ADAPTING DIFFUSION POLICIES TO NOVEL ENVIRONMENTS VIA POLICY-STEERED OPTIMIZATION

Skye Thompson, Sergio Orozco, & George Konidaris
Department of Computer Science
Brown University
Providence, RI, 02906, USA

Eric Rosen & Karl Schmekpeper
Robotics and AI Institute
Cambridge, MA 02142, USA

ABSTRACT

We propose a novel method of adapting generative robot control policies to succeed in unfamiliar environments with novel runtime constraints. We use a model-based planner to optimize for trajectories that obey both novel environmental constraints only observed at inference time, and implicit task constraints learned from demonstrations by the policy model. We achieve this by evaluating a policy-alignment objective that measures the policy model's success at reconstructing noised trajectories. We demonstrate this approach's ability to generalize to two novel simulation environments with obstacles not seen during training.

1 INTRODUCTION

Generative diffusion models have become a popular tool for representing robot policies, thanks to their ability to generate diverse samples in high dimensional action spaces (Janner et al., 2022). By learning to imitate an accessible number of expert demonstrations, diffusion policy models implicitly capture complex constraints about what defines quality behavior for a given skill, including workspace limits, temporal dependencies, and desired end-effector dynamics, without requiring environment models, constraint engineering or reward functions (Chi et al., 2024). But these models struggle to produce *adaptive* behaviors when applied in environments with unfamiliar features—novel obstacles, novel tool shapes, user preferences—that may require novel interactions to execute the skill successfully. Model-based planners and optimizers can produce such behavior by searching in a broader space of possible actions, but are often infeasible for complex skills. Attempting to analytically define desirable skill behavior requires substantial engineering (Ha et al., 2020), and forfeits the efficiency and simplicity of model-free policy learning. Given a policy and a model-based planner, can we leverage both to generate adaptive behavior in unfamiliar environments?

We propose a method, Implicit Policy-Steered Optimization (IPSO) for using learned models and diffusion policies to generate novel, adaptive behaviors at inference time by optimizing jointly over task constraints and a *policy-alignment objective*. By using the learned diffusion process to calculate an objective penalizing unlikely actions under the policy model's action proposal distribution, enable a planner to optimize for behavior that obeys the implicit, demonstrated constraints modeled by the policy, as well as additional constraints in the novel environment, to produce novel behaviors that improve task performance without requiring retraining. We highlight three benefits of this approach:

Capable of producing novel behavior: This approach enables generating behavior which may be difficult or impossible to sample directly from the generative model, even when leveraging other steering techniques like re-sampling or guidance.

Enables composing implicit and explicit task constraints: During optimization, it is possible to compose arbitrarily many additional cost functions, integrating novel information without losing the implicit task constraints present in the demonstrations and captured by the learned policy.

Utilizes independent models: This method integrates completely independent, pretrained, environment and policy models, which are *not* required or assumed to share observation modalities.

We evaluate this approach in two environments—navigating a 2D maze, and solving a PushT task (Chi et al., 2024). The policies are trained with no obstacles, and tested on environments with obstacles. We evaluate this technique using policy and planning models learned with different observation

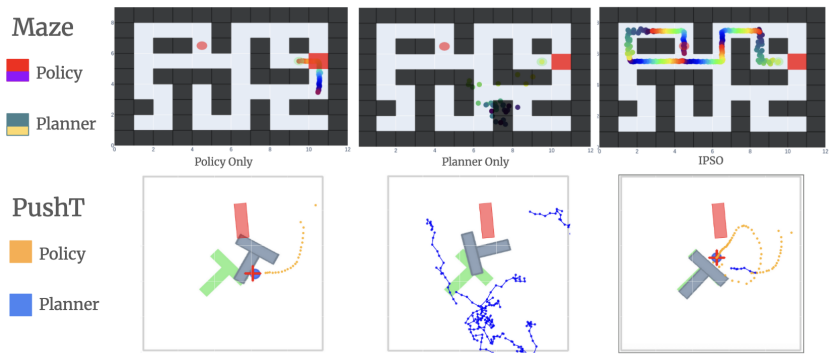

Figure 1: When executed in an environment with an unfamiliar constraint, the learned generative policy fails. Optimization-based planning respects the environmental constraint (the new red obstacle), but fails to respect implicit task constraints learned by the policy (the T-block dynamics and location of maze walls). Optimizing both the environment constraint and policy alignment objective produces novel behavior that obeys the environmental constraint while still progressing the skill.

modalities. We compare to the performance of independent policy and planners, as well as existing policy adaptation methods leveraging classifier guidance (Carvalho et al., 2024) and post-hoc action ranking(Qi et al., 2025), and evaluate the impact of model error on performance. Our method is more consistently able to adapt behavior to the novel environment to succeed in both tasks.

2 Background

A learned diffusion policy, $\pi_\theta(\tau|o^{pol})$, models a distribution over sequences of actions a of length h, where $\tau = (a_0...a_h)$. $O^{pol}(s) = o^{pol}$ is an observation of the environment's state s given by the policy's observation function. Given an observation and an initial noise sample $\tau^K \sim N(0, I)$, the policy predicts a series of denoising steps $\epsilon(\tau^k, o_t^{pol})$ that converges to a sampled action $\tau^0 = (a_0...a_h)$(Chi et al., 2024). Such a policy is trained to generate actions that progress the environment towards a success state without violating any task constraints. These constraints comprise the information that exists implicitly in the mind of the demonstrator(s) providing examples about how to act in the environment, who through their demonstration encodes them into the model. We collectively call them C_{task}. For a successful skill execution, $\sum_{i=0}^{n} C_{task}(s_i, a_i) = 0$

We consider the problem of transferring a skill to a novel environment, in which a different set of constraints, C' hold. We assume that some component of C' is $C'_{task} \approx C_{task}$—the implicit cost of most (s, a) pairs will be similar, otherwise we wouldn't expect executing this skill in this situation to be useful. We label the remainder as G, arising from changes in the environment not present in the training context, such that $C' = C'_{task} + G$. When G can be modeled as a function $G(o^{plan}, a)$, and an environment dynamics model $\mathrm{T}(o_t^{plan}, a_t) \rightarrow o_{t+1}^{plan}$ is available, it's possible to identify favorable action sequences with respect to that objective using an optimization-based planner. But analytically modeling C' in its entirety is infeasible for manipulation skills of any complexity, as C' includes information like temporal dependencies inherent to the skill.

We present a maze environment as an example in Figure 1 Here, policy $\pi_{maze}(o^{pol}, goal)$ has been trained to produce trajectories that avoid the black walls, implicitly capturing C_{task}. The red walls *were not present during training*, and can be modeled as $G(o^{plan}, a) = 1(o^{plan} \in Obstacle)$.

3 Method

Relying solely on either planner or policy in the novel environment will frequently result in failure, as neither can account for the complete set of constraints C' that define skill success—in 1, the planner lacks information about the maze's walls that the policy implicitly captures, while the policy lacks a way to condition on and adapt to the presence of the new red walls. Instead of using either individually, we perform joint optimization across modeled objective G and a *policy-alignment objective*, leveraging the implicit, internal representation C_{task} that our learned policy has captured.

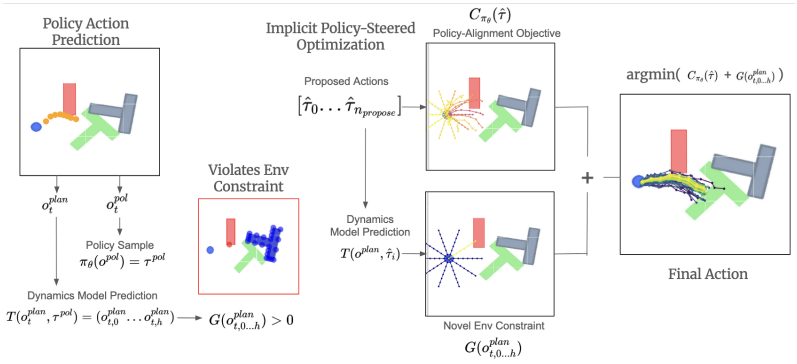

Figure 2: Implicit Policy-Steered Optimization (IPSO) switches from policy execution to optimization when the planning model predicts the policy will violate an environmental constraint.

This optimization only requires that our policy and model share an action representation, allowing us to leverage independent pre-trained models with no need for pretraining.

3.1 Policy-Alignment Objective

Our goal is to identify a function that returns a low cost when a proposal trajectory τ fulfills C_{task}, thus appearing as a plausible output from our pretrained diffusion model. We leverage the observation that a noised plausible trajectory will denoise to an output close to its original form, while an implausible trajectory will denoise to a very different output. Our objective is calculated as:

$$perturb_{\hat{t}}(\tau) = \tau + \sigma_{\hat{t}}\epsilon_{\hat{t}}, \epsilon_{\hat{t}} \sim \mathcal{N}(0, I)$$

$$restore_{\hat{t},\theta}(\tau) = \frac{1}{\sqrt{\alpha_{\hat{t}}}}(\tau - \frac{1-\alpha_{\hat{t}}}{\sqrt{1-\bar{\alpha}_{\hat{t}}}}(\epsilon_\theta(\tau, \hat{t})))$$

$$C_{\pi_\theta}(o^{pol}, \tau) = ||\tau - restore_{\hat{t},\theta}(perturb_{\hat{t}}(\tau))||_2,$$

where $\hat{t} \in (0, 1]$ represents the magnitude of the noise applied (we found a low value, .01, empirically most effective), and ϵ_θ is the diffusion ϵ-model. This is similar to the restoration gap metric described in Lee et al. (2023), which reconstructs the original trajectory by iteratively denoising from $\hat{t}$. For speed, we use the intermediate prediction $\hat{\tau_0}$ from a single step of the reverse process as our reconstruction. The final cost is the L_2 distance between the proposal trajectory and the reconstruction. This objective penalizes noise and variation in scale, direction, and shape between proposal and genuine generated trajectories. It is capable of modeling multi-modal distributions of plausible actions, effectively representing the information captured by the policy.

3.2 Implicit Policy-Steered Optimization

We use our planning model to identify whether a policy-generated action sequence would incur cost $\sum_0^h G(o_i^{plan}, a_i) > g$, by predicting state $o_i^{plan} = T(o_{i-1}^{plan}, a)$ for every action in the sequence. g is a preselected threshold implying violation of a constraint that would result in skill failure, and transition from policy to planner. When planning, we aim to produce behavior that is low cost in both the implicit policy-alignment objective C_{π_θ} introduced in Section 3.1, and the planner objective G. We optimize for $argmin_{\hat{\tau}} C_{\pi_\theta}(o_t^{pol}, \hat{\tau}) + G(T(o_t^{plan}, \hat{\tau}))$ using the Cross-Entropy Method (CEM) (Rubinstein, 1997). We initialize the proposal distribution mean from a set of iteratively noised proposal trajectories, sampled radially from the agent's observed position, selecting the n_{elites} trajectories with lowest cost according to the composed objective. We find this improves performance in multi-modal action distributions, where CEM often collapses to solutions between modes.

4 Experiments and Results

We evaluate in a 2D maze environment and a PushT task. We use a pretrained diffusion policy (For the maze, trained on trajectories navigating the maze without running into the walls. For PushT, we use Lerobot's (Cadene et al., 2024) pre-trained visual diffusion policy, recreating Chi et al. (2024)'s results.) In each environment, at inference time, we add an additional constraint represented by a

Table 1: Policy steering experiment success rates

	Policy	**Planner**	**IPSO**$_{kpt}$	**IPSO**$_{Img}$	**PHR**	**Guided**	**Training**
Maze	17/33	1/33	**25/33**	-	20/33	19/33	33/33
PushT	13/60	0/60	**37/60**	**30/60**	18/60	8/60	44/60

red obstacle, such that $G(o^{plan}, a) = 1(o^{plan} \in Obstacle)$. For the planner-only evaluations, we only provide the location of this obstacle and no other task or goal information. In the maze, the start and goal are randomized for each new trial, the obstacle is placed in an empty grid space within the planning horizon of the policy. In PushT, the initial agent and T positions are randomized for each trial, and the obstacle is placed without overlap within a fixed radius of both the initial T and goal positions. Any violation of G, or inability to complete the task, is a failure.

For each experiment, we assume access to an environment model relevant for evaluating G. For the maze, we use $s' = T(s, a) = s + a$. The PushT environment is a non-prehensile manipulation task with more complex dynamics—to demonstrate both the performance of our method using a learned environment model, and its ability to utilize independently trained policy and planning models without shared representation, we evaluate on two different models: a pre-trained equivariant trained keypoint model (Orozco et al., 2025) and a latent visual dynamics model (Zhou et al., 2025). We compare our method to the independent use of the policy and planner, as well as **Post-hoc Ranking**, where we sample a fixed number (10) of samples τ from the policy, evaluate $G(T(o^{plan}, \tau))$, and select the sample with the lowest cost, following Qi et al. (2025), and **Classifier Guidance**, where we compare to use of an SDF guidance term, as in Carvalho et al. (2024).

As shown in Table 1 IPSO successfully adapts to the presence of a novel environmental constraint in both settings, outperforming naive execution, as well as comparable baselines, which are capable of steering the base generative model, but are not capable of generating novel adaptive behavior. **Training** is the success rate of the base policy in the training environment, without obstacles. IPSO is limited by the base policy's representation of useful actions—it can fail when all proposal trajectories that are low-cost in C_{π_θ} violate G. We also evaluate IPSO and PHR in the presence of model error, using a partially trained keypoint dynamics model (50 vs. 400 epochs, 3e-4 vs. 2e-4 validation error). We find performance degrades for both, but IPSO still outperforms PHR (**21/60** vs 10/60).

5 Related Work

Other examples of diffusion policy steering for robot skill execution include Wang et al. (2025), which demonstrates classifier-guidance based steering to drive policies to resemble user-provided trajectories. Carvalho et al. (2024) steers a dedicated motion-planning policy to avoid novel obstacles at inference time by using an SDF based guidance term. Our method does not require human input, and uses an optimization-based technique rather than guidance. Du & Song (2025) uses a learned, differentiable dynamics model to predict action outcomes and guides the base policy based on outcome similarity to a provided goal in latent image space. Our method uses a dynamics model in calculating an optimization objective and does not require that model to be differentiable . Qi et al. (2025) uses a dynamics model and inference-time objective to select from multiple policy samples. Our method uses the policy to produce a policy-alignment objective for an optimization-based planner, enabling our method to produce novel actions not sampled from the initial policy.

6 Conclusion and Future Work

We find that Implicit Policy-Steered Optimization is a promising method for jointly leveraging a learned policy and model-based planner to generate adaptive behavior in novel environments, without requiring retraining or heavy constraint engineering. In addition to expanding our experiments to evaluate this method on real-world robotics tasks, we're interested in further exploring how the traits of the base policy (multi-modality, state-space coverage) impact its steerability under this and other approaches, as well as integrating more targeted methods for multi-objective optimization.

Acknowledgments

This work was supported in part by ONR REPRISM MURI N00014-24-1-2603, ONR grant 00014-22-1-2592, and NSF GRFP 2439559.

REFERENCES

Remi Cadene, Simon Alibert, Alexander Soare, Quentin Gallouedec, Adil Zouitine, Steven Palma, Pepijn Kooijmans, Michel Aractingi, Mustafa Shukor, Dana Aubakirova, Martino Russi, Francesco Capuano, Caroline Pascal, Jade Choghari, Jess Moss, and Thomas Wolf. Lerobot: State-of-the-art machine learning for real-world robotics in pytorch. https://github.com/huggingface/lerobot, 2024.

Joao Carvalho, An T. Le, Mark Baierl, Dorothea Koert, and Jan Peters. Motion planning diffusion: Learning and planning of robot motions with diffusion models, 2024. URL https://arxiv.org/abs/2308.01557.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion, 2024. URL https://arxiv.org/abs/2303.04137.

Maximilian Du and Shuran Song. Dynaguide: Steering diffusion polices with active dynamic guidance, 2025. URL https://arxiv.org/abs/2506.13922.

Jung-Su Ha, Danny Driess, and Marc Toussaint. A probabilistic framework for constrained manipulations and task and motion planning under uncertainty. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6745–6751, 2020. doi: 10.1109/ICRA40945.2020.9196840.

Michael Janner, Yilun Du, Joshua B. Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis, 2022. URL https://arxiv.org/abs/2205.09991.

Kyowoon Lee, Seongun Kim, and Jaesik Choi. Refining diffusion planner for reliable behavior synthesis by automatic detection of infeasible plans, 2023. URL https://arxiv.org/abs/2310.19427.

Sergio Orozco, Brandon B. May, Tushar Kusnur, George Konidaris, and Laura Herlant. Learning equivariant neural-augmented object dynamics from few interactions. In *Beyond Rigid Worlds: Representing and Interacting with Non-Rigid Objects*, 2025. URL https://openreview.net/forum?id=JAiJpFozaD.

Han Qi, Haocheng Yin, Aris Zhu, Yilun Du, and Heng Yang. Strengthening generative robot policies through predictive world modeling, 2025. URL https://arxiv.org/abs/2502.00622.

Reuven Y. Rubinstein. Optimization of computer simulation models with rare events. *European Journal of Operational Research*, 99(1):89–112, 1997. ISSN 0377-2217. doi: https://doi.org/10.1016/S0377-2217(96)00385-2. URL https://www.sciencedirect.com/science/article/pii/S0377221796003852.

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, 2025. URL https://arxiv.org/abs/2411.16627.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning, 2025. URL https://arxiv.org/abs/2411.04983.