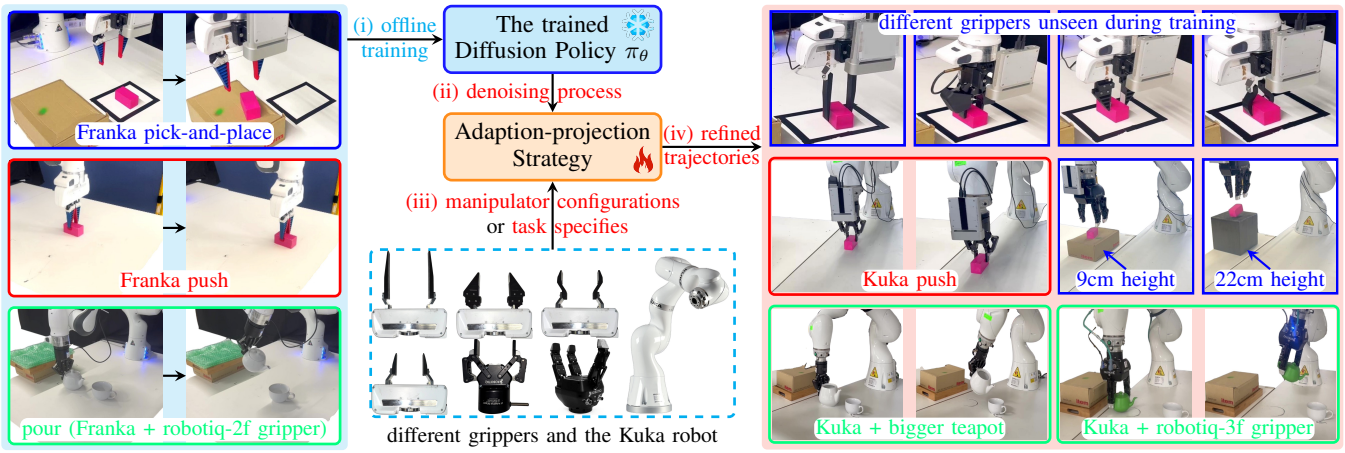
2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Inference-stage Adaptation-projection Strategy Adapts Diffusion Policy to Cross-manipulators Scenarios

Xiangtong Yao[1], Yirui Zhou[1], Yuan Meng[1], Yanwen Liu[1], Zitao Zhang[1], Liangyu Dong[1], Zhenshan Bing[2,1,†], Kai Huang[3], Fuchun Sun[4], Alois Knoll[1]



(a) Demonstrations collection (b) Adaption-projection strategy refines trajectories for vary manipulator configurations or task requirements

Fig. 1: (a) Collect demonstrations via the base manipulator (Franka robot with a specific gripper) to train the diffusion policy π_θ for each task. (b) Our adaptation-projection strategy enables retraining(or fine-tuning)-free adaptation of the policy to new manipulator configurations (Kuka robot, different grippers) and task requirements (obstacle heights) at inference time.

***Abstract*— Diffusion policies are powerful visuomotor models for robotic manipulation, yet they often fail to generalize to manipulators or end-effectors unseen during training and struggle to accommodate new task requirements at inference time. Addressing this typically requires costly data recollection and policy retraining for each new hardware or task configuration. To overcome this, we introduce an adaptation-projection strategy that enables a diffusion policy to perform cost-effective adaptation to novel manipulators and dynamic task settings, entirely at inference time and without retraining or fine-tuning the policy. Our method first trains a diffusion policy in SE(3) space using demonstrations from a base manipulator. During online deployment, it projects the policy's generated trajectories to satisfy the kinematic and task-specific constraints imposed by the new hardware and objectives. Moreover, this projection dynamically adapts to physical differences (e.g., tool-center-point offsets, jaw widths) and task requirements (e.g., obstacle heights), ensuring robust and successful execution. We validate our approach on real-world pick-and-place, pushing, and pouring tasks across multiple manipulators, including the Franka Panda and Kuka iiwa 14, equipped with a diverse array of end-effectors like flexible grippers, Robotiq 2F/3F grippers, and various 3D-printed designs. Our results demonstrate consistently high success rates in these cross-manipulator scenarios, proving the effectiveness and practicality of our adaptation-projection strategy.**

[1] School of Computation, Information and Technology, Technical University of Munich, Garching, Germany. Email: xiangtong.yao@tum.de.
[2] State Key Laboratory for Novel Software Technology and the School of Science and Technology, Nanjing University (Suzhou Campus), China.
[3] Key Laboratory of Machine Intelligence and Advanced Computing, School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China.
[4] Department of Computer Science and Technology, Tsinghua University, Beijing, China.
[†]Corresponding author: Zhenshan Bing bing@nju.edu.cn, bing@in.tum.de

I. Introduction

Deep imitation learning has shown promise in learning manipulation skills from human demonstrations [1], [2], reducing the need for extensive programming efforts compared to rule-based approaches (i.e., motion planning [3]) and reinforcement learning [4]–[6]. Among these, diffusion policies [7], [8] integrate diffusion model [9] into imitation learning scheme and demonstrate powerful capabilities in learning visumotor manipulation skills, generating diverse and adaptable strategies in varying scenarios, such as multi-task manipulations [10], one-shot learning [11]. However, these methods typically assume fixed manipulator configurations (robot and end-effector) throughout training and deployment, struggling to adapt to new manipulator configurations. For instance, when switching from a base gripper to one with a different morphology, the robot's tool-center-point (TCP) may shift (as illustrated in different grippers in Fig. 1), leading to potential collisions or missed grasps if the pretrained policy is naively applied to generate trajectories.

Recent research mitigates this limitation through multi-embodiment learning strategies [12]–[19]. However, these methods often require collecting substantial embodiment-specific data [14], [19], relying on policy fine-tuning to

accommodate new configurations [13], [14], or requiring united task distributions during training and developing [15] synthesis of large-scale heterogeneous gripper datasets [17], training an united policy across multiple manipulators [12], [16], [19], designing a new grasping tool for across-embodiments scenarios [18]. When adapting the trained policy to new manipulator configurations (including robot and end-effector that were unseen during policy training) to satisfy different task requirements (like picking a heavier object), the above approaches incur time-consuming adaptation costs and limit deployment flexibility in real-world scenarios.

We bridge this gap through an integration of diffusion policy [7], [8] with a adaption-projection strategy. This strategy allows the trained policy to fit new manipulator configurations without policy retraining or fine-tuning, as shown in Fig.1. The policy first learns manipulation primitives following diffusion policy, where the training demonstration data is collected via a base configuration in Fig.1(a). During the policy's inference stage, the base configuration is replaced with a new one (different grippers or Kuka robot). To ensure the generative trajectories are valid for new configurations, we first adapt the manipulator-specific dimensional offsets to the observation inputs of the policy, and recast the traditional denoising process as a constrained-optimization projection process. The projection introduces mathematical encoding of task requirements and safety constraints (e.g., object grasping, avoiding collisions) to the denoising process, utilizing a quadratic programming optimization method to incrementally refine an initially noisy trajectory to conform to unseen manipulators and new task requirements (like placing the object into a higher platform), as shown in Fig.1(b). Crucially, this approach does not require policy retraining or fine-turning, preserving flexibility for real-world deployment with minimal overhead. Key contributions are summarized as follows.

- A adaptation-projection strategy is introduced into the inference-stage of the trained diffusion policy, adjusting and refining the generation of trajectories to counteract the TCP offset problem caused by switching manipulator configurations in SE(3) space, ensuring task completion and the satisfaction of new task requirements without retraining and fine-tuning the policy.
- A general formulation of the task and safety constraints is designed for the adaptation-projection process. Moreover, a cumulative trajectory refinement process is utilized to preserve the trajectory's temporal consistency.
- Real-world experiments for varying-difficulty manipulation tasks (pick-and-place, pushing, and pouring tea) in across-grippers and across-manipulator (Franka, Kuka) scenarios show the effectiveness of our method.

II. Related Work

Cross-manipulator Skill Transfer: Transferring robot manipulation knowledge between different embodiments can improve the policy's flexibility and generalizability for adapting to new tasks or hardware settings [14], including the knowledge transitions between different grippers [19], [20] or robot manipulators [12]–[14], [18], [21]. For example, Zakka et al. [20] combine the temporal cycle-consistency method with imitation learning to learn an invariant feature space for different embodiments, empowering target robot manipulators to perform the tasks by imitating video demonstrations of human experts. While the feature correspondence between the source and target domains has to be re-trained if a new source or target domain is introduced. To mitigate this limitation, a dominant approach involves learning aligned latent feature spaces between source and target robots using techniques like adversarial training and cycle consistency [13], [21]. By leveraging large-scale heterogeneous robot data, Wang et al. [14] propose a heterogeneous pre-trained transformer that can perform different tasks across heterogeneous robot manipulators. However, these approaches relies on transfer learning or domain adaption [13], [14], [21], where introducing new hardware configuration (e.g., grippers, manipulators) to the policy require fine-tuning policy [14] or retraining auxiliary networks [13], [21], limiting the flexibility of the policy. Another line of work, such as Mirage [15], uses visual inpainting to transfer policies to unseen robots. This approach, however, struggles with significant changes in background or large tool-center-point (TCP) shifts between robots. Yao et al. [22] propose a learning-optimization diffusion policy that adapts to new grippers without retraining, but their method is limited to translational TCP shifts and is not validated in cross-manipulator scenarios. Our method circumvents these limitations by integrating an adaptation-projection strategy into the inference stage of the diffusion policy [7] that requires no retraining or fine-tuning. The adaptation component explicitly handles geometric discrepancies like TCP shifts (larger than 10cm shifts in translational axis and shift issue in rotational axes), while the projection refines the generated trajectory online to ensure it complies with the safety and task constraints of the new hardware in cross-manipulator scenarios. This enables a retraining(or fine-tuning)-free adaptation to manipulators with new configurations.

Inference-stage Refining Trajectory in Diffusion Policy: Diffusion models have shown promise for solving decision-making tasks, including motion planning [23], [24], imitation learning [7], [8], [25], and reinforcement learning [26], [27]. When solving decision-making tasks through generative models, the policy needs not only to accomplish the task goal but also to satisfy certain task constraints, such as collision avoidance. Some works add a residual loss to the training objectives if the task constraints are consistent during the policy training and inference [24], [28]. A more flexible method is training diffusion models via classifier-free guidance [29], introducing conditioning variables that represent constraints into the policy training, such as physical constraints for guiding human motion generation [30]. However, the model conditioning encourages the generated samples to adhere to task constraints rather than strict guarantee constraints [31]. Alternative post-processing methods draw constraints into the last denoising stage of the sample-generating process and obtain samples that satisfy the constraints by solving

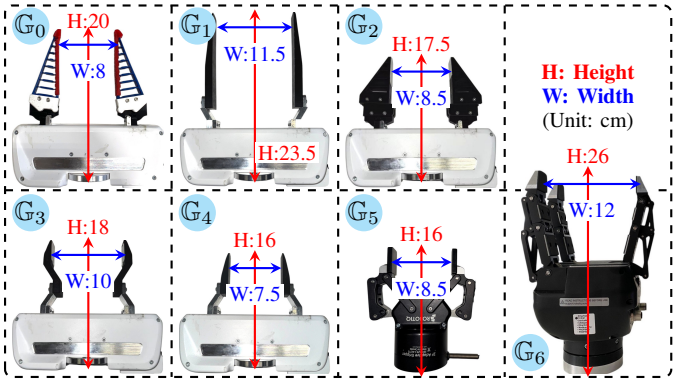


Fig. 2: Gripper variants of different dimensions.

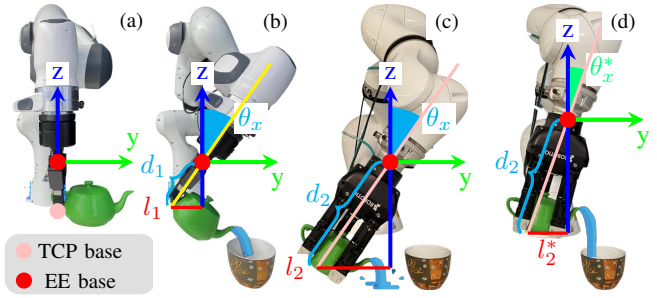


Fig. 3: Adapt robot pose for new manipulator configurations. Base (Franka + Gripper $\mathbb{G}_5$), new (Kuka + Gripper $\mathbb{G}_6$).

an optimization problem [32], [33]. As the optimization problem does not consider the unknown data likelihood, post-processing may result in samples significantly deviating from the data distribution [34]. To mitigate this issue, an iterative projection is integrated into the denoising process, confining each generated sample to the constraints. Some works employ it for sequential decision-making [31], which is time-consuming. Or model-based trajectory control [34], falling short of seamlessly adapting to a new model since introducing a new gripper changes the robot dynamics. Our approach utilizes adaptive projection alongside optimization techniques in the denoising process but mitigates the time consumption and makes the policy seamlessly adaptable to new manipulator configurations and task requirements (i.e., place the object to a higher platform).

III. Methodology

A. Problem Statement

We formalize the challenge of gripper-agnostic manipulation via diffusion policies through three core components:

- **Gripper Configuration**: Let $\mathbb{G} \subset \mathbb{R}^{d_g}$ denote the space of parallel gripper parameters encoding maximum width $w^{\max}$ and gripper height h, as shown in Fig. 2.
- **Observation Domain**: $\mathcal{O} = \mathcal{S}_{\text{sce}} \times \mathcal{S}_{\text{rob}}$ where $\mathcal{S}_{\text{sce}}$ represents scene observations (3D point clouds) and $\mathcal{S}_{\text{rob}} = SE(3) \times [0, g]$ the robot state (end-effector pose $\mathbf{x}_{\text{ee}} \in SE(3)$, gripper width g) $\mathbf{x}_{\text{ee}}$ is gripper-agnostic and different across robots, as shown in Fig. 3. g is specific for grippers.
- **Action Space**: $\mathcal{A} \subset \mathbb{R}^{d_a}$ containing end-effector displacements $\Delta\mathbf{x}_{\text{ee}}$ and gripper commands.

The policy π_θ is trained on demonstrations $\mathcal{D} = \{\tau^{(i)}\}_{i=1}^N$ collected with a base configuration, such as Franka with the gripper $\mathbb{G}_0 \in \mathbb{G}$. Each trajectory $\tau = \{(\mathbf{o}_t, \mathbf{a}_t)\}_{t=0}^T$ satisfies: $\mathbf{o}_t = (\mathcal{S}^0_{\text{sce}}, \mathbf{x}^0_{\text{ee}}, g^0_t)$ and $\mathbf{a}_t \sim \pi_{\text{expert}}(\cdot|\mathbf{o}_t)$,

where superscript 0 indicates $\mathbb{G}_0$ parameters. During deployment with new configuration (Kuka + gripper $\mathbb{G}_i \neq \mathbb{G}_0$), the observation-action distribution shifts due to (1) visual/kinematic differences $\mathcal{O}^i \neq \mathcal{O}^0$ (EE base differs in different robots and gripper kinematics variations result in tool-center-point (TCP) shifts in Fig. 3), and (2) policy mismatch $p_\theta(\mathcal{A}|\mathcal{O}^i) \neq p_\theta(\mathcal{A}|\mathcal{O}^0)$. This manifests as trajectory divergence $\|\tau^{\mathbb{G}_i}_{1:T} - \tau^{\mathbb{G}_0}_{1:T}\|_{\mathcal{W}} > \delta_{\text{tol}}$, where $\mathcal{W}$ is the task-specific metric and δ_{tol} the success threshold, e.g., objects cannot be grasped with shorter grippers, and collisions can result from using longer grippers.

To mitigate these issues, we develop policy π^*_θ that maintains task performance under manipulator variation, combining (1) Adaptation of manipulator configurations for trajectory alignment and (2) Task-satisfied and safety-constrained trajectory projection in the policy inference stage without retraining the policy. The overview framework is shown in Fig. 4.

B. Refining Trajectory Generation

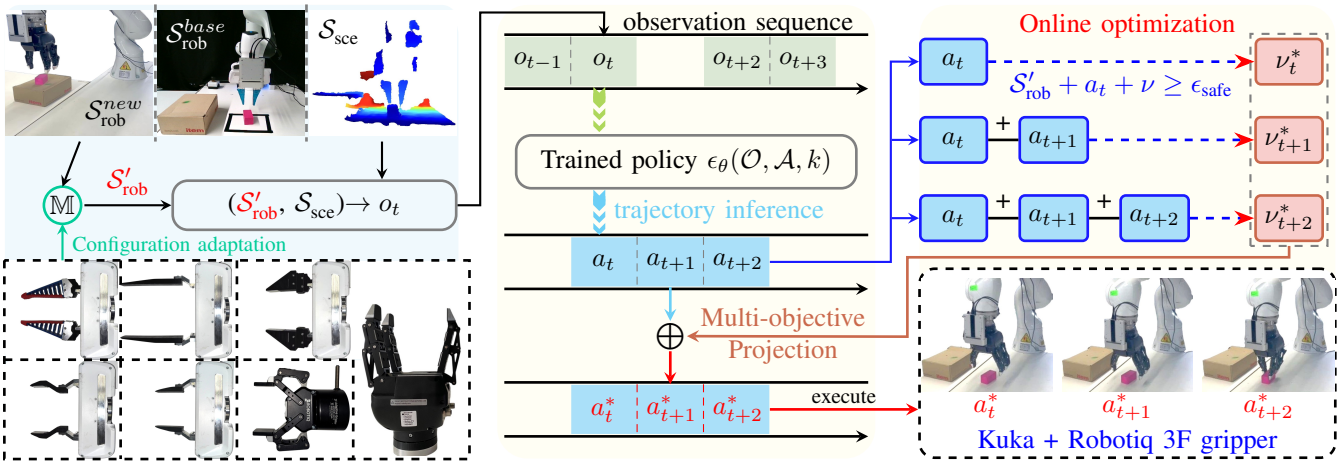
The training scheme of the base policy π_θ is consistent with that of Diffusion Policy [7] (DDPM) with observations $\mathcal{O}$ and MSE training loss. During inference, π_θ employs Denoising Diffusion Implicit Models (DDIM) [35] to generate trajectories. In the denoising process, however, our method introduces an adaptation-projection strategy into DDIM to enforce the generative trajectory to fit different manipulator configurations, ensuring task completion and robot safety.

Manipulator configuration adaptation: As shown in Fig. 3, during identical object manipulation, end-effector base (EE base) variations of different robots and gripper variations induce grasping width differences in the gripper state (g) and the tool-center-point (TCP) discrepancies, primarily along: (1) vertical axis (z): TCP translational offset ($\|d_1 - d_2\|$), (2) rotational axes (θ_x, θ_y): TCP rotational offset. These discrepancies cause inconsistent actions predicted by π_θ across grippers or failure of solving task when the robot perform unadapted trajectories (such as pouring in Fig. 3(c)).

We define mapping expressions that project $\mathbb{G}_{(i)}$ parameters to the $\mathbb{G}_{(0)}$ basis, where $\mathbb{G}_{(0)}$ denotes the base gripper, and $\mathbb{G}_{(i)}$ represent a new gripper of category i:

$$\begin{aligned} z'_{(i)} &= z_{(i)} + \Delta d_{(i)}, \\ g'_{(i)} &= g^{\max}_{(0)} - \alpha_{(i)}(g^{\max}_{(i)} - g_{(i)}), \end{aligned} \tag{1}$$

where $z_{(i)}$ is the measured height of EE base when the end-effector equipped with $\mathbb{G}_{(i)}$, $\Delta d_{(i)} = z_{(0)} - z_{(i)}$ is the TCP offset from the base manipulator configuration, $g_{(i)} \in [g^{\min}_{(i)}, g^{\max}_{(i)}]$ is the real-time grasping width, $\alpha_{(i)} = (g^{\max}_{(0)} - g^{\text{grasp}}_{(0)})/(g^{\max}_{(i)} - g^{\text{grasp}}_{(i)})$ scales widths, g^{grasp} is the width when the gripper grasps the object. The mapping parameters $\{\Delta h_{(i)}, \alpha_{(i)}\}$ are obtained through offline calibration with mechanical measurement of gripper dimensions and EE base offsets across different robots.



(a) Multi-model observation (b) Trajectory inference with safety projection

Fig. 4: Overview of policy. (a) The multi-modal observation consists of robot pose $\mathcal{S}'_{\text{rob}}$, scene point clouds $\mathcal{S}_{\text{sce}}$, Gripper morphological variations and manipulator kinematics variations are encoded into $\mathcal{S}'_{\text{rob}}$. (b) Safety-Constrained trajectory projection via online optimization process, enforcing the executive trajectory to satisfy task and safety constraints.

For the adaptation of rotational axes (taking θ_x as an example, the same applies to θ_y), the key idea is let $d_1 \sin\theta_x = d_2 \sin\theta_x^*$, such that $l_1 = l_2^*$, as shown in Fig. 3(d), the final adapted angle θ_x^* is:

$$\theta_x^* = \arcsin\left(\frac{d_1}{d_2}\sin\theta_x\right), \tag{2}$$

during task performing, to achieve the adapted state θ_x^*, the process is: (1) input the current adapted state $\theta_{t0}^1 = \arcsin(\frac{d_2 \sin\theta_{t0}^2}{d_1})$ to the policy π_θ, obtaining the action $\Delta\theta_{t0}^1$ for the robot 1. (2) The next expected state $\theta_{t1}^1 = \theta_{t0}^1 + \Delta\theta_{t0}^1$. (3) The adapted action $\Delta\theta_{t0}^2 = \arcsin(\frac{d_1 \sin\theta_{t1}^1}{d_2}) - \theta_{t0}^2$ for the robot 2. (4) The robot 2 executes the action $\Delta\theta_{t0}^2$. Reply (1)-(4) until the adapted state θ_x^* is reached in the robot 2.

During policy execution, the transformed pose $\mathcal{S}'_{\text{rob}} = (x, y, z'_{(i)}, \theta'_{(i)}, g'_{(i)})$ is fed to π_θ instead of $\mathcal{S}_{\text{rob}}$ in observations $\mathcal{O}$, maintaining trajectory consistency across grippers and manipulators.

Task-satisfied and Safety-constrained Trajectory Projection While the configuration adaptation of different grippers aligns geometric parameters, visual perception differences from gripper morphology can still induce unsafe trajectory variations. To guarantee constraint satisfaction, we integrate a projection layer into the DDIM denoising process [35]. The modified reverse diffusion step becomes:

$$\mathbf{a}_t^{k-1} = \text{Proj}_{\mathcal{C}}\left(\mu_k(\mathbf{a}_t^k, \epsilon_\theta(\mathbf{a}_t^k, \mathbf{o}_t, k))\right), \tag{3}$$

where $\text{Proj}_{\mathcal{C}}(\cdot)$ enforces safety constraints $\mathcal{C}$ through the following two steps.

*a) **Constraint-aware denoising**:* For efficiency, projection activates only in the final denoising steps ($k \leq 5$). At each step k, we solve a quadratic program problem:

$$\begin{aligned}
&\nu_t^{k*} = \arg\min_{\nu_t^k} \|\nu_t^k\|_2^2 \\
&\text{s.t. } \|\mathcal{S}'_{\text{rob}}(z)_t + \Phi\left(\mathbf{a}_t^k\right) + \nu_t^k\|_1 \geq \epsilon_{\text{safe}}, && \text{(4a)} \\
&\quad \|\arcsin\left[\tfrac{d_1}{d_2}\sin\left(\mathcal{S}'_{\text{rob}}(\theta)_t + \Phi(\mathbf{a}_t^k)\right)\right] - && \text{(4b)} \\
&\qquad \arcsin\left[\tfrac{d_1}{d_2}\sin(\mathcal{S}'_{\text{rob}}(\theta)_t)\right] - \nu_t^k\|_1 \leq \epsilon_{\text{task}},
\end{aligned}$$

where $\Phi(\cdot)$ maps latent actions to Cartesian displacement, which is denormalization in our case, $\epsilon_{\text{safe}} = 0.01$ m (safety margin), $\epsilon_{\text{task}} = 0.05$ rad (task margin), and $\nu_t^k \in \mathbb{R}^6$ is the corrective offset for translational and rotational movements.

*b) **Temporal consistency enforcement**:* To maintain safety over the policy's T_a-step action horizon ($j \in [0, T_a - 1]$), we extend (4a) and (4b) with cumulative constraints:

$$\|\mathcal{S}'_{\text{rob}}(z)_t + \sum_{r=0}^{j} \Phi(\mathbf{a}_{t+r}^k) + \nu_{t+j}^k\|_1 \geq \epsilon_{\text{safe}} \tag{5}$$

$$\begin{aligned}
&\|\arcsin\left[\tfrac{d_1}{d_2}\sin\left(\mathcal{S}'_{\text{rob}}(\theta)_t + \sum_{r=0}^{j}\Phi(\mathbf{a}_{t+r}^k)\right)\right] - \\
&\quad \arcsin\left[\tfrac{d_1}{d_2}\sin(\mathcal{S}'_{\text{rob}}(\theta)_t)\right] - \nu_{t+j}^k\|_1 \leq \epsilon_{\text{task}}
\end{aligned} \tag{6}$$

The projected actions $\mathbf{a}_t^{k*} = \Phi^-[\Phi(\mathbf{a}_t^k) + \nu_t^{k*}]$ guarantee:

$$\mathbb{P}\left(\bigcap_{j=0}^{T_a}\{\mathcal{S}'_{\text{rob}}(z)_{t+j} \geq \epsilon_{\text{safe}}\}\right) = 1, \tag{7}$$

indicating the cumulative trajectory is always safe, with an example of $\mathcal{S}'_{\text{rob}}(z)_{t+1} = \mathcal{S}'_{\text{rob}}(z)_t + \Phi(\mathbf{a}_t^{k*})$.

IV. Experiments

We evaluate the effectiveness of our adaptation-projection strategy in improving the applicability of the trained diffusion policies for pick-and-place, pushing, and pouring tasks across manipulator configurations and task requirements. Each task's demonstrations are collected with a Franka robot equipped with two gripper configurations (the Franka gripper with flexible fingertips or Robotiq-2f gripper) at 2Hz. For

Algorithm 1 Cross-manipulator Trajectory Generation

Require: Novel gripper $\mathbb{G}_i$, Observation $\mathbf{o}_t$, trained noise predicted ϵ_θ **(Inference Stage)**

Ensure: Safe and task-specified trajectory $\tau = \{\mathbf{a}_{t:t+T_a-1}\}$

$\mathcal{S}'_{\text{rob}} \leftarrow$ Equation (1), (2)//**Adaptated Robot State**

$\tilde{\mathbf{o}}_t \leftarrow \mathcal{S}_{\text{sce}} \times \mathcal{S}'_{\text{rob}}$

$\mathbf{a}_t^K \sim \mathcal{N}(0, \mathbf{I})$ // **Diffusion Process:**

repeat

  $k \leftarrow K-1$, and $K \leftarrow K-1$

  $\mathbf{a}_t^k \leftarrow \mathcal{N}\big(\mu_k(\mathbf{a}_t^{k+1}, \epsilon_\theta(\mathbf{a}_t^{k+1}, \tilde{\mathbf{o}}_t, k+1)), 0\big)$

  if $k \leq 5$: // **Multi-objective Projection:**

    for $j \leftarrow 0$ **to** $T_a - 1$ **do**

      $\nu_{t+j}^{k*} \leftarrow \arg\min_{\nu} \|\nu_{t+j}^k\|_2^2$

      s.t. Cumulative constraints (5), (6)

    $\mathbf{a}_{t:t+T_a-1}^k \leftarrow \Phi^-[\Phi(\mathbf{a}_{t:t+T_a-1}^k) + \nu_{t:t+T_a-1}^{k*}]$

until $K = 0$

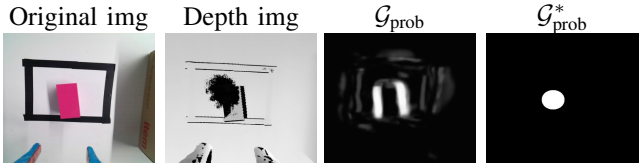
$\tau \leftarrow \text{Decode}(\mathbf{a}_{t:t+T_a-1}^0)$

example, the pick-and-place and pushing tasks are performed with the former gripper, while the pouring task is performed with the latter. Each task policy is trained with around 60 demonstrations, and the training scheme is consistent with the Simple DP3 framework [8] without any modifications.

A. Pick-and-Place Task

We first evaluate the policy's performance in pick-and-place tasks with cross-gripper and cross-manipulator setups. (a) **Baselines**: Diffusion Policy (DP) [7], Diffusion Policy 3D (DP 3D) [8], BC-z [36], Ours w/o AP (Adaptation-Projection strategy), and Ours w/o $\mathcal{G}^*_{\text{prob}}$. (b) **Setup**: Our policy and baselines are evaluated in different task settings, including different grippers, different objects, and placing the object to different-height platforms. Notably, all training demonstrations included the base gripper ($\mathbb{G}_0$), a single object (a pink block), and a fixed-height platform (9 cm). In this task, the visual observation contains global-scenario point clouds acquired via a Realsense L515 camera, and a ego-camera Kinect Azure to collect RGB-D images. A failure trial refers to failing to grasp the object, or dropping it during manipulation, or encountering collisions.

Incorporate gripper-agnostic grasping knowledge: To investigate how switching grippers with different morphologies in the policy's inference stage affect the policy's performance, we incorporate gripper-agnostic grasping knowledge during policy training as a comparison. Specifically, we introduce a gripper-agnostic *grasping probability map* $\mathcal{G}_{\text{prob}}$ as an additional observation component. This map captures object-centric grasp affordances that are independent of end-effector geometry, guiding the policy to focus on relevant object features rather than gripper-specific visual patterns. By decoupling object-related cues from the gripper's appearance, $\mathcal{G}_{\text{prob}}$ enhances the policy's robustness to variations in gripper morphology. We adopt the Generative Grasping CNN (GG-CNN) [37] for $\mathcal{G}_{\text{prob}}$ synthesis from depth images. However, real-world pick-and-place manipulations introduce two key

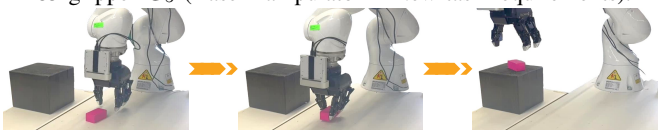


(a) gripper: flexible, EE height: 18cm, object: block

(b) gripper: Robotiq-2f, EE height: 14cm, object: banana

Fig. 5: The gripper-agnostic grasping knowledge. Dynamic changes in robot pose and gripper variants cause changes in visual observations, including RGB-D images and object grasp probabilities, $\mathcal{G}^*_{\text{prob}}$ provides stable visual information.

(a) Picking and placing an unseen banana with Franka + Robotiq 2F-85 gripper $\mathbb{G}_5$ (Base manipulator + New task requirements).



(b) Picking a block with Kuka + Robotiq 3F Gripper $\mathbb{G}_6$ and placing it on a higher platform (22 cm) (New manipulator).

Fig. 6: The rollout of our method for solving the pick-and-place task with unseen gripper and objects.

challenges: (1) the hand-eye camera moving with the robot, causing scale variations in object pixels, and (2) lighting changes disturb depth sensor readings. These factors degrade GG-CNN's output stability, i.e., $\mathcal{G}_{\text{prob}}$, and destabilize policy training and inference performance. To address this issue, our solution involves: (1) threshold filtering: discard pixels with $\mathcal{G}_{\text{prob}} < 0.7$, (2) centroid computation: $\mathbb{O} = \frac{1}{N}\sum_{i=1}^{N}(u_i, v_i)$ for remaining pixels, and (3) region masking: generate $\mathcal{G}^*_{\text{prob}}$ through circular masking ($r = 30$ pixels) about $\mathbb{O}$. We opt for this 2D calculation instead of a 3D point cloud centroid because the ego-camera (Kinect Azure) focuses directly on the object (often from a top-down view), yielding a precise 2D center. In contrast, the global camera (L515) is intended for scene perception; the resulting object point cloud is sparse and, due to the camera angle, concentrated on specific surfaces (e.g., side faces), causing 3D centroid estimates to be inaccurate. The map satisfies $\mathcal{G}^*_{\text{prob}}(u, v) = 1$, if $\|(u, v) - \mathbb{O}\|_2 \leq 30$. This spatial filtering maintains grasp affordance information while eliminating outlier predictions caused by sensor noise, as shown in Fig.5. The policy observation consists of $\mathcal{O}^* = \mathcal{G}^*_{\text{prob}} \times \mathcal{S}_{\text{sce}} \times \mathcal{S}'_{\text{rot}}$.

Table I summarizes the success rates of different policies across-manipulator configurations (e.g., robot and gripper types) and different task requirements (e.g., object types and placement heights). Here, $\mathbb{G}_i$ represents different grippers

TABLE I: Pick-and-place task across grippers (unit: %).

Method	$\mathbb{G}_0$	$\mathbb{G}_1$	$\mathbb{G}_2$	$\mathbb{G}_3$	$\mathbb{G}_4$	$\mathbb{G}_5$	$\mathbb{G}_6$
Case 1: Robot: Franka (seen), Object: Block (seen), Platform height: 9cm							
Diffusion Policy	20.0	0.0	60.0	40.0	0.0	40.0	-
Diffusion Policy 3D	20.0	0.0	60.0	60.0	0.0	0.0	-
DP + AP w/o $\mathcal{G}^*_{\text{prob}}$	100.0	-	-	-	-	-	-
DP 3D + AP w/o $\mathcal{G}^*_{\text{prob}}$	80.0	-	-	-	-	-	-
BC-z w/o AP w/o $\mathcal{G}^*_{\text{prob}}$	20.0	0.0	20.0	20.0	0.0	0.0	-
Ours w/o AP	100.0	0.0	60.0	40.0	0.0	0.0	-
Ours w/o $\mathcal{G}^*_{\text{prob}}$	80.0	20.0	40.0	80.0	100.0	0.0	-
Ours	**100.0**	**80.0**	**100.0**	**80.0**	**100.0**	**100.0**	**-**
Case 2: Robot: Franka (seen), Object: Banana (unseen), Platform height: 9cm							
Diffusion Policy	20.0	0.0	40.0	**60.0**	40.0	20.0	-
Diffusion Policy 3D	20.0	0.0	40.0	40.0	20.0	0.0	-
Ours w/o AP	80.0	0.0	40.0	40.0	0.0	20.0	-
Ours w/o $\mathcal{G}^*_{\text{prob}}$	60.0	0.0	40.0	**60.0**	40.0	0.0	-
Ours	**80.0**	**60.0**	**100.0**	**60.0**	**60.0**	**60.0**	**-**
Case 3: Robot: Kuka (unseen), Object: Block (seen), Platform height: 22cm							
Diffusion Policy	-	-	-	-	-	0.0	0.0
Diffusion Policy 3D	-	-	-	-	-	0.0	0.0
Ours	-	-	-	-	-	**80.0**	**80.0**

- The training data is collected with the base gripper $\mathbb{G}_0$ and a pink block.
- The initial pose and position of the object are identical across tests.
- $\mathbb{G}_i$ indicates different grippers reported in Fig. 2. w/o: without this module. - Every method is validated 5 times for each gripper, totaling 30 evaluations. The Robotiq-3f gripper ($\mathbb{G}_6$) is too heavy to equip on Franka robot, thus $\mathbb{G}_6$ is evaluated on Kuka robot.

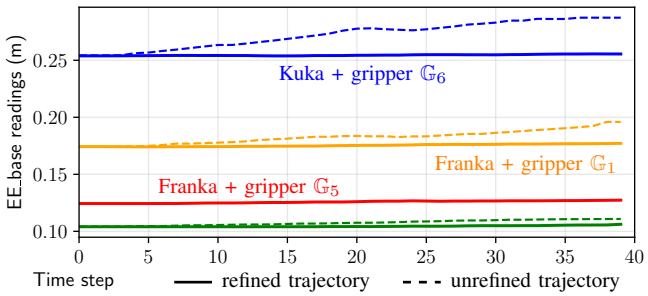
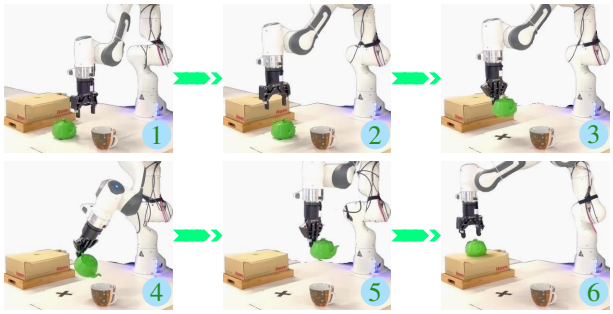


Fig. 7: Result of the pushing task in different configurations.

in Fig. 2 and $\mathbb{G}_0$ is the base gripper. The results showcase the following trends: (1) **Our full method** consistently outperforms the baselines across all different manipulator configurations, demonstrating its effectiveness in adapting to different manipulator and task setups. (2) **Baseline policies (DP, DP 3D, BC-z)** perform poorly, often generating imprecise trajectories that lead to collisions for longer gripper ($\mathbb{G}_1$) and unable grasping for short one ($\mathbb{G}_4$). Their performance is highly sensitive to the specific gripper geometry and they fail to adapt to unseen objects, especially when object positions are randomized. (3) **The ablation study without the Adaptation-Projection strategy (Ours w/o AP)** fails when using grippers with geometries that significantly differ from the base gripper. This highlights the critical role of the projection module in adapting to physical hardware changes. (4) **The ablation study without gripper-agnostic knowledge (Ours w/o $\mathcal{G}^*_{\text{prob}}$)** shows a significant performance drop when grippers appear visually different from the egocentric camera's view or when handling unseen objects. This underscores the importance of decoupling grasp-relevant features from the specific gripper's appearance for robust generalization. (5) **Adapting to New Task Requirements:** Our method seamlessly adapts to new task requirements, such as the varying platform heights in Case 3. We encode



(a) Pouring with Franka + Robotiq 2F-85 (Base manipulator).

(b) Pouring with Kuka + Robotiq 3F (New manipulator).

Fig. 8: Rollout of pouring task with different manipulators.

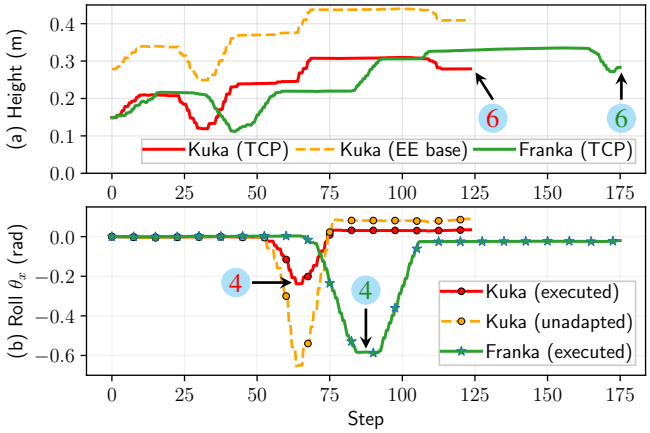


Fig. 9: Result of the pouring task in different manipulators.

these changes directly into our adaptation process by treating the height difference as a TCP shift and applying the translational adaptation from (1). This allows our method to maintain high performance, even with challenging grippers like $\mathbb{G}_6$, whereas baseline methods completely fail to adapt to these changes. Fig. 6 visualizes a successful rollout under these modified task and manipulator configurations.

B. Pushing task

To isolate and evaluate the effect of our adaptation-projection strategy on geometric variations alone, we design a pushing task in which the policy relies exclusively on low-dimensional robot states, without incorporating any visual input. This configuration eliminates confounding effects from visual changes and directly tests the policy's capacity to accommodate various manipulator morphologies. In this task, the robot is required to push an object forward (with the object initially placed on the table and grasped by the

gripper). Without our strategy, switching between different manipulator configurations causes variations in the EE_base readings, leading to out-of-distribution issues in the robot's action-state distribution and resulting in unintended behaviors, such as lifting the object during the pushing motion (see Fig.7). In contrast, our strategy successfully refines the generative trajectories across configurations, ensuring reliable task execution.

C. Pouring tea task

Pouring task is designed to evaluate our method on a more complex, multi-stage manipulation sequence. This requires the robot to pick up a teapot, pour water into a cup with precision, and then place the teapot back on a platform. This task is particularly challenging as it demands accurate control of both position and orientation throughout the entire trajectory. Our adaptation-projection strategy is critical for dynamically refining the generated trajectory to account for TCP shifts between the base and new robot configurations in both translation (1) and rotation (2). This ensures collision-free motion during the pick-and-place phases and enables accurate, spill-free pouring. Fig. 8 demonstrates rollout of pouring in different manipulators, and Fig. 9 shows the corresponding trajectories in different axes. The results demonstrate that our method successfully adapts the trained policy to different manipulators, refining the trajectory in translational (z-axis in Fig. 9(a)) and rotational (θ_x in Fig. 9(b)) axes and ensuring task completions in pouring tasks across various configurations.

V. ACKNOWLEDGMENTS

This research is supported by New Generation Artificial Intelligence-National Science and Technology Major Project (2025ZD0122903).

VI. CONCLUSIONS AND OUTLOOK

This paper introduces an inference-stage adaptation-projection strategy for diffusion policy, transferring manipulation skills across different grippers and robots. This transition does not require retraining or fine-tuning the policy with the new manipulator configuration. Instead, it only needs to introduce the configuration or new task requirements in the policy inference phase, refining the generated trajectories to satisfy safety constraints and task completion. This approach effectively reduces the time and cost of data collection and model training for each new manipulator or task. We validate our method on various manipulation tasks, including pick-and-place, pushing, and pouring, using different robots (including the Franka Panda and Kuka iiwa 14) and parallel grippers (e.g., Robotiq 2F/3F grippers, flexible fingertips, 3D-printed designs). The results demonstrate that our method achieves high success rates in these cross-manipulator scenarios, showcasing its effectiveness and practicality.

Despite the demonstrated effectiveness of our adaptation-projection strategy in handling significant TCP shifts and jaw width variations, this approach presents several limitations. First, the kinematic dimension differences currently require manual measurement and configuration at the inference stage, imposing a minor manual overhead. Future work could explore automatic calibration techniques—for example, leveraging visual point cloud data—to infer geometric discrepancies between grippers and streamline the deployment process. Second, imposing new task objectives (such as placing objects onto platforms of different heights) currently necessitate manual encoding within the adaptation-projection process. This dependency can be mitigated by integrating language-conditioned models [38], [39] (e.g., LLMs, VLMs) to translate human's natural language instructions that contains new task objectives into inference-stage projection constraints. Third, our current method primarily addresses geometric and kinematic constraints, future research aims to incorporate dynamic constraints, including force control and compliance, to enhance adaptability in complex real-world applications. Moreover, the optimization strategy (quadratic programming) of the projection process can integrates with advanced control strategy to achieve real-time obstacle avoidances in the inference stage [40]–[42]. Finally, extending this approach to diverse manipulation tasks and end-effectors, such as deformable object manipulation [43], [44] and multi-fingered hands, would further validate its versatility and broader applicability.

REFERENCES

[1] H. Zhou, Z. Bing, X. Yao, X. Su, C. Yang, K. Huang, and A. Knoll, "Language-conditioned imitation learning with base skill priors under unstructured data," *IEEE Robotics and Automation Letters*, vol. 9, no. 11, pp. 9805–9812, 2024.

[2] F. Di Felice, S. D'Avella, A. Remus, P. Tripicchio, and C. A. Avizzano, "One-shot imitation learning with graph neural networks for pick-and-place manipulation tasks," *IEEE Robotics and Automation Letters*, 2023.

[3] T. Migimatsu and J. Bohg, "Object-centric task and motion planning in dynamic environments," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 844–851, 2020.

[4] X. Yao, Z. Bing, G. Zhuang, K. Chen, H. Zhou, K. Huang, and A. Knoll, "Learning from symmetry: Meta-reinforcement learning with symmetrical behaviors and language instructions," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 5574–5581.

[5] Z. Bing, A. Koch, X. Yao, K. Huang, and A. Knoll, "Meta-reinforcement learning via language instructions," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 5985–5991.

[6] Y. Meng, Z. Bing, X. Yao, K. Chen, K. Huang, Y. Gao, F. Sun, and A. Knoll, "Preserving and combining knowledge in robotic lifelong reinforcement learning," *Nature Machine Intelligence*, vol. 7, no. 2, pp. 256–269, 2025.

[7] 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*, vol. 44, no. 10-11, pp. 1684–1704, 2025.

[8] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu, "3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations," in *Proceedings of Robotics: Science and Systems (RSS)*, 2024.

[9] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[10] X. Ma, S. Patidar, I. Haughton, and S. James, "Hierarchical diffusion policy for kinematics-aware multi-task robotic manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 18 081–18 090.

[11] Z. Xue, S. Deng, Z. Chen, Y. Wang, Z. Yuan, and H. Xu, "Demogen: Synthetic demonstration generation for data-efficient visuomotor policy learning," *arXiv preprint arXiv:2502.16932*, 2025.

[12] J. H. Yang, D. Sadigh, and C. Finn, "Polybot: Training one policy across robots while embracing variability," in *Proceedings of The 7th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, vol. 229. PMLR, 06–09 Nov 2023, pp. 2955–2974.

[13] T. Wang, D. Bhatt, X. Wang, and N. Atanasov, "Cross-embodiment robot manipulation skill transfer using latent space alignment," *arXiv preprint arXiv:2406.01968*, 2024.

[14] L. Wang, X. Chen, J. Zhao, and K. He, "Scaling proprioceptive-visual learning with heterogeneous pre-trained transformers," *Advances in neural information processing systems*, vol. 37, pp. 124 420–124 450, 2024.

[15] L. Y. Chen, K. Hari, K. Dharmarajan, C. Xu, Q. Vuong, and K. Goldberg, "Mirage: Cross-embodiment zero-shot policy transfer with cross-painting," *Robotics: Science and Systems (RSS)*, 2024.

[16] E. Bauer, E. Nava, and R. K. Katzschmann, "Latent action diffusion for cross-embodiment manipulation," *arXiv preprint arXiv:2506.14608*, 2025.

[17] J. Chen, Y. Ke, L. Peng, and H. Wang, "Dexonomy: Synthesizing all dexterous grasp types in a grasp taxonomy," *Robotics: Science and Systems*, 2025.

[18] M. Seo, H. A. Park, S. Yuan, Y. Zhu, and L. Sentis, "Legato: Cross-embodiment imitation using a grasping tool," *IEEE Robotics and Automation Letters*, vol. 10, no. 3, pp. 2854–2861, 2025.

[19] R. Freiberg, A. Qualmann, N. A. Vien, and G. Neumann, "Diffusion for multi-embodiment grasping," *IEEE Robotics and Automation Letters*, vol. 10, no. 3, pp. 2694–2701, 2025.

[20] K. Zakka, A. Zeng, P. Florence, J. Tompson, J. Bohg, and D. Dwibedi, "Xirl: Cross-embodiment inverse reinforcement learning," in *Conference on Robot Learning*. PMLR, 2022, pp. 537–546.

[21] G. Tatiya, J. Francis, and J. Sinapov, "Transferring implicit knowledge of non-visual object properties across heterogeneous robot morphologies," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 11 315–11 321.

[22] X. Yao, Y. Zhou, Y. Meng, L. Dong, L. Hong, Z. Zhang, Z. Bing, K. Huang, F. Sun, and A. Knoll, "Pick-and-place manipulation across grippers without retraining: A learning-optimization diffusion policy approach," *arXiv preprint arXiv:2502.15613*, 2025.

[23] J. Carvalho, A. T. Le, M. Baierl, D. Koert, and J. Peters, "Motion planning diffusion: Learning and planning of robot motions with diffusion models," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 1916–1923.

[24] K. Mizuta and K. Leung, "Cobl-diffusion: Diffusion-based conditional robot planning in dynamic environments using control barrier and lyapunov functions," pp. 13 801–13 808, 2024.

[25] M. Reuss, M. Li, X. Jia, and R. Lioutikov, "Goal conditioned imitation learning using score-based diffusion policies," in *Robotics: Science and Systems*, 2023.

[26] Z. Liang, Y. Mu, M. Ding, F. Ni, M. Tomizuka, and P. Luo, "Adaptdiffuser: Diffusion models as adaptive self-evolving planners," in *International Conference on Machine Learning*. PMLR, 2023, pp. 20 725–20 745.

[27] S. Venkatraman, S. Khaitan, R. T. Akella, J. Dolan, J. Schneider, and G. Berseth, "Reasoning with latent diffusion in offline reinforcement learning," in *The Twelfth International Conference on Learning Representations*, 2024.

[28] J.-H. Bastek, W. Sun, and D. M. Kochmann, "Physics-informed diffusion models," *arXiv preprint arXiv:2403.14404*, 2024.

[29] J. Ho and T. Salimans, "Classifier-free diffusion guidance," *arXiv preprint arXiv:2207.12598*, 2022.

[30] Y. Yuan, J. Song, U. Iqbal, A. Vahdat, and J. Kautz, "Physdiff: Physics-guided human motion diffusion model," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 16 010–16 021.

[31] J. K. Christopher, S. Baek, and F. Fioretto, "Constrained synthesis with projected diffusion models," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[32] F. Mazé and F. Ahmed, "Diffusion models beat gans on topology optimization," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 37, no. 8, 2023, pp. 9108–9116.

[33] G. Giannone, A. Srivastava, O. Winther, and F. Ahmed, "Aligning optimization trajectories with diffusion models for constrained design generation," *Advances in Neural Information Processing Systems*, vol. 36, pp. 51 830–51 861, 2023.

[34] R. Römer, A. von Rohr, and A. Schoellig, "Diffusion predictive control with constraints," pp. 791–803, 2025.

[35] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *International Conference on Learning Representations*, 2021.

[36] E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, and C. Finn, "Bc-z: Zero-shot task generalization with robotic imitation learning," in *Conference on Robot Learning*. PMLR, 2022, pp. 991–1002.

[37] D. Morrison, P. Corke, and J. Leitner, "Learning robust, real-time, reactive robotic grasping," *The International journal of robotics research*, vol. 39, no. 2-3, pp. 183–201, 2020.

[38] X. Yao, H. Zhou, O. Mees, Y. Meng, T. Xiao, Y. Bisk, J. Oh, E. Johns, M. Shridhar, D. Shah, J. Thomason, K. Huang, J. Chai, Z. Bing, and A. Knoll, "Bridging language and action: A survey of language-conditioned robot manipulation," 2025. [Online]. Available: https://arxiv.org/abs/2312.10807

[39] Y. Meng, X. Yao, H. Ye, Y. Zhou, S. Zhang, Z. Bing, and A. Knoll, "Data-agnostic robotic long-horizon manipulation with vision-language-guided closed-loop feedback," *arXiv preprint arXiv:2503.21969*, 2025.

[40] Y. Zhang, G. Tian, L. Wen, X. Yao, L. Zhang, Z. Bing, W. He, and A. Knoll, "Online efficient safety-critical control for mobile robots in unknown dynamic multi-obstacle environments," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024, pp. 12 370–12 377.

[41] Y. Zhang, L. Wen, X. Yao, Z. Bing, L. Kong, W. He, and A. Knoll, "Real-time adaptive safety-critical control with gaussian processes in high-order uncertain models," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 14 763–14 769.

[42] Y. Zhang, L. Wen, Z. Bing, X. Yao, L. Kong, W. He, and A. Knoll, "Adaptive safety-critical control for high-order systems: A real-time gaussian process approach," *IEEE Transactions on Automation Science and Engineering*, vol. 22, pp. 21 296–21 310, 2025.

[43] K. Chen, Z. Bing, F. Wu, Y. Meng, A. Kraft, S. Haddadin, and A. Knoll, "Contact-aware shaping and maintenance of deformable linear objects with fixtures," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 1–8.

[44] K. Chen, Z. Bing, Y. Wu, F. Wu, L. Zhang, S. Haddadin, and A. Knoll, "Real-time contact state estimation in shape control of deformable linear objects under small environmental constraints," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 13 833–13 839.