MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

Closed-Loop Tactile Controller for Tool Manipulation

Shirai, Yuki; Jha, Devesh K.; Raghunathan, Arvind; Hong, Dennis

TR2023-043 May 25, 2023

Abstract

In this paper, we present closed-loop control of a complex manipulation task where a robot uses a tool to interact with objects. Manipulation using a tool leads to complex kinematics and contact constraints that need to be satisfied for generating feasible manipulation trajectories. We first present an open-loop controller design using Non-Linear Programming (NLP) that satisfies these constraints. In order to design a closed-loop controller, we present a pose estimator of objects and tools using tactile sensors. Using our tactile estimator, we design a closed-loop controller based on Model Predictive Control (MPC). The proposed algorithm is verified using a 6 DoF manipulator on tasks using a variety of objects and tools. We verify that our closed-loop controller can successfully perform tool manipulation under several unexpected contacts. All hardware experiment videos could be found at

ICRA 2023 Workshop on Embracing contacts. Making robots physically interact with our world

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

Closed-Loop Tactile Controller for Tool Manipulation

Yuki Shirai[1], Devesh K. Jha[2], Arvind U. Raghunathan[2], and Dennis Hong[1]

***Abstract*— In this paper, we present closed-loop control of a complex manipulation task where a robot uses a tool to interact with objects. Manipulation using a tool leads to complex kinematics and contact constraints that need to be satisfied for generating feasible manipulation trajectories. We first present an open-loop controller design using Non-Linear Programming (NLP) that satisfies these constraints. In order to design a closed-loop controller, we present a pose estimator of objects and tools using tactile sensors. Using our tactile estimator, we design a closed-loop controller based on Model Predictive Control (MPC). The proposed algorithm is verified using a 6 DoF manipulator on tasks using a variety of objects and tools. We verify that our closed-loop controller can successfully perform tool manipulation under several unexpected contacts. All hardware experiment videos could be found at https://youtu.be/VsClK04qDhk.**

I. Introduction

Using contacts efficiently can provide additional dexterity to robots while performing complex manipulation tasks [1], [2], [3]. However, most robotic systems avoid making contact with their environment. This is mainly because contact interactions lead to complex, discontinuous dynamics and thus, planning, estimation, and control of manipulation require careful treatment of these constraints. As a result of these challenges, most of the classical control approaches are not applicable to control of manipulation systems [1], [4], [5], [6], [7]. However, closed-loop control of manipulation tasks is imperative for design of robust, high-performance robotic systems that can effortlessly interact with their environments.

In this paper, we consider tool manipulation where a robot can grasp an external tool that can be used to pivot an external object in the environment. As could be seen in Fig. 1, tool manipulation leads to multiple contact formations between the robot & a tool, the tool & an object, and the object & environment. It is easy to imagine that planning for tool manipulation needs to incorporate all constraints imposed by these contact formations. This makes planning for tool manipulation extremely challenging. Furthermore, the robot can not directly observe all the relevant contact and object states during tool manipulation. This imposes additional complexity during controller design. This makes tool manipulation a challenging, albeit extremely rich system to study closed-loop controller design for manipulation. We present design of planning, estimation, and control for tool manipulation using tactile sensing. The proposed tactile estimator makes use of the contact constraints for pose estimation while the MPC controller enforces the desired contact constraints along the manipulation trajectory.

† Yuki Shirai and Dennis Hong are with the Department of Mechanical and Aerospace Engineering, University of California, Los Angeles, CA, USA 90095 {yukishirai4869,dennishong}@g.ucla.edu

‡Devesh K. Jha and Arvind U. Raghunathan are with Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, USA 02139 {jha,raghunathan}@merl.com

II. Related Work

Our work is inspired by seminal work on manipulation by shared grasping [8] which discusses mechanics of shared grasping and shows impressive demonstrations. The task that we present in this paper is a complex version of shared grasping where the robot uses a tool instead of a rigid end-effector to manipulate objects. This variation leads to additional contact formations. These additional constraints make the problem more complicated to plan, control, and estimate compared to those works.

Model-based planning for tool manipulation was earlier presented in [9]. Learning-based algorithm of grasping for tool manipulation is presented in [10]. In our work, we consider a closed-loop controller and estimator in addition to planning for tool manipulation to robustify the system.

Our work is also closely related to the remarkable previous work on tactile estimation and reactive manipulation presented in [11], [12], [13], [14], [15], [16], [17]. For estimators, [11] shows a pose estimator for tools and [12] presents tactile localization. Learning-based estimator for tool manipulation is presented in [13]. In this work, in addition to a tool through tactile sensors, we estimate and control a pose of an object, introducing additional extrinsic contact. For reactive manipulation, our work is related to [15] where slip detection is used to recompute a new controller. [16] shows the impressive closed-loop controller by simultaneous design of controller and estimator. However, the task in [15] is inherently stable as the object is always grasped by the robot. Also, the tactile sensors can directly estimate the pose of the object, which cannot be done for tool manipulation because tactile sensors are not attached between the object and the end-effector.

III. Mechanics of Tool Manipulation

We explain mechanics of tool manipulation and then discuss Trajectory Optimization (TO) of tool manipulation for designing open-loop trajectories. The notation of variables are summarized in Table I. We define the rotation matrix from frame Σ_A to Σ_B as ${}^A_B R$. We denote $\mathbf{p}_i$ as a position at contact i defined in Σ_W. We denote x- and y-axis as axes in 2D plane and z-axis is perpendicular to the plane.

A. Quasi-Static Mechanics of Tool Manipulation

As is shown in Fig. 1, tool manipulation leads to several contact formations at A, B, and C that would need to be

TABLE I: **Notation of variables**. Σ column indicates the frame of variables. See Fig. 1 and Fig. 2 for graphical definition.

Name	Description	Size	Σ
$\mathbf{w}_E$	reaction wrench at point A	$\mathbb{R}^2$	W
$\mathbf{w}_O$	gravity of object at point O	$\mathbb{R}^2$	W
$\mathbf{w}_{TO}$	wrench from the tool to the object at point B	$\mathbb{R}^2$	T
$\mathbf{w}_T$	gravity of tool at point T	$\mathbb{R}^2$	W
$\mathbf{w}_G$	wrench from the gripper to the tool at patch C	$\mathbb{R}^2$	G
θ_O	orientation of object	$\mathbb{R}^1$	W
θ_T	orientation of tool	$\mathbb{R}^1$	W
θ_G	orientation of gripper	$\mathbb{R}^1$	W
θ_S	relative orientation of frame at center of grasp	$\mathbb{R}^1$	S

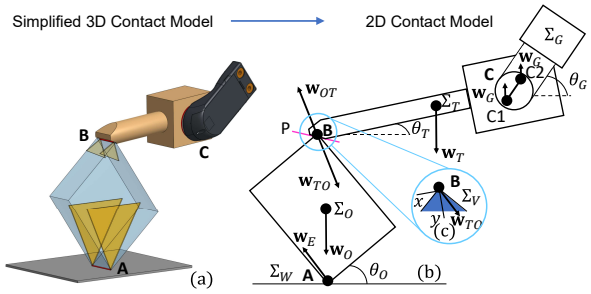

Fig. 1: **Mechanics of tool manipulation.** (a): A simplified 3D contact model for tool manipulation highlighting the three main contact interactions during the task. (b): Free-body diagram of a rigid body and a tool during tool manipulation in 2D. (c): Force from a tool to an object has to lie on a cone defined by the shape of the object.

maintained during manipulation. Additionally, we need to consider quasi-static equilibrium of the tool and the object in the presence of these contacts. The static equilibrium of the object is described as:

$$F_O(\mathbf{w}_E, \mathbf{w}_O, {}^W_T R\mathbf{w}_{TO}) = \mathbf{0}, \quad (1a)$$

$$G_O(\mathbf{w}_E, \mathbf{w}_O, {}^W_T R\mathbf{w}_{TO}, \mathbf{p}_A, \mathbf{p}_B, \mathbf{p}_O) = 0 \quad (1b)$$

where F_O and G_O represent static equilibrium of force and moment, respectively. The static equilibrium of the tool is:

$$F_T(\mathbf{w}_T, {}^W_G R\mathbf{w}_G, {}^W_T R\mathbf{w}_{OT}) = \mathbf{0}, \quad (2a)$$

$$G_T(\mathbf{w}_T, {}^W_G R\mathbf{w}_G, {}^W_T R\mathbf{w}_{OT}, \mathbf{p}_B, \mathbf{p}_T, \mathbf{p}_{G1}, \mathbf{p}_{G2}) = 0 \quad (2b)$$

We approximate patch contact at C as two point contacts with the same force distribution.

B. Contact Model

We first discuss the contact model in 3D then we present the approximated contact model in 2D using Fig. 1. In a simplified 3D setting, the different contact formations could be best described as follows:

1) contact A: line contact.
2) contact B: line or patch contact.
3) contact C: patch contact.

For line contacts A and B, we need to consider generalized friction cones [18] to describe sticking line contact in 3D. However, this work considers manipulation in 2D as shown in Fig. 1 (b) and thus we can argue that there is no moment to break the line contact. Thus, we can approximate line contacts as two point contacts with the same force distribution, leading to the larger coefficients of friction effectively.

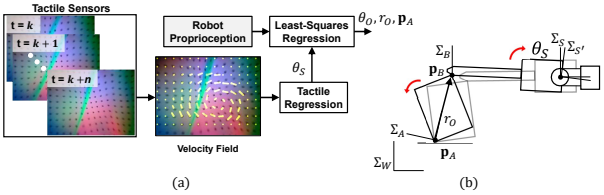

Fig. 2: **Tactile estimator.** (a): Given measurements of robot proprioception and tactile sensors, our method estimates the state of the object and the tool. (b): Schematic showing tool manipulation experiencing rotational slipping by θ_S.

Also for patch contacts at contact C, we need to consider 4D limit surface [19] where we have 3D force $[f_x, f_y, f_z]$ and 1D moment m_z. However, in practice, implementing m_z is difficult, especially for position-controlled manipulators with a force controller with low bandwidth. Thus, this work approximates patch contact at C as two point contacts (see Fig. 1 (b)) with same force distribution.

For point contacts A, B, C_1, C_2, we have the following friction cone constraints:

$$-\mu_i f_y^i \leq f_x^i \leq \mu_i f_y^i, f_y^i \geq 0, \forall i = \{A, B, C_1, C_2\} \quad (3)$$

where μ_i is the coefficient of friction at contact $i = \{A, B, C_1, C_2\}$ and f_x^i, f_y^i are tangential and normal forces for each local coordinate.

C. Contact between Tool and Object

The line contact B is on a certain plane P created by a tool. The plane P is used to discuss the friction cone between the object and the tool since slipping can only occur along the plane P. Thus, by changing the orientation of the tool, the orientation of this plane also changes. Furthermore, different tools have different tip shapes (see Fig. 3). Based on kinematics of the tool, local force definition changes, which is unique to tool manipulation.

Another unique nature of this task is that we need to explicitly consider the feasible region of a force controller. Note that the manipulator can apply forces only along the axes where its motion is constrained. This constraint needs to be explicitly enforced during optimization to generate mechanically feasible force trajectories. Hence, we formulate inequality constraints in vertex frame Σ_V (see Fig. 1 (c)) such that $\mathbf{w}_{TO}$ is constrained by the object:

$$-\rho f_y \leq f_x \leq \rho f_y, f_y \geq 0 \quad (4)$$

where $[f_x, f_x]^\top = {}^V_T R\mathbf{w}_{TO}$. We define Σ_V such that y-axis of Σ_V bisect the angle of vertex B. ρ can be determined by the shape of the object.

D. Trajectory Optimization for Planning

We formulate TO for tool manipulation as follows:

$$\min_{\mathbf{x},\mathbf{u},\mathbf{f}} \sum_{k=1}^{N} (\mathbf{x}_k - \mathbf{x}_g)^\top Q(\mathbf{x}_k - \mathbf{x}_g) + \sum_{k=0}^{N-1} \mathbf{u}_k^\top R\mathbf{u}_k \quad (5a)$$

$$\text{s.t. } (1), (2), (3), (4), \quad (5b)$$

$$\mathbf{x}_0 = \mathbf{x}_s, \mathbf{x}_N = \mathbf{x}_g, \mathbf{x}_k \in \mathcal{X}, \mathbf{u}_k \in \mathcal{U}, \mathbf{f}_k \in \mathcal{F} \quad (5c)$$

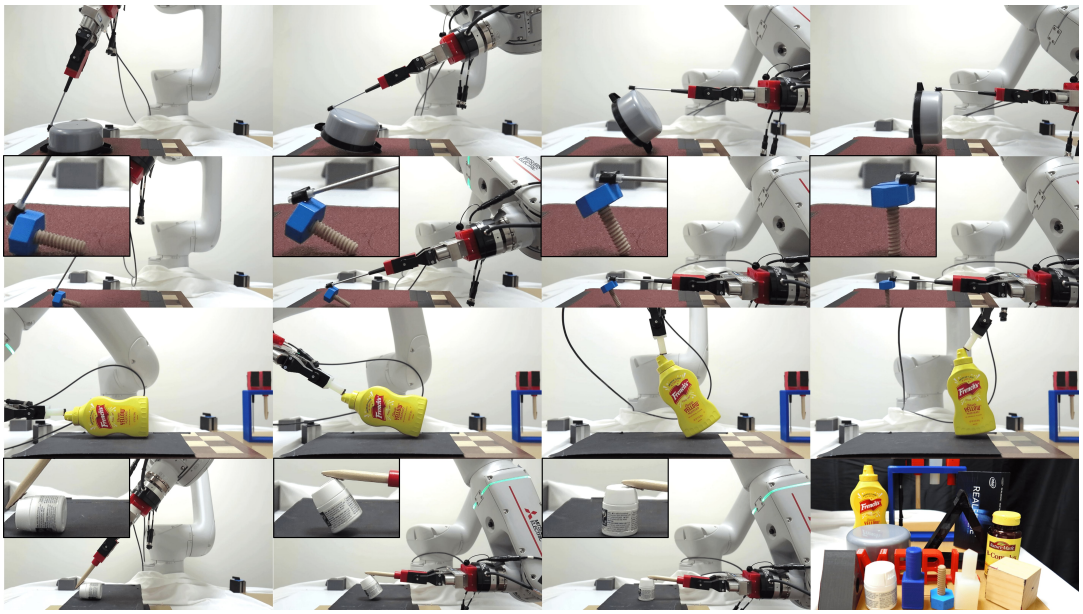

Fig. 3: **Open-loop tool manipulation.** Our controller could successfully perform tool manipulation with different object-tool-environment pairs. The bottom right picture shows the objects and the tools we use in this paper.

where $\mathbf{x}_k = [\theta_{O,k}, \theta_{T,k}, \theta_{G,k}]^\top$, $\mathbf{u}_k = \mathbf{w}_{G,k}$, $\mathbf{f}_k = [\mathbf{w}_{E,k}^\top, \mathbf{w}_{TO,k}^\top]^\top$, $Q = Q^\top \geq 0, R = R^\top > 0$. $\mathcal{X}$, $\mathcal{U}$, and $\mathcal{F}$ are convex polytopes, consisting of a finite number of linear inequality constraints. Based on the solution of (5), we can calculate the pose and force trajectory of the end-effector and we command them during implementation.

IV. Tactile Tool Manipulation

In this section, we present design of our closed-loop controller which makes use of observations from tactile sensors and robot encoders to estimate pose of the system.

A. Tactile Estimator

For our estimator design, we make an assumption that contacts at A and B are sticking. This means that $\mathbf{p}_A$ does not change during pivoting. Using this knowledge, the observed $\mathbf{p}_B$ at $t = k$, denoted as $\bar{\mathbf{p}}_{B,k}$, can be represented as (see Fig. 2 (b)):

$$\left[\bar{\mathbf{p}}_{B,k}^\top, 1\right]^\top = {}^W_S T\, {}^S_{S'} T(\theta_{S,k})\, {}^{S'}_B T \left[\mathbf{0}^\top, 1\right]^\top \tag{6}$$

where $\theta_{S,k}$ is the relative rotation of frame at the center of grasp at $t = k$ (we denote this frame as $\Sigma_{S'}$) with respect to the frame at the reference center of grasp at $t = 0$ (we denote this frame as Σ_S). ${}^S_{S'}T, {}^W_S T$ can be obtained from the tactile sensor and encoders, respectively. ${}^{S'}_B T$ is obtained from the known tool kinematics. We can represent $\mathbf{p}_B$ at $t = k$, denoted as $\mathbf{p}_{B,k}$, also as follows:

$$\left[\mathbf{p}_{B,k}^\top, 1\right]^\top = {}^W_A T(\theta_{O,k}, \mathbf{p}_A)\, {}^A_B T(r_O) [\mathbf{0}^\top, 1]^\top \tag{7}$$

Then, using (6) and (7), with time history of measurements from $t = 0$ to $t = m$, we can do regression:

$$\left\{\theta^*_{O,k}\right\}_{k=0,\ldots,m}, r^*_O, \mathbf{p}^*_A = \arg\min \sum_{k=0}^{m} \|\mathbf{p}_{B,k} - \bar{\mathbf{p}}_{B,k}\|^2 \tag{8}$$

Once contact at $\mathbf{p}_A, \mathbf{p}_B$ slip, the estimator is unable to estimate the state of the object anymore like [12], [16].

B. Tactile Controller

Our online controller based on MPC is as follows:

$$\min_{\mathbf{x},\mathbf{u},\mathbf{f}} \sum_{k=t+1}^{N+t} (\theta_{O,k} - \bar{\theta}_{O,k})^2 + \sum_{k=t}^{N+t-1} \mathbf{u}_k^\top R \mathbf{u}_k \tag{9a}$$

$$\text{s.t. } (5b), (5c) \tag{9b}$$

where $\bar{\theta}_{O,k}$ represent the reference trajectory computed offline using (5).

V. Results

A. Experiment Setup

For the planner and controller, we use IPOPT [20] with pyrobocop [21] interface to solve TO. MPC is run with a large horizon of $N = 160$, and thus can only achieve a control rate of 2 Hz.

For the hardware experiments, we use a Mitsubishi Electric Assista industrial manipulator arm equipped with a WSG-32 gripper. For the closed-loop experiments, the gripper is equipped with GelSlim 3.0 [22] sensors. We use a stiffness controller to track the reference force trajectory [23], [24]. As shown in Fig. 3, we test our framework with 12 different objects, 4 different tools, and 5 different environments (i.e., friction surfaces).

B. Open-Loop Controller

In this experiment, we show that our open-loop controller (5) generates successful trajectories for different objects, tools, and environments. Note that our framework works even for non-rectangle objects as long as the shape of the object can be approximated as a rectangle in 2D. The results are shown in Fig. 3. More results are shown in the supplementary

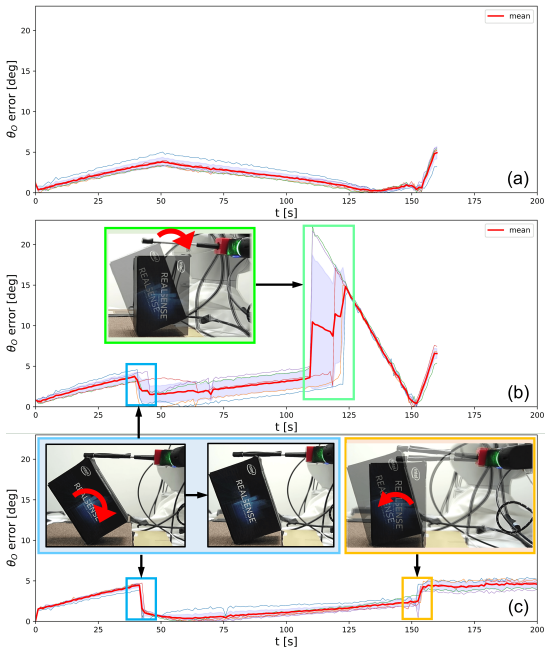

Fig. 4: **Evaluation of the tactile estimator.** We show the time history of error of θ_O for 5 trials (a) with the open-loop controller under no disturbance, (b) with the open-loop controller under disturbance, and (c): with the closed-loop controller under disturbance. The red line shows the mean of and the blue region shows the 95% confidence interval. We added disturbance around $t = 40$ s for (b) and (c) (see the blue box). For (c), we added another disturbance around $t = 150$ s (see the orange box). The contact is lost around $t = 110$ s for (b) (see the green box).

video. There are several failure modes of the open-loop controller (see [25]).

C. Tactile Estimator Results

In this section, we discuss the results of our tactile estimator. To test the accuracy of our estimator, we perform three different kinds of experiments– the open-loop controller with no external disturbance, the open-loop controller with external disturbance, and the closed-loop controller with external disturbance. In all these experiments, the robot is trying to pivot the same box with the same tool. We perform 5 trials for each experiment. All results are shown in Fig. 4.

Our estimator works with the open-loop controller under no disturbances as shown in Fig. 4 (a) but does not work under disturbances as shown in Fig. 4 (b). Since our estimator assumes that contact is always maintained, once contact is broken (see Fig. 4 (b) around $t = 110$ s), the estimator diverges. In contrast, Fig. 4 (c) shows that our estimator works under disturbances since our MPC controller can react to the disturbance and maintain the desired contact state.

D. Tactile Controller Results

We demonstrate the effectiveness of our tactile controller to recover from different unexpected contacts in Sec V-B.

We first discuss recovery from slipping of the tool in the gripper fingers, i.e., non-zero θ_S (see Fig. 2 for definition of θ_S) at $t = 0$. We implemented the open- and closed-loop controllers with the above disturbance at $t = 0$. We did this experiment for 5 trials per controller. We declare

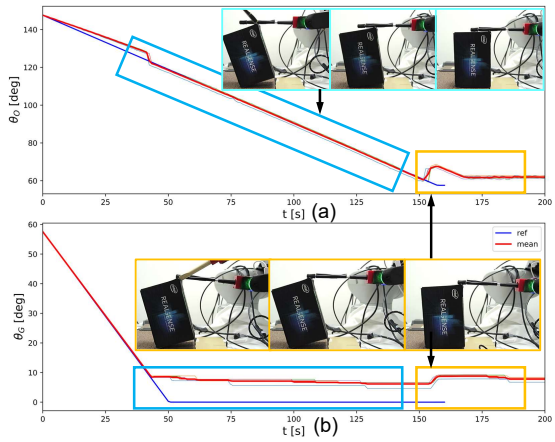

Fig. 5: **Evaluation of the closed tactile controller.** We show time history of (a) θ_O and (b) θ_G, with the closed-loop controller under disturbances at $t = 40, 150$ s. The blue line is the reference trajectory computed offline and the red trajectory is the mean of the 5 trajectories computed online.

TABLE II: **Evaluation of the closed-loop tactile controller with disturbances**. The number of successful pivoting attempts of the box over 5 trials for different disturbances are summarized.

Box	Disturbance 1			Disturbance 2	
	5°	10°	15°	$t = 40$ s	$t = 150$ s
Open-loop	4/5	0/5	0/5	0/5	N/A
Close-loop	5/5	5/5	5/5	5/5	5/5

failure if the contact is broken. The result is summarized in Table II (Disturbance 1). For $\theta_S = 5°$, both the open- and the closed-loop controllers could complete the task. However, for $\theta_S = 10°, 15°$, we observed that the open-loop controller lost the contact between the tool and object around $t = 110$ s (see Fig. 4 (b)) while the closed-loop controller could conduct the pivoting.

Next, we discuss how the closed-loop controller reacts to different unexpected contacts during the trajectory. In these experiments, we add disturbance to the object (see blue and orange box in Fig. 4 (c)) around $t = 40$ s and $t = 150$ s. We conducted 5 trials. Fig. 5 (a) shows that the closed-controller could successfully track the reference trajectory even under these unexpected contacts. The reactive control efforts can be observed around $t = 40, 150$ s in Fig. 5 (b). The robot changes its gripper orientation to maintain the constraints discussed in Sec III. The results discussed here are also summarized in Table II (Disturbance 2). Some additional results could be found in [25].

VI. Conclusions

Closed-loop control of manipulation remains elusive. This is because contacts lead to complex, discontinuous constraints that need to be carefully handled. In this paper, we presented tactile tool manipulation. More specifically, we presented the design and implementation of a closed-loop controller to control the complex mechanics of tool manipulation using tactile sensors and NLP. Through extensive experiments, we demonstrate that the proposed method provides robustness against parametric uncertainties as well as unexpected contact events during manipulation.

REFERENCES

[1] M. T. Mason, "Toward robotic manipulation," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, pp. 1–28, 2018.

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

[3] Y. Shirai, X. Lin, A. Schperberg, Y. Tanaka, H. Kato, V. Vichathorn, and D. Hong, "Simultaneous contact-rich grasping and locomotion via distributed optimization enabling free-climbing for multi-limbed robots," in *Proc. 2022 IEEE/RSJ Int. Conf. Intell. Rob. Syst.*, 2022, pp. 13 563–13 570.

[4] T. Marcucci, R. Deits, M. Gabiccini, A. Bicchi, and R. Tedrake, "Approximate hybrid model predictive control for multi-contact push recovery in complex environments," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, 2017, pp. 31–38.

[5] Y. Shirai, D. K. Jha, A. Raghunathan, and D. Romeres, "Chance-constrained optimization in contact-rich systems for robust manipulation," *arXiv preprint arXiv:2203.02616*, 2022.

[6] Y. Shirai, D. K. Jha, A. U. Raghunathan, and D. Romeres, "Robust pivoting: Exploiting frictional stability using bilevel optimization," in *Proc. 2022 IEEE Int. Conf. Robot. Automat.*, 2022, pp. 992–998.

[7] Y. Shirai, D. K. Jha, and A. U. Raghunathan, "Covariance steering for uncertain contact-rich systems," in *2023 International Conference on Robotics and Automation (ICRA)*. IEEE, accepted.

[8] Y. Hou, Z. Jia, and M. Mason, "Manipulation with shared grasping," in *Robotics: Science and Systems*, 2020.

[9] R. Holladay, T. Lozano-Pérez, and A. Rodriguez, "Force-and-motion constrained planning for tool use," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 7409–7416.

[10] K. Fang, Y. Zhu, A. Garg, A. Kurenkov, V. Mehta, L. Fei-Fei, and S. Savarese, "Learning task-oriented grasping for tool manipulation from simulated self-supervision," *The International Journal of Robotics Research*, vol. 39, no. 2-3, pp. 202–216, 2020.

[11] G. Izatt, G. Mirano, E. Adelson, and R. Tedrake, "Tracking objects with point clouds from vision and touch," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 4000–4007.

[12] D. Ma, S. Dong, and A. Rodriguez, "Extrinsic contact sensing with relative-motion tracking from distributed tactile measurements," in *Proc. 2021 IEEE Int. Conf. Robot. Automat.*, 2021, pp. 11 262–11 268.

[13] Z. Qin, K. Fang, Y. Zhu, L. Fei-Fei, and S. Savarese, "Keto: Learning keypoint representations for tool manipulation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 7278–7285.

[14] F. R. Hogan, E. R. Grau, and A. Rodriguez, "Reactive planar manipulation with convex hybrid mpc," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 247–253.

[15] F. R. Hogan, J. Ballester, S. Dong, and A. Rodriguez, "Tactile dexterity: Manipulation primitives with tactile feedback," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 8863–8869.

[16] N. Doshi, O. Taylor, and A. Rodriguez, "Manipulation of unknown objects via contact configuration regulation," in *Proc. 2022 IEEE Int. Conf. Robot. Automat.*, 2022, pp. 2693–2699.

[17] S. Dong, D. K. Jha, D. Romeres, S. Kim, D. Nikovski, and A. Rodriguez, "Tactile-RL for insertion: Generalization to objects of unknown geometry," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 6437–6443.

[18] M. Erdmann, "On a representation of friction in configuration space," *Int. J. Robo. Res.*, vol. 13, no. 3, pp. 240–271, 1994.

[19] N. Xydas and I. Kao, "Modeling of contact mechanics and friction limit surfaces for soft fingers in robotics, with experimental results," *Int. J. Robo. Res.*, vol. 18, no. 9, pp. 941–950, 1999.

[20] A. Wächter and L. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, May 2006.

[21] A. U. Raghunathan, D. K. Jha, and D. Romeres, "Pyrobocop: Python-based robotic control & optimization package for manipulation," in *Proc. 2022 IEEE Int. Conf. Robo. Auto.*, 2022, pp. 985–991.

[22] I. H. Taylor, S. Dong, and A. Rodriguez, "Gelslim 3.0: High-resolution measurement of shape, force and slip in a compact tactile-sensing finger," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 10 781–10 787.

[23] N. Hogan, "Impedance Control: An Approach to Manipulation: Part II—Implementation," *Journal of Dynamic Systems, Measurement, and Control*, vol. 107, no. 1, pp. 8–16, 03 1985.

[24] D. K. Jha, D. Romeres, S. Jain, W. Yerazunis, and D. Nikovski, "Design of adaptive compliance controllers for safe robotic assembly," *arXiv preprint arXiv:2204.10447*, 2022.

[25] Y. Shirai, D. K. Jha, A. U. Raghunathan, and D. Hong, "Tactile tool manipulation," in *Proc. 2023 IEEE Int. Conf. Robot. Automat.*, 2023.