# Characterizing Manipulation Robustness Through Energy Margin and Caging Analysis

Yifei Dong ⬤, Xianyi Cheng ⬤, *Member, IEEE*, and Florian T. Pokorny, *Member, IEEE*

*Abstract*—To develop robust manipulation policies, quantifying robustness is essential. Evaluating robustness in general manipulation, nonetheless, poses significant challenges due to complex hybrid dynamics, combinatorial explosion of possible contact interactions, global geometry, etc. This paper introduces an approach for evaluating manipulation robustness through energy margins and caging-based analysis. Our method assesses manipulation robustness by measuring the energy margin to failure and extends traditional caging concepts for dynamic manipulation. This global analysis is facilitated by a kinodynamic planning framework that naturally integrates global geometry, contact changes, and robot compliance. We validate the effectiveness of our approach in simulation and real-world experiments of multiple dynamic manipulation scenarios, highlighting its potential to predict manipulation success and robustness.

*Index Terms*—Dexterous manipulation, in-hand manipulation, contact modeling, manipulation planning.

## I. INTRODUCTION

**H**UMAN manipulation is intriguing due to its dexterity, simplicity, and remarkable robustness. Consider picking up a thin object from a table: this involves complex interactions between the object, table, and hand, yet it remains robust and fast. The robustness of human manipulation is often attributed not to precise control over or knowledge of contact interactions but to factors like hand kinematic configurations, environment geometry, joint compliance, material softness, and so on. An empirical study in [1] discovers similar robustness in in-hand manipulation with a soft robotic hand.

Nonetheless, there is a lack of methods in the literature for evaluating such general robustness. Existing research studying similar robustness problems can generally be categorized into three areas: grasp quality metrics, control stability analysis (including contraction analysis [2], Lyapunov functions [3] and barrier functions [4]), and caging. Most quality metrics in grasping [5] and non-prehensile manipulation [6] often define

robustness through object position or wrench space margins based on first-order analysis. Control stability methods assess robustness by examining system evolution from initial states, offering a local perspective that struggles to account for the robustness under the combinatorial explosion of possible contact interactions, the resulting hybrid dynamics, and global geometric properties. In contrast, caging approaches, which confine an object within a bounded space to prevent escape, offer a more global geometric analysis of robustness.

This paper proposes an approach to characterize robustness in manipulation via energy margins (Fig. 1). Our approach expands on prior quasi-static cage [7] and soft fixture [8] analyses by adapting classical caging concepts to a kinodynamic context, where dynamically feasible escape paths are considered. Utilizing forward simulation enables natural consideration of contact changes, robot joint stiffness, complex geometries, and system kinodynamics. This methodology introduces a new lens by evaluating the robustness of manipulation strategies through energy margins and broadens the applicability of caging concepts to dynamic scenarios.

The main contributions of this paper are: 1) We propose to characterize more general robustness in manipulation with energy margins and caging analysis. This approach evaluates manipulation strategies that consider global geometry, kinodynamics, and complex contact events. 2) We develop a sampling-based kinodynamic planning framework for dynamic caging analysis and energy margin calculation, designed to be inherently efficient in dynamic, contact-rich environments. 3) Through experimental validation in both real-world and simulated settings, we demonstrate the practicality and effectiveness of our approach, showcasing its benefits over a wrench-based baseline method.

## II. RELATED WORK

### A. Caging

Classical caging focuses on preventing an object from escaping using geometric constraints. This foundational concept, as introduced by Kuperberg [9] and further developed by Rimon et al. [10], [11], Maeda et al. [12], explores how robots can effectively constrain objects, without necessitating force closure or form closure. Rodriguez et al. [13] demonstrate utilizing caging to achieve immobilizing grasps. Energy-bounded caging [14], [15] relaxes complete geometric caging and considers the work done by external forces, such as gravity [14] and planar friction [15], as a source to cage objects. Caging has also been leveraged in the robust manipulation of biological cells using optical tweezers [16], [17]. Our previous work on soft fixtures [8] presents sampling-based methods to search for upper-bound estimates of escape energy in higher-dimensional configuration spaces, such as simplified deformable objects. Our
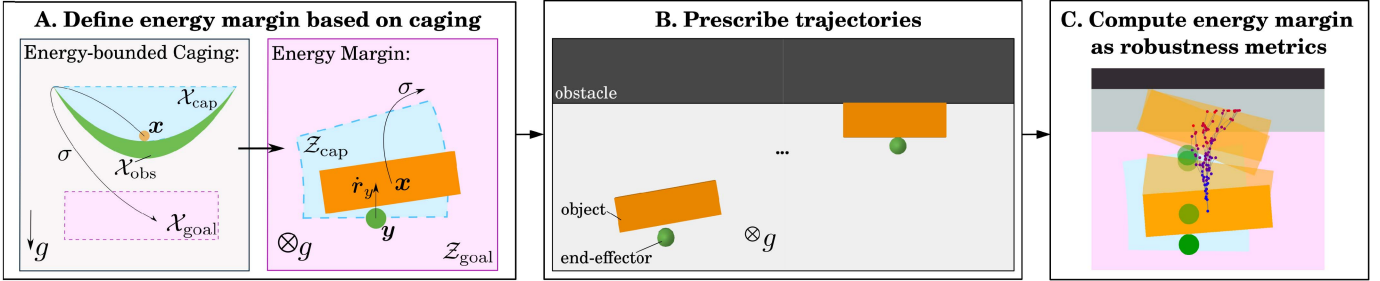
Fig. 1.    Inspired by caging analysis (A, left), we propose to characterize the robustness of manipulation actions using energy margins (A, right) from failure regions (pink). Given prescribed trajectories of manipulation tasks (B, planar pushing as an example), we use sampling-based kinodynamic planners to compute the metrics (C), which are used to predict manipulation robustness and success under uncertainty.

approach presents a new perspective that enables caging analysis to be used to characterize general manipulation robustness. It also extends caging into dynamic settings, bridging the gap between static caging theories and the dynamic nature of robotic manipulation.

### B. Robustness and Quality Metrics

Most research in manipulation robustness focuses on grasp quality measures. A comprehensive review by Roa et al. [5] categorizes classical quality measures based on contact configurations and wrench space profiles. Many metrics are derived from Grasp Wrench Space (GWS) [18], which constructs a wrench cone formed by grasping contacts, such as the volume of the GWS [19], the largest perturbation wrench the grasp can resist in any direction [20], etc. Lin et al. [21] introduce a task-specific grasp quality criterion based on the distribution of task disturbance, exemplifying the importance of task-specific optimization of grasps. Xu et al. [22] propose minimal work as a grasp quality metric to plan grasps for 3D deformable hollow objects considering wrench resistance. Machine learning approaches are also employed to predict grasp success based on data labels derived from analytic quality metrics [23]. For non-prehensile manipulation, a framework of quasi-static wrench-based analysis is presented for balancing grasps in [24]. Makapunyo et al. [25] employ sampling-based methods to determine a quality metric for partial caging. While previous work mostly focuses on first-order and instantaneous analysis to quantify how good an action is under position or object wrench uncertainty, we use caging to characterize robustness through a more global analysis, with the ability to take into account many hard-to-model effects, such as contact changes and friction uncertainty.

### III. PRELIMINARIES

### A. Nomenclature

We denote by $\mathcal{X} \subset \mathrm{SE}(3)$ the configuration space (C-space) of a 3D rigid object. The set of time derivatives of the object configuration is denoted by $\dot{\mathcal{X}} \subset T\mathrm{SE}(3)$, where $T\mathrm{SE}(3)$ is the tangent space associated with the Lie group $\mathrm{SE}(3)$. The C-space of a robot end-effector and surrounding static obstacles is denoted by $\mathcal{Y} \subset \mathrm{SE}(3) \times \mathbb{R}^{n_r}$, where $\mathrm{SE}(3)$ denotes the base pose, and $n_r$ refers to the number of joints. Its set of time derivatives is denoted by $\dot{\mathcal{Y}} \subset T\mathrm{SE}(3) \times \mathbb{R}^{n_r}$. $\mathcal{Y}$ might

degenerate when, for example, the base of the end-effector is fixed, and then $\mathcal{Y} \subset \mathbb{R}^{n_r}$. For planar manipulation, $\mathcal{X}$ and $\mathcal{Y}$ degenerate to $\mathrm{SE}(2)$ and $\mathrm{SE}(2) \times \mathbb{R}^{n_r}$. An element $\boldsymbol{x} \in \mathcal{X}$, described as $\boldsymbol{x} = (\boldsymbol{r}_x, \boldsymbol{q}_x)$, encompasses the position $\boldsymbol{r}_x$ of the Center of Mass (CoM) and orientation $\boldsymbol{q}_x$ (a unit quaternion). An element $\dot{\boldsymbol{x}} \in \dot{\mathcal{X}}$ is described as $\dot{\boldsymbol{x}} = (\dot{\boldsymbol{r}}_x, \dot{\boldsymbol{q}}_x)$, which comprises the CoM linear velocity $\dot{\boldsymbol{r}}_x$ and the derivative of quaternion $\dot{\boldsymbol{q}}_x$. Similarly for $\boldsymbol{y} \in \mathcal{Y}$ and $\dot{\boldsymbol{y}} \in \dot{\mathcal{Y}}$, we have $\boldsymbol{y} = (\boldsymbol{r}_y, \boldsymbol{q}_y, \boldsymbol{\alpha}_y)$ and $\dot{\boldsymbol{y}} = (\dot{\boldsymbol{r}}_y, \dot{\boldsymbol{q}}_y, \dot{\boldsymbol{\alpha}}_y)$, where $\boldsymbol{\alpha}_y$ and $\dot{\boldsymbol{\alpha}}_y$ refer to the joint position and velocity of the end-effector. $\boldsymbol{r}_y$ and $\boldsymbol{q}_y$ denote the position and orientation of the end-effector.

### B. Cage and Soft Fixture

Here, we revisit the concept of caging. The free C-space $\mathcal{X}_{\mathrm{free}}$ indicates a set of configurations for which the object does not penetrate any of the bodies (the end-effector or obstacles) in the workspace.

*Definition 1:* A *cage* occurs when an object configuration $\boldsymbol{x}_{\mathrm{init}} \in \mathcal{X}_{\mathrm{free}}$ is situated in a bounded path component of $\mathcal{X}_{\mathrm{free}}$.

An object in a cage thus indicates its limited capacity to freely move beyond a certain proximity to its starting configuration. Caging was relaxed as partial caging by allowing escape paths through narrow passages in the free space [25]. Partial caging further extends to energy-bounded caging [14] and soft fixtures [8] that consider both geometric and potential energy constraints, as introduced below.

*Definition 2:* In scenarios where only conservative forces act within a quasi-static system, an object at an initial configuration $\boldsymbol{x}_{\mathrm{init}} \in \mathcal{X}_{\mathrm{free}}$ possesses potential energy $E(\boldsymbol{x}_{\mathrm{init}})$. Its *escape energy* $\mathcal{E}(\boldsymbol{x}_{\mathrm{init}})$ refers to the supremum of $e$ such that the path component $\mathcal{PC}_{\boldsymbol{x}_{\mathrm{init}}}(\mathcal{X}_e(\boldsymbol{x}_{\mathrm{init}}))$ of a sublevel set $\mathcal{X}_e(\boldsymbol{x}_{\mathrm{init}})$ containing $\boldsymbol{x}_{\mathrm{init}}$ is bounded:

$$\mathcal{E}(\boldsymbol{x}_{\mathrm{init}}) = \sup \left\{ e \geq 0 : \mathcal{PC}_{\boldsymbol{x}_{\mathrm{init}}}(\mathcal{X}_e(\boldsymbol{x}_{\mathrm{init}})) \text{ is bounded} \right\}, \quad (1)$$

where

$$\mathcal{X}_e(\boldsymbol{x}_{\mathrm{init}}) = \{\boldsymbol{x} \in \mathcal{X}_{\mathrm{free}} : E(\boldsymbol{x}) \leq E(\boldsymbol{x}_{\mathrm{init}}) + e\}. \quad (2)$$

The object is deemed to be in a *soft fixture* if $\mathcal{E}(\boldsymbol{x}_{\mathrm{init}}) > 0$. Note that $\mathcal{E}(\boldsymbol{x}_{\mathrm{init}})$ is defined only if $\mathcal{PC}_{\boldsymbol{x}_{\mathrm{init}}}(\mathcal{X}_0(\boldsymbol{x}_{\mathrm{init}}))$ is bounded.

In practice, sampling-based approaches can approximate a close upper bound of $\mathcal{E}(\boldsymbol{x}_{\mathrm{init}})$ by utilizing a goal region (Fig. 1-A, left) sufficiently far away from the obstacles with relatively lower energy values and searching for geometrically feasible escape paths with lowest energy cost [8].

## C. Assumptions

We introduce the assumptions that this paper relaxes compared to our prior work [8].

*1) System Motion:* While the previous work only considers quasi-static conditions, here we account for the movement of both the object and the end-effector, while keeping other environmental elements static. We introduce $\mathcal{Z} = \mathcal{X} \times \dot{\mathcal{X}} \times \mathcal{Y} \times \dot{\mathcal{Y}}$, where $\mathcal{Z}_{\text{free}}$ represents penetration-free space. In this context, the energy function $E$ is the system's total mechanical energy (sum of kinetic and potential energy).

*2) Non-Conservative Forces:* Given the existence of non-conservative forces, such as friction, the previous definition of escape energy in Def. 1 is problematic. Consider the example of planar pushing (Fig. 1-A, right), assuming the tabletop with a rough surface is infinitely large. The escape energy $\mathcal{E}(\boldsymbol{x}_{\text{init}})$ is always infinite in this scenario because the object overcomes friction along any escape path to a point that is arbitrarily far away. To utilize escape energy for more general cases like this, we define a *capture set* $\mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}}) \subset \mathcal{Z}$ instead,

$$\mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}}) = \{\boldsymbol{z} \in \mathcal{Z} : f(\boldsymbol{z}) \leq c_1(\boldsymbol{z}_{\text{init}}), g(\boldsymbol{z}) = c_2(\boldsymbol{z}_{\text{init}})\}. \tag{3}$$

The constraints above are task-specific, refraining the system from transitioning to failure modes. For instance, in the pushing task (Fig. 1-A, right), $\mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}})$ contains positional constraints that prevent the object from slipping away from the end-effector. A system state $\boldsymbol{z} \in \mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}})$ indicates the object as dynamically controllable by the end-effector for achieving specific tasks. The effort of the object to escape from $\mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}})$ thus indicates manipulation robustness, implying an energy margin from failure. Several examples of task-specific $\mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}})$ can be found in Section VI-A and Fig. 2.

*3) Dynamic Escape Paths:* We consider dynamically feasible escape paths that lead the object out of the capture set $\mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}})$. A dynamically feasible path $\sigma : [0, 1] \rightarrow \mathcal{Z}_{\text{free}}$, such that $\sigma(0) = \boldsymbol{z}_{\text{init}}$ and $\sigma(1) = \boldsymbol{z}_{\text{goal}}$, is generated by applying random wrenches to the object to mimic perturbation or uncertainty during the manipulation. Here, $\boldsymbol{z}_{\text{goal}} \in \mathcal{Z}_{\text{goal}}$ and $\mathcal{Z}_{\text{goal}}$ is a goal set of escape.

Besides the assumptions above, we only consider rigid objects under potential energy fields, though our framework can potentially be extended to scenarios with articulated or multiple objects with more computational costs caused by the increased degrees of freedom of the system.

## D. Problem Statement

We tackle the following problem: Given a manipulation task objective and the state $\boldsymbol{z}_{\text{init}}$ of a rigid object and an end-effector in motion, which is subject to the assumptions in Section III-C, 1) Define metrics to characterize robustness in manipulation based on energy margins to failure (Section IV). 2) Compute the robustness metrics using kinodynamic motion planning and evaluate their performance (Sections V and VI).

## IV. ENERGY MARGIN DEFINITIONS

We quantify robustness with two methods based on energy margin: the effort of escape (the ease of failure) and the capture score (the probability of staying safe).

## A. Effort of Escape

The effort of escape calculates the minimal effort required to exit the capture set.

*Definition 3:* The minimal *effort of escape* is the minimal integral of the absolute power $|\dot{W}_{\text{ext}}(\sigma(t))|$ of the external work $W_{\text{ext}}(\sigma(t))$ required to be done to help the object escape from the capture set $\mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}})$:

$$\Omega_{\text{esc}}(\boldsymbol{z}_{\text{init}}) = \min_{\sigma \in \Sigma(\boldsymbol{z}_{\text{init}}, \mathcal{Z}_{\text{goal}})} \int_0^1 |\dot{W}_{\text{ext}}(\sigma(t))| dt, \tag{4}$$

where

$$\dot{W}_{\text{ext}}(\sigma(t)) = \frac{dE(\sigma(t))}{dt} - \dot{W}_{\text{fri}}(\sigma(t)). \tag{5}$$

$\frac{dE(\sigma(t))}{dt}$ is the instantaneous rate of change of the mechanical energy of the system, and $W_{\text{fri}}(\sigma(t))$ represents the work done by friction at the instantaneous state of the system. We denote $\Sigma(\boldsymbol{z}_{\text{init}}, \mathcal{Z}_{\text{goal}})$ as the set of all escape paths starting from $\boldsymbol{z}_{\text{init}}$ terminating in $\mathcal{Z}_{\text{goal}}$, where $\mathcal{Z}_{\text{goal}} = \mathcal{Z}_{\text{cap}}^c(\boldsymbol{z}_{\text{init}})$ (the complement set of the capture set).

Equation (5) follows the law of conservation of energy. We employ the absolute value of the power $|\dot{W}_{\text{ext}}(\sigma(t))|$ as a measure of the total control input that we want to minimize because, for example, both speeding up and slowing down a vehicle require fuel, and we might want to minimize the total fuel used over a journey. Since paths in the path space $\Sigma(\boldsymbol{z}_{\text{init}}, \mathcal{Z}_{\text{goal}})$ can hardly be enumerated or optimized analytically given the complexity introduced by friction, we employ kinodynamic sampling-based planners to filter candidate paths that upper bounds $\Omega_{\text{esc}}(\boldsymbol{z}_{\text{init}})$.

## B. Capture Score

The capture score estimates the likelihood of remaining within the capture set. We consider an energy cost field and its correlating probabilistic distribution of the system. Specifically, a sequence of random wrenches applied on the object at $\boldsymbol{z}_{\text{init}}$ lead it to $\boldsymbol{z}^0$ with a cost of total effort $c^0$. By repeating the process $M$ times from $\boldsymbol{z}_{\text{init}}$, the system terminates in a list of states $\{\boldsymbol{z}^0, \ldots, \boldsymbol{z}^M\}$ with corresponding costs $\{c^0, \ldots, c^M\}$. Data pairs in $\{(\boldsymbol{z}^0, c^0), \ldots, (\boldsymbol{z}^M, c^M)\}$ constitute an energy cost field that demonstrates the state space reachability in terms of energy cost from $\boldsymbol{z}_{\text{init}}$. We provide a probabilistic interpretation of the likelihood of reaching each state $\boldsymbol{z}^m \in \mathcal{Z}$ by introducing a probability mass function $L : \mathcal{Z} \rightarrow \mathbb{R}^+$,

$$L(\boldsymbol{z}^m) = \frac{e^{-\lambda(c^m - c_{\text{min}})}}{\sum_{m=0}^M e^{-\lambda(c^m - c_{\text{min}})}}, \tag{6}$$

where $c_{\text{min}} = \min_{0 \leq m \leq M} c^m$ and $\lambda$ is a hyper-parameter. The softmax function in (6) is ideal for modeling hybrid continuous-to-discrete mappings in such stochastic systems [26]. The approximated likelihood function indicates some states with lower costs are probabilistically more reachable.

*Definition 4:* The *capture score* is defined as the sum of likelihood values of samples in a capture set $\mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}})$,

$$\Omega_{\text{cap}}(\boldsymbol{z}_{\text{init}}) = \sum_{m=0}^M \delta(\boldsymbol{z}^m \in \mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}})) \cdot L(\boldsymbol{z}^m), \tag{7}$$

where $\delta(\cdot)$ is an indicator function that equals 1 if the condition inside the brackets is satisfied and 0 otherwise.

Similarly, we define a task success set $\mathcal{Z}_{\text{suc}} \subset \mathcal{Z}$ independent of $z_{\text{init}}$, indicating the set of states symbolizing the object accomplishes the manipulation objective. A capture score of success is thus given by

$$\Omega_{\text{suc}}(z_{\text{init}}) = \sum_{m=0}^{M} \delta(z^m \in \mathcal{Z}_{\text{suc}}) \cdot L(z^m), \qquad (8)$$

which is essentially a predictor for fulfilling a task-specific objective from the state $z_{\text{init}}$. In practice, we employ kinodynamic motion planners rather than repetitive Monte Carlo rollouts from $z_{\text{init}}$. Sampling-based kinodynamic motion planners are in general more time- and memory-efficient by using strategies such as biased sampling and caching explored nodes.

## V. COMPUTING ENERGY MARGIN THROUGH KINODYNAMIC PLANNING

We develop kinodynamic motion planning algorithms to compute the scores in Section IV. We calculate the effort of escape $\Omega_{\text{esc}}(z_{\text{init}})$ by an iterative tree search algorithm (Section V-B). We calculate the capture scores $\Omega_{\text{cap}}(z_{\text{init}})$ and $\Omega_{\text{suc}}(z_{\text{init}})$ by growing an expansive tree and approximating its correlating probabilistic distribution (Section V-C).

### A. Planning Objectives

We consider a system with an object, an end-effector, and the environment. The system state at time $k$ is

$$z(k) = \begin{bmatrix} x^\top(k) & \dot{x}^\top(k) & y^\top(k) & \dot{y}^\top(k) \end{bmatrix}^\top. \qquad (9)$$

A manipulation trajectory is denoted by $\{(k, z(k)) : k \in [0, K]\}$, complying with the system dynamics $z'(k) = F(z(k), \phi(k))$. $\phi$ denotes the robot control. The system dynamics is rolled out through physics simulation. For each state $z(k)$ along the trajectory, Algorithm 1 computes the minimal effort of escape $\Omega_{\text{esc}}(z(k))$, and Algorithm 2 computes the capture scores $\Omega_{\text{cap}}(z(k))$ and $\Omega_{\text{suc}}(z(k))$. We assume that the end-effector is not actuated in the short period when computing the metrics so as not to introduce extra energy sources to the system.

---

**Algorithm 1:** Compute Effort of Escape.

1  Initialize, $P$, $A$, $n$
2  $\sigma_0 \leftarrow A(P_\infty)$                              ▷ Find a first escape path
3  **if** $\sigma_0$ *does not exists* **then**
4  |    $P$ has no solution
5  |    **Return** $\infty$                                  ▷ Infinite effort to escape
6  $c_0 \leftarrow C(\sigma_0)$                        ▷ EQ. (10). Initial effort of escape
7  **for** $i = 1, 2, \ldots, n$ **do**
8  |    $\sigma_i \leftarrow A(P_{c_{i-1}})$                   ▷ Find a path of lower effort
9  |    $c_i \leftarrow C(\sigma_i)$            ▷ EQ. (10). Update effort upper bound
10 $\Omega_{\text{esc}}(z_{\text{init}}) \leftarrow c_n$                    ▷ EQ. (4). Final effort of escape
11 **Return** $\Omega_{\text{esc}}(z_{\text{init}})$

---

### B. Computing Effort of Escape

We frame finding the minimum effort to escape from the capture set $\mathcal{Z}_{\text{cap}}(z_{\text{init}})$ as an optimal kinodynamic motion planning problem $P$ of which the goal set is the complement set of the capture set ($\mathcal{Z}_{\text{goal}} = \mathcal{Z}_{\text{cap}}^c(z_{\text{init}})$). We employ the AO-$x$

meta algorithms [27], which are asymptotically optimal motion planners that decompose $P$ into feasible motion planning problems in the state-cost space. They iteratively find an upper bound approaching the minimum effort of escape by running as subroutines feasible kinodynamic planner $x$, such as Expansive Space Tree (EST) [28] or Rapidly-exploring Random Tree (RRT) [29] algorithms.

The optimal motion planning problem $P = (Q, \mathcal{Z}_{\text{free}}, \mathcal{U}, z_{\text{init}}, \mathcal{Z}_{\text{goal}}, \mathcal{Z}_b, \mathcal{U}_b, G)$ produces a trajectory $\sigma(t) : [0, 1] \to \mathcal{Z}_{\text{free}}$ and control $u(t) : [0, 1] \to \mathcal{U}$ that minimizes the objective functional:

$$C(\sigma) = \int_0^1 Q(\sigma(t), u(t)) dt = \int_0^1 |\dot{W}_{\text{ext}}(\sigma(t))| dt. \qquad (10)$$

Here, $Q$ denotes the incremental cost (terminal cost is 0) following (4). $C(\sigma)$ thereby refers to the total effort of transitioning from $\sigma(0)$ to $\sigma(1)$. $\mathcal{U}_b$ is the set of control constraints and $u \in \mathcal{U}_b \subset \mathcal{U}$. $\mathcal{Z}_b$ refers to the set of kinematic constraints and $z \in \mathcal{Z}_b \subset \mathcal{Z}_{\text{free}}$. The dynamics is subjected to $z' = G(z, u)$. The control input $u = \begin{bmatrix} f^\top & \tau^\top \end{bmatrix}^\top$ refers to a wrench (force $f$ applied at its CoM and torque $\tau$) applied on the object, mimicking an external perturbation.

Given the optimal motion planning problem $P$, a feasible motion planning problem $P_{\bar{c}} = (\mathcal{Z}_{\text{free}} \times \mathbb{R}^+, \mathcal{U}, (z_{\text{init}}, 0), \hat{\mathcal{Z}}_{\text{goal}}^{\bar{c}}, \mathcal{Z}_b \times \mathbb{R}^+, \mathcal{U}_b, \hat{G})$ is solved at each iteration. $P_{\bar{c}}$ augments the state in $P$ by an auxiliary cost variable, which measures the accumulated cost from the root $z_{\text{init}}$, i.e. cost-to-come. Here, $\hat{\mathcal{Z}}_{\text{goal}}^{\bar{c}} = \{(z, c) : z \in \mathcal{Z}_{\text{goal}}, c \in [0, \bar{c}]\}$ denotes the augmented goal set with the range of cost space upper-bounded by $\bar{c}$, and $\hat{G}$ refers to the augmented dynamics

$$\hat{z}' = \begin{bmatrix} z' \\ c' \end{bmatrix} = \begin{bmatrix} G(z, u) \\ Q(z, u) \end{bmatrix}. \qquad (11)$$

We solve $P$ by converting it to a series of feasible motion planning subproblems $P_{\bar{c}}$ and iteratively lowering the cost space upper bound $\bar{c}$ utilizing a well-behaved, probabilistically complete feasible motion planner $A$ (EST or RRT). Informally, a planning algorithm is well-behaved [27] if it can determine a feasible solution in finite time if it exists, and incrementally improves the cost of any found solution by a nonneglegible fraction in each further iteration.

Specifically, we first run the subroutine algorithm $A(P_\infty)$ with infinite cost space upper bound $\bar{c} = \infty$ to find the first escape path $\sigma_0$ by growing a tree rooted at $z_{\text{init}}$ (Algorithm 1, Line 2). The tree expands a new node $z'$ by enforcing a randomized control $u$ based on algorithm-specific strategies from an existing node $z$, complying with the system dynamics $G$. The forward integration is performed in a physical simulator that allows rebounding after a collision. A new extension is valid if $z'$ abides by the augmented kinematic constraints $(z', c(z')) \in \mathcal{Z}_b \times \mathbb{R}^+$. If a path does not exist, it indicates the effort of escape is infinite $\Omega_{\text{esc}}(z_{\text{init}}) = \infty$ (Line 3-5). Once we found a path $\sigma_0$ with cost $c_0 = C(\sigma_0)$, we solve a second subproblem $P_{c_0}$ (Line 8) that shrinks the cost space by lowering the upper bound from $\infty$ to $c_0$. The tree in the previous subproblem is cached and optionally pruned (discarding the nodes with costs higher than $c_0$) such that we do not grow a new tree from scratch. $A(P_{c_0})$ results in a new escape path $\sigma_1$ with cost $c_1$ (Line 9), and so on and so forth. Given the well-behavedness of $A$, the cost $c_i$ approaches the optimal cost $c^*$ as the total number of iterations approaches infinity. Therefore, we can approximately upper-bound the minimum

---

**Algorithm 2:** Compute Capture Scores.

1   Initialize, $P_\infty$, $\mathcal{Z}_{\text{cap}}(z_{\text{init}})$, $\mathcal{Z}_{\text{suc}}$, $M$
2   ▷ Obtain energy cost field by growing a tree
3   $\{(z^0, c^0), ..., (z^M, c^M)\} \leftarrow \text{EST}(P_\infty, M)$
4   ▷ Approximate probabilistic distribution
5   $c_{\min} = \min_{0 \le m \le M} c^m$
6   $S \leftarrow \sum_{m=0}^{M} e^{-\lambda(c^m - c_{\min})}$
7   **for** $m = 1, 2, \ldots, M$ **do**
8      $L(z^m) \leftarrow \frac{1}{S} e^{-\lambda(c^m - c_{\min})}$        ▷ EQ. (6).
9   ▷ Compute scores
10   $\Omega_{\text{cap}}(z_{\text{init}}) \leftarrow \sum_{m=0}^{M} \delta(z^m \in \mathcal{Z}_{\text{cap}}(z_{\text{init}})) \cdot L(z^m)$   ▷ EQ. (7).
11   $\Omega_{\text{suc}}(z_{\text{init}}) \leftarrow \sum_{m=0}^{M} \delta(z^m \in \mathcal{Z}_{\text{suc}}) \cdot L(z^m)$   ▷ EQ. (8).
12   **Return** $\Omega_{\text{cap}}(z_{\text{init}})$, $\Omega_{\text{suc}}(z_{\text{init}})$

---



Fig. 2.   Examples of manipulation primitives. The capture sets $\mathcal{Z}_{\text{cap}}$ (blue) and task success sets $\mathcal{Z}_{\text{suc}}$ (gray) are marked.

effort of escape $\Omega_{\text{esc}}(z_{\text{init}})$ in finitely many $(n)$ iterations. In practice, we consider EST or RRT as the subroutine algorithms $A$. RRT randomly samples a state in the state space and extends toward the sampled state from the nearest neighbor node in the tree. EST prioritizes extensions in the area with less density of existing nodes in the tree.

### C. Computing Capture Scores

To approximate the capture scores, we construct an energy cost field through kinodynamic tree expansion. Achieving an evenly distributed sampling of state space around the initial state $z_{\text{init}}$ is critical for this purpose. EST is thereby selected for its efficacy in promoting uniform distribution across the state space through inverse density weighting, unlike RRT, which may lead to uneven exploration. As detailed in Algorithm 2, we consider the cost-augmented unbounded feasible motion planning problem $P_\infty$ (Line 2-3) with the same settings as in Section V-B except a sufficiently far away goal set $\mathcal{Z}_{\text{goal}} = \mathcal{Z}_\infty \subset \mathcal{Z}$ with unconstrained kinematics $\mathcal{Z}_b = \mathcal{Z}_{\text{free}}$. EST terminates after growing $M$ nodes in the tree. We thus obtain an energy cost field of $M$ state-cost pairs, $\{(z^0, c^0), \ldots, (z^M, c^M)\}$. We thereafter approximate the probabilistic distribution (Line 4-8) and compute the scores $\Omega_{\text{cap}}(z_{\text{init}})$ and $\Omega_{\text{suc}}(z_{\text{init}})$ (Line 9-11) following the procedure in Section IV-B.

## VI. EVALUATION

This section aims to validate the practicality and effectiveness of our metrics and algorithms by conducting simulation and real-world experiments. Our key assumption is that energy margins and caging analysis can effectively characterize robustness in manipulation. To verify this, we designed four simulated manipulation tasks and used our metrics to predict their robustness and task success. More specifically, we generated multiple trajectories of the tasks and recorded their system kinematic information and ground-truth labels of robustness and success. We computed the metrics offline, used them to make predictions and compared the results with the ground-truth labels. Through the experiments, we validated the reliable prediction capability of our metrics in environments with complex contact events. Furthermore, we verify the efficiency of our kinodynamic planning algorithms and the robustness against different model parameter errors.
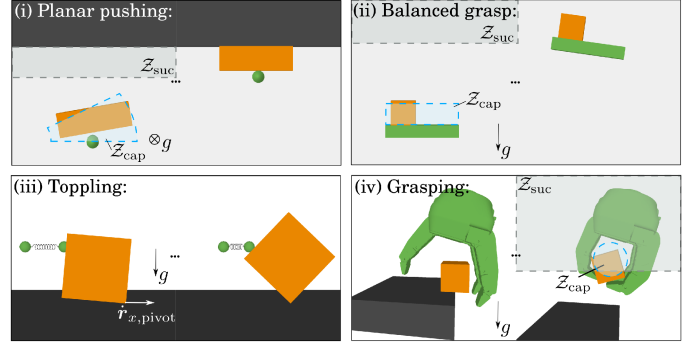
### A. Task Description

Inspired by the manipulation primitive taxonomy in [30], [31], we design examples covering planar pushing, balanced transport, toppling, and grasping from a table.

*1) Planar Pushing:* We consider the planar manipulation problem of pushing an object on a horizontal plane towards a wall (Fig. 2-i). The task is considered successful if a state $z \in \mathcal{Z}_{\text{suc}}$ lies in the task success set $\mathcal{Z}_{\text{suc}} = \{z \in \mathcal{Z} : r_x \in \mathcal{R}_{\text{suc}}\}$, and $\mathcal{R}_{\text{suc}} \subset \mathbb{R}^2$ refers to the set of object positions where the longest edge of the convex hull of the object aligns with the wall. The capture set $\mathcal{Z}_{\text{cap}}(z_{\text{init}}) = \{z \in \mathcal{Z} : r_x \in \mathcal{R}_{\text{cap}}(z_{\text{init}})\}$ includes state $z \in \mathcal{Z}$ such that the CoM position of the object $r_x$ lies inside $\mathcal{R}_{\text{cap}}(z_{\text{init}}) \subset \mathbb{R}^2$. $\mathcal{R}_{\text{cap}}(z_{\text{init}})$ indicates the circular sector region that the end-effector will sweep over given the current instantaneous rotation center of the end-effector.

*2) Balanced Transport:* We aim to balance a cube on a steep slope with a rectangular support surface so that it avoids the failure mode of falling off and is transported to a target region (Fig. 2-ii). Similar to the selection of the capture set $\mathcal{Z}_{\text{cap}}(z_{\text{init}})$ and the task success set $\mathcal{Z}_{\text{suc}}$ above, we add constraints on the object CoM position $r_x$ such that the capture condition (maintaining on the support surface, $\mathcal{R}_{\text{cap}}(z_{\text{init}})$) and the task success condition (reaching a goal region $\mathcal{R}_{\text{suc}}$) are satisfied.

*3) Toppling:* We consider the 2D task of toppling a box on the table using a spring-like fingertip manipulator (Fig. 2-iii). The task is accomplished when the box rotates by $\pi/2$ rad pivoting on one corner in contact with the table, i.e. $\mathcal{Z}_{\text{suc}} = \{z \in \mathcal{Z} : p(q_x) = \pi/2\}$, where $p(q_x)$ is the SO(2) orientation of the box in Euler angle. It often fails when sliding occurs on the table contact. The capture set is defined as $\mathcal{Z}_{\text{cap}}(z_{\text{init}}) = \{z \in \mathcal{Z} : \|\dot{r}_{x,\text{pivot}}\| \le 0.1\}$, i.e. the magnitude of the sliding velocity of the table contact is smaller than 0.1 m/s.

*4) Grasping from a Table:* The task of grasping a box lying on the desktop is considered here, using a Robotiq three-finger adaptive robot gripper with $n_r = 12$ revolute joints in the fingers (Fig. 2-iv). The task is successful when the box of edge length 10 cm is lifted above the desktop by 10 cm, while it fails if it slips and falls from the gripper. The position-controlled compliant gripper poses an external energy bound for the box to escape. Constraints on the object CoM position $r_x$ are considered similarly as in the first two examples in selecting the sets $\mathcal{Z}_{\text{cap}}(z_{\text{init}})$ and $\mathcal{Z}_{\text{suc}}$.

Several other tasks with evaluations of simultaneously manipulating multiple objects, including pushing and scooping, can be found in our recent workshop contribution.[1]

### B. Baseline

We compare our approach with a manipulation score in [6], which is designed to measure the robustness to maintain desired contact modes (sticking, sliding, etc) against disturbance forces and avoid failure contact modes (e.g. avoid slipping for sticking contacts). However, our energy margin approach quantifies manipulation robustness more globally and allows changes of contacts. To make a fair comparison, we design a hybrid force-based score $\Omega_{\text{force}}$ with a weighted sum of the engaging score $\Omega_{\text{engage}}$, the sticking score $\Omega_{\text{stick}}$ and a kinematic heuristic $\Omega_{\text{dist}}$ - the minimum distance between any two points on the object and the end-effector. $\Omega_{\text{engage}}$ uses the magnitude of the contact normal force $\lambda_{\perp}$ to evaluate how much normal disturbance forces a contact can withstand, $\Omega_{\text{engage}} = \lambda_{\perp}$. $\Omega_{\text{stick}}$ refers to the minimal amount of disturbance force required to transition the sticking contact into sliding, which is the distance between the contact force and the friction cone edges (i.e., a wrench margin away from failure):

$$\Omega_{\text{stick}} = \left(\mu\lambda_{\perp} - |\lambda_{\parallel}|\right)\cos(\arctan\mu), \tag{12}$$

where $\mu$ refers to the static friction coefficient.

### C. Data Generation and Implementation

We collected trajectories and per-frame system states of the 4 tasks to compute their energy margins offline and compare them with ground-truth robustness and success labels.

*1) Software:* Our code can be found here.[2] Pybullet served as the platform for forward simulation. All experiments are performed on an Intel Core i9-12900H processor with 14 cores and speeds up to 5.0 GHz.

*2) Ground-truth Data Generation:* In the simulation, we generated and recorded 50 trajectories for each of the four examples in Fig. 2 by randomizing the initial states of the object and the end-effector, the friction coefficients, etc. 10 frames ($K = 10$) are evenly selected from each trajectory and the system states across the trajectory $z_k, k \in \{1, \ldots, K\}$ are recorded. As an instance, two of such scripted trajectories with three states each are illustrated in Fig. 3. The hybrid force-based score $\Omega_{\text{force}}(z_k)$ is computed by recording the friction coefficient $\mu$, the contact normal force $\lambda_{\perp}$ and lateral force $\lambda_{\parallel}$ in each frame. A trajectory is labeled 1 (successful, e.g. Fig. 3, Trajectory 2) if the task objective is accomplished in the last recorded frame, i.e. $z_K \in \mathcal{Z}_{\text{suc}}$, otherwise 0 (Fig. 3, Trajectory 1); and a state in a frame $z_k$ is labeled 1 (captured) if the state is contained in the capture set for the current and subsequent $\hat{k}$ recorded frames $z_{k'} \in \mathcal{Z}_{\text{cap}}(z_{k'})$ for $k' \in \{k, \ldots, k + \hat{k}\}$. We collected 108 successful trajectories and 92 failed trajectories.

*3) Implementation Details:* To analyze the effectiveness of our proposed metrics in predicting the robustness and success of manipulation tasks, we utilized the Area Under the receiver operating characteristic Curve (AUC) and the Average Precision (AP) as evaluation tools. We run the algorithms in Section V and obtain $\Omega_{\text{esc}}(z_k)$, $\Omega_{\text{cap}}(z_k)$ and $\Omega_{\text{suc}}(z_k)$ for each recorded state

[1]https://yifeidong0.github.io/assets/pdf/ICRA2024_Multi_Object.pdf
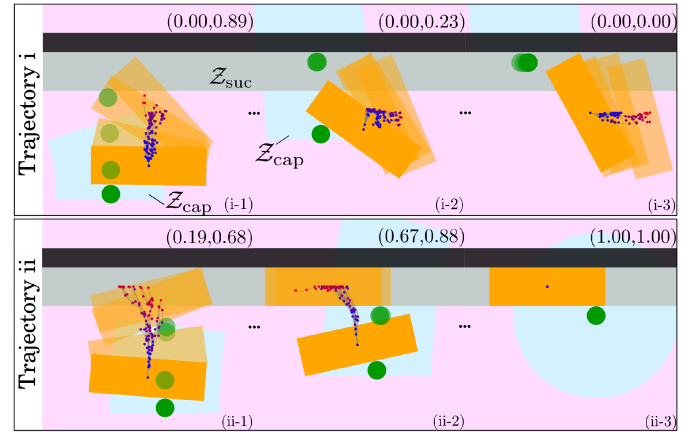[2]https://github.com/yifeidong0/EnergyMarginDynCage



Fig. 3. Qualitative evaluation for the planar pushing example. Data in the subfigures indicate $(\Omega_{\text{suc}}, \Omega_{\text{cap}})$. (i) and (ii) refer to a failed and a successful trajectory of pushing a rectangular box (yellow) to the wall (black). Three screenshots are taken along each trajectory, for which we run Algorithm 2 and visualize the expansive tree after running 100 iterations. The nodes on the tree are shown in colored dots (CoM position of the box) ranging from blue to red, implying the energy cost field. More reddish dots indicate nodes with higher cost-to-come $c$. Box and circular end-effector (green) configurations of three random nodes on the tree are visualized in partial transparency. Note the capture set $\mathcal{Z}_{\text{cap}}$ (light blue) and the task success set $\mathcal{Z}_{\text{suc}}$ (gray).

TABLE I
ENERGY MARGINS EVALUATION IN SIMULATION (AUC/AP)

| Task | $\Omega_{\text{cap}}$ | $\Omega_{\text{esc}}$ (**EST**) | $\Omega_{\text{esc}}$ (**RRT**) | $\Omega_{\text{force}}$ (**baseline**) | $\Omega_{\text{suc}}$ |
|------|------|------|------|------|------|
| Pushing | **0.97/0.99** | 0.92/0.98 | 0.92/**0.99** | 0.86/0.95 | 0.98/0.98 |
| Balancing | **0.98/0.99** | 0.93/0.98 | 0.94/0.98 | 0.94/0.96 | 0.93/0.95 |
| Toppling | **0.94/0.93** | 0.93/0.91 | 0.93/0.92 | 0.77/0.62 | 0.89/0.92 |
| Grasping | **1.00/1.00** | 0.99/0.99 | 0.97/0.99 | 0.73/0.88 | 0.99/0.99 |

The bold values are significant here to emphasize highest scores.

$z_k$ in each trajectory of the examples. The hybrid force-based score $\Omega_{\text{force}}(z_k)$ is computed by recording the friction coefficient $\mu$, the contact normal force $\lambda_{\perp}$ and lateral force $\lambda_{\parallel}$ in each frame. We thereafter compare the scores with ground-truth labels using AUC and AP, where higher values indicate better performance across all possible classification thresholds. In Fig. 3, we demonstrated the expansive trees after running Algorithm 2. The distribution of the energy cost field with respect to the capture set $\mathcal{Z}_{\text{cap}}$ and the task success set $\mathcal{Z}_{\text{suc}}$ directly indicates the corresponding quality score values and energy margins. To demonstrate the capability of predicting task success using the capture score of success $\Omega_{\text{suc}}(z_k)$, we designed a trajectory-level score $\Omega_{\text{suc}}(z, \bar{k})$ which is the weighted average of $\Omega_{\text{suc}}(z_k)$ of only the last $\bar{k}$ states with an increasing weight from $k = 1$ to $k = \bar{k}$. We thereby have, for each scenario, a dataset comprising 500 state-level data points $(\Omega_{\text{esc}}(z_k), \Omega_{\text{cap}}(z_k))$ and 50 trajectory-level data points $\Omega_{\text{suc}}(z, \bar{k})$ with corresponding labels.

### D. Quantitative Analysis

*1) Overall Performance:* Upon examination of the dataset for each example in simulation (Fig. 2), AUC and AP were computed for the robustness predictions ($\Omega_{\text{cap}}, \Omega_{\text{esc}}$ and $\Omega_{\text{force}}$, Table I left) and the success predictions ($\Omega_{\text{suc}}$, Table I rightmost column). The results indicate a high level of predictive capability of our metrics. The AUC/AP values for the robustness predictions using our methods ($\Omega_{\text{cap}}, \Omega_{\text{esc}}$) consistently exceed 0.9
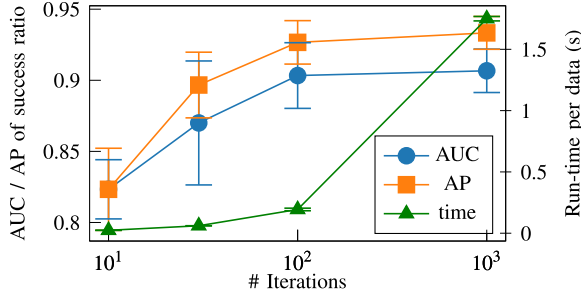
Fig. 4. Time efficiency and performance (AUC/AP values for $\Omega_{\text{suc}}(\boldsymbol{z}, \bar{k})$) of Algorithm 2 in a planar pushing task.



Fig. 6. Real-world experiments settings. (i) The planar pushing task using an Interbotix WidowX-200 robot arm with a top-view camera. (ii) 5 3D-printed objects and 2 end-effectors. (iii) Replication in the simulation of the scene in (i) for computing the energy margins. (iv) and (v) Screenshots of a failed/successful trajectory from the camera view.
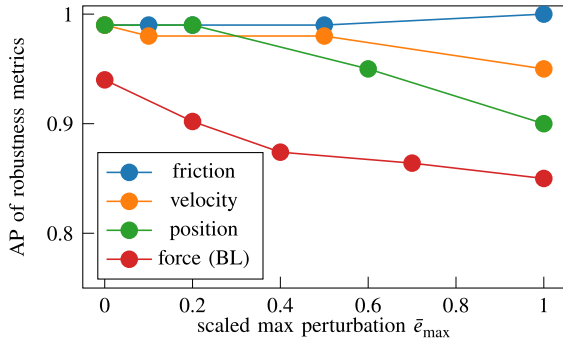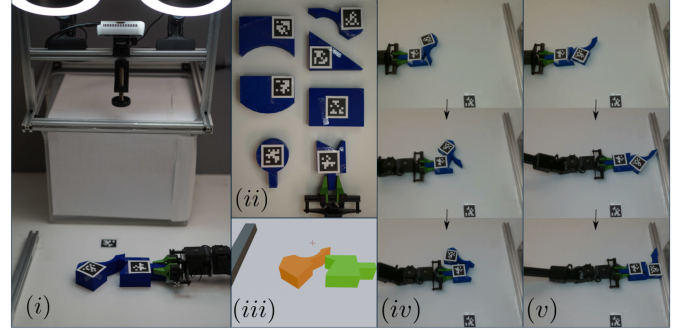


Fig. 5. Robustness evaluation of Algorithm 2 under various modeling errors: This plot shows the algorithm performance against different estimation errors of various types, including friction coefficient ($\mu$), velocity ($\dot{\boldsymbol{x}}, \dot{\boldsymbol{y}}$), relative position ($\boldsymbol{r}_x, \boldsymbol{r}_y$), and contact forces ($\lambda_{\parallel}, \lambda_{\perp}$). The graph demonstrates how AP values for the capture score $\Omega_{\text{cap}}$ (blue, orange and green) and the force-based baseline (BL) score $\Omega_{\text{force}}$ (red) vary given increasing maximal error thresholds ($\bar{e}_{\max}$).

across all examples, demonstrating strong discriminative power over the baseline $\Omega_{\text{force}}$. Here, $\Omega_{\text{esc}}$ is computed by employing both RRT and EST as subroutine algorithms $A$ for Algorithm 1. The values for the success predictions given $\bar{k} = 5$ indicate reliable predictions of successful trajectories using the score $\Omega_{\text{suc}}$ we proposed.

*2) Algorithm Efficiency:* We examined the time efficiency of Algorithm 2 with respect to the number of iterations and the task success predictions (AUC/AP values for $\Omega_{\text{suc}}(\boldsymbol{z}, \bar{k})$) as an ablation study. Fig. 4 illustrates that it takes about 1.75 sec to finish 1000 iterations in the simulated planar pushing task (Fig. 2-i). The AUC/AP values increase given more iterations of the algorithm, which is expected because of more nodes on the tree and better state space coverage. AUC/AP values reach 0.9 given only 100 iterations and about 0.2 sec of runtime, which confirms the time efficiency of our algorithm.

*3) Robustness against Different Model Parameter Errors:* A study was conducted (in Fig. 5) on the robustness of our algorithms given different model parameter errors, such as the positions $\boldsymbol{r}_x, \boldsymbol{r}_y$ and the velocities $\dot{\boldsymbol{x}}, \dot{\boldsymbol{y}}$ of the object and the end-effector, the contact forces $\lambda_{\parallel}, \lambda_{\perp}$, the friction coefficient $\mu$, etc. For instance, we simulated perturbed friction coefficient $\hat{\mu}$ by adding a perturbation $\mu_e$ randomly, uniformly sampled from the interval $[0, e_{\max}]$ on the true values $\mu$, where $e_{\max}$ is the maximal perturbation, i.e. $\hat{\mu} = \mu_e + \mu$.

For the simulated planar pushing task, we randomized the perturbed values $\hat{\mu}$ given increasing thresholds $e_{\max} \in$

$\{0, e_{\max}^1, \dots, e_{\max}^s\}$ for each trajectory $\boldsymbol{z}$. We then ran Algorithm 2 and obtained the capture score $\Omega_{\text{cap}}(\boldsymbol{z}_k)$ and the AP values with respect to scaled maximal thresholds $\bar{e}_{\max} \in \{0, e_{\max}^1/e_{\max}^s, \dots, 1\}$ given the ground-truth labels (Fig. 5). We generated the perturbed positions $\hat{\boldsymbol{r}}_x, \hat{\boldsymbol{r}}_y$ and the perturbed velocities $\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}$ for each state-level data point $\boldsymbol{z}_k$ and the corresponding AP curves of velocity and position. Similarly, the perturbed contact forces $\hat{\lambda}_{\parallel}, \hat{\lambda}_{\perp}$ are generated for the force-based score $\Omega_{\text{force}}(\boldsymbol{z}_k)$ with the AP curve of force. Since different types of error have different units, we scaled their maximal thresholds $e_{\max}^s$ to $\bar{e}_{\max}$ so they can be shown on one figure. While higher modeling errors do affect the prediction capability, our methods (the friction, velocity, and position curves) demonstrate better robustness against model parameter errors than the baseline (the force curve). This is because our method inherently captures many hard-to-model effects through rollout and energy analysis.

*E. Real-World Experiment*

*1) Setup and Data Collection:* We conducted real-world experiments using an Interbotix WidowX-200 robot arm (Fig. 6-i) for the planar pushing task of various geometric shapes of 3D-printed objects (five shapes) and end-effectors (two shapes) (Fig. 6-ii). The arm has a position error between 5-8 mm, providing a testbed to evaluate our scores under system uncertainty as in Section VI-D-3. We used a joystick to manually control the end-effector positions to perform the planar pushing task in the horizontal desktop plane. Visual markers were attached to the object and the end-effector to record their poses over time using a Realsense D415 camera.

We considered ten combinations of objects and end-effectors. For each combination, we collected 12 trajectories, with 100 recorded states for each trajectory. The velocities were computed from the poses and the frequency of the recording (33 Hz). A human observer provided success labels of the task objective. Failed (Fig. 6-iv) and successful (Fig. 6-v) trajectories are illustrated. Given the recorded states and system kinematic and dynamic parameters (mass, moment of inertia, friction coefficients, geometric information, etc.), the trajectories were replicated in simulation (Fig. 6-iii). We use the simulation to compute energy margins offline.

TABLE II
REAL-WORLD EXPERIMENTS RESULT (AUC/AP)

| Object | End-effector | Robust $\Omega_{\text{cap}}$ | Success $\Omega_{\text{suc}}$ |
|---|---|---|---|
| Rectangle | Jaw | 0.85 / 0.89 | 0.94 / 0.95 |
| | Round | 0.89 / 0.98 | 1.00 / 1.00 |
| Triangle | Jaw | 0.94 / 0.99 | 0.69 / 0.62 |
| | Round | 0.93 / 0.99 | 0.94 / 0.93 |
| Convex | Jaw | 0.91 / 0.92 | 1.00 / 1.00 |
| | Round | 0.84 / 0.81 | 0.91 / 0.88 |
| Concave | Jaw | 0.93 / 0.96 | 0.67 / 0.78 |
| | Round | 0.97 / 0.99 | 0.94 / 0.93 |
| Irregular | Jaw | 0.95 / 0.99 | 0.47 / 0.36 |
| | Round | 0.94 / 0.99 | 0.93 / 0.87 |

*2) Geometry Effect Evaluation:* Statistical analysis results of the AUC/AP values for the scores $\Omega_{\text{cap}}(z_k)$ and $\Omega_{\text{suc}}(z, \bar{k})$, are summarized in Table II following the same procedure as in the simulation (Section VI-C) given $\bar{k} = 60$, $K = 100$. The effort of escape $\Omega_{\text{esc}}$ is not discussed here due to its inferior performance compared to the capture score in Table I. The results presented mostly exceed 0.85, which supports the discrimination and predictivity power of our scores and is in line with the results in the simulation. The performance of the capture score of success $\Omega_{\text{suc}}(z)$ of 3 of the 10 combinations of geometries (the jaw end-effector with the triangle, concave and irregular objects) is below the average though. A reason for the failure could be that the sim-to-real gap is especially sensitive to these cases with complicated geometric shapes.

## VII. CONCLUSION

In this paper, we propose to quantify manipulation robustness through energy margins to failure and compute them using kinodynamic motion planning algorithms. We believe this approach is a step towards robustness characterization of general dexterous manipulation. Our approach currently has certain limitations, such as reliance on dynamic rollouts in simulation, which requires system modeling and causes computational costs. Additionally, our analysis is confined to rigid objects and utilizes an empirically defined capture set. Looking ahead, we aim to improve our method by overcoming these limitations and applying our robustness characterization in motion planning for general manipulation.

## ACKNOWLEDGMENT

## REFERENCES

[1] A. Bhatt, A. Sieler, S. Puhlmann, and O. Brock, "Surprisingly robust in-hand manipulation: An empirical study," *Robot.: Sci. Syst.*, 2021, doi: 10.15607/rss.2021.xvii.089.

[2] H. Tsukamoto, S.-J. Chung, and J.-J. E. Slotine, "Contraction theory for nonlinear stability analysis and learning-based control: A tutorial overview," *Annu. Rev. Control*, vol. 52, pp. 135–169, 2021.

[3] R. Tedrake et al., "LQR-trees: Feedback motion planning on sparse randomized trees," *Robotics: Sci. Syst.*, vol. 2009, pp. 17–24, 2009.

[4] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *Proc. IEEE 18th Eur. Control Conf.*, 2019, pp. 3420–3431.

[5] M. A. Roa and R. Suárez, "Grasp quality measures: Review and performance," *Auton. Robots*, vol. 38, pp. 65–88, 2015.

[6] Y. Hou and M. T. Mason, "Criteria for maintaining desired contacts for quasi-static systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2019, pp. 6555–6561.

[7] S. Makita and W. Wan, "A survey of robotic caging and its applications," *Adv. Robot.*, vol. 31, no. 19-20, pp. 1071–1085, 2017 .

[8] Y. Dong and F. T. Pokorny, "Quasi-static soft fixture analysis of rigid and deformable objects," in *Proc. Int. Conf. Robot. Automat.*, 2024.

[9] W. Kuperberg, "Problems on polytopes and convex sets," in *Proc. DIMACS Workshop Polytopes*, 1990, pp. 584–589.

[10] E. Rimon and A. Blake, "Caging 2D bodies by 1-parameter two-fingered gripping systems," in *Proc. IEEE Int. Conf. Robot. Automat.*, 1996, pp. 1458–1464.

[11] E. Rimon and A. Blake, "Caging planar bodies by one-parameter two-fingered gripping systems," *IEEE Int. J. Robot. Res.*, vol. 18, no. 3, pp. 299–318, 1999.

[12] Y. Maeda and T. Asamura, "Sensorless in-hand caging manipulation," in *Proc. 14th Int. Conf. Intell. Auton. Syst.*, 2017, pp. 255–267.

[13] A. Rodriguez, M. T. Mason, and S. Ferry, "From caging to grasping," *IEEE Int. J. Robot. Res.*, vol. 31, no. 7, pp. 886–900, 2012.

[14] J. Mahler, F. T. Pokorny, Z. McCarthy, A. F. van der Stappen, and K. Goldberg, "Energy-bounded caging: Formal definition and 2-D energy lower bound algorithm based on weighted alpha shapes," *IEEE Robot. Automat. Lett.*, vol. 1, no. 1, pp. 508–515, Jan. 2016.

[15] J. Mahler, F. T. Pokorny, S. Niyaz, and K. Goldberg, "Synthesis of energy-bounded planar caging grasps using persistent homology," *IEEE Trans. Automat. Sci. Eng.*, vol. 15, no. 3, pp. 908–918, Jul. 2018.

[16] A. Thakur, S. Chowdhury, P. Švec, C. Wang, W. Losert, and S. K. Gupta, "Indirect pushing based automated micromanipulation of biological cells using optical tweezers," *Int. J. Robot. Res.*, vol. 33, no. 8, pp. 1098–1111, 2014.

[17] Q. M. Ta and C. C. Cheah, "Caging and transportation of biological cells using optical tweezers," in *Proc. IEEE Int. Conf. Real-Time Comput. Robot.*, 2020, pp. 363–368.

[18] N. S. Pollard, "Synthesizing Grasps From Generalized Prototypes," in *Proc. Int. Conf. Robot. Automat*, 1996, vol. 3, pp. 2124–2130.

[19] A. T. Miller and P. K. Allen, "Examples of 3D grasp quality computations," in *Proc. IEEE Int. Conf. Robot. Automat.*, 1999, pp. 1240–1246.

[20] C. Ferrari et al., "Planning optimal grasps," in *Proc. Int. Conf. Robot. Automat.*, 1992, vol. 3, no. 4, pp. 2290–2295.

[21] Y. Lin and Y. Sun, "Task-based grasp quality measures for grasp synthesis," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2015, pp. 485–490.

[22] J. Xu, M. Danielczuk, J. Ichnowski, J. Mahler, E. Steinbach, and K. Goldberg, "Minimal work: A grasp quality metric for deformable hollow objects," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 1546–1552.

[23] A. Saxena, L. L. Wong, and A. Y. Ng, "Learning grasp strategies with partial shape information," in *Proc. AAAI Conf. Artif. Intell.*, 2008, vol. 3, no. 2, pp. 1491–1494.

[24] R. Krug, Y. Bekiroglu, D. Kragic, and M. A. Roa, "Evaluating the quality of non-prehensile balancing grasps," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 4215–4220.

[25] T. Makapunyo, T. Phoka, P. Pipattanasomporn, N. Niparnan, and A. Sudsang, "Measurement framework of partial cage quality based on probabilistic motion planning," in *Proc. IEEE Int. Conf. Robot. Automat*, 2013, pp. 1574–1579.

[26] N. R. Ahmed, E. M. Sample, and M. Campbell, "Bayesian multicategorical soft data fusion for human–robot collaboration," *IEEE Trans. Robot.*, vol. 29, no. 1, pp. 189–206, Feb. 2013.

[27] K. Hauser and Y. Zhou, ";Asymptotically optimal planning by feasible kinodynamic planning in a state–cost space," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1431–1443, Dec. 2016.

[28] D. Hsu, J.-C. Latombe, and R. Motwani, "Path Planning in Expansive Configuration Spaces," in *Proc. IEEE Int. Conf. Robot. Automat*, 1997, vol. 3, pp. 2719–2726.

[29] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," *Res. Rep.*, vol. 9811, 1998.

[30] J. Liu, F. Feng, Y. C. Nakamura, and N. S. Pollard, "A taxonomy of everyday grasps in action," in *Proc. IEEE Int. Conf. Humanoid Robot.*, 2014, pp. 573–580.

[31] F. Ruggiero, V. Lippiello, and B. Siciliano, "Nonprehensile dynamic manipulation: A survey," *IEEE Robot. Automat. Lett.*, vol. 3, no. 3, pp. 1711–1718, Jul. 2018.