CRONOS: Benchmarking Multi-Task Robotic Manipulation for Reset-Free Reinforcement Learning At Scale

Po-Yi Wu [*1] **Djengo Cyun-Jyun Fang** [*1] **Dian Cheng** [1] **Tsung-Wei Ke** [1]

Abstract

Reinforcement learning (RL) is promising for adapting robot policies to unstructured real-world environments. However, standard RL pipelines rely on episodic training with frequent scene resets, which is impractical for real-world deployment due to the need for substantial human intervention. We introduce CRONOS, a simulation benchmark for studying reset-free multi-task RL under long-horizon interactions and constrained reset budgets. To reflect realistic deployment, CRONOS leverages high-fidelity physics simulators, adopts shared-scene multi-task settings, targets the adaptation of state-of-the-art robot policies, and formalizes reset-free learning under a fixed reset budget. CRONOS highlights five research topics: (1) learning effectiveness under varying reset budgets, (2) algorithm design for automatic resets, (3) biases in pre-trained policies, (4) the impact of individual learning factors, and (5) robustness to distributional drifts. We show that naively fine-tuning pre-trained policies fails in reset-free settings; however, these challenges can be mitigated through intelligent reset allocation and by addressing biases in pre-trained models. Finally, we demonstrate that reset-free training enhances long-horizon manipulation and improves generalization to out-of-distribution object configurations and task sequences. Project page: https://embodiedai-ntu.github.io/cronos/index.html.

1. Introduction

The ability to learn and adapt within their environments is essential for deploying robots in unstructured and continually changing real-world environments. Recently, behavior cloning (BC) has demonstrated remarkable results of robots trained to perform complex daily tasks (Black et al., 2024; Team et al., 2025). However, this learning framework alone is insufficient for robust real-world robot deployment. BC policies often fail under novel distributions–such as unseen scene layouts, object categories, or dynamics (Barreiros et al., 2025)–while collecting demonstrations to exhaustively cover such failures is impractical. Reinforcement learning (RL) is a compelling alternative, that enables policies to adapt through trial-and-error interaction with the environment. Recent studies show that RL fine-tuning can substantially improve generalization and robustness of BC policies in novel scenarios (Intelligence et al., 2025a).

Despite this promise, RL has so far been demonstrated primarily in simulation (Yu et al., 2020; Tunyasuvunakool et al., 2020) or small-scale real-world studies (Levine et al., 2016; Yahya et al., 2017; Kalashnikov et al., 2018; Ankile et al., 2025). Scaling RL to real-world deployment remains a major bottleneck due to several unrealistic assumptions in standard RL pipelines, such as reliance on episodic training, hand-crafted reward functions and human-specified task suites. This paper focuses specifically on the limitation of *episodic training*, which assumes frequent scene resets to prevent agents from being trapped in low-reward states during training. While scene reset is free in simulation, it incurs substantial human intervention in real-world settings, thereby preventing RL from being deployed at scale in real-world robotic systems.

To reduce these laborious needs, a wide range of reset-free RL algorithms have been proposed, including manually scripting the reset robot policy (Levine et al., 2016; Sharma et al., 2020), instrumenting the reset mechanism (Zeng et al., 2020; Kalashnikov et al., 2018; Kalashnkov et al., 2021), or tackling scene resets through a multi-task learning framework (Gupta et al., 2021) that treats scene resets as additional tasks and trains policies to reset (Eysenbach et al., 2018; Zhu et al., 2020). With these approaches, we aim to answer if they are capable of adapting state-of-the-art robot policies, such as vision-language-action models (VLAs; Kim et al., 2024; Li et al., 2024a; Qu et al., 2025), under non-episodic training settings.

Since real-world experiments are often time-consuming and costly, we propose to leverage simulation benchmarks to

[1]Department of Computer Science and Information Engineering, National Taiwan University. Correspondence to: Djengo Cyun-Jyun Fang <b09501048@csie.ntu.edu.tw>.

Preprint. March 19, 2026.

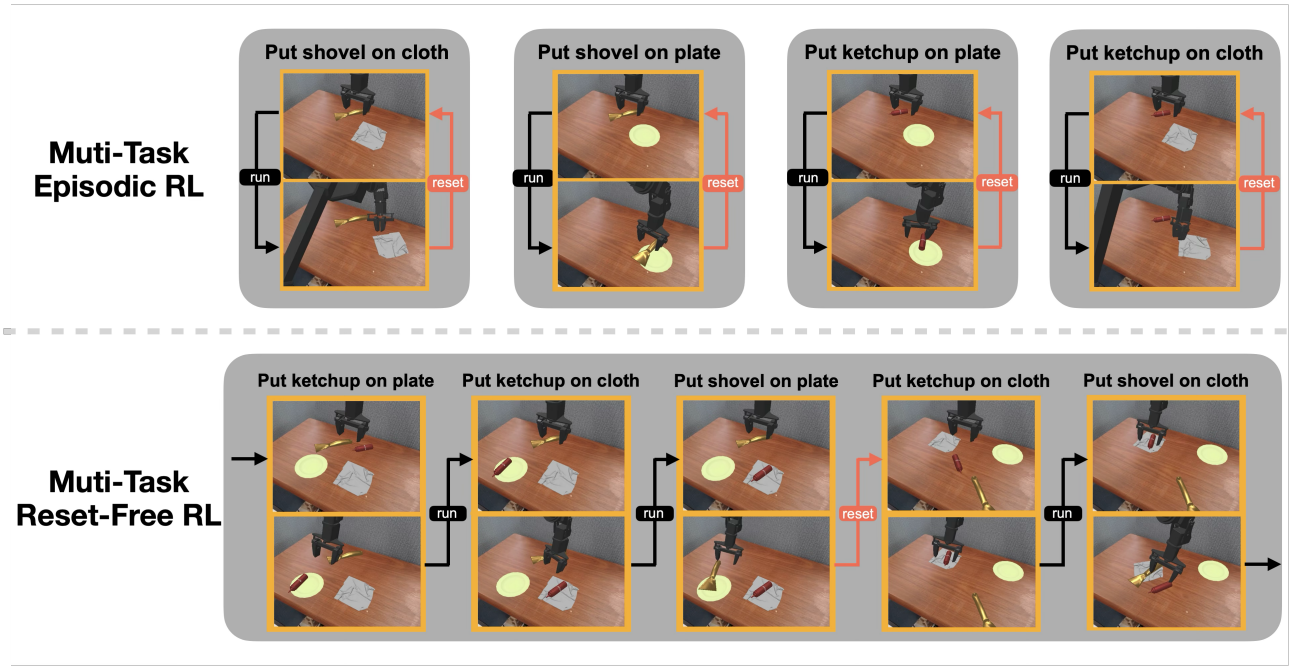


Figure 1. **Comparison between common episodic multi-task RL and our shared-scene reset-free multi-task RL (RF-MTRL). Top:** Common episodic multi-task RL isolates each task into separate scenes and conducts frequent scene resets to prevent agents from being trapped in low-reward states. **Bottom:** We propose CRONOS to study RF-MTRL. This simulation benchmark shares scenes across tasks, which reduces reset overhead and more closely reflects real-world deployment. Meanwhile, agents must determine not only the ordering of training tasks but also *when* and *how* to reset scenes given a fixed reset budget. This enables more realistic studies for RF-MTRL.

facilitate systematic study on reset-free RL. This paper introduces Continual Robotic Operations in NOn-episodic Settings (CRONOS)–a simulation benchmark carefully designed to reflect realistic deployment while addressing key limitations of existing benchmarks. First, CRONOS enhances sim-to-real alignment by building upon high-fidelity simulators–SimplerEnv (Li et al., 2024b) and its extension (Liu et al., 2025). In contrast, existing benchmarks rely on outdated simulators (Balsells et al., 2023) with simplified physics and limited visual realism, resulting in huge sim-to-real gap. Second, CRONOS adopts a realistic multi-task learning setup by sharing scenes across tasks, in contrast to prior benchmarks that isolate each task into a separate scene (Yang et al., 2024). This design reduces scene reset overhead and more closely reflects real-world deployment. Third, CRONOS targets recent advances in robot learning by evaluating reset-free RL fine-tuning of state-of-the-art pre-trained policies, such as OpenVLA (Kim et al., 2024), rather than policies trained from scratch (Gupta et al., 2021). Finally, following EARL (Sharma et al., 2021), CRONOS formalizes reset-free training as learning under a fixed budget of scene resets, enabling systematic analysis across varying levels of human intervention and encouraging the development of policies that learn how and when to reset. We present an overview comparison between common multi-task episodic RL and our shared-scene multi-task reset-free RL in Figure 1.

Using CRONOS, we conduct an initial study to investigate: (1) the effectiveness of reset-free RL under varying levels of human intervention, (2) the role and latent biases of pre-trained BC policies, (3) the development of algorithms that determine when and how to reset the scene, (4) the impact of individual learning factors–task ordering or curriculum learning–in reset-free RL, and (5) robustness to out-of-distribution (OOD) conditions, including novel object placements and task sequences.

Based on our experiments, we present the following key findings in reset-free multi-task RL (RF-MTRL). First, naively fine-tuning BC policies fails in reset-free settings. Second, addressing biases in pre-trained policies and implementing intelligent reset allocation–such as curriculum learning–are critical for enhancing RF-MTRL. Third, strategic task ordering is a vital component for learning efficiency. Finally, RF-MTRL enforces long-horizon manipulation and OOD generalization compared to episodic baselines.

2. Preliminaries

In this section, we present a general formulation of multi-task RL problems, and extend it to reset-free multi-task RL problems (Gupta et al., 2021) without loss of generality.

Multi-task RL. Multi-task RL problems consider learning optimal policies π^* for a set of tasks sampled from a task

distribution p_z. A task $z \sim p_z(\cdot)$ defines a Markov decision process (MDP) $\mathcal{M}_z = (\mathcal{S}, \mathcal{A}, P, R_z, \rho, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, P denotes the transition dynamics of the environment, R_z is the reward function of the task, ρ is the initial state distribution, and γ denotes the discount factor. Let $J(\pi)$ be the expected return across all tasks with respect to policy π, which is written as:

$$J(\pi) = \mathbb{E}_{z\sim p_z(\cdot)} \mathbb{E}_{\substack{s_0\sim\rho, a_h\sim\pi(\cdot|s_h;z) \\ s_{h+1}\sim P(\cdot|s_h,a_h)}} \left[\sum_{h=0}^{H} \gamma^h R_z(s_h, a_h) \right] \tag{1}$$

where policy π conditions on task z and H is the maximum time horizon. The RL objective is to learn a policy maximizing the expected return average across all tasks.

Reset-free Multi-Task RL (RF-MTRL). Following Sharma et al., 2021, we extend this formulation to *reset-free deployment settings*. Evaluation remains episodic, with trajectories rolled out from the initial state for each task. In contrast, training trajectories are rolled out continually across tasks with no or only occasional scene resets. Therefore, the evaluation objective and MDP are identical to Eqn. 1 and $\mathcal{M}_z$ for task z, whereas the training objective and MDP differ.

To model occasional scene resets, we re-define the transition dynamics $\hat{P}$ as $\hat{P}(\cdot|s_h, a_h) = (1-\varepsilon)P(\cdot|s_h, a_h) + \varepsilon\rho(\cdot)$, where the next state follows the environment dynamics P with probability $1-\varepsilon$, and is instead sampled from the initial-state distribution ρ with probability ε. This formulation generally covers different levels of human intervention: common episodic training uses larger ε, while occasional scene reset and non-episodic training (Zhu et al., 2020) use smaller and zero ε. The MDP during reset-free training becomes $\hat{\mathcal{M}}_z = (\mathcal{S}, \mathcal{A}, \hat{P}, R_z, \rho, \gamma)$.

Since training trajectories continually performs a sequence of tasks with occasional resets, we define the training objective of RF-MTRL as follows:

$$\hat{J}(\pi) = \mathbb{E}_{s_0\sim\rho} \mathbb{E}_{\substack{z_h\sim p_z(\cdot), a_h\sim\pi(\cdot|s_h;z_h) \\ s_{h+1}\sim\hat{P}(\cdot|s_h,a_h)}} \left[\sum_{h=0}^{H} \gamma^h R_{z_h}(s_h, a_h) \right] \tag{2}$$

where z_h denotes the task at time h.

Evaluation. To quantify the learning efficiency of reset-free RL algorithms, we first define a learning algorithm $\mathbb{A}$ that continually updates its policy based on accumulated experience. Formally, at time t, $\mathbb{A}$ maps the history of transitions $(s_i, a_i, s_{i+1})_{i=0}^{t-1}$ to (1) an action a_t used for data collection, and (2) a current policy estimate π_t, which is evaluated separately.

We evaluate $\mathbb{A}$ along two dimensions: interaction efficiency and reset efficiency. Interaction efficiency, commonly used in prior work (Sharma et al., 2021; Gupta et al., 2021), measures how quickly the algorithm approaches optimal performance as a function of environment interactions. We formalize this via the cumulative performance gap with a metric function $\mathbb{D}$:

$$\mathbb{D}(\mathbb{A}) = \sum_{t=0}^{T} \left(J(\pi^*) - J(\pi_t) \right), \tag{3}$$

where T is the interaction horizon, policy π_t is estimated by algorithm $\mathbb{A}$ at interaction step t, $J(\pi)$ is defined in Eqn. 1, and $\pi^* = \arg\max_\pi J(\pi)$ denotes the optimal policy.

Since scene resets are often significantly more costly than environment interactions, we additionally measure reset efficiency, which captures how effectively an algorithm learns under limited resets. Following the similar formulation as Eqn. 3, we define a metric function $\mathbb{C}$:

$$\mathbb{C}(\mathbb{A}) = \sum_{k=0}^{K} \big(J(\pi^*) - J(\pi_k) \big), \tag{4}$$

where K is the maximum number of scene resets and π_k denotes the policy estimated by algorithm $\mathbb{A}$ immediately after the k-th reset.

3. Challenges in Reset-free Multi-task RL

We outline five major challenges in RF-MTRL that motivates the design of CRONOS and our study.

(C1) Learning effectiveness under varying levels of human intervention. Scene resets are laborious and incurs a major bottleneck for scaling real-world RL deployment. Therefore, it is critical to evaluate how efficiently algorithms learn under varying reset budgets. CRONOS provides a general reset-efficiency metric (Eqn. 4) to enable this analysis.

(C2) Algorithm design for automatic reset. An effective reset-free RL algorithm requires two components: a learning framework operating under a limited reset budget, and an autonomous mechanism allocating that budget. Here we focus on the latter: deciding *when* and *how* to reset. The *when* decision involves detecting unrecoverable or persistently low-reward states, while the *how* decision concerns executing reset behaviors that return the agent to high-value or recoverable states.

(C3) Biases in pre-trained BC policy. While BC pre-training is known to improve RL sample efficiency (Rajeswaran et al., 2017), the conditions under which RL can effectively adapt BC policies—-especially in reset-free, multi-task settings—-remain an open question.

(C4) The impact of individual learning factors. Beyond core algorithmic choices, several auxiliary factors can substantially affect learning efficiency in reset-free RL settings.

We highlight two such factors, though this list is not exhaustive. (1) Task ordering. In sequential multi-task settings, the terminal states induced by earlier tasks constrain the feasibility and difficulty of subsequent tasks. Consequently, task ordering directly shapes the effective state distribution encountered during training and can dominate learning efficiency. Prior work typically relies on manually specified task sequences (Gupta et al., 2021; Zhu et al., 2020), leaving principled, automated task-ordering strategies largely unexplored. (2) Curriculum learning. This considers which tasks to be learned or how often scenes to be reset, in the early training stage. As shown in prior RL work (Wan et al., 2023), well-designed curricula can dramatically improve sample efficiency, yet their role in RF-MTRL remains understudied. The impact of these factors are rarely examined.

(C5) Robustness to out-of-distribution conditions. The capability of OOD generalization is critical for policy deployment. Existing simulation benchmarks primarily evaluate OOD generalization along visual, semantic, or execution dimensions (Liu et al., 2025). In contrast, CRONOS shares scenes across tasks and minimizes scene resets, which naturally induces sequential multi-task learning. We conjecture that this setup enforces policy's generalization to unseen task orderings. In this paper, CRONOS measures robustness along the dimension of task sequences and object positions.

4. CRONOS

CRONOS is designed to facilitate the rigorous analysis of RF-MTRL through simulation. To ensure realism, CRONOS adopts settings that closely reflect the challenges of real-world deployment.

CRONOS is built upon realistic simulation benchmarks–SimplerEnv (Li et al., 2024b) and its extension (Liu et al., 2025)–to ensure sim-to-real alignment. SimplerEnv was originally designed to approximate real-world robotic task suites derived from Bridge (Walke et al., 2023) using a high-fidelity physics simulator. It was subsequently extended with greater diversity in object categories, object placements, instruction phrasings, and scene appearance, enabling evaluation across visual, semantic, and execution variations. Prior studies (Li et al., 2024b; Liu et al., 2025) report that policies trained in this simulator can transfer to real-world settings with limited performance degradation, making it a reasonable testbed for studying reset-free RL.

Scene layout. While both SimplerEnv and its extension (Liu et al., 2025) incorporate multi-task training, their settings are unrealistic as they isolate tasks into individual scenes. This increases scene diversity, thereby requiring more intensive human supervision to create or reset environments. In real-world deployment, such manual effort should be minimized. Consequently, CRONOS implements a more natural and efficient setup by placing multiple objects in a single scene and sharing that scene across tasks. To construct a scene, CRONOS selects a subset of objects and receptacles from Liu et al., 2025, uses an RGB camera with 640×480 resolution, and deploys an 8-DoF WidowX-250S robotic arm. The robot is position-controlled via 6-DoF end-effector pose commands and gripper signals.

Task suites. CRONOS focuses on typical pick-and-place tasks, each defined by an object, a target receptacle, and a corresponding language instruction. The benchmark adopts a realistic shared-scene multi-task setting, where policies must learn to perform multiple tasks concurrently within the same environment. Specifically, CRONOS configures a scene with N objects and M receptacles, constituting $M \times N$ distinct tasks: "put the [object] on the [receptacle]. While the current version trains and tests models within the same scene, we will expand the benchmark to support scene randomization, facilitating the study of visual, semantic and execution generalization.

Training. Following prior work (Liu et al., 2025), we employ sparse rewards: a value of 0.1 for grasping or maintaining possession of the correct object, and 1.0 for successful task completion. Additionally, each object or receptacle is placed in one of 64 random positions. For all experiments, we use Proximal Policy Optimization (PPO; Schulman et al., 2017) as the base RL framework. During the rollout collection phase, task transitions occur every 80 interaction steps, while the scene reset frequency is determined by the specific algorithm. Other training parameters–such as training curriculum and task ordering–are left to the algorithmic design of each RF-MTRL method.

Independent-task evaluation. CRONOS adopts *reset-free deployment* settings (Sharma et al., 2021), where evaluation remains episodic. To evaluate each task, the scene is reset to a fixed initial configuration, and the agent is allowed to interact with the environment for up to 80 steps. A rollout is considered successful if the agent completes the task before the episode terminates. We use *success rate*–the proportion of successful completions across all trials–as the primary metric. As discussed in Sec. 2, CRONOS evaluates performance along two dimensions: success rate relative to interaction steps (Eqn. 3) and relative to the number of scene resets (Eqn. 4). Finally, we report both in-domain learning efficiency on the training set and out-of-distribution generalization on a held-out set, each consisting of 64 object-receptacle configurations.

Sequential-task evaluation. In real-world deployments, robots must often perform multiple tasks in sequence without human intervention. To reflect these scenarios, CRONOS incorporates a sequential evaluation setup that measures a policy's ability to transition between and accomplish tasks consecutively. This setup is inspired by

prior work–CALVIN (Mees et al., 2022)–which focuses on multi-task imitation learning, whereas CRONOS addresses reset-free multi-task reinforcement learning.

During testing, a policy is instructed to perform multiple tasks in sequence. The policy must complete each task within 80 steps; otherwise, it fails the current and all subsequent tasks in the chain. We incorporate 10 task orderings, each evaluated across 64 object-receptacle configurations. Consistent with the single-task setup, we assess in-domain and out-of-distribution performance on the training and held-out sets, reporting the mean and standard deviation of success rates across all rollouts.

5. Baseline Methods

We propose several baseline methods to tackle challenges–**C1, C2, C3, C4**–outlined in Section 3. Their results provide a systematic study on policy adaptation via RF-MTRL.

Periodic and heuristic reset. We propose two baseline algorithms for automatic reset (**C2**), enabling us to study learning effectiveness under varying reset budgets (**C1**). The first is *periodic reset*, which resets the scene every T interaction steps. In this paper, we evaluate various reset frequencies with $T \in \{80, 320, 1280, 2560\}$, where $T = 80$ corresponds to standard episodic training. The second algorithm is *heuristic reset*, which triggers a reset whenever the scene enters an unrecoverable state–such as when objects or receptacles are displaced outside the robot's workspace. This prevents the agent from wasting interaction steps in impossible scenarios, allowing the heuristic baseline to strike a better balance between interaction and reset budgets.

Learning-based reset. Another established approach to automatic reset (**C2**) is the use of learned reset policies (Eysenbach et al., 2018). We define 'put the [object] on the table' as the reset task and fine-tune the policy to perform both pick-and-place and reset tasks. This baseline alternates rollout collection between the primary pick-and-place task and the reset task; accordingly, we refer to this as the *learned reset* baseline.

VLAs and episodic end-effector reset. In real-world deployment, fine-tuning pre-trained models often better satisfies application requirements, as it is more effective than training from scratch for obtaining robust policies (Rajeswaran et al., 2017). To reflect this, we focus on fine-tuning state-of-the-art VLA models under reset-free settings (**C3**). We select OpenVLA (Kim et al., 2024)–a model widely adopted in recent RL fine-tuning literature (Zhang et al., 2024; Lu et al., 2025)–to ensure the reproducibility of our study. Surprisingly, we found that reset-free fine-tuning of OpenVLA is initially ineffective; the policy often samples low-quality actions that sweep objects off the table during task transitions. We hypothesize that this stems from a distributional drift in the robot's end-effector configuration between the pre-training and fine-tuning phases. Specifically, VLAs are pre-trained on expert demonstrations (O'Neill et al., 2024) where the end-effector typically begins at a fixed pose. In contrast, reset-free settings involve more diverse starting configurations. Consequently, we propose to episodically reset the end-effector, but not the scene, when fine-tuning OpenVLA. While we address end-effector drift, we recognize that other biases impacting RL sample efficiency in this context are yet to be fully uncovered.

Random and heuristic task ordering. To analyze the impact of training task ordering (**C4**), we compare random ordering against a heuristic approach. This heuristic determines subsequent tasks based on a cyclic task graph (Gupta et al., 2021), which avoids consecutively picking the same object or placing it onto the same receptacle. See the Appendix E for further details.

Curriculum Learning. To analyze the impact of reset allocation (**C2**,**C4**), we devise a curriculum learning approach that performs more frequent scene resets during the early stages of training and reduces the frequency as training progresses. This examines whether scene resets should be allocated adaptively based on the agent's learning progress.

6. Experiments

We conduct experiments as an initial study for tackling challenges described in Section 3. Specifically, our experiments are designed to answer the following questions:

Q1: Which biases in pre-trained BC policies affect fine-tuning efficiency under RF-MTRL settings?
Q2: How do RF-MTRL baselines perform on CRONOS?
Q3: What are the impacts of learning factors, such as task ordering and curriculum learning, on RF-MTRL?
Q4: What is learning efficiency of RF-MTRL under varying levels of human intervention?
Q5: How robust are RF-MTRL policies to OOD conditions?
Q6: How do RF-MTRL baselines scale to more complex scenes and expanded task suites?

For Q1-Q5, we configure a scene with two objects (shovel and ketchup bottle) and two receptacles (plate and cloth), constituting four distinct tasks. To answer Q6, we introduce a more complex environment featuring three objects (toy bear, shovel and ketchup bottle) and three receptacles (plate, cloth and carpet), resulting in a total of nine pick-and-place tasks. We refer to the Appendix D for more training details.

6.1. Biases in Pre-trained BC Policies

We first investigate the feasibility of fine-tuning BC policies (OpenVLA) under reset-free multi-task settings. We compare model performance under episodic training against

T	Episodic e.e reset	Curriculum learning	Learned scene reset	Heuristic scene reset	Success rate @ 1.3M interaction
80	–	–	–	–	$0.897_{\pm 0.027}$
1280	–	–	–	–	$0.023_{\pm 0.030}$
1280	✓	–	–	–	$0.326_{\pm 0.179}$
1280	✓	✓	–	–	$0.578_{\pm 0.054}$
1280	✓	✓	✓	–	$0.580_{\pm 0.036}$
1280	✓	✓	–	✓	$0.649_{\pm 0.090}$
1280	✓	✓	✓	✓	$0.631_{\pm 0.065}$

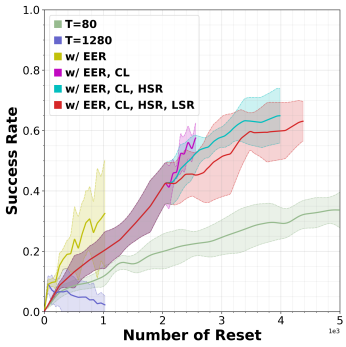


Table 1. **Performance of RF-MTRL approaches.** We analyze the impact of episodic end-effector resets (EER), curriculum learning (CL), learned scene resets (LSR), and heuristic scene resets (HSR) on RF-MTRL. All methods are trained for a maximum of 1.3M interaction steps across five random seeds, and tested on the training set of object-receptacle configurations. For curriculum learning, we set $T = 320$ to the first 0.65M interaction steps, and increase it to 1280 afterwards. **Left:** We measure interaction efficiency by calculating the average success rate achieved by the final checkpoints. **Right:** We measure reset efficiency by calculating the average success rate relative to the total number of scene resets.

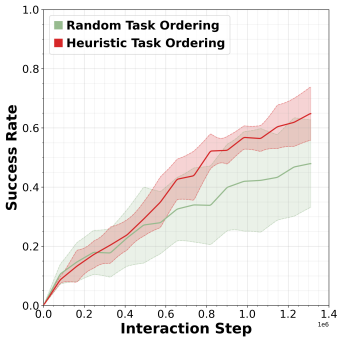


Figure 2. **Impact of task ordering.** We evaluate performance of heuristic and random task ordering. We use independent-task evaluation, reporting the average success rate relative to total interaction steps. Models are tested on the training set of object-receptacle configurations. Heuristic task ordering outperforms random task ordering.

reset-free training. Specifically, the former employs episodic scene resets ($T = 80$), while the latter utilizes periodic scene resets ($T = 1280$). In this experiment, all models are trained for a maximum of 1.3M interaction steps across 5 random seeds. Using independent-task evaluation, we test the final checkpoints on the training set of object-receptacle configurations and report the average success rate across all tasks and seeds.

The results are summarized in Table 1. Surprisingly, we find that directly fine-tuning OpenVLA under reset-free settings is ineffective, yielding only a 2.3% average success rate compared to 89.7% achieved via episodic training. As discussed in Section 5, our analysis suggests that these BC policies overfit the end-effector configuration distributions found in pre-training expert demonstrations, where the end-effector is always reset to a fixed pose at the beginning of each task. We validate this hypothesis by implementing episodic end-effector resets, which significantly improves performance, increasing the absolute success rate by 30.3%.

6.2. Efficacy of RF-MTRL Approaches

We next investigate the performance of existing RF-MTRL approaches on CRONOS, considering two primary baselines: (1) learned scene reset and (2) heuristic scene reset. Following the training and evaluation protocol established in Section 6.1, we summarize the results in Table 1.

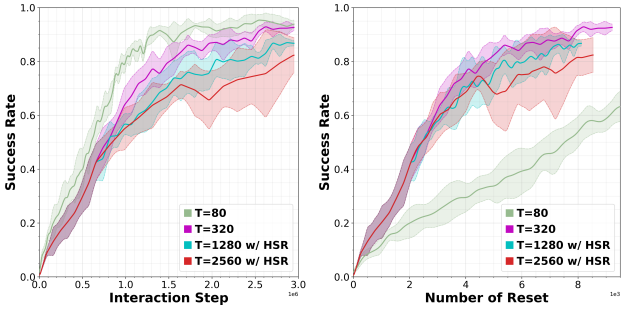
While the learned scene reset approach avoids additional reset overhead during training, it yields only a trivial performance gain of 0.2% in average success rate. We hypothesize that RL sample efficiency degrades as the number of learning tasks increases. Given a fixed interaction budget, a policy required to master both pick-and-place and reset tasks achieves lower performance than a policy dedicated solely to pick-and-place tasks.

In contrast, our heuristic scene reset baseline proves effective, providing an absolute performance gain of 7.1% in success rate. However, this improvement incurs a cost: the baseline consumes twice the reset budget compared to the model without heuristic resets (6000 vs. 2000 resets, as shown in the right figure of Table 1). These results indicate that while the heuristic approach enhances interaction efficiency, it does not necessarily improve reset efficiency. This trade-off exemplifies the fundamental challenge of balancing interaction and reset efficiency, emphasizing the necessity for more intelligent reset algorithms in RF-MTRL.

6.3. Impact of Curriculum Learning

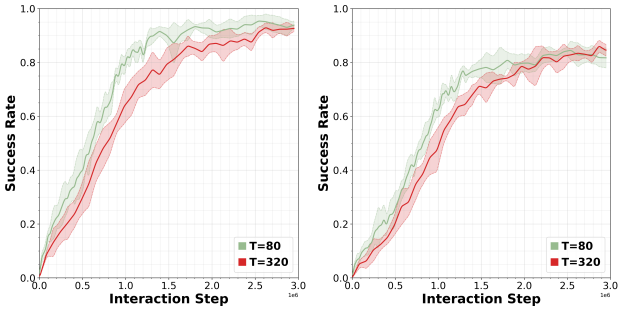
Curriculum learning offers an alternative strategy for allocating scene resets by prioritizing more frequent resets during the early stages of training. Specifically, we set $T = 320$ for the first 0.65M interaction steps, subsequently increasing T to 1280 for the remainder of training. Following the same protocol as Section 6.1, our results in Table 1

demonstrate that this approach significantly enhances interaction efficiency, bringing an absolute gain of 25.2% in the average success rate. Notably, this strategy achieves a superior trade-off between interaction and reset efficiency; it requires only 60% more resets to match the success rate of the non-curriculum baseline (1600 vs. 1000 resets at a 32.6% success rate, as shown in the right figure of Table 1).

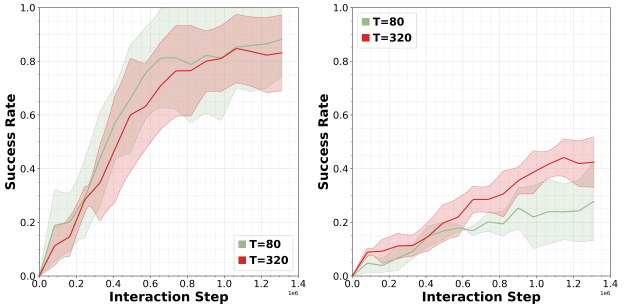


(a) Independent task (#. interaction steps) (b) Independent task (#. resets)

Figure 3. **Learning efficiency under varying reset budgets.** We consider four baselines: standard episodic training ($T = 80$), periodic reset ($T = 320$) with episodic end-effector resets, and a combination of periodic reset ($T \in \{1280, 2560\}$) and heuristic reset (HSR) integrated with episodic end-effector resets and curriculum learning. All models are trained using heuristic task ordering and evaluated on the training set of object configurations. **From left to right**: We evaluate independent-task manipulation relative to total interaction steps (a) and the total number of scene resets (b). Non-episodic baselines achieve superior reset efficiency.



(a) Training configurations (b) Held-out configurations



(c) Training configurations (d) Held-out configurations

Figure 4. **Robustness to OOD object-receptacle configurations.** We compare an episodic baseline ($T = 80$) against a non-episodic baseline–periodic reset ($T = 320$) with episodic end-effector resets. Both models are tested on the training set–consisting of 64 (top row) or 1 (bottom row) object-receptacle configurations–and the held-out set, consisting of 64 object-receptacle configurations. When training on fewer configurations, the periodic reset approach demonstrates stronger OOD generalization to the episodic baseline.

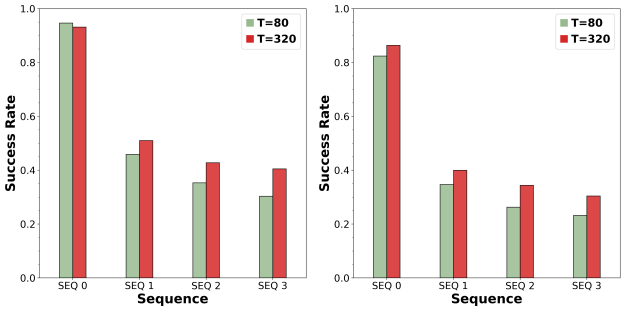
6.4. Impact of Task Ordering

We investigate how task ordering influences learning dynamics in reset-free multi-task training by comparing our proposed heuristic task ordering against a random ordering baseline. We follow the same training and evaluation protocol as Section 6.1. Specifically, we select the baseline that adopts periodic reset ($T = 1280$), episodic end-effector reset, curriculum learning and heuristic scene resets. We report the average success rate relative to total interaction steps. As shown in Figure 2, heuristic task ordering achieves higher learning efficiency than random ordering under reset-free multi-task settings. These results suggest that strategic task sequencing is a critical component for optimizing performance in RF-MTRL.

6.5. Learning Efficiency Under Varying Reset Budgets

We study learning effectiveness under varying levels of human intervention (reset budgets) by evaluating four primary baselines: (1) standard episodic training ($T = 80$), (2) periodic reset ($T = 320$) with episodic end-effector resets, and (3) a combination of periodic and heuristic resets ($T \in \{1280, 2560\}$) integrated with episodic end-effector resets and curriculum learning. To ensure convergence, all methods are trained for 2.9M interaction steps. We perform both independent-task and sequential-task evaluations to assess the agent's capability in both isolated and chained manipulation scenarios. Our analysis focuses on two dimensions: interaction efficiency and reset efficiency, measuring performance relative to total interaction steps and the total number of scene resets, respectively.

Our results are summarized in Figure 3. In independent-task evaluation, while the standard episodic baseline ($T = 80$) achieves the highest success rate for a fixed number of interaction steps, it exhibits poor reset efficiency, resulting in a 38% performance gap under an 8,000-reset budget. Conversely, we observe a degradation in learning efficiency as reset frequency decreases ($T \in \{1280, 2560\}$). Notably, these baselines consume a similar number of resets as the $T = 320$ baseline due to the heuristic reset mechanism, yet they fail to match its performance. These results underscore the ongoing need for more optimal RF-MTRL algorithms that maximize the utility of each interaction.



(a) Training configurations (b) Held-out configurations

Figure 5. **Robustness to OOD task sequences.** We compare an episodic baseline ($T = 80$) against a non-episodic baseline–periodic reset ($T = 320$) with episodic end-effector resets. The periodic reset baseline is trained with heuristic task ordering. We adopt sequential-task evaluation, using 10 task sequences. Notably, 9 out of 10 task sequences used for evaluation are unseen during training. In addition, both models are tested on the training (left, in-domain) and the held-out (right, OOD) set, each consisting of 64 object-receptacle configurations. The non-episodic approach not only enhances long-horizon manipulation, but also demonstrates stronger generalization to OOD task sequences during inference.

6.6. Robustness to distributional shifts

We evaluate the OOD generalization of the learned policies under reset-free settings. In this study, we focus specifically on distributional drifts in two areas: object-receptacle configurations and testing task sequences.

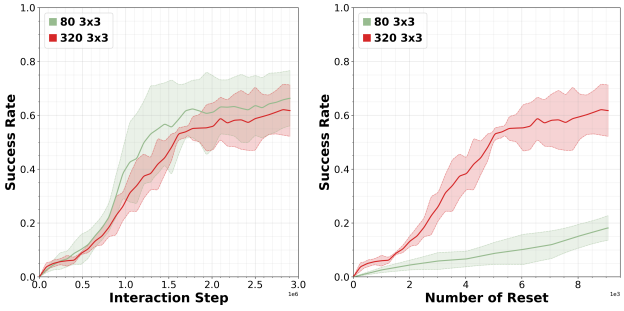
For object configurations, we hypothesize that RF-MTRL facilitates stronger generalization than episodic training when policies are trained on a limited number of configurations–a performance gap that narrows as the diversity of training configurations increases. We compare two baselines: (1) standard episodic training ($T = 80$) and (2) periodic reset ($T = 320$) with episodic end-effector resets. Both models are trained for 1.3M interaction steps across five random seeds, with final checkpoints evaluated using the independent-task protocol. To test our hypothesis, we train these models on either 64 configurations or a single, fixed object-receptacle configuration. The periodic reset baseline uses heuristic task ordering during training.

Figure 4 summarizes our results. The columns represent evaluation on the training and held-out configurations, while the rows differentiate between models trained on 64 and 1 configuration(s). As expected, when trained on a single configuration, the periodic reset method achieves performance comparable to the episodic baseline on training data but yields a substantial performance gain on held-out configurations. This performance gap narrows as the training diversity increases. These results are encouraging: they suggest that even without manual efforts to increase scene variety, practitioners can simply allow the scene to evolve naturally via RF-MTRL to achieve robust real-world fine-tuning.

Figure 6. **Training scene for complex object–receptacle configurations.** We scale the environment from a 2×2 to a 3×3 setup, introducing three objects and three receptacles, resulting in 9 tasks. This increases environment complexity for multi-task learning.

For testing task sequences, we conjecture that RF-MTRL approaches inherently excel at sequential task execution, as they are trained to handle the continuous state transitions that episodic baselines rarely encounter. We evaluate the episodic and periodic reset baselines–both trained on 64 configurations–on 10 different task orderings. Notably, for the periodic reset baseline, 9 out of 10 task sequences used for evaluation were unseen during training. Across these orderings, we report the average success rate of every task performed within the first 80^{th} steps (SEQ 0), the 80^{th} to 160^{th} steps (SEQ 1), the 160^{th} to 240^{th} steps (SEQ 2) and the last 80^{th} steps (SEQ 3). The model performance is expected to degrade as interaction steps increase due to the accumulation of action errors over time.

As shown in Figure 5, the periodic reset baseline outperforms the episodic baseline. This is expected, as our multi-object scenes naturally support sequential task execution in reset-free settings. These findings mirror how humans learn to perform sequences of tasks in complex environments, suggesting a promising direction for designing task suites in both simulation and the real world.



(a) Independent task (# interaction steps) (b) Independent task (# resets)

Figure 7. **Performance of a more complex task suite.** We evaluate the independent-task baseline under a larger task set consisting of three objects and three receptacles. Performance is measured with respect to (a) the number of interaction steps and (b) the number of scene resets. The results show that while both approaches achieve comparable performance when measured by interaction steps, the reset-efficient setting demonstrates improved robustness under a limited reset budget.

6.7. Scalability to More Complex Task Suites

In real-world deployments, robots are expected to handle a wide range of tasks. Therefore, we validate the scalability of current RF-MTRL baselines to more complex scenes and expanded task suites. Specifically, we configure a scene with three objects and three receptacles, resulting in a suite of nine combinatorial pick-and-place tasks (Figure 6).

We evaluate two baselines: (1) standard episodic training ($T = 80$), and (2) periodic reset ($T = 320$) with episodic end-effector resets. Both methods are trained for 2.9M interaction steps and evaluated using the independent-task protocol. We summarize our results in Fig 7. While the non-episodic method ($T = 320$) achieves competitive performance against the episodic baseline ($T = 80$) over a fixed number of interaction steps, it demonstrates significantly superior reset efficiency–achieving an absolute performance gain of 44% in success rate under a fixed reset budget of 9000 resets. These results confirm that current RF-MTRL baselines scale effectively to more complex scenarios.

7. Related Work

Fine-tuning VLA with RL. Vision-Language Action (VLA) models (Intelligence et al., 2025b; Kim et al., 2024; O'Neill et al., 2024; Ankile et al., 2025; Team et al., 2024) learn generalist robot policies across different tasks by leveraging pre-trained vision-and-language foundation models. Despite their strong representational capacity, VLA models inherit a key limitation of behavior cloning methods: poor robustness to out-of-distribution (OOD) scenarios. Recent work has shown that reinforcement learning fine-tuning can substantially improve VLA generalization in perception, semantics, and execution (Liu et al., 2025). Several approaches further enhance long-horizon generalization through task decomposition and dense rewards (Zhang et al., 2025), action chunking with value estimation (Huang et al., 2025), world-model-based verification (Li et al., 2025), or exploration-driven strategies (Lu et al., 2025). However, all existing methods fine-tune VLA models under episodic training assumption, while our benchmark explores the VLA fine-tuning under a non-episodic setting.

Reset-free RL. Real-world reinforcement learning is fundamentally constrained by the cost of human intervention for environment resets, and prior benchmarks show that episodic algorithms do not directly extend to non-episodic settings (Liu et al., 2023; Sharma et al., 2021). Existing approaches address this limitation via model-based methods that improve exploration efficiency without resets (Yang et al., 2025; Treven et al., 2024), joint learning of task and reset policies with explicit consideration of irreversible states (Lee & Seo, 2024; Patil et al., 2024; Sharma et al., 2023; Kim et al., 2023; Walke et al., 2022; Sharma et al., 2022; 2021; Zhang & Weihs, 2023; Xie et al., 2022; Eysenbach et al., 2018), or by formulating reset-free learning as multi-task learning where auxiliary tasks implicitly induce resets (Gupta et al., 2022; Xu et al., 2022; Gupta et al., 2021; Ha et al., 2020). Hybrid formulations that integrate these perspectives have also been explored (Yang et al., 2024). In contrast, our benchmark explicitly quantifies reset efficiency by measuring the number of resets and highlights the core challenges of reset-free learning.

8. Conclusion

The contributions of this paper are: (1) we introduce CRONOS, a simulation benchmark designed to enable systematic studies of RF-MTRL, (2) we formalize two distinct evaluation metrics to quantify the interaction efficiency and reset efficiency of RF-MTRL algorithms, (3) we focus on the fine-tuning of state-of-the-art pre-trained BC policies, a setting that better reflects real-world deployment, and (4) we present an extensive analysis revealing the necessity of mitigating policy biases, utilizing intelligent reset allocation, and implementing strategic task ordering in RF-MTRL.

References

Ankile, L., Jiang, Z., Duan, R., Shi, G., Abbeel, P., and Nagabandi, A. Residual off-policy rl for finetuning behavior cloning policies. *arXiv preprint arXiv:2509.19301*, 2025.

Balsells, M., Villasevil, M. T., Wang, Z., Desai, S., Agrawal, P., and Gupta, A. Autonomous robotic reinforcement learning with asynchronous human feedback. In *Conference on Robot Learning*, pp. 774–799. PMLR, 2023.

Barreiros, J., Beaulieu, A., Bhat, A., Cory, R., Cousineau, E., Dai, H., Fang, C.-H., Hashimoto, K., Irshad, M. Z., Itkina, M., et al. A careful examination of large behavior models for multitask dexterous manipulation. *arXiv preprint arXiv:2507.05331*, 2025.

Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., et al. $\pi 0$: A vision-language-action flow model for general robot control, 2024. *arXiv preprint arXiv:2410.24164*, 2024.

Eysenbach, B., Gu, S., Ibarz, J., and Levine, S. Leave no trace: Learning to reset for safe and autonomous reinforcement learning. In *International Conference on Learning Representations*, 2018.

Gupta, A., Yu, J., Zhao, T. Z., Kumar, V., Rovinsky, A., Xu, K., Devlin, T., and Levine, S. Reset-free reinforcement learning via multi-task learning: Learning dexterous manipulation behaviors without human intervention. In

2021 IEEE International Conference on Robotics and Automation (ICRA), pp. 6664–6671. IEEE, 2021.

Gupta, A., Lynch, C., Kinman, B., Peake, G., Levine, S., and Hausman, K. Demonstration-bootstrapped autonomous practicing via multi-task reinforcement learning, 2022. URL https://arxiv.org/abs/2203.15755.

Ha, S., Xu, P., Tan, Z., Levine, S., and Tan, J. Learning to walk in the real world with minimal human effort, 2020. URL https://arxiv.org/abs/2002.08550.

Huang, D., Fang, Z., Zhang, T., Li, Y., Zhao, L., and Xia, C. Co-rft: Efficient fine-tuning of vision-language-action models through chunked offline reinforcement learning, 2025. URL https://arxiv.org/abs/2508.02219.

Intelligence, P., Amin, A., Aniceto, R., Balakrishna, A., Black, K., Conley, K., Connors, G., Darpinian, J., Dhabalia, K., DiCarlo, J., et al. $\pi_{0.6}$: a vla that learns from experience. *arXiv preprint arXiv:2511.14759*, 2025a.

Intelligence, P., Black, K., Brown, N., Darpinian, J., Dhabalia, K., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Galliker, M. Y., Ghosh, D., Groom, L., Hausman, K., Ichter, B., Jakubczak, S., Jones, T., Ke, L., LeBlanc, D., Levine, S., Li-Bell, A., Mothukuri, M., Nair, S., Pertsch, K., Ren, A. Z., Shi, L. X., Smith, L., Springenberg, J. T., Stachowicz, K., Tanner, J., Vuong, Q., Walke, H., Walling, A., Wang, H., Yu, L., and Zhilinsky, U. $\pi_{0.5}$: a vision-language-action model with open-world generalization, 2025b. URL https://arxiv.org/abs/2504.16054.

Kalashnikov, D., Irpan, A., Pastor, P., Ibarz, J., Herzog, A., Jang, E., Quillen, D., Holly, E., Kalakrishnan, M., Vanhoucke, V., et al. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on robot learning*, pp. 651–673. PMLR, 2018.

Kalashnkov, D., Varley, J., Chebotar, Y., Swanson, B., Jonschkowski, R., Finn, C., Levine, S., and Hausman, K. Mt-opt: Continuous multi-task robotic reinforcement learning at scale. *arXiv*, 2021.

Kim, J., Cho, D., and Kim, H. J. Demonstration-free autonomous reinforcement learning via implicit and bidirectional curriculum, 2023. URL https://arxiv.org/abs/2305.09943.

Kim, M., Pertsch, K., Karamcheti, S., Xiao, T., Balakrishna, A., Nair, S., Rafailov, R., Foster, E., Lam, G., Sanketi, P., Vuong, Q., Kollar, T., Burchfiel, B., Tedrake, R., Sadigh, D., Levine, S., Liang, P., and Finn, C. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Lee, J., Duan, J., Fang, H., Deng, Y., Liu, S., Li, B., Fang, B., Zhang, J., Wang, Y. R., Lee, S., et al. Molmoact: Action reasoning models that can reason in space. *arXiv preprint arXiv:2508.07917*, 2025.

Lee, S.-H. and Seo, S.-W. Self-supervised curriculum generation for autonomous reinforcement learning without task-specific knowledge, 2024. URL https://arxiv.org/abs/2311.09195.

Levine, S., Finn, C., Darrell, T., and Abbeel, P. End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17(39):1–40, 2016.

Li, H., Ding, P., Suo, R., Wang, Y., Ge, Z., Zang, D., Yu, K., Sun, M., Zhang, H., Wang, D., and Su, W. Vla-rft: Vision-language-action reinforcement fine-tuning with verified rewards in world simulators, 2025. URL https://arxiv.org/abs/2510.00406.

Li, Q., Liang, Y., Wang, Z., Luo, L., Chen, X., Liao, M., Wei, F., Deng, Y., Xu, S., Zhang, Y., et al. Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. *arXiv preprint arXiv:2411.19650*, 2024a.

Li, X., Hsu, K., Gu, J., Pertsch, K., Mees, O., Walke, H. R., Fu, C., Lunawat, I., Sieh, I., Kirmani, S., et al. Evaluating real-world robot manipulation policies in simulation. *arXiv preprint arXiv:2405.05941*, 2024b.

Liu, B., Zhu, Y., Gao, C., Feng, Y., Liu, Q., Zhu, Y., and Stone, P. Libero: Benchmarking knowledge transfer for lifelong robot learning. *arXiv preprint arXiv:2306.03310*, 2023.

Liu, J., Gao, F., Wei, B., Chen, X., Liao, Q., Wu, Y., Yu, C., and Wang, Y. What can rl bring to vla generalization? an empirical study. *arXiv preprint arXiv:2505.19789*, 2025.

Lu, G., Guo, W., Zhang, C., Zhou, Y., Jiang, H., Gao, Z., Tang, Y., and Wang, Z. Vla-rl: Towards masterful and general robotic manipulation with scalable reinforcement learning. *arXiv preprint arXiv:2505.18719*, 2025.

Mees, O., Hermann, L., Rosete-Beas, E., and Burgard, W. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022.

O'Neill, A., Rehman, A., Maddukuri, A., Gupta, A., Padalkar, A., Lee, A., Pooley, A., Gupta, A., Mandlekar, A., Jain, A., et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6892–6903. IEEE, 2024.

Patil, D., Rajendran, J., Berseth, G., and Chandar, S. Intelligent switching for reset-free rl, 2024. URL https://arxiv.org/abs/2405.01684.

Qu, D., Song, H., Chen, Q., Yao, Y., Ye, X., Ding, Y., Wang, Z., Gu, J., Zhao, B., Wang, D., et al. Spatialvla: Exploring spatial representations for visual-language-action model. *arXiv preprint arXiv:2501.15830*, 2025.

Rajeswaran, A., Kumar, V., Gupta, A., Vezzani, G., Schulman, J., Todorov, E., and Levine, S. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. *arXiv preprint arXiv:1709.10087*, 2017.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Sharma, A., Ahn, M., Levine, S., Kumar, V., Hausman, K., and Gu, S. Emergent real-world robotic skills via unsupervised off-policy reinforcement learning. *arXiv preprint arXiv:2004.12974*, 2020.

Sharma, A., Xu, K., Sardana, N., Gupta, A., Hausman, K., Levine, S., and Finn, C. Autonomous reinforcement learning: Formalism and benchmarking. *arXiv preprint arXiv:2112.09605*, 2021.

Sharma, A., Ahmad, R., and Finn, C. A state-distribution matching approach to non-episodic reinforcement learning. In *International Conference on Machine Learning*, pp. 19645–19657. PMLR, 2022.

Sharma, A., Ahmed, A. M., Ahmad, R., and Finn, C. Self-improving robots: End-to-end autonomous visuomotor reinforcement learning, 2023. URL https://arxiv.org/abs/2303.01488.

Team, G. R., Abeyruwan, S., Ainslie, J., Alayrac, J.-B., Arenas, M. G., Armstrong, T., Balakrishna, A., Baruch, R., Bauza, M., Blokzijl, M., et al. Gemini robotics: Bringing ai into the physical world. *arXiv preprint arXiv:2503.20020*, 2025.

Team, O. M., Ghosh, D., Walke, H., Pertsch, K., Black, K., Mees, O., Dasari, S., Hejna, J., Kreiman, T., Xu, C., Luo, J., Tan, Y. L., Chen, L. Y., Sanketi, P., Vuong, Q., Xiao, T., Sadigh, D., Finn, C., and Levine, S. Octo: An open-source generalist robot policy, 2024. URL https://arxiv.org/abs/2405.12213.

Treven, L., Dorfler, F., Coros, S., and Krause, A. Neorl: Efficient exploration for nonepisodic rl. *Advances in Neural Information Processing Systems*, 37:74966–74998, 2024.

Tunyasuvunakool, S., Muldal, A., Doron, Y., Liu, S., Bohez, S., Merel, J., Erez, T., Lillicrap, T., Heess, N., and Tassa, Y. dm_control: Software and tasks for continuous control. *Software Impacts*, 6:100022, 2020. ISSN 2665-9638. doi: https://doi.org/10.1016/j.simpa.2020.100022. URL https://www.sciencedirect.com/science/article/pii/S2665963820300099.

Walke, H., Yang, J., Yu, A., Kumar, A., Orbik, J., Singh, A., and Levine, S. Don't start from scratch: Leveraging prior data to automate robotic reinforcement learning, 2022. URL https://arxiv.org/abs/2207.04703.

Walke, H. R., Black, K., Zhao, T. Z., Vuong, Q., Zheng, C., Hansen-Estruch, P., He, A. W., Myers, V., Kim, M. J., Du, M., et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning*, pp. 1723–1736. PMLR, 2023.

Wan, W., Geng, H., Liu, Y., Shan, Z., Yang, Y., Yi, L., and Wang, H. Unidexgrasp++: Improving dexterous grasping policy learning via geometry-aware curriculum and iterative generalist-specialist learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3891–3902, 2023.

Xie, A., Tajwar, F., Sharma, A., and Finn, C. When to ask for help: Proactive interventions in autonomous reinforcement learning, 2022. URL https://arxiv.org/abs/2210.10765.

Xu, K., Hu, Z., Doshi, R., Rovinsky, A., Kumar, V., Gupta, A., and Levine, S. Dexterous manipulation from images: Autonomous real-world rl via substep guidance, 2022. URL https://arxiv.org/abs/2212.09902.

Yahya, A., Li, A., Kalakrishnan, M., Chebotar, Y., and Levine, S. Collective robot reinforcement learning with distributed asynchronous guided policy search. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 79–86. IEEE, 2017.

Yang, J., Mark, M. S., Vu, B., Sharma, A., Bohg, J., and Finn, C. Robot fine-tuning made easy: Pre-training rewards and policies for autonomous real-world reinforcement learning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4804–4811. IEEE, 2024.

Yang, Z., Moerland, T. M., Preuss, M., Plaat, A., and Hu, E. S. Reset-free reinforcement learning with world models, 2025. URL https://arxiv.org/abs/2408.09807.

Yu, T., Quillen, D., He, Z., Julian, R., Hausman, K., Finn, C., and Levine, S. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on robot learning*, pp. 1094–1100. PMLR, 2020.

Zeng, A., Song, S., Lee, J., Rodriguez, A., and Funkhouser, T. Tossingbot: Learning to throw arbitrary objects with residual physics. *IEEE Transactions on Robotics*, 36(4): 1307–1319, 2020.

Zhang, H., Zhuang, Z., Zhao, H., Ding, P., Lu, H., and Wang, D. Reinbot: Amplifying robot visual-language manipulation with reinforcement learning, 2025. URL https://arxiv.org/abs/2505.07395.

Zhang, Z. and Weihs, L. When learning is out of reach, reset: Generalization in autonomous visuomotor reinforcement learning, 2023. URL https://arxiv.org/abs/2303.17600.

Zhang, Z., Zheng, K., Chen, Z., Jang, J., Li, Y., Han, S., Wang, C., Ding, M., Fox, D., and Yao, H. Grape: Generalizing robot policy via preference alignment. *arXiv preprint arXiv:2411.19309*, 2024.

Zhu, H., Yu, J., Gupta, A., Shah, D., Hartikainen, K., Singh, A., Kumar, V., and Levine, S. The ingredients of real-world robotic reinforcement learning. *arXiv preprint arXiv:2004.12570*, 2020.

A. Limitations

We identify the following limitations of our proposed simulation benchmark:

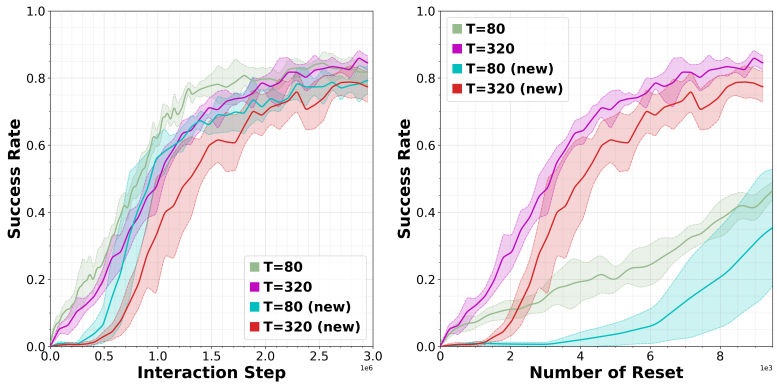
Limited evaluation on generalization. Our current protocol for evaluating OOD generalization does not yet account for the dimensions of object categories, object quantities, physical properties, scene appearances, and lighting conditions–factors that are critical to faithfully mirroring the complexity of real-world deployment. We intend to expand the benchmark with a significantly more diverse range of environments following its public release.

Limited robot policy diversity. This study evaluates a single BC policy, OpenVLA, despite the recent release of numerous alternative VLAs (Qu et al., 2025; Black et al., 2024; Lee et al., 2025). It remains unclear whether these models exhibit the same failure modes or latent biases observed in OpenVLA. A more comprehensive study involving a broader range of policy architectures is required to determine the generalizability of our findings across different VLA foundations.

Narrow task complexity and reversibility. While pick-and-place represents a fundamental building block of manipulation, it does not capture the full spectrum of challenges in reset-free learning, such as irreversible states (e.g., spilling liquid or breaking an object). In our current setup, the "reset" of one task is often the "start" of another, but real-world scenarios may involve "dead-ends" where a robot cannot recover without high-level human intervention. Future iterations of CRONOS should incorporate more complex, non-reversible tasks to better test the limits of autonomous recovery and failure detection.

B. More Experiments

We additionally validate the performance of the RF-MTRL baselines in a distinct scene where the objects and receptacles exhibit different geometries, surface properties, and contact dynamics compared to those in the main experiment (Figure 9). Specifically, we configure the scene with a novel set of objects (a toy bear and a plastic bottle) and receptacles (a newspaper and a carpet). Following the same training and testing protocol detailed in Section 6.5, we study learning effectiveness under varying levels of human intervention (reset budgets). We adopt two baselines: (1) standard episodic training ($T = 80$), and (2) periodic reset ($T = 320$) with episodic end-effector resets. As summarized in Figure 8, our results are consistent with those from the primary experiment (Figure 3). For a fixed number of interaction steps, the non-episodic method achieves performance competitive with the episodic baseline, while outperforming it by a large margin when constrained to a fixed reset budget.



(a) Independent task (#. interaction steps)

(b) Independent task (#. resets)

Figure 8. **Learning efficiency under varying reset budgets.** We configure a novel scene with a different set of objects (a toy bear and a plastic bottle) and receptacles (a newspaper and a carpet) from those used in the main experiment. We consider two baselines: standard episodic training ($T = 80$), and periodic reset ($T = 320$) with episodic end-effector resets. All models are trained using heuristic task ordering and evaluated on the training set of object configurations. **From left to right**: We evaluate independent-task manipulation relative to total interaction steps (a) and the total number of scene resets (b). Non-episodic baselines achieve superior reset efficiency.

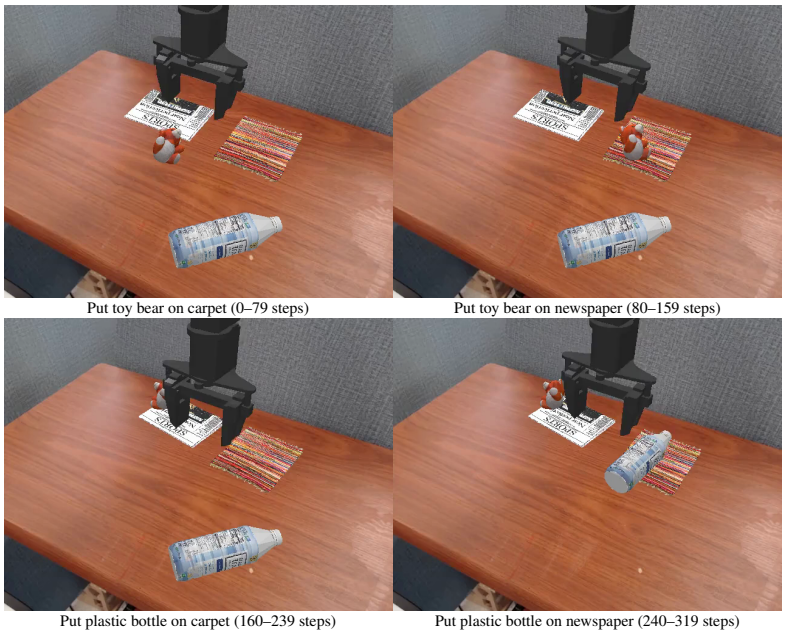


Figure 9. **Training scene for diverse object–receptacle configurations.** We apply the proposed training framework to diverse object–receptacle pairs within a shared scene, including different objects (toy bear, plastic bottle) and surfaces (carpet, newspaper). This demonstrates that the framework can be consistently applied across varying object and environment configurations.

C. RF-MTRL Algorithm

Our full algorithm, summarized in Algorithm 1, illustrates the complete training procedure of RF-MTRL. Training progresses over a continuous interaction stream and is structured around task-switch intervals. At each task switch, the agent may trigger one or more reset mechanisms according to the specified reset strategies. Environment resets are not tied to episode terminations, but are instead applied selectively, reflecting the reset-free training protocol under a limited reset budget.

Algorithm 1 Multi-task Reset-Free Reinforcement Learning (RF-MTRL)

Input: task set $\mathcal{T}$, policy π_θ, value V_ϕ
Input: total interaction steps S, task switch interval K, update interval U
Input: fixed reset interval H, reset budget B
(Optional) end-effector reset $\mathcal{R}_{\text{ee}}$, unsuitable-state detector $\mathbb{I}_{\text{bad}}(s)$
Initialize environment state s_0, step counter $n \leftarrow 0$
Initialize reset counter $b \leftarrow 0$, buffer $\mathcal{D} \leftarrow \emptyset$
while $n < S$ **and** $b < B$ **do**
  Task Switch: sample instruction $\tau \sim \textsc{TaskScheduler}(\mathcal{T})$
  for $t = 1$ to K **do**
    Observe o_t and instruction τ
    Sample action $a_t \sim \pi_\theta(a \mid o_t, \tau)$
    Step environment: $s \leftarrow f(s, a_t)$
    Store (o_t, τ, a_t, r_t, s) into buffer $\mathcal{D}$
    $n \leftarrow n + 1$
    if $(n \bmod U) = 0$ **then**
      Compute returns and advantages using V_ϕ
      Update policy π_θ and value V_ϕ on $\mathcal{D}$
      Clear buffer $\mathcal{D} \leftarrow \emptyset$
    end if
  end for
  If end-effector reset enabled then
    Apply end-effector reset $\mathcal{R}_{\text{ee}}$
  If unsuitable reset enabled and $\mathbb{I}_{\text{bad}}(s) = 1$ **then**
    reset invisible target; $b \leftarrow b + 1$
  If $(n \bmod H) = 0$ **then**
    Apply scene reset; $b \leftarrow b + 1$
end while

D. Hyperparameter details

We detail the training configuration and experiment setup used in our experiments. Table 2 reports the hyperparameters used across all experiments, covering PPO optimization, the vision-language-action (VLA) model configuration based on OpenVLA, and additional training settings related to reset-free multi-task learning.

Hyperparameter	**Value**
PPO	
PPO epochs per update	1
Clip ratio (ϵ)	0.2
Discount factor (γ)	0.99
GAE parameter (λ)	0.95
Entropy coefficient	0.0
Gradient accumulation steps	20
Vision-language-Action (VLA) model	
Backbone	OpenVLA-7B
LoRA rank	32
Optimizer	AdamW
Learning rate (VLA backbone)	1×10^{-4}
Learning rate (value head)	3×10^{-3}
Adam β_1, β_2	0.9, 0.999
Action sampling temperature (train)	1.0
Action sampling temperature (eval)	0.6
Other Training Settings	
Number of parallel environments	64
Training horizon (steps)	320
Instruction switch interval	80
Training update interval	160

Table 2. Training hyperparameters used in all experiments.

E. Heuristic task ordering

In our experiments, we specify a fixed task ordering: **putting the ketchup bottle on the yellow plate** $\rightarrow$ **putting the ketchup bottle on the cloth** $\rightarrow$ **putting the kitchen shovel on the yellow plate** $\rightarrow$ **putting the kitchen shovel on the cloth**. This ordering ensures that consecutive tasks are different, thereby reducing the likelihood of repeatedly executing an already successful task.

F. Training Process

We visualize the training dynamics of reset-free reinforcement learning under three different reset strategies: end-effector reset, learned scene reset, and heuristic scene reset. Figure 10 illustrates representative training trajectories for each strategy. These visualizations highlight how different reset mechanisms influence long-horizon interaction, scene stability, and training continuity in reset-free multi-task settings.

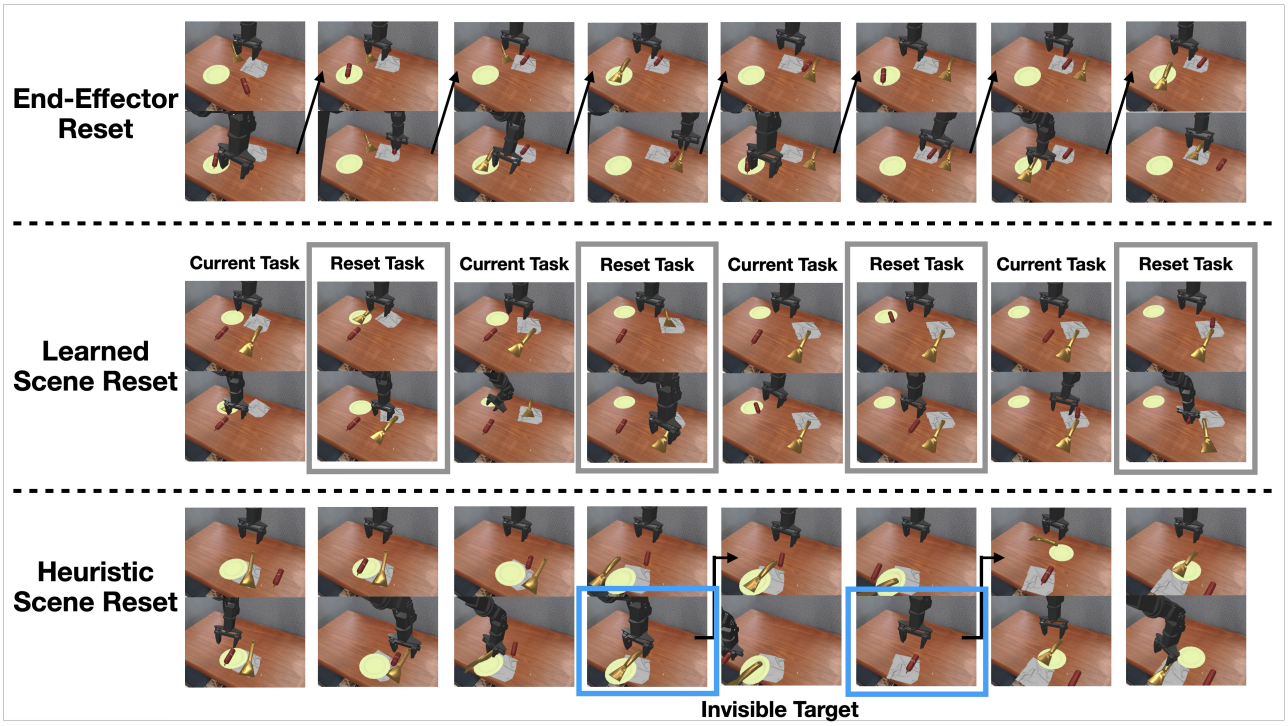


Figure 10. **Training dynamics under different reset strategies.** Visualization of training trajectories under three reset mechanisms: end-effector reset (top), learned scene reset (middle), and heuristic scene reset (bottom). End-effector reset restores the robot configuration while preserving the scene state, reducing distributional drift during long-horizon interaction. Learned scene reset alternates between task execution and explicit reset tasks, restoring scenes without human intervention. Heuristic reset selectively intervenes when the environment enters unrecoverable states (e.g., targets leaving the workspace), enabling continued training while avoiding excessive resets.

G. Full Experiment Result

In this section, we present the complete set of experimental results, including independent task success rates across all tasks and evaluation protocols. The results include analyses of learning efficiency under varying reset budgets (Figure 11), an ablation study of RF-MTRL design choices (Figure 12), robustness to out-of-distribution (OOD) object–receptacle configurations (Figure 13), and robustness to OOD task sequences (Figure 14).

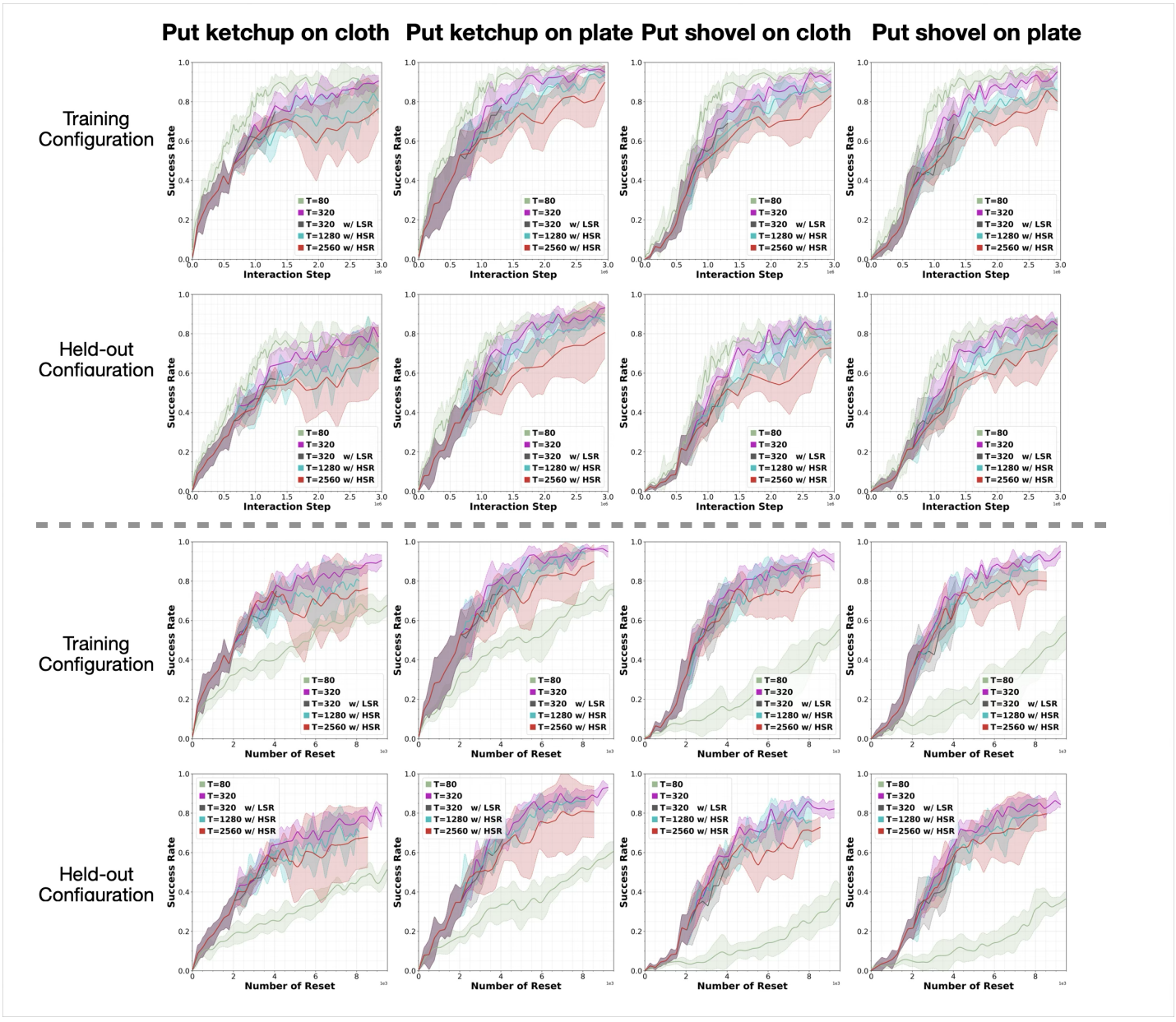


Figure 11. **Learning efficiency under varying reset budgets (Independent Task).** We compare four baselines: standard episodic training ($T = 80$), periodic reset ($T = 320$) with episodic end-effector resets, and periodic reset with longer horizons ($T \in \{1280, 2560\}$) combined with heuristic scene resets (HSR) and curriculum learning. All models are trained using heuristic task ordering and evaluated on the training set of object configurations. **Independent task evaluation**: each manipulation task is evaluated independently rather than sequentially within a shared rollout. **From left to right**: we report success rate as a function of total interaction steps (a) and the total number of scene resets (b). Non-episodic baselines achieve higher reset efficiency while maintaining comparable learning performance.

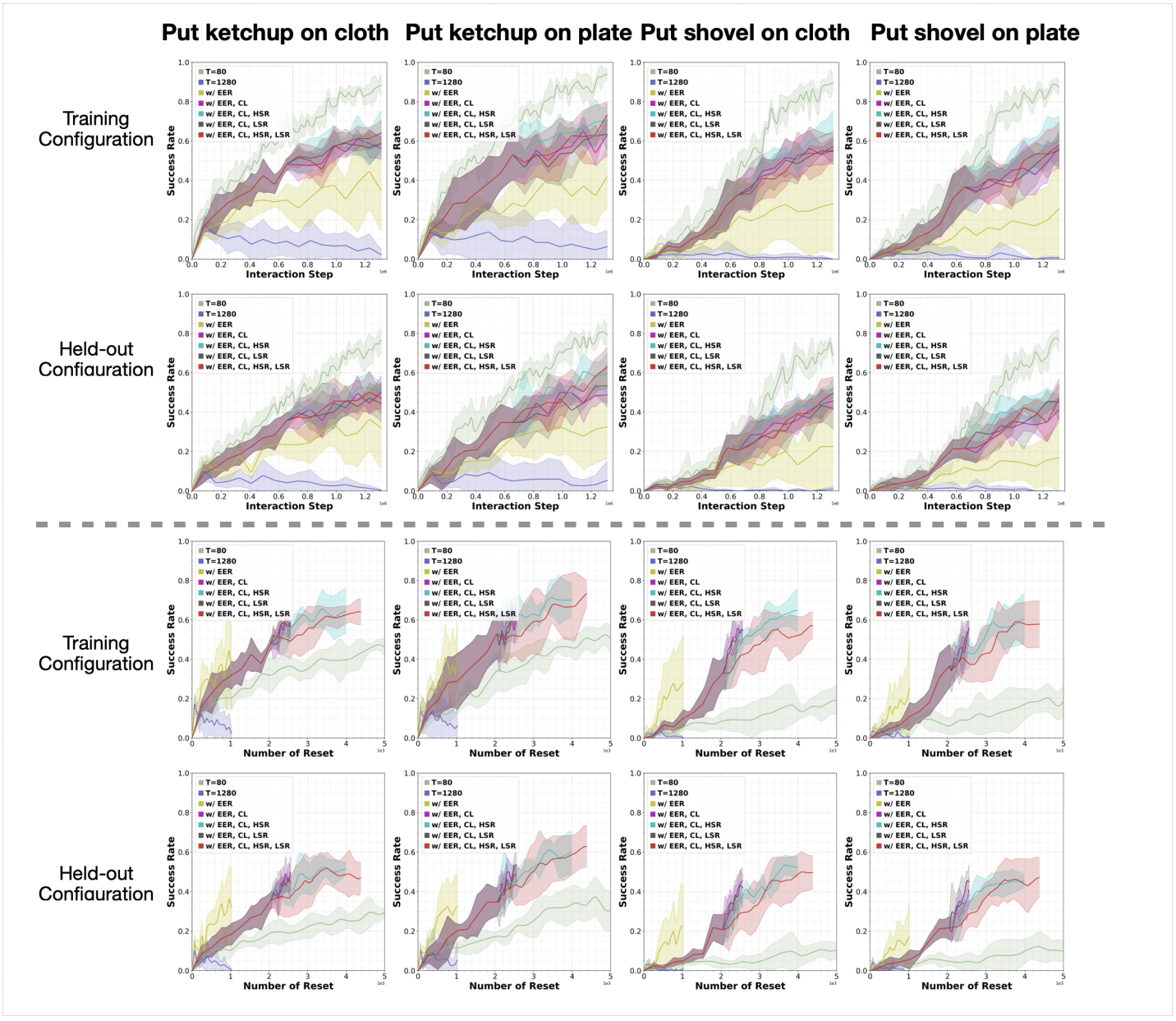


Figure 12. **Performance of RF-MTRL approaches (Independent Task).** We evaluate the effect of episodic end-effector resets (EER), curriculum learning (CL), heuristic scene resets (HSR), and learned scene resets (LSR) across four manipulation tasks. All models are trained for up to 1.3M interaction steps with five random seeds. Columns correspond to different object–receptacle tasks. Top rows show performance measured by interaction steps, while bottom rows measure reset efficiency with respect to the number of scene resets. Results are reported on both training configurations and held-out configurations. Combining EER, CL, and scene reset strategies consistently improves learning stability and reset efficiency across tasks.

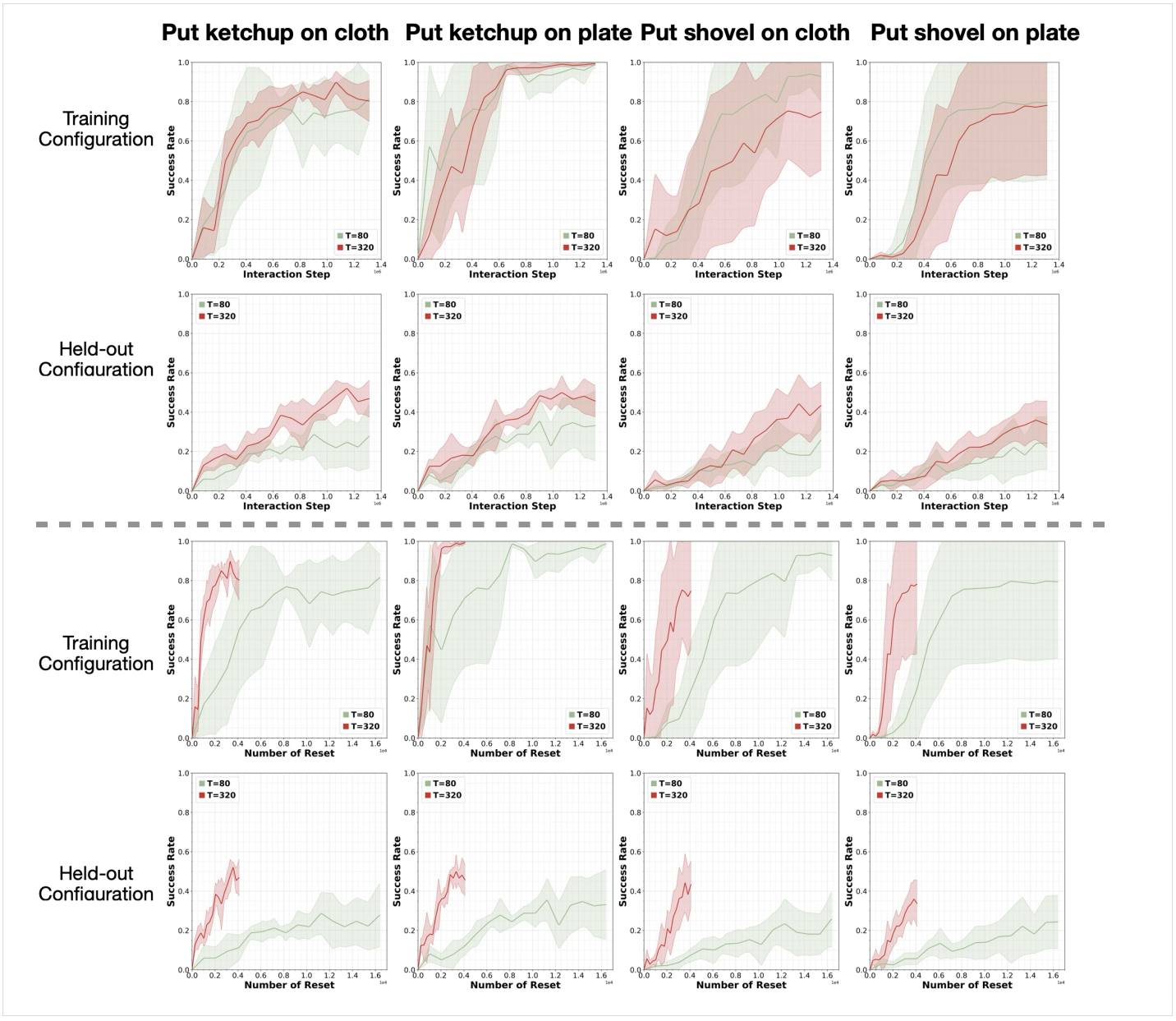


Figure 13. **Robustness to OOD object-receptacle configurations (Independent Task).** We compare an episodic baseline ($T = 80$) with a non-episodic baseline—periodic reset ($T = 320$) with episodic end-effector resets. Both models are evaluated using an independent task protocol, where each manipulation task is assessed independently rather than sequentially within a shared rollout. Performance is measured on the training set—consisting of either 64 (top row) or 1 (bottom row) object–receptacle configurations—and on a held-out set of 64 unseen configurations. When trained on fewer configurations, the periodic reset approach demonstrates stronger out-of-distribution (OOD) generalization compared to the episodic baseline.

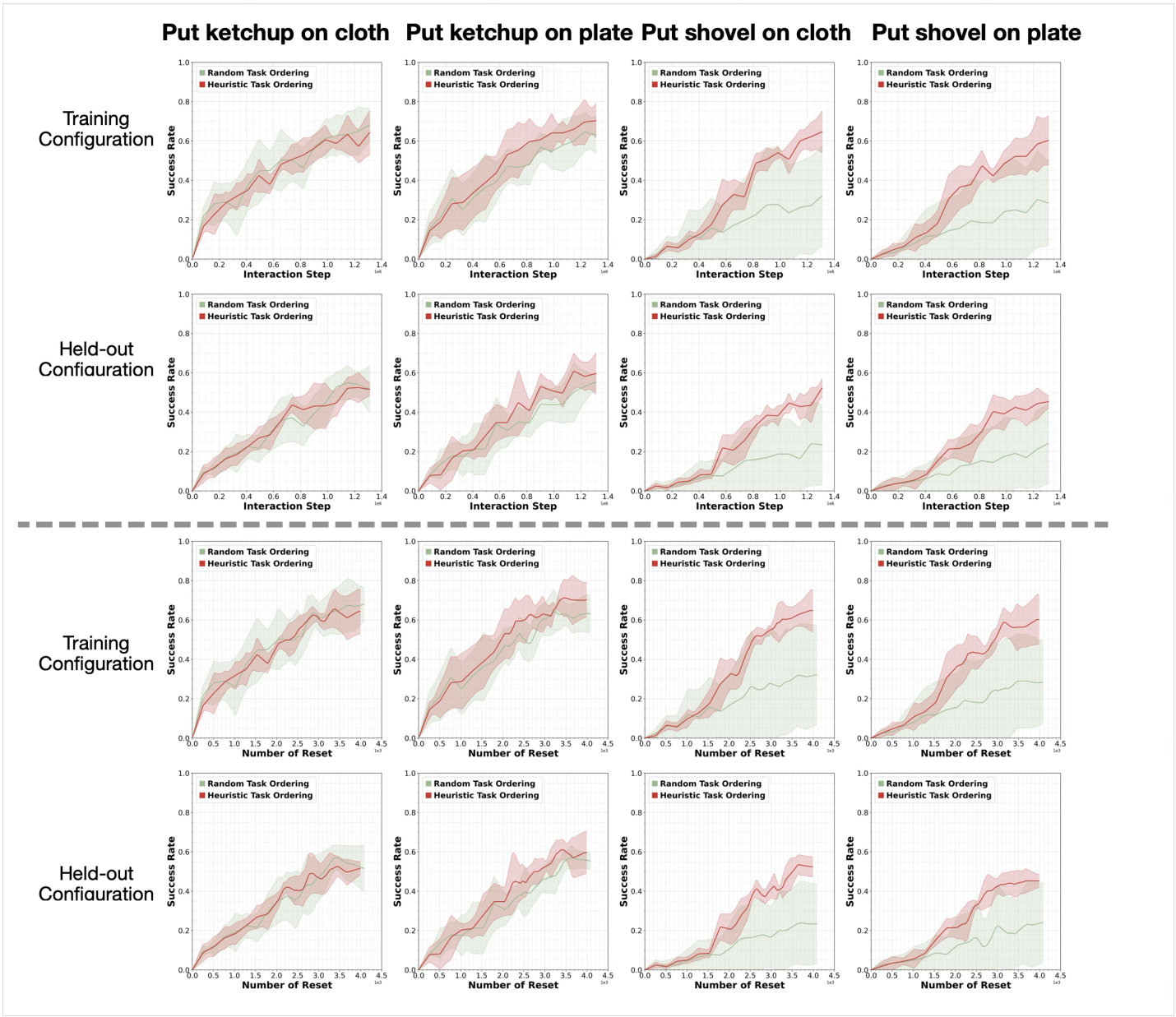


Figure 14. **Robustness to OOD task sequences (Independent Task).** We evaluate the impact of heuristic and random task ordering on learning performance. All models are trained with $T = 1280$ using heuristic scene resets (HSR). Both models are evaluated using an independent task protocol, where each manipulation task is assessed independently rather than sequentially within a shared rollout. Columns correspond to different object–receptacle tasks, while rows report results on the training configurations and held-out configurations. The top half measures success rate with respect to interaction steps, and the bottom half measures reset efficiency with respect to the number of scene resets. Heuristic task ordering consistently achieves higher success rates and demonstrates improved robustness to out-of-distribution (OOD) task sequences compared to random ordering.