.16 ± 0.02**	1.65 ± 0.33
PT-DQN	0.39 ± 0.02	0.00 ± 0.00	0.00 ± 0.00	0.07 ± 0.02	1.64 ± 0.02
FAME	**14.54 ± 0.58**	**18.72 ± 0.52**	**1.69 ± 0.17**	**0.16 ± 0.03**	**0.72 ± 0.13**





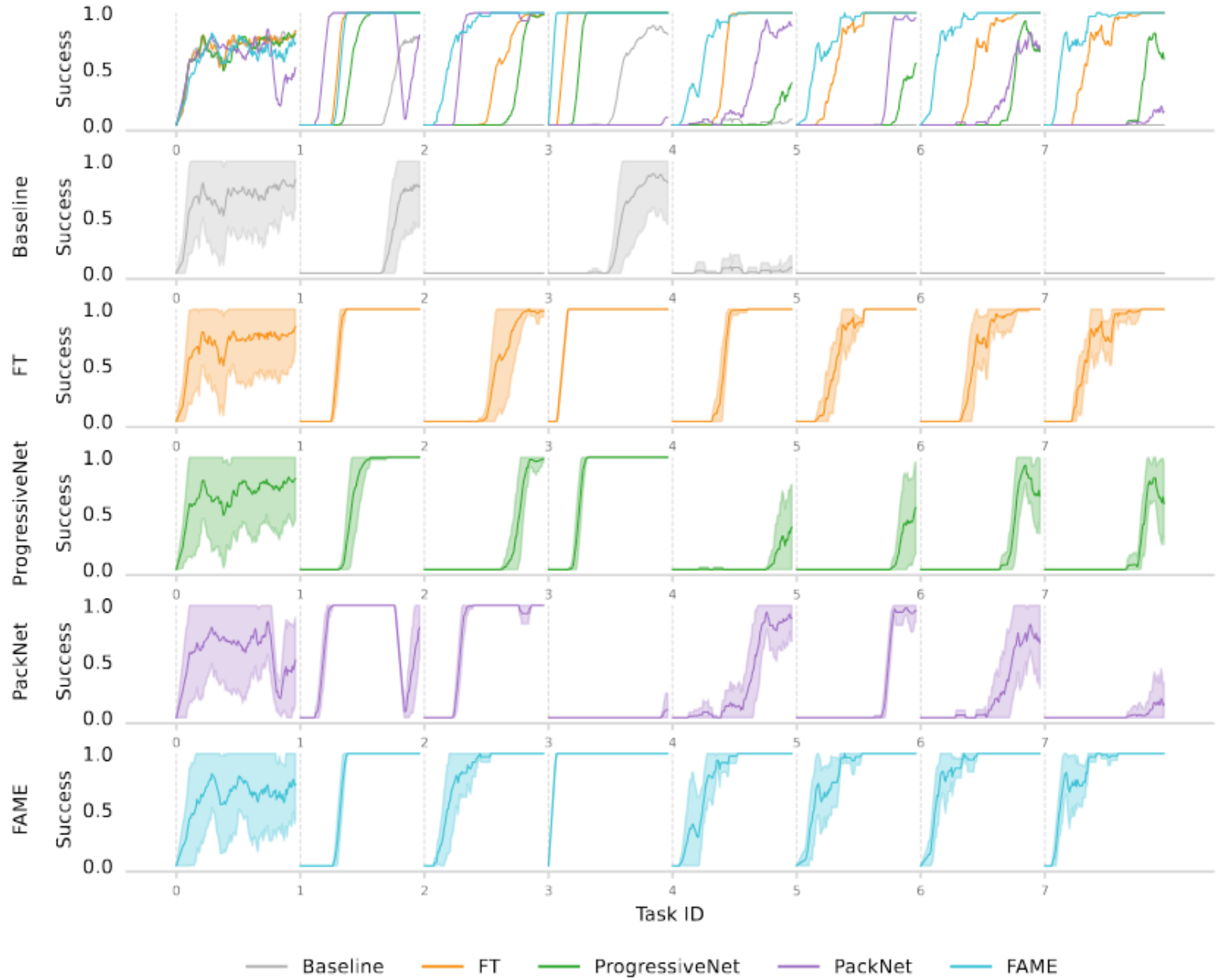


Figure 6: Learning curves of the fast learner in FAME on the Freeway environment averaged over 3 seeds.

Experiments: Meta-world

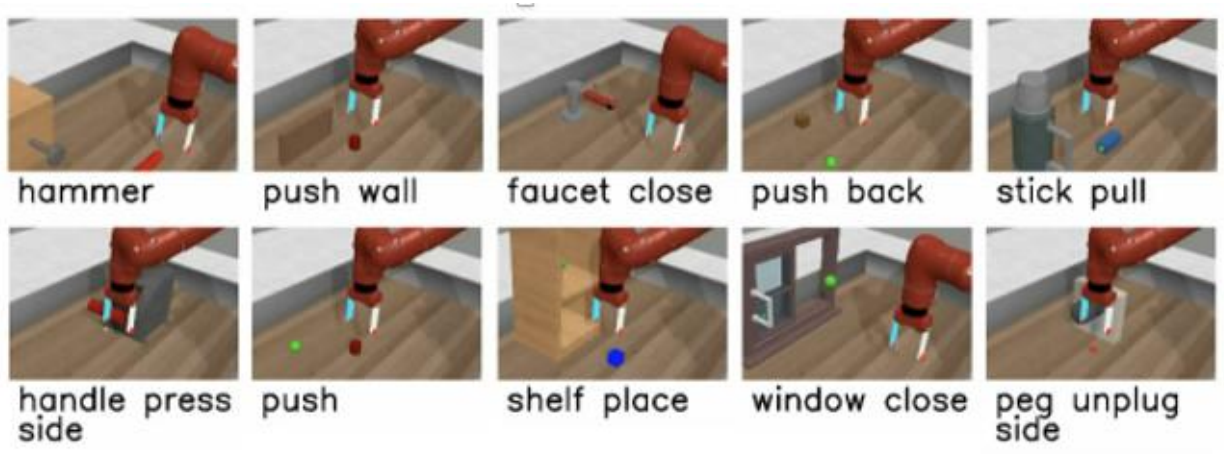
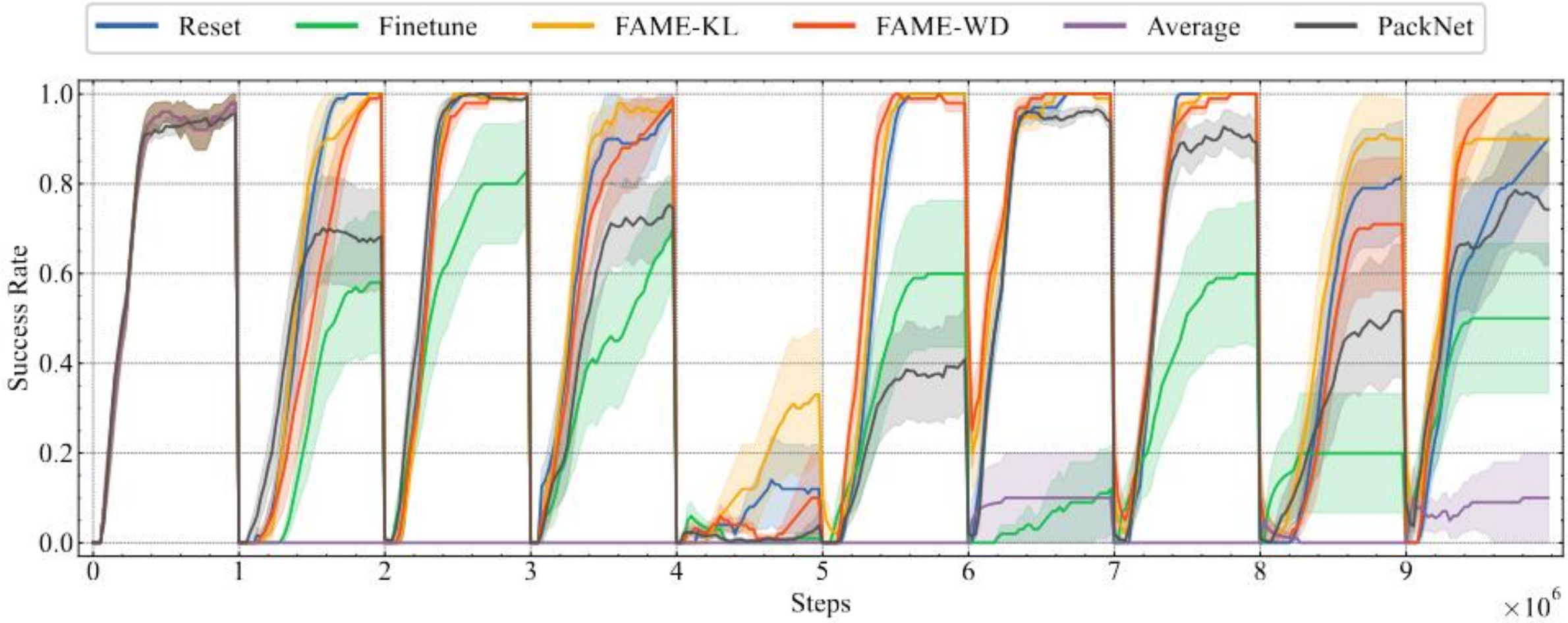


Table 3: Main results on **Meta-World** on Average Performance (*Ave. Perf*), Forward Transfer (*FT*), and Forgetting averaged over 3 sequences. Results are presented as averages and standard errors across 10 seeds.

Methods	**Avg. Perf** ↑	**FT** ↑	**Forgetting** ↓
Reset	0.093 ± 0.017	0.000 ± 0.000	0.710 ± 0.030
Finetune	0.037 ± 0.011	-0.265 ± 0.028	0.427 ± 0.033
Average	0.013 ± 0.007	-0.530 ± 0.024	0.070 ± 0.022
PackNet	0.491 ± 0.025	-0.194 ± 0.018	$\mathbf{0.000 \pm 0.000}$
FAME-KL	0.733 ± 0.026	$\mathbf{0.022 \pm 0.015}$	0.073 ± 0.019
FAME-WD	$\mathbf{0.767 \pm 0.024}$	-0.003 ± 0.014	0.023 ± 0.015


Reset
Finetune
FAME-KL
FAME-WD
Average
PackNet
Success Rate
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
5
6
7
8
9
Steps
$\times 10^6$

Principled Fast and Meta Knowledge Learners for Continual Reinforcement Learning

Thank you!