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

Dexterous Planar Pushing under Uncertain Object Properties: A Contact-Aware Goal-Oriented Approach

Yongseok Lee and Keehoon Kim, *Senior Member, IEEE*

***Abstract*— Robotic pushing is a versatile non-prehensile manipulation skill that enables robots to handle ungraspable objects without specialized tools. This paper introduces a contact-aware, goal-oriented pushing framework that achieves dexterous and robust manipulation by explicitly allowing free-motion of the end-effector. Central to our approach is the contact-aware generalized velocity–motion model (C-GVMM), which captures the relationship between pusher velocity and slider motion across all contact modes, including separation. Unlike prior methods that rely on predefined trajectories or fixed contact-mode sequences, our framework enables seamless transitions among sticking, sliding, and separating modes. Building upon C-GVMM, we employ Model Predictive Path Integral (MPPI) control to generate goal-directed actions, and UKF-based online estimation to handle the uncertain object properties in real-world setting. We validate our approach through both numerical simulations and real-robot experiments, demonstrating that the framework accomplishes diverse pushing tasks with more optimal pusher and slider motion with high success rates. These results demonstrate the practical viability of the proposed approach for real-world robotic pushing tasks.**

I. INTRODUCTION

Robotic systems are increasingly integrated into daily applications to improve efficiency and autonomy. A foundational skill enabling such integration is robotic manipulation, which involves physically interacting with and changing the state of objects in the environment [1]. Manipulation strategies are broadly categorized into prehensile and non-prehensile approaches [2]. Prehensile manipulation employs stable force-closure grasps, but often suffers from geometric constraints imposed by the end-effector design. In contrast, non-prehensile manipulation avoids grasping altogether and instead exploits contact interactions such as pushing and toppling, significantly expanding the range of objects and scenarios that robots can handle [3].

Robotic pushing has been extensively studied for both robotic arms [4], [5], [6], [7], [8] and quadrupedal platforms [9], [10], [11], due to its intuitive and versatile nature. Pushing plays a crucial role in enabling pre-grasp manipulation [12], dense object packing [13], and large-scale environment rearrangement tasks [9]. Despite these practical benefits, robotic pushing remains fundamentally challenging because of its hybrid and mode-dependent contact dynamics combined with constraints imposed by underactuated control. During pushing, interactions continuously switch between sticking, sliding, and separation modes, each governed by distinct physical laws such as friction cones and surface contacts. These abrupt switches introduce discontinuities that complicate both system modeling and real-time control. As a result, achieving precise and reliable goal-oriented pushing under unpredictable hybrid behaviors is notably challenging.

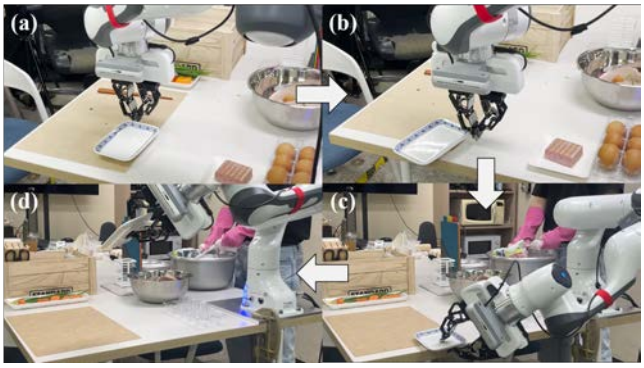

Fig. 1. The results of autonomous tableware cleaning with proposed contact-aware dexterous goal-oriented pushing manipulation. The robot can handle ungraspable thin object without any predefined trajectory and contact-mode under uncertain object properties.

A core requirement for overcoming the challenges in robotic pushing is sophisticated contact-mode reasoning: the robot must actively determine which contact mode (stick, slide, or separate) to engage at each moment to accomplish manipulation tasks accurately. Recent advances in model-free pushing approaches [14], [7] have demonstrated scalability and dexterity; however, these methods often struggle to generalize effectively to unseen objects. Conversely, model-based optimization approaches offer better generalization under model uncertainties such as unknown mass distribution and surface friction by incorporating online parameter estimation technique [15]. Despite this, many existing frameworks rely on predefined contact-mode sequences for mode-transition pushing [4], [6], which inherently limits dexterity and restricts the robot's reachable manipulation space.

Equally important is explicitly enabling free-motion of the end-effectors. By allowing separation and subsequent re-approach motions, the robot can reset contact geometries, access multiple object surfaces, and avoid undesirable jamming or stuck configurations. This capability enriches available manipulation strategies and strengthens robustness for execution in complex real-world environments [6]. Motivated by these insights, we propose a contact-aware, goal-oriented pushing framework that eliminates the restrictive non-separation assumption and leverages free-motion to

This work was supported by the Industrial Strategic Technology Development Program (RS-2024-00442029) funded by the Ministry of Trade, Industry and Energy (MOTIE, Korea)

All authors are with Department of Mechanical Engineering, Pohang University of Science and Technology (POSTECH), 37673, Gyeong-buk, South Korea. Email: {dldydtjr2000, khk}@postech.ac.kr

achieve both dexterous and robust manipulation.

In this paper, we introduce a robotic pushing framework enabling dexterous, goal-oriented manipulation without dependence on predefiend trajectories or contact-mode sequences, explicitly including phases of free motion. To this end, we first derive a *contact-aware generalized velocity–motion model* (C-GVMM) that captures the relationship between the pusher's velocity and the induced motion of the slider across all contact modes, including separation mode. This model enables reliable prediction of object behavior without enforcing the commonly assumed non-separation constraint. Building upon C-GVMM, we employ Model Predictive Path Integral (MPPI) control as a goal-oriented pushing strategy capable of effectively handling contact-rich, non-differentiable dynamics and cost functions. Leveraging C-GVMM's flexibility in accommodating contact-mode transitions, MPPI can simply generate control inputs by sampling candidate velocities, simulating trajectories, evaluating costs, and aggregating outcomes through importance weighting. Also, we leveraged the UKF-based online model estimation scheme [15] to handle the uncertain object properties. Extensive numerical simulations and real-robot experiments demonstrate that the proposed framework achieves dexterous pushing behaviors while naturally incorporating contact-free motions.

The contributions of this work are threefold:

- **(Contact-Aware Pushing Motion Model)** We propose a contact-aware pushing motion model that automatically infers contact modes (sticking, sliding, separating) and validate its accuracy on MIT pushing dataset [16].
- **(Goal-Oriented Dexterous Pushing Framework)** We achieve accurate and dexterous goal-oriented robotic pushing by integrating the contact-aware pushing motion model with an MPPI-based control and UKF-based online parameter estimation.
- **(Real-World Validation)** We validate the effectiveness of our approach through comprehensive numerical experiments and real-robot demonstrations.

II. Related Work

In this section, we review work on robotic pushing in two threads: (i) pushing mechanics and (ii) goal-oriented pushing strategies.

A. Pushing Mechanics

A central objective in robotic pushing is to model how contact interactions induce object motion. The canonical quasi-static planar formulation is built on the *limit surface* (LS) of frictional wrenches under uniform pressure and Coulomb friction, introduced by Goyal et al. [17]. The LS yields two widely used analytical models: (i) *force–motion models* (FMM), which map contact wrench to object twist [18]; and (ii) *velocity–motion models* (VMM), which map pusher velocity to object twist [19]. These models were verified and utilized in various planar pushing works.

While these models provide valuable structure and insight, their fidelity can degrade in practice due to simplifying assumptions (e.g., spatially uniform pressure, isotropic friction) [16]. To address this gap, two complementary approaches have emerged. *Data-driven* approaches learn from empirical push data to capture stochastic, configuration-dependent effects [20], [21], [22]. These methods often require substantial offline datasets and may struggle to extrapolate beyond observed regimes. In contrast, *online model adaptation* updates the model during execution such as LSTM-based [23], online updated GP-based [24], and UKF-based online parameter estimation [15]. In this paper, we adopt this latter paradigm, leveraging UKF-based online parameter estimation to handle real-world uncertainty on physics properties.

B. Goal-Oriented Pushing Manipulation

Goal-oriented pushing has been approached either via a plan–track pipeline or by directly synthesizing closed-loop goal-reaching policies. In plan–track, trajectories are first generated under hybrid, underactuated contact dynamics and then tracked. Planning formulations such as Dubins style reductions under persistent sticking [25], complementarity-based full-horizon optimization allowing stick–slip [5], graph-of-convex-sets relaxations for shortest paths [26], and kinodynamic sampling with contact awareness [27]. Tracking commonly employs MPC like mixed-integer MPC [4] or nonlinear MPC [5], and differential dynamic programming (DDP) [6]. Despite strong nominal performance, these methods remain sensitive to model mismatch; online parameter identification can mitigate this sensitivity [15] but still presumes precomputed nominal trajectory and non-separation assumption which restrict the dexterity.

Direct goal-oriented methods bypass explicit nominal trajectories by synthesizing closed-loop goal-reaching policies. One example is visuomotor policies [28], [29], [14] or DRL policy [30] that map observations to desired pushing actions directly. These results also extend to pushing with mobile and quadrupedal platforms in large workspaces [9], [10]. Alternatively, sampling-based optimal control, especially model predictive path integral (MPPI) [31], has emerged as a versatile goal-oriented pushing action generator. Cong et al. [23] combined LSTM-based forward models with MPPI, yielding reactive policies. Lee et al. [8], which is highly related to our work, combined online estimation models into long-term MPPI to achieve high accuracy in goal-oriented pushing under the non-separation assumption. However, the pushing manipulation under non-separation assumptions defected the dexterity of the robot system [6]. In this paper, we extended this work by allowing separation motion without any contact-mode planning.

III. Contact-aware dexterous robotic pushing

This section begins with a concise review of planar pushing mechanics, including the generalized version of conventional analytical pushing motion models. Then, the compact input formulation of these models is introduced in detail. Finally, we introduce a robust online model estimation

TABLE I
NOMENCLATURES

Symbol	Description
$\{\mathcal{S}\}$	World frame fixed to the ground
$\{\mathcal{B}\}$, $\{\mathcal{F}\}$	Body frame fixed to the reference/CoF
$\boldsymbol{p}_{sb} = [x_{sb}\ y_{sb}]^\top$	Position of the reference of the object in $\{\mathcal{S}\}$
θ_{sb}	Orientation of the object in $\{\mathcal{S}\}$
ϕ_c	Azimuth of the contact point in $\{\mathcal{B}\}$
$\boldsymbol{r}(\phi_c)$	Radial vector of the object contour in $\{\mathcal{B}\}$
$\boldsymbol{V}_{sb}$, $\boldsymbol{V}_{sf}$	Body twist of $\{\mathcal{B}\}/\{\mathcal{F}\}$ w.r.t $\{\mathcal{S}\}$
$\boldsymbol{F}_f$	Surface frictional wrench on object in $\{\mathcal{F}\}$
$\boldsymbol{f}_{bc}$, $\boldsymbol{v}_{bc}$	Contact force/velocity in $\{\mathcal{B}\}$
$\boldsymbol{\mu}_c$, $\boldsymbol{\mu}_g$	Contact friction coefficients

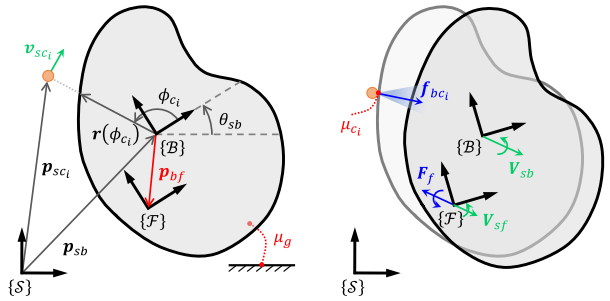

Fig. 2. Illustration of the pushing system: **(a)** kinematics of the pushing system. **(b)** free body diagram and motions of a pushed object.

scheme that continuously estimates those parameters online, thereby maintaining accurate pushing motion predictions in the presence of unknown or changing object properties.

A. Conditional contact friction constraints

Table I and Fig. 2 explain the notations used in this paper. Note that this paper employed a single-contact pushing for experiments in Sec IV, but we derive the generalized multi-contact pushing motion model (denoted by $\{c_i\}_{i=1}^{N_c}$ where N_c is the number of contacts) in this section.

The first step in pushing motion modeling is to define the contour shape of the object. For a star-shaped object, the contour shape can be expressed as a radial distance function $\boldsymbol{r}(\phi_{c_i})$ for a given azimuth ϕ_{c_i}. Then, the improper rotation matrix $\mathbf{N}(\phi_{c_i})$, which contains the normal and tangent information of the object contour, can be computed as

$$\mathbf{N}(\phi_{c_i}) = \begin{bmatrix} \hat{\boldsymbol{n}}_{c_i}^\top \\ \hat{\boldsymbol{t}}_{c_i}^\top \end{bmatrix} = \frac{1}{\|\partial_\phi \boldsymbol{r}\|_2} \begin{bmatrix} -\partial_\phi r_y & \partial_\phi r_x \\ \partial_\phi r_x & \partial_\phi r_y \end{bmatrix}_{\phi=\phi_{c_i}} \tag{1}$$

The pushing motion is induced by the external contact force $\boldsymbol{f}_{bc_i} = [f_{bc_i x}\ f_{bc_i y}]^\top = \mathbf{N}_i \cdot \boldsymbol{f}_{c_i}$. Here, the normal and tangential friction force $\boldsymbol{f}_{c_i} \triangleq [f_{n_i}\ f_{t_i}]^\top$ is constrained

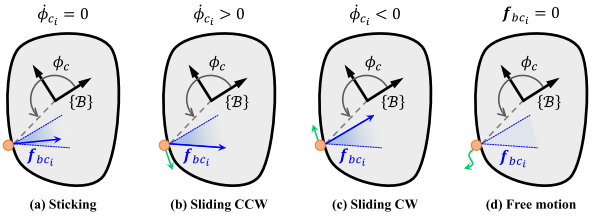

Fig. 3. The conditional contact force constraints induced from friction cone depend on various contact mode. **(a) sticking (b), (c) sliding CCW/CW (d) free motion**

depends on the four different contact modes as

$$\mathcal{C}: \begin{cases} \mathcal{C}_1: \dot{\phi}_{c_i} = 0,\ |f_{t_i}| < \mu_p f_{n_i} & \textbf{(sticking)} \\ \mathcal{C}_2: \dot{\phi}_{c_i} > 0,\ f_{t_i} = \mu_p f_{n_i} & \textbf{(sliding CCW)} \\ \mathcal{C}_3: \dot{\phi}_{c_i} < 0,\ f_{t_i} = -\mu_p f_{n_i} & \textbf{(sliding CW)} \\ \mathcal{C}_4: f_{n_i} = 0,\ f_{t_i} = 0 & \textbf{(free motion)} \end{cases}. \tag{2}$$

B. Contact-aware generalized velocity-motion model (C-GVMM)

Under the quasi-static pushing assumption, the motion equation of the pushing system can be derived from the ellipsoidal-approximated limit surface [32]. The limit surface represents a convex bounding set of all possible frictional wrenches $\boldsymbol{F}_f = [f_x\ f_y\ m_z]^\top$ as

$$H(\boldsymbol{F}_f) = \frac{1}{2}\boldsymbol{F}_f^\top \mathbf{L} \boldsymbol{F}_f, \tag{3}$$

where $\mathbf{L} = \mathrm{diag}\left(2/f_{\max}^2,\ 2/f_{\max}^2,\ 2/m_{\max}^2\right)$ and $f_{\max}$, $m_{\max}$ are the maximum frictional force and moment. By the principle of maximal dissipation, $\boldsymbol{V}_{sf}$ and $\boldsymbol{F}_f$ have the following relationship:

$$\boldsymbol{V}_{sf} = -s\nabla H(\boldsymbol{F}_f) = -s\mathbf{L}\boldsymbol{F}_f = -\tilde{\mathbf{L}}\boldsymbol{F}_f \tag{4}$$

where s is a proportionality constant. Note that the diagonal matrix $\tilde{\mathbf{L}}$ can be parametrized into $\alpha^2 \cdot \mathrm{diag}\left(1,\ 1,\ \beta^2\right)$ which are the model parameters that depend on the object properties.

Let an Adjoint matrix $\mathbf{Ad}_{ab} := \begin{bmatrix} \mathbf{I}_{2\times 2} & [y_{ab}\ -x_{ab}]^\top \\ \mathbf{0}_{1\times 2} & 1 \end{bmatrix}$ and Jacobian matrix $\mathbf{J}_{ab}^* := \begin{bmatrix} \mathbf{I}_{2\times 2} & [-y_{ab}\ x_{ab}]^\top \end{bmatrix}$ with respect to the frame $\{\mathcal{A}\}$ and $\{\mathcal{B}\}$, where $\mathbf{I}_{n\times n}$ is an n-dimensional identity matrix. With the model parameter vector $\boldsymbol{\lambda}_f \triangleq [\alpha\ \beta\ \boldsymbol{p}_{bf}^\top]^\top$, we can derive a relationship between $\boldsymbol{V}_{sb}$ and $\{\boldsymbol{f}_{c_i}\}_{i=1}^{N_c}$ as

$$\begin{aligned} \boldsymbol{V}_{sb} &= \mathbf{Ad}_{bf} \cdot \boldsymbol{V}_{sf} \\ &= -\mathbf{Ad}_{bf} \cdot \tilde{\mathbf{L}} \cdot \boldsymbol{F}_f \\ &= \mathbf{Ad}_{bf} \cdot \tilde{\mathbf{L}} \cdot \mathbf{J}_{fc}^{*\top} \cdot \boldsymbol{f}_{bc} \\ &= \underbrace{\left(\mathbf{Ad}_{bf} \cdot \tilde{\mathbf{L}} \cdot \mathbf{Ad}_{bf}^\top\right)}_{\alpha,\ \beta,\ \boldsymbol{p}_{bf}} \cdot \sum_{i=1}^{N_c} \left(\underbrace{\left(\mathbf{J}_{bc_i}^{*\top} \cdot \mathbf{N}_i\right)}_{\phi_{c_i}} \cdot \boldsymbol{f}_{c_i} \right) \\ &\triangleq \sum_{i=1}^{N_c} \mathbf{C}_i(\phi_{c_i}; \boldsymbol{\lambda}_f) \cdot \boldsymbol{f}_{c_i}\ , \end{aligned} \tag{5}$$

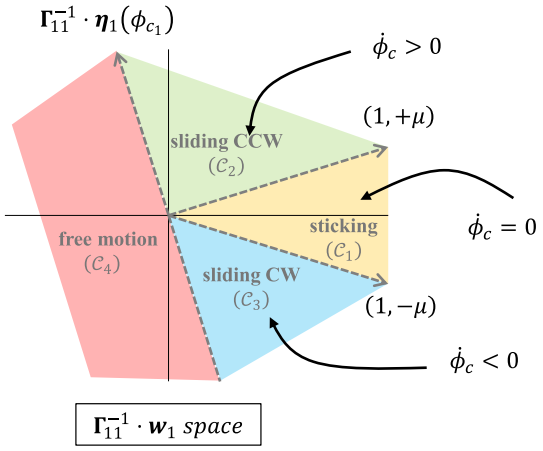

Fig. 4. The illustration of the criterion to determine the contact modes.

and it is well known pushing force-motion model.

Let's assume the non-separation behavior (i.e. always maintain contact), then the relationship between contact forces $\boldsymbol{f}_{c_i}$ and contact velocity $\boldsymbol{v}_{sc_i}$ can be formulated as

$$\begin{aligned} \boldsymbol{v}_{sc_i} &= \mathbf{R}_{2\times2}(\theta_{sb}) \left(\mathbf{J}_{bc_i}^* \cdot \boldsymbol{V}_{sb} + \frac{\partial \boldsymbol{r}}{\partial \phi} \cdot \dot{\phi}_{c_i} \right) \\ &= \mathbf{R}_{2\times2}(\theta_{sb}) \left(\mathbf{J}_{bc_i}^* \cdot \sum_{j=1}^{N_c} \left(\mathbf{C}_j \boldsymbol{f}_{c_j} \right) + \partial_\phi \boldsymbol{r} \cdot \dot{\phi}_{c_i} \right) , \end{aligned} \tag{6}$$

where $\mathbf{R}_{2\times2} \in \mathrm{SO}(2)$ is a 2-dimensional rotation matrix. Note that the model parameter α canceled in this formulation. Then, we can compute the mode-decision vectors $\boldsymbol{w}_i$ as

$$\begin{aligned} \boldsymbol{w}_i &\triangleq \mathbf{N}_i \mathbf{R}_{2\times2}^\top(\theta_{sb}) \cdot \boldsymbol{v}_{sc_i} \\ &= \mathbf{N}_i \cdot \left(\mathbf{J}_{bc_i}^* \cdot \sum_{j=1}^{N_c} \left(\mathbf{C}_j \boldsymbol{f}_{c_j} \right) + \partial_\phi \boldsymbol{r} \cdot \dot{\phi}_{c_i} \right) \\ &= \sum_{j=1}^{N_c} \boldsymbol{\Gamma}_{ij} \boldsymbol{f}_{c_j} + \boldsymbol{\eta}_i(\phi_{c_i}) \cdot \dot{\phi}_{c_i} \ . \end{aligned} \tag{7}$$

Through the (5) and (7), we can obtain the resultant motion $\boldsymbol{V}_{sb}$ from given pusher velocities $\boldsymbol{v}_{sc_i}$ with a conditional constraint (2). For example, in a simple single contact pushing scenario ($N_c = 1$), the (7) can be rewritten as

$$\begin{aligned} \boldsymbol{\Gamma}_{11}^{-1}\boldsymbol{w}_1 &= \underbrace{\begin{bmatrix} 1 & 1 \\ +\mu_{c_1} & -\mu_{c_1} \end{bmatrix} \begin{bmatrix} f_{c_1}^+ \\ f_{c_1}^- \end{bmatrix}}_{= \boldsymbol{f}_{c_1}} + \boldsymbol{\Gamma}_{11}^{-1}\boldsymbol{\eta}_1 \cdot \dot{\phi}_{c_1} \\ (\text{if } \mathcal{C}_1) \ &= \begin{bmatrix} 1 \\ +\mu_{c_1} \end{bmatrix} f_{c_1}^+ + \begin{bmatrix} 1 \\ -\mu_{c_1} \end{bmatrix} f_{c_1}^- \\ (\text{if } \mathcal{C}_2) \ &= \begin{bmatrix} 1 \\ +\mu_{c_1} \end{bmatrix} f_{c_1}^+ + \boldsymbol{\Gamma}_{11}^{-1}\boldsymbol{\eta}_1 \cdot (+\dot{\phi}_{c_1}) \\ (\text{if } \mathcal{C}_3) \ &= \begin{bmatrix} 1 \\ -\mu_{c_1} \end{bmatrix} f_{c_1}^- - \boldsymbol{\Gamma}_{11}^{-1}\boldsymbol{\eta}_1 \cdot (-\dot{\phi}_{c_1}) \\ (\text{if } \mathcal{C}_4) \ &= \text{otherwise} \ . \end{aligned} \tag{8}$$

Algorithm 1: Contact-aware GVMM (C-GVMM)

Input: $\boldsymbol{x} = [\boldsymbol{q}_{sb}^\top \boldsymbol{p}_{sc}^\top]^\top$, $\boldsymbol{u} = \boldsymbol{v}_{sc}$
Output: $\dot{\boldsymbol{x}} = [\dot{\boldsymbol{q}}_{sb}^\top \dot{\boldsymbol{p}}_{sc}^\top]^\top$
Param: $\boldsymbol{\lambda} = [\beta \ \boldsymbol{p}_{bf}^\top]^\top$
Given: object shape $\boldsymbol{r}(\cdot)$

$\phi_c \leftarrow \mathrm{arctan2}\,(y_{sc} - y_{sb},\ x_{sc} - x_{sb}) - \theta_{sb}$
$\boldsymbol{w}_1 \leftarrow \mathbf{N}_1 \mathbf{R}_{2\times2}^\top(\theta_{sb}) \cdot \boldsymbol{v}_{sc_1}$
$\boldsymbol{\eta}_1 \leftarrow \mathbf{N}_1 \cdot \partial_\phi \boldsymbol{r}(\phi_c)$
$\boldsymbol{\Gamma}_{11} \leftarrow \mathbf{N}_1 \mathbf{J}_{bc_1}^* \mathbf{C}_1$
if $\|\boldsymbol{p}_{sc} - \boldsymbol{p}_{sb}\|_2 > \|\boldsymbol{r}(\phi_c)\|_2$ **then**
  $[f_{c_1}^+ \ f_{c_1}^-]^\top = \mathbf{0}_{2\times1}$
else
  $\boldsymbol{c}_{stick} \leftarrow \begin{bmatrix} 1 & 1 \\ \mu_{c_1} & -\mu_{c_1} \end{bmatrix}^{-1} \boldsymbol{\Gamma}_{11}^{-1}\boldsymbol{w}_1$
  $\boldsymbol{c}_{ccw} \leftarrow \begin{bmatrix} 1 & \\ +\mu_{c_1} & +\boldsymbol{\Gamma}_{11}^{-1}\boldsymbol{\eta}_c \end{bmatrix}^{-1} \boldsymbol{\Gamma}_{11}^{-1}\boldsymbol{w}_1$
  $\boldsymbol{c}_{cw} \leftarrow \begin{bmatrix} 1 & \\ -\mu_{c_1} & -\boldsymbol{\Gamma}_{11}^{-1}\boldsymbol{\eta}_c \end{bmatrix}^{-1} \boldsymbol{\Gamma}_{11}^{-1}\boldsymbol{w}_1$
  if $\boldsymbol{c}_{stick} > \mathbf{0}_{2\times1}$ **then**
    $[f_c^+ \ f_c^-]^\top \leftarrow \boldsymbol{c}_{stick}$
  else if $\boldsymbol{c}_{ccw} \geq \mathbf{0}_{2\times1}$ **then**
    $[f_c^+ \ (\cdot)]^\top \leftarrow \boldsymbol{c}_{ccw}$
    $f_c^- \leftarrow 0$
  else if $\boldsymbol{c}_{cw} \geq \mathbf{0}_{2\times1}$ **then**
    $[f_c^- \ (\cdot)]^\top \leftarrow \boldsymbol{c}_{cw}$
    $f_c^+ \leftarrow 0$
  else
    $[f_c^+ \ f_c^-]^\top \leftarrow \mathbf{0}_{2\times1}$
  end
end
$\boldsymbol{f}_{c_1} \leftarrow \begin{bmatrix} 1 & 1 \\ +\mu_{c_1} & -\mu_{c_1} \end{bmatrix} \begin{bmatrix} f_{c_1}^+ \\ f_{c_1}^- \end{bmatrix}$
$\dot{\boldsymbol{q}}_{sb} \leftarrow \mathbf{R}_z(\theta_{sb}) \cdot \mathbf{C}(\phi_{c_1}; \boldsymbol{\lambda}) \cdot \boldsymbol{f}_{c_1}$
$\dot{\boldsymbol{x}} \leftarrow [\dot{\boldsymbol{q}}_{sb}^\top \ \boldsymbol{u}^\top]^\top$

Here, the $f_{c_1}^+$, $f_{c_1}^-$, $\dot{\phi}_{c_1}$ if $\mathcal{C}_2$, and $-\dot{\phi}_{c_1}$ if $\mathcal{C}_3$ must be a non-negative, so we can compute the reliable contact forces and sliding rates from given velocity input $\boldsymbol{v}_{sc_1}$ directly, then the resultant motion $\boldsymbol{V}_{sb}$ can be computed from (5). The criterion of (8) can be illustrated as Fig. 4. Note that the process in (8) can be simply extended into multi-contact cases in the same sense.

The overall algorithm of the C-GVMM is summarized in Algorithm 1. We set a state $\boldsymbol{x} \triangleq [\boldsymbol{p}_{sb}^\top \ \theta_{sb} \ \boldsymbol{p}_{sb}^\top]^\top = [\boldsymbol{q}_{sb}^\top \ \boldsymbol{p}_{sb}^\top]^\top$ and input $\boldsymbol{u} \triangleq \dot{\boldsymbol{p}}_{sc}$. Note that the model parameter of the C-GVMM is $\boldsymbol{\lambda} = [\beta \ \boldsymbol{p}_{bf}^\top]^\top$, which has one reduced dimension than the force-motion model in (5).

C. Goal-oriented robotic pushing framework

For a goal-oriented pushing control, we leveraged long-horizon MPPI, which is well-known goal-oriented motion generator [33], [34], [8] that can available for a non-differentiable dynamics and cost function such as pushing motion model in Sec III-B. For a given desired goal state

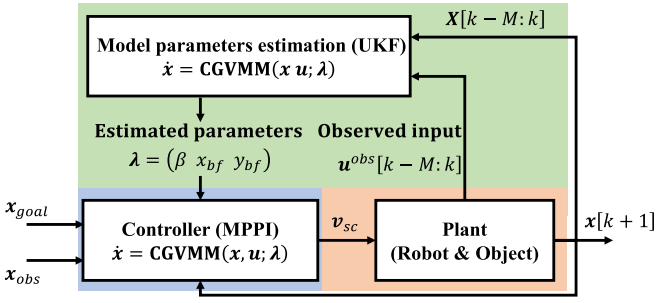

Fig. 5. Overall of proposed robotic pushing framework. The model parameters are estimated using previous M steps observations and fed into the MPPI module.

$\boldsymbol{x}_g$, the optimal robot motion $\dot{\boldsymbol{p}}^*_{sc}$ is computed through MPPI with a state cost function as

$$\text{StateCost}(\cdot) = \phi(\boldsymbol{x}_T) + \sum_{t=0}^{T-1} l(\boldsymbol{x}_t) + P(\boldsymbol{x}_t) \tag{9a}$$

$$\phi(\boldsymbol{x}) = q_f \cdot d^2(\boldsymbol{q}_{sb}, \boldsymbol{q}_g) \tag{9b}$$

$$l(\boldsymbol{x}) = q_{r_1} \cdot d^2(\boldsymbol{q}_{sb}, \boldsymbol{q}_g) \tag{9c}$$

$$+ q_{r_2} \cdot \text{ReLU}(\|\boldsymbol{p}_{sc} - \boldsymbol{p}_{sb}\|_2 - r_s) \tag{9d}$$

$$P(\boldsymbol{x}) = \begin{cases} C_{pen} & \textbf{else if } \text{isCollision}(\boldsymbol{q}_{sb}) \\ 0 & \textbf{otherwise} \end{cases} \tag{9e}$$

where $d(\boldsymbol{q}_1, \boldsymbol{q}_2) \triangleq \sqrt{|\Delta \boldsymbol{p}\|_2^2 + c \cdot (1 - \cos \Delta\theta)}$ a decoupled distance[1] with a scaling constant c. The cost $P(\boldsymbol{x})$ was embedded to penalize some unwanted state-dependent behaviors. The input clamping process of MPPI was proceeded as

$$\dot{\boldsymbol{p}}_{sc} \leftarrow \text{clip}(\dot{\boldsymbol{p}}_{sc},\ \frac{1}{\sqrt{2}}[-v_{sc}^{max}, v_{sc}^{max}]) \quad . \tag{10}$$

Also, we used monotonically increasing prediction time $\{\Delta t_k \triangleq \Delta t_0 \cdot e^{2k/N_p}\}_{k=1}^{N_p}$ to guarantee long-horizon prediction with a lower number of prediction steps N_p [33].

To handle the model uncertainties induced from the unknown physical properties, we leveraged the online model parameter estimation scheme using an unscented Kalman filter (UKF; see green region in Fig. 5). Please refer [8] for details of implementation and hyperparameter setting.

IV. EXPERIMENTS AND RESULTS

The evaluation proceeded in three stages. First, we validated the proposed contact-aware pushing motion model (Sec. III-B) using the MIT pushing dataset [16] to confirm its reliability. Second, we conducted numerical simulations under diverse pushing conditions to assess the framework's ability to manipulate objects effectively without relying on predefined contact modes or trajectories. Finally, we performed real-robot experiments and demonstrations to verify the framework's applicability in real-world scenarios.

All algorithms described in Sec. III were implemented in `Python` using `PyTorch` [35] to leverage GPU-accelerated parallel computation. The hyperparameter settings used throughout the experiments are summarized in Table II.

[1]The $d(\cdot, \cdot)$ is not a distance function because it does not satisfy the trigonometric inequality, but for convenience, specify it as a distance function in this paper.

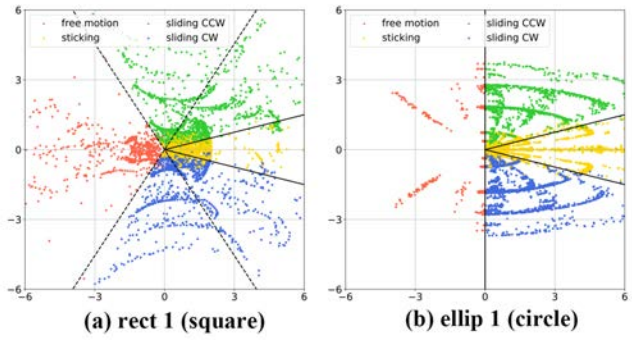

Fig. 6. The result of observed contact-modes in MIT pushing dataset. Note that the black dashed line ($\boldsymbol{\Gamma}_{11}^{-1} \cdot \boldsymbol{\eta}(\phi_{c_1})$ in Fig. 4) is varing under different ϕ_{c_1}.

TABLE II
HYPERPARAMETERS SETTING

Symbol	Description	Value
N_p	Total prediction steps	25
Δt_0	Initial prediction time step	0.04
q_f, q_{r_1}, q_{r_2}	Gains of state cost (9)	5000, 100, 10^4
c, r_s, C_{pen}	Constants of state cost (9)	0.1, 0.2, 10^{13}
v_{sc}^{max}	EE's speed limit	0.1 m/s
α_c	Constant value of α	1.0
$\boldsymbol{\Sigma}$	Covariance of sampled noise	$\text{diag}(0.12\ 0.12)^2$

A. Model validation

We first validated the proposed contact-aware pushing motion model (Sec. III-B) using the MIT pushing dataset [16], which consists of diverse pushing interactions recorded with high-fidelity pose tracking. The dataset was resampled and low-pass filtered to 25 Hz, and experiments were conducted on both square (*rect 1*) and circular (*ellip 1*) objects. For the validation, the contact friction coefficient was set to $\mu_c = 0.25$, and the center of friction was assumed at $\boldsymbol{p}_{bf} = [0\ 0]^\top$.

As shown in Fig. 6, the observed contact modes represented by the sign of (sign of $\dot{\phi}_c$) aligned well with the criterion defined in Fig. 4. The boundaries between the sliding (green and blue) and sticking (yellow) are well separated by vectors $[1\ \mu_c]^\top$ and $[1\ -\mu_c]^\top$ (black line), and the boundaries between the sliding and free motion (red) also well described by a vector $\boldsymbol{\Gamma}_{11}^{-1} \cdot \boldsymbol{\eta}_1(\phi_{c_1})$. These results experimentally confirm the accuracy and validity of the C-GVMM formulation for modeling contact-aware pushing dynamics.

B. Numerical experiments

We evaluated the proposed contact-aware dexterous goal-oriented pushing framework in numerical simulation across three benchmark tasks: re-positioning (**Task I**; long re-pos), re-orienting (**Task II**; re-ori), and obstacle avoidance (**Task III**; O.A.). Two object shapes, a rectangle and a T-shape,

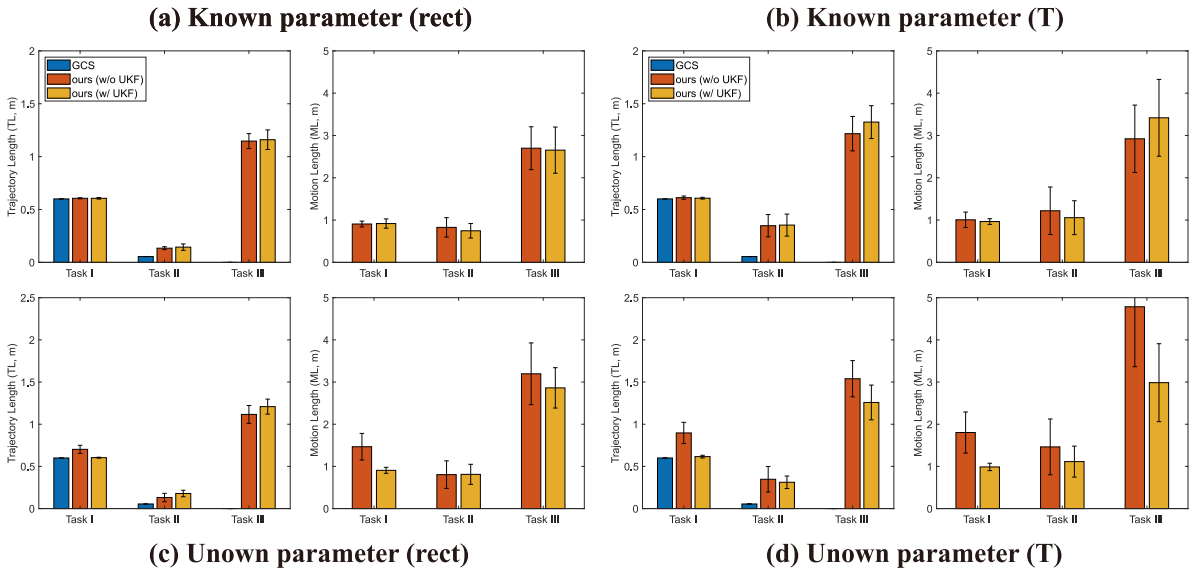

Fig. 7. The results of real-robot experiments. The blue region is a start pose, and red is a goal pose. The plots on the right side show the full behavior of the slider (gray) and pusher (cyan circle), and we can observe that the robot freely move around the object.

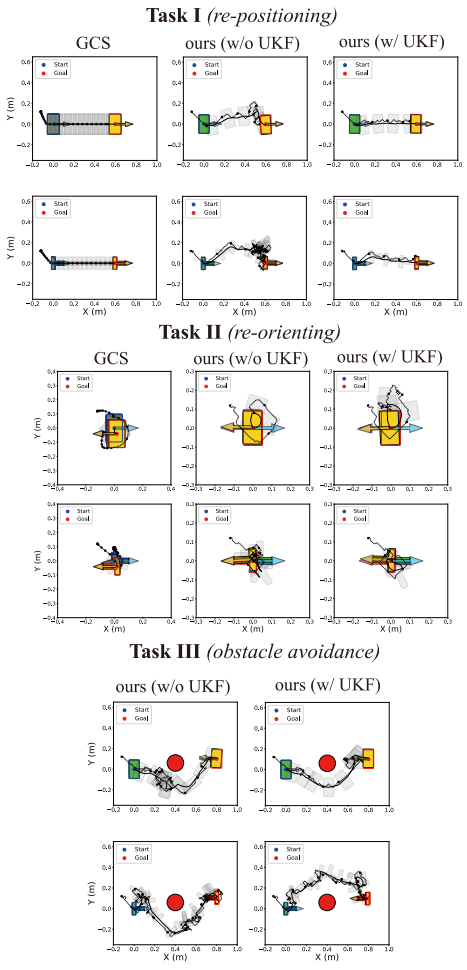

Fig. 8. The results of the numerical experiments. The black lines, black dashed line illustrate the trajectory of object and contact points respectively. In **Task III**, the red circle is an obstacle.

were tested in each task, and the true model parameters were set to $\boldsymbol{\lambda}_{true} = [25.0 \ \ -0.01 \ \ 0.02]^\top$. Two scenarios were considered for model parameter knowledge: (i) Known Parameters, where the prediction model exactly matches the true parameters ($\boldsymbol{\lambda}_{pred} = \boldsymbol{\lambda}_{true}$), and (ii) Unknown Parameters, where the prediction model parameters differ from the true values $\boldsymbol{\lambda}_{pred} = [15 \ 0.0 \ 0.0]^\top$. We compared our approach combining MPPI with UKF-based online parameter estimation (ours w/ UKF) against MPPI with fixed parameters (ours w/o UKF) and a state-of-the-art offline pushing planner (GCS [26]).

We repeated 30 trials for each condition and shape, and the results of the total trajectory length (**TL** (m)) and success rate (**SR** (%)) are reported in Table **??**. Note that success was declared if the object was reached toward the goal within $N_{max} = 1500$ steps with $\|[x \ y]^{sg}_{sb}\|_2 < 8$ mm and $|\theta^{sg}_{sb}| < 3°$. Note that the GCS planner was evaluated only for Tasks I and II, as it does not support obstacle avoidance.

The results are depicted in Fig. 7 and Fig. 8. In the known parameter scenarios (Fig. 7 (a)-(b)), both 'ours w/o UKF' and 'ours w/ UKF' have accomplished the pushing tasks with similar pushing performances. However, in the unknown parameter scenarios (Fig. 7 (c)-(d)), the 'ours w/ UKF' outperformed the 'ours w/o UKF' because the uncertain object properties make the pushing motion unpredictable while the 'ours w/ UKF' tried to estimate the best model parameters and update them in every execution step. These findings demonstrate that the proposed framework effectively achieves accurate, dexterous pushing with adaptive contact-mode transitions under uncertainty.

C. Real-robot experiments

We evaluated the proposed contact-aware, goal-oriented pushing framework on a 7-DoF robotic manipulator (Panda, Franka Emika) in real-world settings. A low-level impedance

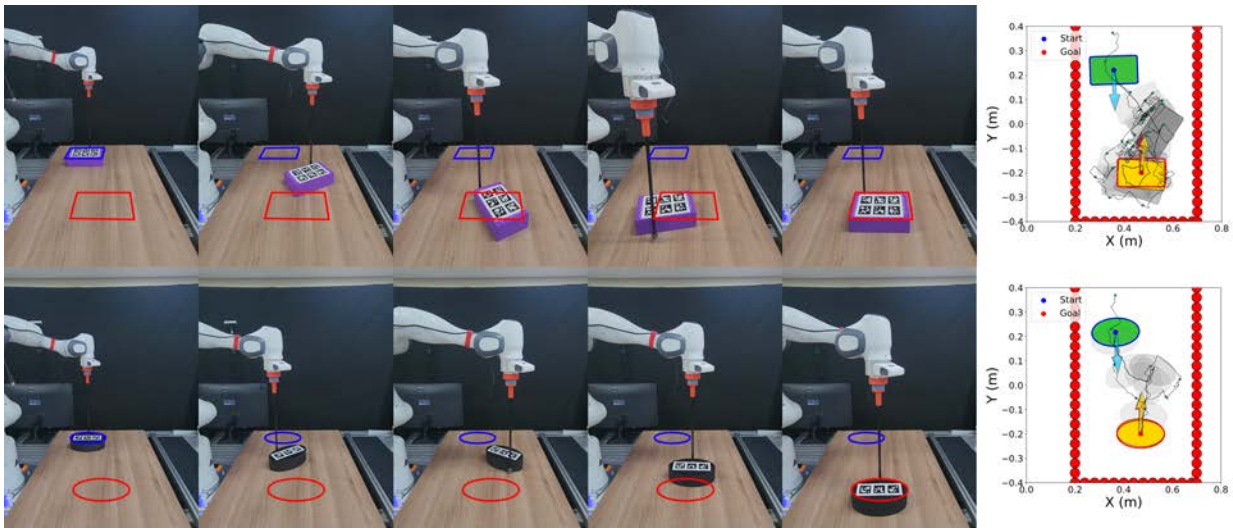

Fig. 9. The results of real-robot experiments. The blue region is a start pose, and red is a goal pose. The plots on the right side show the full behavior of the slider (gray) and pusher (cyan circle), and we can observe that the robot freely move around the object.

controller running at 1 kHz, augmented with a QP-based safety filter [36], was employed to safely track the desired task-space velocity. Object poses were estimated at 15 Hz using ArUco markers and two global RGB-D cameras (RealSense D435, Intel). The pushed objects included a 200 mm × 120 mm grid box and an ellipse.

The results of the real-robot experiments are depicted in Fig. 9. We performed re-arranging the object in confined pushing workspace (highlighted by red circles in Fig. 9). Each condition was executed over five trials. Consistent with the numerical experiments, the proposed framework successfully accomplished both tasks while naturally incorporating contact-free motions.

D. Demonstration: Pushing and Grasping

We further validated the applicability of the proposed pushing framework through a pushing-and-grasping demonstration using an adaptive gripper (OMEGA gripper, MARCH Bionics Co. Ltd.). The scenario was designed as a tableware-cleaning task involving both graspable objects (chopsticks, cup, bowl) and an ungraspable object (flat dish). The robot was tasked with autonomously clearing the tableware from the workspace.

To estimate object poses, we employed SAM2 [37] and FoundationPose [38], which enabled accurate 6D pose tracking using pre-scanned 3D models. A key advantage of our framework is its ability to handle contact-free motions, eliminating the need for exact modeling of contact kinematics, such as CAD-based end-effector geometry. Therefore, we were able to apply the same framework as in Sec. IV-C without additional modeling effort.

As illustrated in Fig. 1 and the supplementary video, the robot successfully executed the task, demonstrating that the proposed contact-aware framework enables reliable pushing and grasping in real-world applications.

V. CONCLUSIONS

This work introduced a contact-aware robotic pushing framework capable of performing goal-oriented pushing without relying on explicit trajectory or contact-mode planning. We first formulated the contact-aware generalized velocity motion model (C-GVMM), which captures the relationship between the pusher's velocity and the induced motion of the slider, including transitions into separation motion. This model enables reliable prediction of object behavior without the need for any predefined contact-mode sequences that are commonly assumed in existing pushing studies.

Building on this formulation, we employed model predictive path integral (MPPI) control with an UKF-based online parameter estimation as an accurate goal-oriented pushing strategy to address the challenges of contact-rich, non-differentiable dynamics and cost functions. By leveraging the flexibility of C-GVMM in handling contact-mode transitions, MPPI can be directly applied without any tricks for contact-mode transitions. Our comprehensive numerical simulations and real-robot experiments verified that the proposed framework achieves dexterous pushing tasks without predefined trajectories or contact-mode specifications, while naturally incorporating contact-free motions.

Although C-GVMM was derived for multi-contact interactions, this study focused only on single-contact scenarios. Extending the framework to multi-contact pushing is a promising direction, as it can substantially improve robustness and versatility under real-world uncertainties, including variations in contact location, physical parameters, and sensory observations. As future work, we aim to integrate multi-contact capabilities into our framework to enable inherently robust and more dexterous contact-aware robotic manipulation.

REFERENCES

[1] A. Billard and D. Kragic, "Trends and challenges in robot manipulation," *Science*, vol. 364, no. 6446, p. eaat8414, 2019.

[2] M. R. Cutkosky *et al.*, "On grasp choice, grasp models, and the design of hands for manufacturing tasks." *IEEE Transactions on robotics and automation*, vol. 5, no. 3, pp. 269–279, 1989.

[3] M. T. Mason, "Progress in nonprehensile manipulation," *Int. J. Robot. Res.*, vol. 18, no. 11, pp. 1129–1141, 1999.

[4] F. R. Hogan and A. Rodriguez, "Reactive planar non-prehensile manipulation with hybrid model predictive control," *Int. J. Robot. Res.*, vol. 39, no. 7, pp. 755–773, 2020.

[5] J. Moura, T. Stouraitis, and S. Vijayakumar, "Non-prehensile planar manipulation via trajectory optimization with complementarity constraints," in *Proc. IEEE Int. Conf. Robot. Automat.,*, 2022, pp. 970–976.

[6] T. Xue, H. Girgin, T. S. Lembono, and S. Calinon, "Guided optimal control for long-term non-prehensile planar manipulation," in *Proc. IEEE Int. Conf. Robot. Automat.,*. IEEE, 2023, pp. 4999–5005.

[7] N. Dengler, J. D. A. Ferrandis, J. Moura, S. Vijayakumar, and M. Bennewitz, "Learning goal-directed object pushing in cluttered scenes with location-based attention," *arXiv preprint arXiv:2403.17667*, 2024.

[8] Y. Lee and K. Kim, "Goal-driven robotic pushing manipulation under uncertain object properties," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 11 904–11 910.

[9] S. Jeon, M. Jung, S. Choi, B. Kim, and J. Hwangbo, "Learning whole-body manipulation for quadrupedal robot," *IEEE Robot. Autom. Lett.*, vol. 9, no. 1, pp. 699–706, 2023.

[10] I. Dadiotis, M. Mittal, N. Tsagarakis, and M. Hutter, "Dynamic object goal pushing with mobile manipulators through model-free constrained reinforcement learning," *arXiv preprint arXiv:2502.01546*, 2025.

[11] Y. Feng, C. Hong, Y. Niu, S. Liu, Y. Yang, and D. Zhao, "Learning multi-agent loco-manipulation for long-horizon quadrupedal pushing," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 14 441–14 448.

[12] K. Hang, A. S. Morgan, and A. M. Dollar, "Pre-grasp sliding manipulation of thin objects using soft, compliant, or underactuated hands," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 662–669, 2019.

[13] C. Song and A. Boularias, "Object rearrangement with nested non-prehensile manipulation actions," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.* IEEE, 2019, pp. 6578–6585.

[14] 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*, p. 02783649241273668, 2023.

[15] Y. Lee and K. Kim, "Accurate robotic pushing manipulation through online model estimation under uncertain object properties," *IEEE Robot. Autom. Lett.*, 2024.

[16] K.-T. Yu, M. Bauza, N. Fazeli, and A. Rodriguez, "More than a million ways to be pushed. a high-fidelity experimental dataset of planar pushing," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 30–37.

[17] S. Goyal, A. Ruina, and J. Papadopoulos, "Planar sliding with dry friction part 1. limit surface and moment function," *Wear*, vol. 143, no. 2, pp. 307–330, 1991.

[18] J. Zhou, R. Paolini, J. A. Bagnell, and M. T. Mason, "A convex polynomial force-motion model for planar sliding: Identification and application," in *Proc. IEEE Int. Conf. Robot. Automat.,*, 2016, pp. 372–377.

[19] K. M. Lynch, H. Maekawa, and K. Tanie, "Manipulation and active sensing by pushing using tactile feedback." in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, vol. 1, 1992, pp. 416–421.

[20] M. Bauza and A. Rodriguez, "A probabilistic data-driven model for planar pushing," in *Proc. IEEE Int. Conf. Robot. Automat.,*, 2017, pp. 3008–3015.

[21] A. Ajay, J. Wu, N. Fazeli, M. Bauza, L. P. Kaelbling, J. B. Tenenbaum, and A. Rodriguez, "Augmenting physical simulators with stochastic neural networks: Case study of planar pushing and bouncing," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 3066–3073.

[22] A. Kloss, S. Schaal, and J. Bohg, "Combining learned and analytical models for predicting action effects from sensory data," *Int. J. Robot. Res.*, vol. 41, no. 8, pp. 778–797, 2022.

[23] L. Cong, M. Grner, P. Ruppel, H. Liang, N. Hendrich, and J. Zhang, "Self-adapting recurrent models for object pushing from learning in simulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 5304–5310.

[24] G. Wang, K. Ren, and K. Hang, "Uno push: Unified nonprehensile object pushing via non-parametric estimation and model predictive control," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.* IEEE, 2024, pp. 9893–9900.

[25] J. Zhou, Y. Hou, and M. T. Mason, "Pushing revisited: Differential flatness, trajectory planning, and stabilization," *Int. J. Robot. Res.*, vol. 38, no. 12-13, pp. 1477–1489, 2019.

[26] B. P. Graesdal, S. Y. C. Chia, T. Marcucci, S. Morozov, A. Amice, P. A. Parrilo, and R. Tedrake, "Towards tight convex relaxations for contact-rich manipulation," in *Proc. Robot.: Sci. Syst.*, 2024.

[27] Y. Jiang, Y. Jia, and X. Li, "Contact-aware non-prehensile manipulation for object retrieval in cluttered environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.* IEEE, 2023, pp. 10 604–10 611.

[28] J. K. Li, W. S. Lee, and D. Hsu, "Push-net: Deep planar pushing for objects with unknown physical properties." in *Proc. Robot.: Sci. Syst.*, vol. 14, 2018, pp. 1–9.

[29] A. Rezazadeh and C. Choi, "Kinet: Unsupervised forward models for robotic pushing manipulation," *IEEE Robot. Autom. Lett.*, 2023.

[30] J. D. A. Ferrandis, J. Moura, and S. Vijayakumar, "Nonprehensile planar manipulation through reinforcement learning with multimodal categorical exploration," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2023, pp. 5606–5613.

[31] G. Williams, P. Drews, B. Goldfain, J. M. Rehg, and E. A. Theodorou, "Information-theoretic model predictive control: Theory and applications to autonomous driving," *IEEE Trans. Robot.*, vol. 34, no. 6, pp. 1603–1622, 2018.

[32] S. H. Lee and M. Cutkosky, "Fixture planning with friction," 1991.

[33] M. Bhardwaj, B. Sundaralingam, A. Mousavian, N. D. Ratliff, D. Fox, F. Ramos, and B. Boots, "Storm: An integrated framework for fast joint-space model-predictive control for reactive manipulation," in *Proc. Conf. Robot Learn.* PMLR, 2022, pp. 750–759.

[34] H. Xue, C. Pan, Z. Yi, G. Qu, and G. Shi, "Full-order sampling-based mpc for torque-level locomotion control via diffusion-style annealing," *arXiv preprint arXiv:2409.15610*, 2024.

[35] A. Paszke, "Pytorch: An imperative style, high-performance deep learning library," *arXiv preprint arXiv:1912.01703*, 2019.

[36] D. Lee, D. Ko, W. K. Chung, and K. Kim, "Quadratic programming-based task scaling for safe and passive robot arm teleoperation," *IEEE/ASME Trans. Mechatronics*, vol. 27, no. 4, pp. 1937–1945, 2022.

[37] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer, "Sam 2: Segment anything in images and videos," 2024. [Online]. Available: https://arxiv.org/abs/2408.00714

[38] B. Wen, W. Yang, J. Kautz, and S. Birchfield, "Foundationpose: Unified 6d pose estimation and tracking of novel objects," *arXiv preprint arXiv:2312.08344*, 2023.