In-Context Adaptation for Generalizable Imitation Learning

Junlin Xie[1], **Xu Luo**[1*], **Hao Wu**[1*], **Ji Zhang**[3], **Youguang Xing**[1], **Lianli Gao**[1], **Jingkuan Song**[2]
[1]UESTC, [2]Tongji University, [3]Southwest Jiaotong University
junlinxie0601@outlook.com

Abstract: While imitation learning on large-scale robot data produces robot policies with impressive task performance, these policies are typically *reactive* and lack the ability to adapt to novel conditions at test time. This limitation stands in stark contrast to Large Language Models (LLMs), which excel at in-context learning and adaptation. In this work, we take the first steps toward bridging this gap, exploring how imitation learning can instill in-context adaptation into robot policies. We specifically address the challenge of varying action dynamics, a scenario requiring online inference and adjustment. Our experiments with Diffusion Policy reveal that enabling such adaptation hinges on two critical components: conditioning the policy on histories of both observations and actions, and training on a diverse sampling of action dynamics. The resulting method successfully generalizes to unseen, out-of-distribution dynamics in context, representing a key advancement toward behavioral generalization in imitation learning.

Keywords: In-Context Adaptation, Imitation Learning, Zero-Shot Generalization

1 Introduction

The ability to swiftly adapt to new tasks and environments is a defining characteristic of human intelligence and a significant milestone in the pursuit of artificial general intelligence [1, 2]. While the current paradigm of imitation learning on large-scale expert data has produced generalist robot policies with impressive generalization capabilities [3, 4, 5, 6, 7, 8], these policies are predominantly reactive. They lack the memory and mechanisms to improve on the fly, thus often failing when faced with novel scenarios where reactive control is inadequate for generalization.

In contrast, Large Language Models (LLMs) have demonstrated a remarkable capacity not only for strong generalization from pretrained domain knowledge, but also for adapting to new tasks from experience—given a few interactions with environments [9]. Crucially, this adaptation occurs entirely "in-context" without requiring gradient-based weight updates, a process analogous to human episodic memory [10]. A natural question arises: *Can this powerful paradigm of in-context adaptation be unlocked for robotics?* The hope is that, by processing a recent context history of interactions, the policy can implicitly infer the environment's latent dynamics and then modulate its behavior accordingly.

In fact, the principle of in-context adaptation has been successfully leveraged to address the sim-to-real challenge in state-based robotic tasks, primarily through reinforcement learning (RL). In this line of work, the prevailing approaches [11, 12, 13, 14, 15] combine training on a wide distribution of simulated environments (domain randomization) with architectures that possess memory, such as LSTMs or Transformers. This enables the policy to learn an adaptation mechanism online, effectively adjusting to the discrepancies between simulation and reality. Nevertheless, applying this methodology to current vision-based systems like Vision-Language-Action (VLA) models [4] presents significant hurdles. Unlike the physical parameters targeted by domain randomization in

*: denotes equal contribution.

RemembeRL Workshop @ 9th Conference on Robot Learning (CoRL 2025), Seoul, Korea.

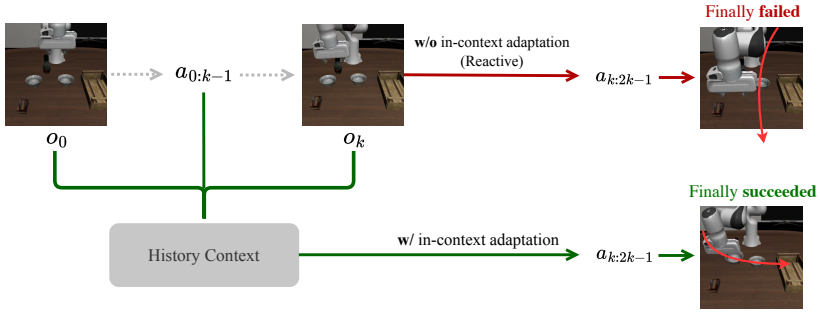

Figure 1: **An illustration of in-context adaptation.** When encountering a new environment with unknown dynamics, the policy with in-context adaptation leverages its history of observations ($o_{0:k}$) and actions ($a_{0:k-1}$) to infer the new dynamics and successfully adapt its subsequent actions ($a_{k:2k-1}$). In contrast, a policy without this capability fails to adjust its behavior.

robotics, the visual complexity of the real world creates a sim-to-real gap that is far more difficult to close [16]. Furthermore, the reliance on RL, which often requires millions of environmental interactions for convergence, makes training directly in the real world an impractical endeavor.

In this paper, we explore the possibility of directly instilling the capability for in-context adaptation into vision-based robot policies via imitation learning. Unlike RL, imitation learning is substantially more sample-efficient and inherently safer during data collection, rendering it a more viable pathway for real-world robotics. As an initial step, we design a controlled experimental setting where the model's generalization ability can be achieved only through online adaptation. Specifically, we introduce variations in the robot's action dynamics during evaluation, simulating a common deployment challenge where a policy operates under potentially different physical constraints for different trials. In this scenario, the robot must infer the underlying dynamics from its contextual history (i.e., recent observations and actions) and modulate its behavior accordingly to complete the given tasks. Within this framework, our investigation centers on two primary research questions: (1) How does history contexts influence the policy's adaptive performance? and (2) What characteristics must the training data possess to enable generalization to unseen action dynamics at test time?

Our results demonstrate that for a policy to adapt in context, it must have access to both historical observations and actions to identify the characteristics of a new environment. Furthermore, we find that the training data must be sufficiently randomized along the specific axes of variation to which the robot must adapt, enabling generalization to unseen dynamics. While this study is exploratory in nature, the results offer initial validation for the core hypothesis that imitation learning is sufficient for acquiring complex adaptive behaviors. This work provides the necessary groundwork for subsequent research to scale this methodology to larger datasets and a wider spectrum of real-world environmental variations.

2 Related Work

Recent efforts on imitation learning of robots focus on building large-scale robot datasets [17, 18, 19, 20, 21, 22] and the training of high-capacity models [17, 4, 23, 24, 25, 26, 27] on these datasets, especially Vision-Language-Action models [4, 26, 5, 28, 6, 7]. A critical characteristic of most of these models is that they are trained end-to-end for outputting task planning [29, 30] or raw low-level actions in response to immediate sensory observations without accessing histories of memory. This is partly due to long contexts of histories as input introduce comparable computation cost at inference, especially for large VLA models [21], and that recent works [31, 32] empirically found that image-conditioned policies degrade with history. However, this kind of "reactive" control can

be insufficient for generalization tasks where adaptation is required to figure out some unknown environmental factors or improve existing skills on the fly.

Introducing memories or context of high-dimensional observations like vision into robot policies has been a long-standing problem in robot learning. Under the assumption that historical observations are highly redundant, several works discard parts of the past information via adversarial regularization [33], information bottlenecks [34], or selecting salient subsets through techniques like keyframes [35] and motion tracks [36]. However, such methods may fail in temporally complex tasks. Some recent vision-based robot policies use consecutive history visual observations as inputs [31, 37, 17, 38], but they only use history to improve general performance and do not consider self-adaptation capabilities.

Recent works [39, 40] consider injecting in-context learning capabilities into vision-based robot policies through imitation learning. Different from in-context adaptation, the context of policies is filled with expert-level demonstrations instead of past experience of interactions. We argue that the ability of in-context adaptation is more demanding in real-world scenarios where demonstrations are hard to obtain, and is more natural for robots which can obtain knowledge from environment by themselves [9], which is more of a sign of intelligence.

Our work is closely related to Behavioral Exploration (BE) by Wagenmaker et al. [41], which applies the concept of in-context adaptation to imitation learning from offline data. Methodologically, both our approach and BE condition a transformer-based diffusion policy on a history of past observations to enable online adaptation. The fundamental distinction, however, lies in the purpose of this adaptation. BE adapts its policy based on its history to intentionally maximize coverage over the expert demonstration space and perform targeted exploration. Conversely, our work employs adaptation to infer the underlying dynamics from its contextual history and modulate its behavior accordingly to complete the given tasks, aiming for zero-shot generalization rather than comprehensive exploration.

3 Problem Setup

We frame our problem of in-context adaptation within the context of imitation learning in a distribution of environments, formalized as a collection of Markov Decision Processes (MDPs). Each task or episode corresponds to interacting with a specific MDP, $\mathcal{M}_\epsilon = (\mathcal{S}, \mathcal{A}, P_\epsilon, p_0)$, sampled from a family of MDPs parameterized by a latent dynamics variable $\epsilon \in \mathbb{R}^D$. At the beginning of each episode, ϵ is drawn from a fixed distribution and remains constant throughout the episode. All MDPs in this family share the same state space $\mathcal{S}$, action space $\mathcal{A}$, and initial state distribution p_0. Their distinction lies in the transition kernel, $P_\epsilon : \mathcal{S} \times \mathcal{A} \to \Delta_{\mathcal{S}}$, where the same state s and action a can potentially lead to different probabilities of future states.

The variation in dynamics is manifested as a perturbation on the actions. The policy, π, outputs an intended action a_t at timestep t. However, the action executed in the environment, a_t^{exec}, is a function of the policy's output and the latent variable for that episode: $a_t^{\text{exec}} = f(a_t, \epsilon)$. The environment then transitions to the next state according to this executed action, $s_{t+1} \sim P_\epsilon(s_t, a_t) \equiv P(s_t, a_t^{\text{exec}})$. This setup models various real-world scenarios where a discrepancy exists between the intended and executed action, such as errors from inverse kinematics or physical limitations of the robot.

The core challenge is that the dynamics parameter ϵ is latent, placing the policy in a partially observable environment where its visual observation o_t is only an incomplete view of the true state s_t. Consequently, a purely reactive policy $\pi(a_t|o_t)$ is insufficient. To succeed, the policy must be history-aware, using the sequence of past observations and actions to infer the latent dynamics parameter ϵ and adapt its behavior in-context. As an initial exploration, we instantiate this framework with a simple additive perturbation on the first action dimension (i.e., the x-direction displacement). Specifically, $a_t^{\text{exec}} = a_t + \epsilon$, where the noise vector is $\epsilon = [\epsilon_x, 0, \ldots, 0]^T$ with $\epsilon_x \sim \mathcal{U}[-\delta, \delta]$. The policy must use its interaction history to implicitly estimate ϵ_x and adjust its subsequent actions to counteract the perturbation.

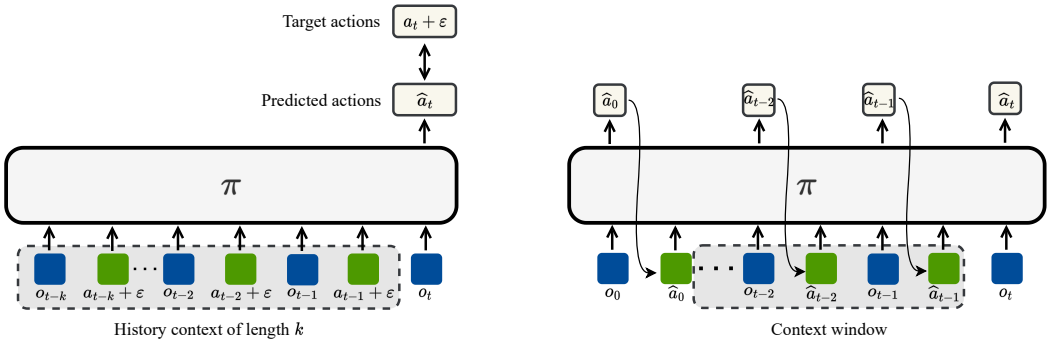

Figure 2: **Training and inference of our method that triggers in-context adaptation. (Left)** During training, the policy is given a history context of observations and perturbed actions ($a + \epsilon$) and learns to predict a compensated target action. **(Right)** During inference, the policy uses a context window of its own past actions and observations to continually adapt its behavior in an environment with unknown dynamics.

4 Method

Our goal is to train a policy that can adapt its behavior in-context to unseen action dynamics during deployment. To achieve this, we build upon a history-conditioned diffusion policy (§4.1) and introduce a specific training strategy that exposes the model to a wide distribution of action dynamics (§4.2). This enables the policy to learn to infer the latent dynamics from its recent interaction history and modulate its following actions accordingly.

4.1 History-Conditioned Diffusion Policy

We adopt Diffusion Policy (DP) [31], as it is renowned for its capacity to model complex, multi-modal action distributions. To effectively process historical context, a capability central to our goal and well-aligned with the architecture of large language models, we utilize a transformer as the backbone of our policy network. The policy, denoted as π_θ, is explicitly conditioned on a history of both past observations and past actions to predict a chunk of future actions. Formally, at each timestep t, the policy takes as input the sequence of the last $K+1$ observations, $o_{t-K:t}$, and the last K executed actions, $a_{t-K:t-1}$. It then outputs a distribution over the sequence of the next H actions, $a_{t:t+H-1}$, where H is known as the prediction horizon. The policy is trained as a conditional denoising diffusion model, learning to reverse a noising process that gradually corrupts expert actions into Gaussian noise. This training paradigm allows the policy to capture the underlying structure of the expert action distribution, conditioned on its recent interaction history.

4.2 Learn to Adapt In-Context

Our training strategy is designed to instill the capability for in-context adaptation by teaching the policy to infer and counteract latent environmental dynamics from a short history of interactions. To this end, we augment the training data by simulating a wide distribution of action dynamics, a principle analogous to domain randomization. This procedure compels the policy to actively use its history context to identify the specific dynamics of an episode and modulate its behavior accordingly, rather than overfitting to any single, deterministic condition.

The training process, illustrated in Figure 2 (Left), is structured as follows. For each expert trajectory $(\dots, o_t, a_t, \dots)$ sampled from the dataset $\mathcal{D}$, we first sample a latent dynamics parameter $\epsilon \in \mathbb{R}^D$ from a predefined distribution (e.g., $\epsilon \sim \mathcal{U}([-\delta, \delta]^D)$). This parameter remains fixed for the entire trajectory, representing a consistent but unobserved physical perturbation. The policy π_θ must learn to compensate for this dynamic. It is conditioned on a history context c_t comprising past observa-

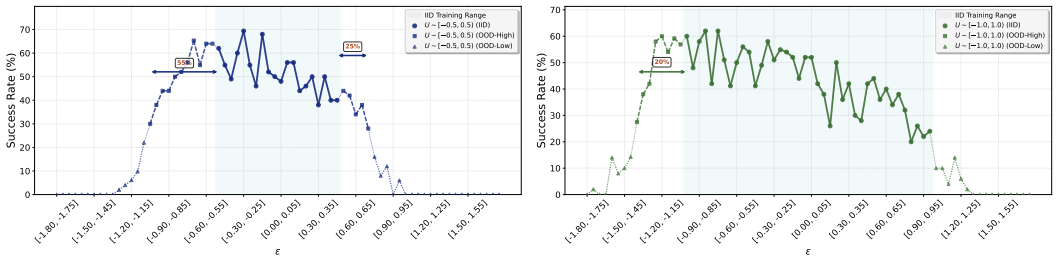

Figure 3: **Comparison of success rate when trained with different perturbation ranges.** In both scenarios, the policy is conditioned on a history of 2 observations and 1 past action. **(Left.) Policy A.** Training with a narrow perturbation range, $\epsilon \sim \mathcal{U}[-0.5, 0.5)$. **(Right.) Policy B.** Training with a wider perturbation range, $\epsilon \sim \mathcal{U}[-1, 1)$. The shaded region indicates the in-distribution range. We plot the i.i.d success rate (solid line) and the o.o.d success rate (bold dashed line), and the red percentages stand for relative extrapolation distance

tions and the *executed* actions that resulted from the perturbation: $c_t = (o_{t-K:t}, a^{\text{exec}}_{t-K:t-1})$. The historical actions fed to the policy are constructed by applying the perturbation to the ideal expert actions, i.e., $a^{\text{exec}}_i = a_i + \epsilon$.

The policy's objective is to produce a *compensated* action $\hat{a}_t$ that matches the perturbed expert action. Therefore, the training target for the policy is the compensated action $a^{\text{target}}_t = a_t + \epsilon$. The policy is trained to minimize the prediction error, governed by a loss function such as:

$$\mathcal{L}(\theta) = \mathbb{E}_{\tau\sim\mathcal{D},\epsilon\sim p(\epsilon)}\left[\sum_t \|\pi_\theta(c_t) - (a_t + \epsilon)\|^2\right]$$

By training over a diverse range of sampled ϵ, the model learns a general mechanism to implicitly infer the underlying dynamic from its history and produce the correctly compensated action.

During inference, as depicted in Figure 2 (Right), the policy is deployed in an environment with an unknown but fixed action dynamic ϵ_{test}. At each timestep t, the policy conditions on a sliding context window of its own recent interactions, comprising the latest observations and the actions it previously *commanded*: $c_t = (o_{t-K:t}, \hat{a}_{t-K:t-1})$. The policy then outputs the next action $\hat{a}_t$, which the environment executes. At the start of an episode, the history buffer is empty and the policy's behavior is unadapted. As the robot interacts with the world, the growing context c_t accumulates evidence about the effects of the unknown dynamic ϵ_{test}. This allows the policy to implicitly deduce the perturbation and progressively refine its outputs to better counteract it, thereby achieving the task goal. This adaptation occurs entirely in-context, without requiring any gradient-based weight updates.

5 Experiments and Analyses

To validate our approach, we conduct a series of simulation experiments to evaluate the in-context adaptation capabilities of our policy. Our experiments use LIBERO[42] benchmark, choosing the '*pick up the black bowl on the left and put it in the tray*' as our task. We process the expert data following the OpenVLA, filtering out unsuccessful trials and removing no-op actions, which yields a final dataset of 49 trajectories.

For training, we adopt the hyperparameter configuration of DP but use only third-person images, ensuring partial observability. A dedicated projector aligns the actions in the history with the transformer's input space. During inference, the policy generates a compensated action $\hat{a}$. To simulate unknown dynamics in the environment, we execute the action $\hat{a} - \epsilon$ in the simulator, where ϵ is the perturbation for that episode. The simulator which can execute precise actions combined with sampled perturbations jointly build a realistic action dynamic setting.

When evaluating generalization, as a policy's success rate depends heavily on the perturbation range used during training and testing, it is misleading to compare them directly. To overcome this, we introduce a normalized metric called relative extrapolation distance. This metric quantifies to what extent the extrapolation can be extended beyond the training distribution when out-of-distribution success rates fall to 50% average in-distribution ones. This metric is a strong indicator of a policy's adaptability, as it measures how well it withstands increasingly severe perturbations.

We begin by presenting the generalization of our in-context adaptation strategy in §5.1, and then answer the two questions raised in §1. We investigate what training data characteristics are required for generalization (Q_2) in §5.2, and how the history context influences the policy's performance (Q_1) in §5.3 and §5.4.

5.1 In-Context Adaptation can Generalize well

To evaluate whether our training strategy endows the policy with a generalizable in-context adaptation capability, we investigate its ability to extrapolate to action dynamics. Therefore, we train two policies, policy A and policy B, on different uniform distributions of the perturbation parameter ϵ. Our goal is to determine if the policies can generalize to values of ϵ they have never encountered.

The results, presented in Fig. 3, demonstrate that both policies achieve stable in-distribution success rates of approximately 40%, confirming the effectiveness of our training method within the seen dynamics. More importantly, both policies exhibit a remarkable capacity to extrapolate to out-of-distribution fields. policy A achieves a relative extrapolation distance of 55%, while policy B achieves 20%. Intriguingly, although policy B is exposed to a wider training distribution, its relative extrapolation distance is considerably smaller than that of policy A. This finding suggests that a broader training distribution does not necessarily lead to better extrapolation performance. We will further analyze this in §5.4.

In the preceding experiments, the action dynamic parameter, ϵ, was held constant throughout each test episode. This setup represents an idealized scenario, however in real-world applications, a robot's dynamics can fluctuate within a single task execution due to various factors. For instance, prolonged motor operation can lead to increased thermal noise, introducing variations of dynamic. Similarly, mechanical components may experience a "running-in" period, where friction changes as parts become more lubricated over time, causing a drift in dynamic.

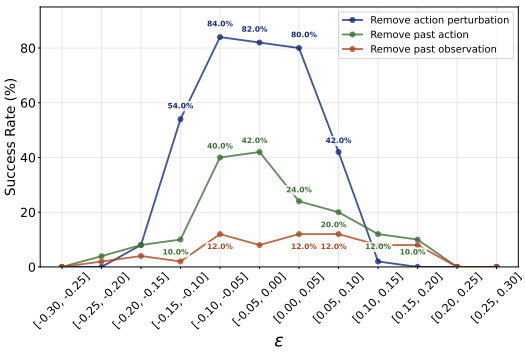

Figure 4: **History components ablation study.** Ablation study on the performance impact of removing key components during training: past actions $a^{\text{exec}}_{t-K:t-1}$, past observations $o_{t-K:t-1}$ and action perturbations ϵ. The x-axis stands for the range of perturbation ϵ.

To assess our policy's robustness in more realistic conditions, we designed experiments where the action dynamics are no longer static but vary at each timestep. We evaluated the policy under two time-varying dynamic distributions, Gaussian noise which simulates the effect of continuous, random electronic or thermal noise, and power-law drift that models a decaying drift, akin to a mechanical running-in process. Specially, in each timestep t, the perturbation follows $\epsilon_t \sim \mathcal{N}(0, 0.3^2)$ in Gaussian sampling and $\epsilon_t = -0.2 \cdot t^{-1/4}$ in power-law drift. Remarkably, despite never being exposed to time-varying dynamics during training, the policy achieves a success rate of 46.7% under the Gaussian noise and 55.1% under the power-law drift. These results strongly indicate that our training strategy enables the policy to learn a true, generalizable adaptation mechanism.

5.2 Ablation for the History Components

In this section, we investigate the essential characteristics of training data required to enable in-context adaptation. Our central hypothesis is that two factors are critical: the completeness of the history and the diversity of dynamics experienced during training.

First, for the policy to infer the latent action dynamics, the history context, $c_t = (o_{t-K:t}, a^{\text{exec}}_{t-K:t-1})$, must contain all necessary information. The sequence of past observations, $o_{t-K:t}$, implicitly captures the state transitions that result from the executed actions. Meanwhile, the past executed actions, $a^{\text{exec}}_{t-K:t-1}$, represent the robot's intended actions within its logical action space. A discrepancy between the intended action and the actual state transition reveals the effect of the underlying action dynamics, ϵ. The absence of either past actions or past observations would prevent the policy from correlating its commands with their outcomes, thereby making it impossible to infer the latent dynamics. As shown in Fig. 4, policies fail to generalize if they are not given access to the history of past actions or observations. Despite being trained on a wide $(-0.5, 0.5)$ perturbation range, they cannot handle even the small disturbances within $(-0.1, 0.1)$.

Second, to ensure the policy can adapt to arbitrary action dynamics when deployment, it must be exposed to a sufficiently diverse range of dynamics during training, like domain randomization. By training the policy on trajectories perturbed by a wide distribution of ϵ, we compel the model to learn a general adaptation mechanism rather than rote responses to some perturbations. For instance, when trained without any dynamic variations, the policy performs well only near zero perturbation and fails to generalize beyond a narrow $(-0.15, 0.1)$ range.

5.3 History Context: More isn't always Better

In our previous experiments, the history context was consistently set to two observations and one past action. A natural question arises: would a longer history context, which provides more information, enhance the policy's ability to infer latent dynamics? To investigate this, we explored how varying the length of the history context affects adaptation performance.

First, we examined the standard single-variable additive perturbation $a^{\text{exec}} = a + \epsilon$. Intuitively, a longer history might be expected to improve performance. However, as shown in the Fig. 5(**Left**) , we observed that extending the history context did not yield better results. In fact, some policies exhibited a slight performance degradation. This finding suggests that for simple, single-variable dynamics, a short history already provides sufficient information for the policy to infer the dynamic. In this case, additional history context may introduce unnecessary complexity and hinder the network optimization.

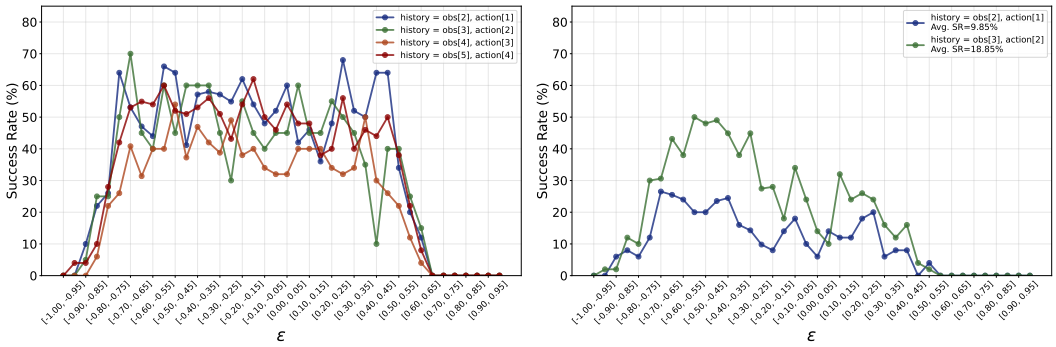

Figure 5: **Left.** Comparison of in-context adaptation training strategy with different history context size. A longer history context does not bring performance improvement. **Right.** (Dual variable perturbation, e.g. $k \times a + b$, etc.) However, when incorporating more than one variable, it performs better.

To further test this hypothesis, we designed a more challenging scenario with a two-variable perturbation, defined as $a^{\text{exec}} = k \times a + b$. Here, the policy must infer both a multiply factor (k) and an additive bias (b). To identify two unknown variables, the policy must implicitly solve a set of equations, which requires more data points from a longer interaction history. As predicted, the results for this task, shown in the right panel of Fig. 5, demonstrate that a longer history context leads to better in-context adaptation. This confirms that the optimal history length is coupled with the complexity of the latent dynamics. While a short context is sufficient for simple dynamics, a longer one becomes crucial for adapting to more complex environmental variations.

5.4 Not all ϵ works well

In our experiments, we observed that the generalization performance of the policy is sensitive to the range of the perturbation ϵ. This suggests that the relationship between the magnitude of the action, a, and the perturbation, ϵ, is a critical factor for adaptation. We hypothesize that a significant mismatch in scale between these two signals can degrade the learning process.

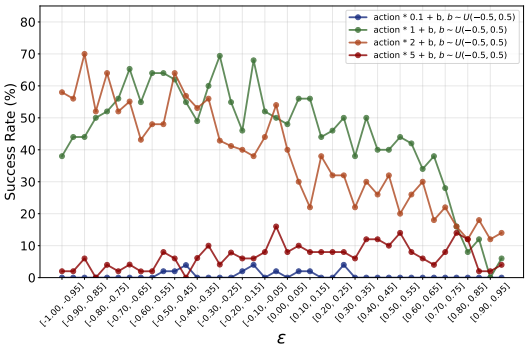

Figure 6: **The effect of the action-to-perturbation magnitude ratio on in-context adaptation.**

To investigate this hypothesis, we designed an experiment to analyze the impact of the relative scale between actions and perturbations. We scaled the expert action data by factors of $0.1, 1, 2, 5$, respectively, and trained new policies on these modified datasets while keeping the perturbation distribution fixed at $\epsilon \sim \mathcal{U}[-0.5, 0.5)$. As illustrated in Fig. 6, when the action magnitudes were either significantly smaller ($0.1\times$) or larger ($5\times$) than the perturbation range, the policies failed to learn almost entirely, with success rates dropping to near zero. In contrast, the policy trained with the original action data ($1\times$) achieved robust performance.

This result highlights that the effective learning of in-context adaptation is contingent on a balanced signal-to-noise ratio between the actions and the dynamic perturbations. When the perturbation ϵ is excessively large relative to the action a, it dominates the training target $a + \epsilon$, causing the policy to neglect the underlying action signal. Conversely, when ϵ is too small, it may be treated as negligible noise, preventing the policy from learning the adaptation mechanism. Therefore, for the policy to effectively infer and counteract unknown dynamics, the training data must present the action signal and the perturbation signal at comparable scales. This finding underscores the importance of careful data normalization and curriculum design when training policies for in-context adaptation.

6 Conclusion

Our work demonstrates that imitation learning can instill in-context adaptation into robot policies, enabling them to generalize to novel action dynamics at test time. The key findings indicate that this capability is contingent upon two critical components: the policy must be conditioned on a complete history of both observations and actions, and it must be trained across a diverse sampling of action dynamics. The resulting policy not only adapted to unseen, fixed perturbations but also successfully generalized to time-varying dynamics. We provide foundational evidence that imitation learning is sufficient for acquiring complex adaptive behaviors, paving the way for future research to scale this methodology to more factors and more varied real-world applications.

References

[1] J. Bauer, K. Baumli, F. Behbahani, A. Bhoopchand, N. Bradley-Schmieg, M. Chang, N. Clay, A. Collister, V. Dasagi, L. Gonzalez, et al. Human-timescale adaptation in an open-ended task space. In *International Conference on Machine Learning*, pages 1887–1935. PMLR, 2023.

[2] M. R. Morris, J. Sohl-Dickstein, N. Fiedel, T. Warkentin, A. Dafoe, A. Faust, C. Farabet, and S. Legg. Position: Levels of agi for operationalizing progress on the path to agi. In *Forty-first International Conference on Machine Learning*, 2024.

[3] J. Gao, S. Belkhale, S. Dasari, A. Balakrishna, D. Shah, and D. Sadigh. A taxonomy for evaluating generalist robot policies. *arXiv preprint arXiv:2503.01238*, 2025.

[4] B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR, 2023.

[5] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

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

[7] P. Intelligence, K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

[8] P. Atreya, K. Pertsch, T. Lee, M. J. Kim, A. Jain, A. Kuramshin, C. Eppner, C. Neary, E. Hu, F. Ramos, et al. Roboarena: Distributed real-world evaluation of generalist robot policies. *arXiv preprint arXiv:2506.18123*, 2025.

[9] A. Setlur, M. Y. Yang, C. Snell, J. Greer, I. Wu, V. Smith, M. Simchowitz, and A. Kumar. e3: Learning to explore enables extrapolation of test-time compute for llms. *arXiv preprint arXiv:2506.09026*, 2025.

[10] J.-A. Li, C. Zhou, M. Benna, and M. G. Mattar. Linking in-context learning in transformers to human episodic memory. *Advances in neural information processing systems*, 37:6180–6212, 2024.

[11] O. M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, et al. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020.

[12] I. Akkaya, M. Andrychowicz, M. Chociej, M. Litwin, B. McGrew, A. Petron, A. Paino, M. Plappert, G. Powell, R. Ribas, et al. Solving rubik's cube with a robot hand. *arXiv preprint arXiv:1910.07113*, 2019.

[13] A. Kumar, Z. Fu, D. Pathak, and J. Malik. Rma: Rapid motor adaptation for legged robots. *arXiv preprint arXiv:2107.04034*, 2021.

[14] I. Radosavovic, T. Xiao, B. Zhang, T. Darrell, J. Malik, and K. Sreenath. Real-world humanoid locomotion with reinforcement learning. *Science Robotics*, 9(89):eadi9579, 2024.

[15] I. Radosavovic, S. Kamat, T. Darrell, and J. Malik. Learning humanoid locomotion over challenging terrain. *arXiv preprint arXiv:2410.03654*, 2024.

[16] C. Li, R. Zhang, J. Wong, C. Gokmen, S. Srivastava, R. Martín-Martín, C. Wang, G. Levine, W. Ai, B. Martinez, et al. Behavior-1k: A human-centered, embodied ai benchmark with 1,000 everyday activities and realistic simulation. *arXiv preprint arXiv:2403.09227*, 2024.

[17] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

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

[19] H. Bharadhwaj, J. Vakil, M. Sharma, A. Gupta, S. Tulsiani, and V. Kumar. Roboagent: Generalization and efficiency in robot manipulation via semantic augmentations and action chunking. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4788–4795. IEEE, 2024.

[20] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

[21] A. O'Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain, 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)*, pages 6892–6903. IEEE, 2024.

[22] Q. Bu, J. Cai, L. Chen, X. Cui, Y. Ding, S. Feng, S. Gao, X. He, X. Huang, S. Jiang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv:2503.06669*, 2025.

[23] S. Liu, L. Wu, B. Li, H. Tan, H. Chen, Z. Wang, K. Xu, H. Su, and J. Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. *arXiv preprint arXiv:2410.07864*, 2024.

[24] R. Doshi, H. Walke, O. Mees, S. Dasari, and S. Levine. Scaling cross-embodied learning: One policy for manipulation, navigation, locomotion and aviation. *arXiv preprint arXiv:2408.11812*, 2024.

[25] L. Wang, X. Chen, J. Zhao, and K. He. Scaling proprioceptive-visual learning with heterogeneous pre-trained transformers. *Advances in Neural Information Processing Systems*, 37: 124420–124450, 2024.

[26] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

[27] A. Mandlekar, Y. Zhu, A. Garg, J. Booher, M. Spero, A. Tung, J. Gao, J. Emmons, A. Gupta, E. Orbay, et al. Roboturk: A crowdsourcing platform for robotic skill learning through imitation. In *Conference on Robot Learning*, pages 879–893. PMLR, 2018.

[28] K. Pertsch, K. Stachowicz, B. Ichter, D. Driess, S. Nair, Q. Vuong, O. Mees, C. Finn, and S. Levine. Fast: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025.

[29] M. Zawalski, W. Chen, K. Pertsch, O. Mees, C. Finn, and S. Levine. Robotic control via embodied chain-of-thought reasoning. *arXiv preprint arXiv:2407.08693*, 2024.

[30] L. X. Shi, B. Ichter, M. Equi, L. Ke, K. Pertsch, Q. Vuong, J. Tanner, A. Walling, H. Wang, N. Fusai, et al. Hi robot: Open-ended instruction following with hierarchical vision-language-action models. *arXiv preprint arXiv:2502.19417*, 2025.

[31] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, page 02783649241273668, 2023.

[32] R. Zheng, Y. Liang, S. Huang, J. Gao, H. Daumé III, A. Kolobov, F. Huang, and J. Yang. Tracevla: Visual trace prompting enhances spatial-temporal awareness for generalist robotic policies. *arXiv preprint arXiv:2412.10345*, 2024.

[33] C. Wen, J. Lin, T. Darrell, D. Jayaraman, and Y. Gao. Fighting copycat agents in behavioral cloning from observation histories. *Advances in Neural Information Processing Systems*, 33: 2564–2575, 2020.

[34] S. Seo, H. Hwang, H. Yang, and K.-E. Kim. Regularized behavior cloning for blocking the leakage of past action information. *Advances in Neural Information Processing Systems*, 36: 2128–2153, 2023.

[35] C. Wen, J. Lin, J. Qian, Y. Gao, and D. Jayaraman. Keyframe-focused visual imitation learning. *arXiv preprint arXiv:2106.06452*, 2021.

[36] J. Ren, P. Sundaresan, D. Sadigh, S. Choudhury, and J. Bohg. Motion tracks: A unified representation for human-robot transfer in few-shot imitation learning. *arXiv preprint arXiv:2501.06994*, 2025.

[37] H. Huang, F. Liu, L. Fu, T. Wu, M. Mukadam, J. Malik, K. Goldberg, and P. Abbeel. Otter: A vision-language-action model with text-aware visual feature extraction. *arXiv preprint arXiv:2503.03734*, 2025.

[38] O. M. Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, T. Kreiman, C. Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024.

[39] L. Fu, H. Huang, G. Datta, L. Y. Chen, W. C.-H. Panitch, F. Liu, H. Li, and K. Goldberg. In-context imitation learning via next-token prediction. *arXiv preprint arXiv:2408.15980*, 2024.

[40] K. Sridhar, S. Dutta, D. Jayaraman, and I. Lee. Ricl: Adding in-context adaptability to pre-trained vision-language-action models. *arXiv preprint arXiv:2508.02062*, 2025.

[41] A. Wagenmaker, Z. Zhou, and S. Levine. Behavioral exploration: Learning to explore via in-context adaptation. In *Forty-second International Conference on Machine Learning*, 2025.

[42] B. Liu, Y. Zhu, C. Gao, Y. Feng, Q. Liu, Y. Zhu, and P. Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023.