Self-Guided Action Diffusion

Rhea Malhotra
Stanford University
rheamal@stanford.edu

Yuejiang Liu
Stanford University
yuejiang.liu@stanford.edu

Chelsea Finn
Stanford University
cbfinn@stanford.edu

***Abstract*—Recent works have shown the promise of inference-time search over action samples for improving generative robot policies. In particular, optimizing cross-chunk coherence via bidirectional decoding has proven effective in boosting the consistency and reactivity of diffusion policies. However, this approach remains computationally expensive as the diversity of sampled actions grows. In this paper, we introduce *self-guided action diffusion*, a more efficient variant of bidirectional decoding tailored for diffusion-based policies. At the core of our method is to guide the proposal distribution at each diffusion step based on the prior decision. Experiments in simulation tasks show that the proposed self-guidance enables near-optimal performance at negligible inference cost. Notably, under a tight sampling budget, our method achieves up to 70% higher success rates than existing counterparts on challenging dynamic tasks. See project website here.**

I. Introduction

Imitation learning from large-scale human demonstrations has shown great promise in developing generalist robot policies. Notably, recent policies have demonstrated remarkable capabilities in solving challenging tasks across environments [4, 5, 25, 6, 3]. However, as the volume of demonstrations increases, two key challenges emerge: (i) inherent behavioral diversity among demonstrations [24, 12, 16, 1], and (ii) complex action dependencies spanning multiple time steps [28, 7].

To address these challenges, existing methods often model the distribution of action chunks, aiming to capture temporal dependencies within each chunk of demonstrations Lee et al. [14], Zhao et al. [28], Chi et al. [7]. Yet, when dependencies extend beyond individual chunks, maintaining cross-chunk consistency remains difficult. Liu et al. [15] have recently proposed to tackle these cross-chunk dependencies via test-time search. Despite its effectiveness, this decoding strategy becomes computationally inefficient as the diversity of sampled actions grows.

In this paper, we introduce Self-Guided Action Diffusion (Self-GAD), a more efficient test-time inference method by intervening in the proposal distribution. At the core of our method is a guided diffusion objective that leverages previous action predictions to balance exploration and exploitation. Empirically, our method achieves near-optimal performance with fewer samples, especially when demonstrations exhibit high diversity. Under tight sampling budgets, our method attains 70% higher success rates than competitive baselines on challenging manipulation tasks.

II. Related Work

a) Temporal Dependencies: Existing inference-time methods, including receding horizon, attempt to balance temporal and dynamic factors by employing intermediate action horizons. Temporal ensembling techniques enhance temporal coherence by averaging action chunk predictions over time, but earlier prediction averages become obsolete in rapidly changing contexts [15]. Modeling multimodal action distributions hinder policy learning with oscillatory behaviors [9, 17, 20]. While diffusion-based policies capture multimodal distributions, they struggle to generate smooth, temporally consistent trajectories in single-sample and single-action-horizon settings, critical for closed-loop stability.

b) Policy Steering: Inference-time policy adaptation is critical for aligning robotic policies with task objectives. Inference-Time Policy Steering (ITPS) optimizes sample alignment with user intent while preserving constraints within the data manifold [23]. Techniques like trajectory sketches, point goals, and physical corrections dynamically adjust policies to mitigate distribution shifts. Others re-rank actions via offline RL-derived value functions to improve robustness despite noisy training [18]. Classifier-guided sampling steers generation via gradient-based optimization, leveraging diffusion model features to enhance performance [8, 26, 10]. Gradient-based steering in diffusion denoising improves semantic segmentation through stop-gradient operations and enhances robotic planning by incorporating physics-informed diffusion steps [11, 22]. We introduce latent guidance for dynamically steering closed-loop inference using prior knowledge, optimizing performance by adapting proposal distributions. By modulating prior weight, our approach narrows distributions when reusability is high and broadens them to encourage exploration when reusability is low, achieving near-optimal performance while significantly reducing sample requirements at inference.

III. Approach

Our problem formulation involves a dataset of demonstrations $D = \{\tau_i\}_{i=1}^{N}$, where each trajectory τ_i consists of state-action pairs:

$$\tau_i = \{(s_1, a_1), (s_2, a_2), \ldots, (s_T, a_T)\}.$$

At each time step t, the demonstrated action a_t is influenced by both the observed state s_t and latent variables z_t. Action chunking captures the joint distribution of future actions

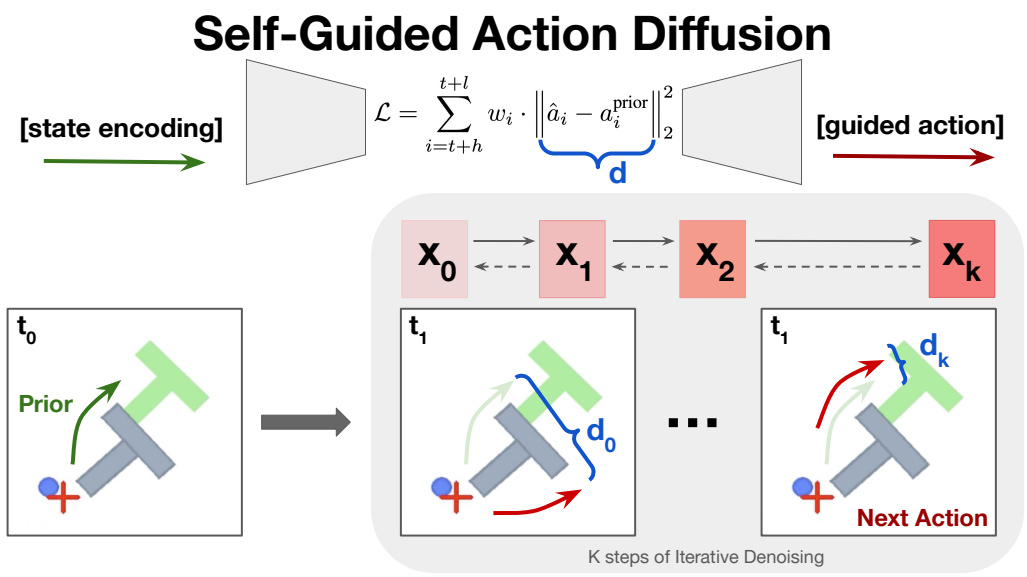

Fig. 1: **Self-Guided Action Diffusion:** We compute a weighted Euclidean distance between predicted and prior trajectories, applying gradients to guide model outputs interpolated between two denoising paths. The green arrow represents the prior action, while the red arrow denotes the predicted action iteratively refined through diffusion denoising, with guidance applied at each step.

conditioned on past states, where c is the *context length* (past states) and l is the *prediction length* (future actions) [28, 27]:

$$\pi(a_{t:t+l} \mid s_{t-c:t}).$$

The policy minimizes the divergence between a learned policy π and the expert policy π^*:

$$\pi = \arg\min_{\pi} \sum_{\tau \in D} \sum_{s_{t-c:t}} \sum_{a_{t:t+l}} L\big(\pi(a_{t:t+l} \mid s_{t-c:t}), \pi^*(a_{t:t+l} \mid s_{t-c:t})\big). \quad (1)$$

At deployment, h steps of the predicted action sequence are executed, with $h \in [1, l]$, forming a (c, h)-policy.

Denoising diffusion generates samples from the data distribution $p_{\text{data}}(s, a)$ by iteratively denoising a sample of pure white noise. The process involves diffusing $p_{\text{data}}(s, a)$ into a sequence of smoothed densities:

$$p(s, a; \sigma) = p_{\text{data}}(s, a) * \mathcal{N}(s, a; 0, \sigma^2 I).$$

For large $\sigma_{\max}$:

$$p(s, a; \sigma_{\max}) \approx \mathcal{N}(s, a; 0, \sigma_{\max}^2 I),$$

The sample is evolved backward to lower noise levels using a probability flow ODE:

$$\frac{dx_{t:t+l}}{d\sigma} = -\sigma \nabla_{x_{t:t+l}} \log p(x_{t:t+l}; \sigma) d\sigma,$$

which maintains the property $x_{t:t+l} \sim p(x_{t:t+l}; \sigma)$ for every $\sigma \in [0, \sigma_{\max}]$. Upon reaching $\sigma = 0$, we obtain

$$x_{t:t+l} \sim p(x_{t:t+l}; 0) = p_{\text{data}}(x_{t:t+l}).$$

The ODE is solved numerically by stepping along the trajectory defined by Equation (1). At each step, we evaluate the score function $\nabla_{x_{t:t+l}} \log p(x_{t:t+l}; \sigma)$, which can be approximated using a neural network $D_\theta(x_{t:t+l}; \sigma)$, trained for denoising:

$$\theta = \arg\min_{\theta} \mathbb{E}_{y \sim p_{\text{data}}, \sigma \sim p_{\text{train}}, n \sim \mathcal{N}(0, \sigma^2 I)} \| D_\theta(y + n; \sigma) - y \|_2^2.$$

Our hypothesis is that prior reusability estimates improve decoding strategies, optimizing both proposal distributions and sample efficiency. By leveraging gradients from the policy's likelihood function, actions are guided towards prior predictions (Figure 1).

We exclude the boundary actions outside the horizon steps and compute the loss to penalize deviations between the prior and current state-action trajectories, ensuring smooth transitions in action execution. We apply a weighted gradient update to the predicted states and actions, where the guidance weight β modulates the influence of the prior. This weight is tuned via grid search to balance adherence to prior trajectories with flexibility for adaptation.

$$(\hat{s}_t, \hat{a}_t) \leftarrow (\hat{s}_t, \hat{a}_t) + \beta \nabla_{(\hat{s}_t, \hat{a}_t)} \mathcal{L}$$

The trajectory deviation loss function $\mathcal{L}$ applies exponentially decaying weights to each timestep in the overlapping region between the generated trajectory and the prior:

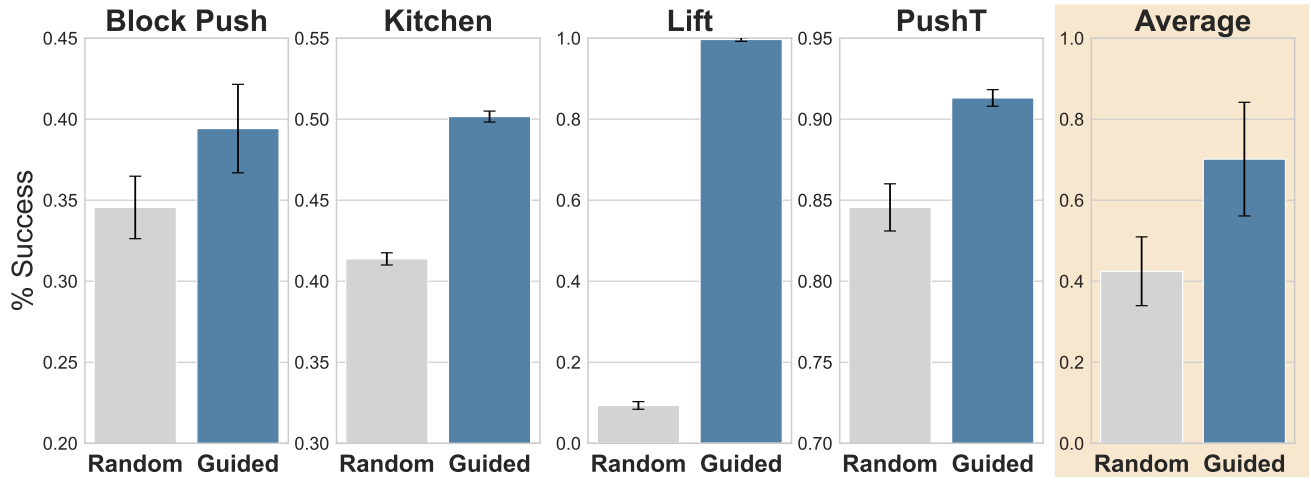

Fig. 2: **Comparison of Sampling Methods in Single-Sample Settings:** Self-Guided Action Diffusion outperforms Random sampling in single-sample, single-horizon tasks, achieving an average success rate 71.4% higher across all Robomimic benchmarks.

$$\mathcal{L} = \sum_{i=t+h}^{t+l} w_i \cdot \left\| \hat{a}_i - a_i^{\text{prior}} \right\|_2^2, \quad \text{where} \quad w_i = 0.5^{i-(t+h)}$$

Diffusion flow-matching architectures similarly sample actions through iterative denoising, integrating a learned velocity field that guides samples toward the data manifold. We explore the robotic foundation model, GR00T-N1, which leverages a Diffusion Transformer (DiT). The denoising tokens are conditioned on proprioceptive state and action history, cross-attended with multimodal visual and textual embeddings from the Eagle-2 vision-language model (VLM) to predict denoised motor actions. Flow-matching minimizes the discrepancy between a predicted velocity field and the ideal denoising direction. We demonstrate that Self-GAD as a plug-in guidance method improves general robotic foundation models closed-loop performance.

Further, we implement environmental perturbations that challenge vanilla diffusion policies to maintain state-action coherence. Noise that persists across multiple timesteps can lead to spurious correlations between states and actions, degrading policy performance [21]. Continuous noise is modeled as a constant velocity applied to goal-position target objects, introducing uniform linear shifts in (x, y) pose. Increasing action horizon and action chunking further degrade Diffusion Policy performance under perturbations, highlighting the necessity of closed-loop control in dynamic environments.

IV. EXPERIMENTS

In this section, we present a series of experiments designed to evaluate the performance of Self-Guided Action Diffusion across various inference settings. Specifically, we aim to address the following research questions:

- How does Self-GAD compare to baseline random sampling in single-sample closed-loop settings?
- How does Self-GAD improve sample efficiency relative to coherence sampling, and what are the trade-offs in sample count?
- How does Self-GAD perform under challenging conditions such as stochastic environments or diverse demonstrations?
- How does Self-GAD perform on state of the art robotic foundation models (eg. GR00T-N1-2B)?

A. *Self-GAD outperforms Baseline*

We evaluate the performance of Self-GAD in closed-loop, single-sample settings across the Robomimic benchmark tasks, BlockPush, Franka Kitchen, Lift, and PushT [17, 29]. Our results demonstrate that, with optimally tuned β, Self-GAD consistently outperforms random sampling, achieving superior performance in a single-sample setting (Figure 2). The baseline for comparison remains consistent across coherence sampling and vanilla BID, where single-sample performance is equivalent to a random draw.

B. *Sample Efficiency of Self-GAD*

Next, we evaluate the sample efficiency of Self-GAD relative to Coherence Sampling, which requires significantly more samples to achieve comparable performance. Specifically, we assess the impact of sample count $\{1, 4, 8, 12, 16\}$ to find that Self-GAD attains near-optimal performance with significantly fewer samples. In contrast, Coherence Sampling requires up to 16 samples to achieve comparable success rates, demonstrating the superior sample efficiency of Self-GAD (Figure 3). Policies operating with fewer samples were particularly vulnerable to performance degradation when subjected to overly constrained guidance weights. Notably, the single-sample advantage of Self-GAD enables significantly faster inference while maintaining robust performance.

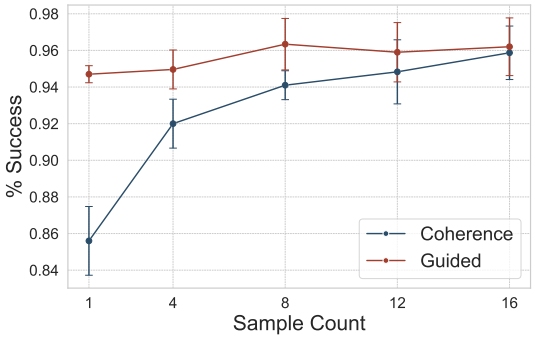

Fig. 3: **Sample Efficiency of Self-GAD:** Self-GAD achieves near-optimal performance with a single sample, maintained from 16 samples in PushT.

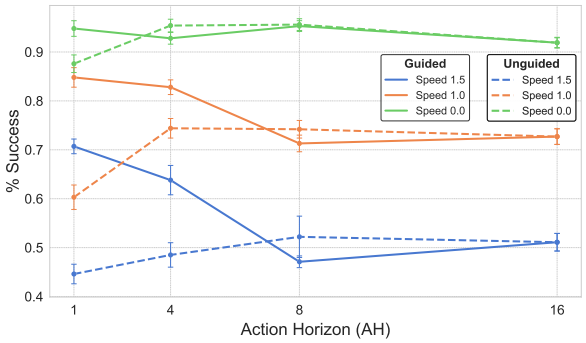

Fig. 4: **Self-GAD in Dynamic Settings Across Action Horizons:** In the dynamic PushT task, Self-GAD enhances single-sample closed-loop performance, with greater gains in high-variability environments.

C. Robustness to Unseen Dynamics

To evaluate the robustness of Self-GAD in temporally noisy, dynamic environments, we analyze its performance in scenarios where closed-loop execution and consistency are critical. We introduce a dynamically moving target object with a fixed speed increment of $\{0, 1, 1.5\}$ applied to both the x and y positions of PushT. We evaluate performance across action horizons $\{1, 4, 8, 16\}$, where longer horizons improve consistency but lack closed-loop adaptability for rapid reactions to dynamically moving objects.

Our findings indicate that Self-Guided Action Diffusion within a single horizon is optimal, effectively enabling closed-loop execution with guided benefits. The advantage of Self-GAD is further amplified in environments with stronger dynamic shifts, where the prior and current state become misaligned in the absence of guidance (Figure 4). At a speed of 1.5, Self-GAD achieves a 26.5% performance improvement, while in the static PushT setting, performance improves by 9%. Vanilla coherence sampling struggles with diverse demonstrations and dynamic movements, leading to significantly lower success rates until action horizon 8, where strategy consistency becomes critical. By action horizon 16, both Self-GAD and unguided Coherence Sampling achieve comparable performance.

D. Robustness to Dataset Variability

While prior work has primarily focused on closed-loop inference-time methods for handling noise at test time, the impact of dataset variability and multimodality learned during training remains largely unexplored. However, these factors are critical for developing generalizable robot policies. In this section, we evaluate Self-GAD 's ability to adapt to dataset variability and scripted multimodality in the RoboMimic pick-and-place square task, using datasets with incrementally varying trajectories.

To systematically introduce dataset variability, we apply perturbation parameters that modify task difficulty and trajectory modalities. These perturbations include variations in object positions, grasp offsets, and peg/nut placements, where

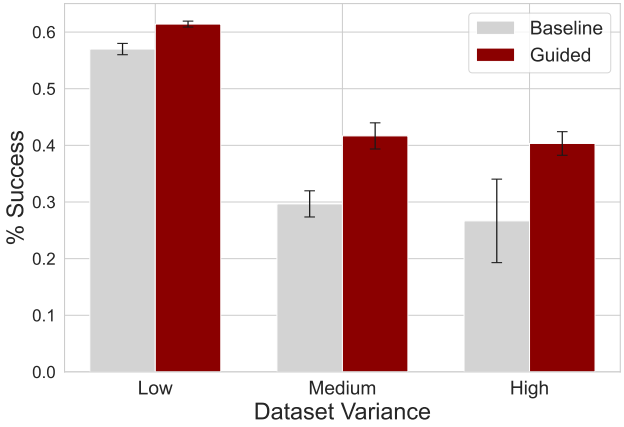

Fig. 5: **Guidance Enhances Robustness to Dataset Variance:** In robomimic square task rollouts with dataset variance (object positioning, orientation, and trajectories), guidance enhances consistency in single-sample settings, with its advantage amplifying in high-variance conditions, especially as learned action diversity grows.

increasing variance leads to more challenging learned trajectories (Table I). Object pose information is extracted directly from simulation data, ensuring accurate and reproducible environment setups. We create a structured dataset for RoboMimic tasks, logging successful trajectories in an HDF5 dataset. We training a diffusion policy across three levels of increasing scripted dataset variance, enabling robust policy learning under progressively more challenging conditions.

Self-GAD improves performance by 6.4%, 11.2%, and 14.3% for low, medium, and high variance settings, respectively, highlighting the growing significance of guided consistency in increasingly variable datasets (Figure 5).

E. Self-GAD Enhanced Robotic Foundation Models

To demonstrate its compatibility with generalized frameworks, we extend Self-GAD to a large-scale foundation model. We integrate self guidance within the diffusion transformer of GR00T-N1 [2], finetuned to tasks across the RoboCasa

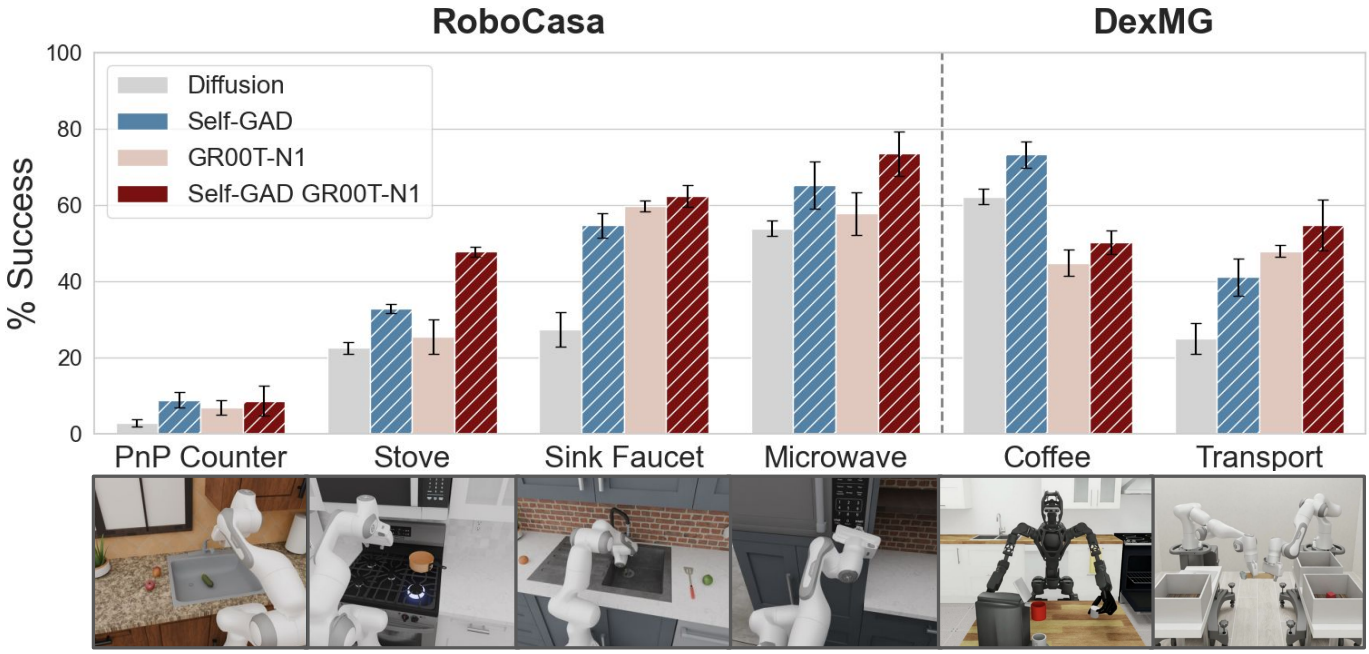

Fig. 6: **Self-GAD in Generalized Robotic Foundation Models.** We fine-tune GR00T-N1-2B on 100 demonstrations per task in single action horizon settings (PnP Counter to Cab, Turn Stove On, Turn Sink Faucet On, Turn Microwave Off, Coffee, and Transport). Self-GAD boosts success in both RoboCasa and DexMG, by 28.4% and 12% respectively.

benchmark and the DexMimicGen Cross-Embodiment Suite (DexMG). By adapting the GR00T-N1 checkpoint to new post-training datasets, we convert a generalist foundation model into a task-specialized policy for comparable performance to diffusion and Self-GAD.

RoboCasa consists of simulated interactive kitchen scenes, including pick-and-place, door manipulation, button pressing, and turning levers [19]. DexMG features bimanual dexterous manipulation tasks executed by dual-arm Panda robots with parallel-jaw grippers and a GR-1 humanoid equipped with dexterous hands [13].

In RoboCasa and DexMG, we confirm that Self-GAD improves vanilla diffusion success rates by 48.2% and 17.2%, respectively. Integrating Self-GAD into GR00T-N1 leads to a 28.4% improvement in task success on finetuned RoboCasa benchmarks and a 12% gain on DexMG. Across both benchmarks, Self-GAD consistently outperforms diffusion and GR00T-N1 baselines. In most settings, Self-GAD GR00T-N1 achieved highest performance, demonstrating improved sample efficiency and robustness even under limited data (Figure 6).

V. DISCUSSION

We demonstrate the sample efficacy of Self-Guided Action Diffusion to maintain consistency, leveraging inference-time gradient-based guidance. This sampling paradigm dynamically adjusts sampling distributions based on task-specific contexts, which exceeds sampling benchmarks. Despite its advantages over baselines, Self-Guided Action Diffusion is constrained by its reliance on manual tuning across different settings. Future work aims to address this limitation by developing adaptive, on-the-fly tuning mechanisms that leverage environmental history and noise patterns. Our results indicate that each task setting has an optimal guidance weight, which can be tuned on-the-fly to maximize performance and efficiency. This adaptive approach is critical in highly dynamic, non-uniform environments where abrupt changes in acceleration or fluctuating state transitions challenge traditional sampling techniques.

REFERENCES

[1] Suneel Belkhale, Yuchen Cui, and Dorsa Sadigh. Data quality in imitation learning. *Advances in Neural Information Processing Systems*, 36, 2024.

[2] Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

[3] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. pi0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

[4] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[5] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen

Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

[6] Chi-Lam Cheang, Guangzeng Chen, Ya Jing, Tao Kong, Hang Li, Yifeng Li, Yuxiao Liu, Hongtao Wu, Jiafeng Xu, Yichu Yang, et al. Gr-2: A generative video-language-action model with web-scale knowledge for robot manipulation. *arXiv preprint arXiv:2410.06158*, 2024.

[7] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *arXiv preprint arXiv:2303.04137*, 2023.

[8] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[9] Pete Florence, Corey Lynch, Andy Zeng, Oscar A Ramirez, Ayzaan Wahid, Laura Downs, Adrian Wong, Johnny Lee, Igor Mordatch, and Jonathan Tompson. Implicit behavioral cloning. In *Conference on Robot Learning*, pages 158–168. PMLR, 2022.

[10] Vincent Tao Hu, Yunlu Chen, Mathilde Caron, Yuki M Asano, Cees GM Snoek, and Bjorn Ommer. Guided diffusion from self-supervised diffusion features. *arXiv preprint arXiv:2312.08825*, 2023.

[11] Ye Huang, Di Kang, Shenghua Gao, Wen Li, and Lixin Duan. High-level feature guided decoding for semantic segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, 2024.

[12] Xiaogang Jia, Denis Blessing, Xinkai Jiang, Moritz Reuss, Atalay Donat, Rudolf Lioutikov, and Gerhard Neumann. Towards diverse behaviors: A benchmark for imitation learning with human demonstrations. *arXiv preprint arXiv:2402.14606*, 2024.

[13] Zhenyu Jiang, Yuqi Xie, Kevin Lin, Zhenjia Xu, Weikang Wan, Ajay Mandlekar, Linxi Fan, and Yuke Zhu. Dexmimicgen: Automated data generation for bimanual dexterous manipulation via imitation learning. *arXiv preprint arXiv:2410.24185*, 2024.

[14] Seungjae Lee, Yibin Wang, Haritheja Etukuru, H. Jin Kim, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. Behavior generation with latent actions. *arXiv preprint arXiv:2403.03181*, 2024.

[15] Yuejiang Liu, Jubayer Ibn Hamid, Annie Xie, Yoonho Lee, Maximilian Du, and Chelsea Finn. Bidirectional decoding: Improving action chunking via closed-loop resampling. *arXiv preprint arXiv:2408.17355*, 2024.

[16] Corey Lynch, Mohi Khansari, Ted Xiao, Vikash Kumar, Jonathan Tompson, Sergey Levine, and Pierre Sermanet. Learning latent plans from play. In *Conference on robot learning*, pages 1113–1132. PMLR, 2020.

[17] Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. *arXiv preprint arXiv:2108.03298*, 2021.

[18] Mitsuhiko Nakamoto, Oier Mees, Aviral Kumar, and Sergey Levine. Steering your generalists: Improving robotic foundation models via value guidance. *arXiv preprint arXiv:2410.13816*, 2024.

[19] Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. Robocasa: Large-scale simulation of everyday tasks for generalist robots. *arXiv preprint arXiv:2406.02523*, 2024.

[20] Nur Muhammad Shafiullah, Zichen Cui, Ariuntuya Arty Altanzaya, and Lerrel Pinto. Behavior transformers: Cloning k modes with one stone. *Advances in neural information processing systems*, 35:22955–22968, 2022.

[21] Gokul Swamy, Sanjiban Choudhury, Drew Bagnell, and Steven Wu. Causal imitation learning under temporally correlated noise. In *International Conference on Machine Learning*, pages 20877–20890. PMLR, 2022.

[22] Tsun-Hsuan Johnson Wang, Juntian Zheng, Pingchuan Ma, Yilun Du, Byungchul Kim, Andrew Spielberg, Josh Tenenbaum, Chuang Gan, and Daniela Rus. Diffusebot: Breeding soft robots with physics-augmented generative diffusion models. *Advances in Neural Information Processing Systems*, 36:44398–44423, 2023.

[23] Yanwei Wang, Lirui Wang, Yilun Du, Balakumar Sundaralingam, Xuning Yang, Yu-Wei Chao, Claudia Perez-D'Arpino, Dieter Fox, and Julie Shah. Inference-time policy steering through human interactions. *arXiv preprint arXiv:2411.16627*, 2024.

[24] Ziyu Wang, Josh S Merel, Scott E Reed, Nando de Freitas, Gregory Wayne, and Nicolas Heess. Robust imitation of diverse behaviors. *Advances in Neural Information Processing Systems*, 30, 2017.

[25] Hongtao Wu, Ya Jing, Chilam Cheang, Guangzeng Chen, Jiafeng Xu, Xinghang Li, Minghuan Liu, Hang Li, and Tao Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. *arXiv preprint arXiv:2312.13139*, 2023.

[26] Xiaomeng Xu, Huy Ha, and Shuran Song. Dynamics-guided diffusion model for robot manipulator design. *arXiv preprint arXiv:2402.15038*, 2024.

[27] Guangyu Zhao, Kewei Lian, Haowei Lin, Haobo Fu, Qiang Fu, Shaofei Cai, Zihao Wang, and Yitao Liang. Optimizing latent goal by learning from trajectory preference. *arXiv preprint arXiv:2412.02125*, 2024.

[28] Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

[29] Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Soroush Nasiriany, and Yifeng Zhu. robosuite: A modular simulation framework and benchmark for robot learning. *arXiv preprint arXiv:2009.12293*, 2020.

APPENDIX

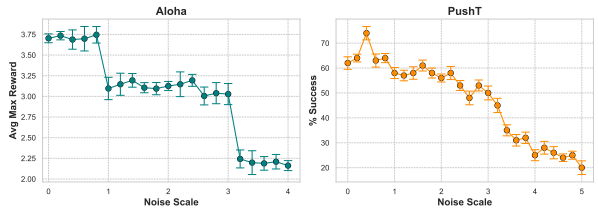

Fig. 7: **Diffusion & ACT Policy Degradation Under Inference-Time Noise:** Increasing inference-time Gaussian noise degrades maximum reward in both ACT (Aloha pick-place, discrete) and PushT (diffusion, continuous) tasks, highlighting their sensitivity to unseen dynamics and environmental noise.

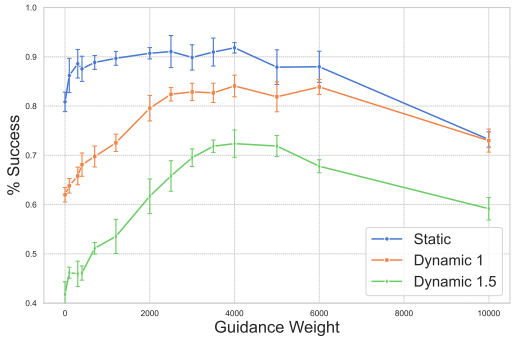

Fig. 8: **Self-GAD under Dynamic Conditions:** Self-GAD reduces the performance gap between static and dynamic environments, with optimal guidance weights (around 4000 consistently).

Perturbation Metric	Low Variance	Medium Variance	High Variance
Offset Scale	0.08	0.25	0.3
Grasp Position Variance	0.008	0.008	0.009
Pick Rotation Variance	0.008	0.008	0.009
Peg Lateral Variance	0.008	0.008	0.009
Peg Height Variance	0.04	0.05	0.055
Nut Height Variance	0.04	0.05	0.055

TABLE I: **Perturbation metrics for dataset variance in the RoboMimic pick-and-place task.** Higher variance settings introduce greater variability in object positioning, grasp offsets, and peg/nut placements, increasing the complexity of learned trajectories.