Caging in Motion: Characterizing Robustness in Manipulation through Energy Margin and Dynamic Caging Analysis

Yifei Dong[†], Xianyi Cheng[‡] and Florian T. Pokorny[†]

Abstract— To develop robust manipulation policies, quantifying robustness is essential. Evaluating robustness in general dexterous manipulation, nonetheless, poses significant challenges due to complex hybrid dynamics, combinatorial explosion of possible contact interactions, global geometry, etc. This paper introduces "caging in motion", an approach for analyzing 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 a global analysis of 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 the simulation and real-world experiments of multiple dynamic manipulation scenarios, highlighting its potential to predict manipulation success and robustness.

I. INTRODUCTION

Human 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 current 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 the *caging in motion* approach to

†: The authors are with the division of Robotics, Perception and Learning, KTH Royal Institute of Technology, 10044 Stockholm, Sweden, {`yifeid`, `fpokorny`}`@kth.se`. ‡: The author is with Carnegie Mellon University, Pittsburgh, PA, 15213, USA. `xianyic@andrew.cmu.edu`

characterize robustness in manipulation via energy margins. 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 has laid the groundwork for robotic non-prehensile manipulation, focusing on preventing an object from escaping using minimal constraints. This foundational concept, as introduced by Kuperberg [9] and further developed by Rimon et al. [10], [11], has been pivotal in exploring how robotic grasps can effectively cage objects, without necessitating force closure or form closure. Rodriguez et al. [12] demonstrate utilizing caging to achieve immobilizing grasps. Energy-bounded caging [13] relaxes the complete caging in the presence of external forces such as gravity. Planar energy-bounded cages [14] are synthesized by identifying configurations that maximize the energy required for the object to escape. [15] proposes whole-hand manipulation via caging by considering compliant underactuated hands as obstacles and introducing an object energy field. Our previous work on soft fixture [8] presents practical sampling-based methods to search for upper-bound estimates of escape energy applicable in higher-dimensional configuration spaces, such as simplified deformable objects. While classical caging has offered foundational insights into object restraint, the "cage in motion" approach we propose here represents a new perspective. It extends traditional caging into dynamic

settings, bridging the gap between static caging theories and the dynamic realities of robotic manipulation. On the other hand, our work first brings the caging analysis to characterize manipulation robustness in motion.

B. Robustness and Quality Metrics

Parallel to the evolution of caging, grasp quality measures that quantify the goodness of a grasp have been a significant line of research. A comprehensive review by Roa et al. [5] categorizes quality measures based on contact points and hand configurations, providing a systematic framework for evaluating grasp quality. Grasp Wrench Space (GWS) [16] has been a useful tool to determine the quality of a grasp. Several metrics derived from the GWS can quantify the "goodness" of a grasp, such as the volume of the GWS [17], the largest perturbation wrench the grasp can resist in any direction [18], etc. More recently, Lin et al. [19] 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. [20] 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 [21]. This line of research thoroughly investigates wrench-based quality metrics for prehensile grasping, especially force closure grasps. For non-prehensile manipulation, a framework of quasi-static wrench-based analysis is presented for balancing grasps in [22]. Makapunyo et al. [23] employ sampling-based methods to determine a quality metric for partial caging. In contrast to the previous literature, our work aims to provide a new paradigm for robustness characterization that is independent of wrench information. We focus instead on analyzing energy margin and dynamic caging within kinodynamic settings. Thereby, we aim to encapsulate a broader spectrum of manipulation scenarios, offering a more global and adaptable framework for assessing manipulation robustness and success in dynamically evolving environments.

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 $\mathrm{SE}(2) \times \mathbb{R}^{n_r}$, respectively. 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 (a unit quaternion) $\boldsymbol{q}_x$. 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
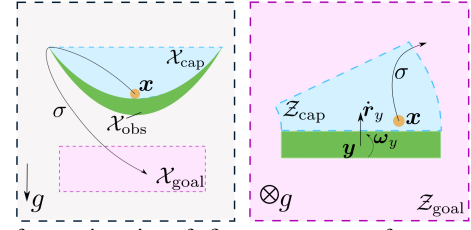

Fig. 1: Left: quasi-static soft fixture concepts of an escape path σ that leads an object (orange) at $\boldsymbol{x}$ inside a bowl (green, regarded as a static obstacle $\mathcal{X}_{\mathrm{obs}}$) to a goal region $\mathcal{X}_{\mathrm{goal}}$. A capture set (blue, $\mathcal{X}_{\mathrm{cap}}$) is constructed by a bounded potential energy sublevel set with a minimum energy threshold value. Right: "caging in motion" — an end-effector (with pose $\boldsymbol{y}$, linear and angular velocity $\dot{\boldsymbol{r}}_y$, $\boldsymbol{\omega}_y$) is employed to guide an object (orange) towards a target area (top left corner) on a table. A capture set denotes the region anticipated to be swept by the end-effector in the forthcoming time interval.

$\dot{\boldsymbol{y}} \in \dot{\mathcal{Y}}$, 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, respectively.

B. Cage and Soft Fixture

Here we revisit the fundamental 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 the configuration of an object $\boldsymbol{x}_{\mathrm{init}} \in \mathcal{X}_{\mathrm{free}}$ 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 [23]. Partial caging further extends to energy-bounded caging [13] 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:

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

where

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

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, left) sufficiently far away from the obstacles with relatively lower energy values and searching for geometrically feasible escape paths with low energy cost leading to it [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 wiping a table (Fig. 1, right), assuming the tabletop with a rough surface is infinitely large. The escape energy $\mathcal{E}(x_{\text{init}})$ is always infinite in this scenario because the object overcomes friction along any escape path to arbitrarily far away. To utilize escape energy in this context, we define a *capture set* $\mathcal{Z}_{\text{cap}}(z_{\text{init}}) \subset \mathcal{Z}$ instead, considering the object as dynamically controllable by the end-effector for achieving specific tasks. The effort of the object to escape from $\mathcal{Z}_{\text{cap}}(z_{\text{init}})$ thus indicates manipulation robustness, implying an energy margin from failure.

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

D. Problem Statement

In this work, we tackle the following problem: Given a specific manipulation task objective and the state z_{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 quantify energy margins to failure for characterizing robustness in dexterous manipulation (Section IV). 2. Compute the energy margin metrics using kinodynamic motion planning and evaluate their performance (Section V and VI).

IV. ENERGY MARGIN FOR CAGING IN MOTION

This section quantifies manipulation robustness with two methods based on energy margin: the effort of escape and capture score. The former focuses on the ease of failure, and the latter on the probability of staying safe.

A. Effort of Escape

We first measure the energy margin to failure by the minimum effort to escape out of 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 $\mathcal{Z}_{\text{cap}}(z_{\text{init}})$:

$$\Omega_{\text{esc}}(z_{\text{init}}) = \min_{\sigma \in \Sigma(z_{\text{init}}, \mathcal{Z}_{\text{goal}})} \int_0^1 |\dot{W}_{\text{ext}}(\sigma(t))| dt, \quad (3)$$

where

$$\dot{W}_{\text{ext}}(\sigma(t)) = \frac{dE(\sigma(t))}{dt} - \dot{W}_{\text{fri}}(\sigma(t)). \quad (4)$$

$\frac{dE(\sigma(t))}{dt}$ 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 and velocity of the system. We denote $\Sigma(z_{\text{init}}, \mathcal{Z}_{\text{goal}})$ as the set of all escape paths starting from z_{init} terminating in $\mathcal{Z}_{\text{goal}}$, where $\mathcal{Z}_{\text{goal}} = \mathcal{Z}_{\text{cap}}^c(z_{\text{init}})$ (the complement set of the capture set).

Eq. (4) follows the first law of thermodynamics and the thermal equilibrium assumption. 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 $\Sigma(z_{\text{init}}, \mathcal{Z}_{\text{goal}})$ 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 $\Omega_{\text{esc}}(z_{\text{init}})$.

B. Capture Score

While the effort of escape assesses the minimal effort required to exit the capture set, the capture score estimates the likelihood of remaining within the capture set.

We first consider an energy cost field and its correlating probabilistic distribution of the system. Specifically, a sequence of random wrenches applied on the object at z_{init} lead it to z^0 with a cost of total effort c^0. By repeating the process M times from z_{init}, the system terminates in a list of states $\{z^0, ..., z^M\}$ with corresponding costs $\{c^0, ..., c^M\}$. Data pairs in $\{(z^0, c^0), ..., (z^M, c^M)\}$ constitute an energy cost field that demonstrates the state space reachability in terms of energy cost from z_{init}. We can thus build a probability mass function $L : \mathcal{Z} \to \mathbb{R}^+$ from the energy cost field, approximating the likelihood of reaching a state $z^m \in \mathcal{Z}$ with a softmax function,

$$L(z^m) = \frac{e^{-\lambda(c^m - c_{\min})}}{\sum_{m=0}^M e^{-\lambda(c^m - c_{\min})}}, \quad (5)$$

where $c_{\min} = \min_{0 \leq m \leq M} c^m$ and λ is a hyper-parameter. 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 $\mathcal{Z}_{\text{cap}}(z_{\text{init}})$,

$$\Omega_{\text{cap}}(z_{\text{init}}) = \sum_{m=0}^M \delta(z^m \in \mathcal{Z}_{\text{cap}}(z_{\text{init}})) \cdot L(z^m), \quad (6)$$

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_{init}, indicating the set of states symbolizing the object accomplishes the manipulation objective. A capture

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** σ_0 *does not exists* **then**
4   P has no solution
5   **Return** ∞ ▷ Infinite cost to escape
6 $c_0 \leftarrow C(\sigma_0)$ ▷ EQ. (9). Initial cost
7 **for** $i = 1, 2, \ldots, n$ **do**
8      $\sigma_i \leftarrow A(P_{c_{i-1}})$ ▷ Find a path of lower cost
9      $c_i \leftarrow C(\sigma_i)$ ▷ EQ. (9). Update cost upper bound
10 $\Omega_{\mathrm{esc}}(\boldsymbol{z}_{\mathrm{init}}) \leftarrow c_n$ ▷ EQ. (3). Final cost
11 **Return** $\Omega_{\mathrm{esc}}(\boldsymbol{z}_{\mathrm{init}})$

---

score of success is thus given by

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

which is essentially a predictor for fulfilling a task-specific objective from the state $\boldsymbol{z}_{\mathrm{init}}$. In practice, we employ kinodynamic motion planners rather than repetitive Monte Carlo rollouts from $\boldsymbol{z}_{\mathrm{init}}$. 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 energy margins for characterizing robustness as defined in Section IV. We calculate the effort of escape $\Omega_{\mathrm{esc}}(\boldsymbol{z}_{\mathrm{init}})$ by an iterative tree search algorithm with an optimality guarantee (Section V-B). The capture scores $\Omega_{\mathrm{cap}}(\boldsymbol{z}_{\mathrm{init}})$ and $\Omega_{\mathrm{suc}}(\boldsymbol{z}_{\mathrm{init}})$ are obtained 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

$$\boldsymbol{z}(k) = \begin{bmatrix} \boldsymbol{x}^\top(k) & \dot{\boldsymbol{x}}^\top(k) & \boldsymbol{y}^\top(k) & \dot{\boldsymbol{y}}^\top(k) \end{bmatrix}^\top. \qquad (8)$$

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

B. Compute Effort of Escape

We frame finding the minimum effort to escape from the capture set $\mathcal{Z}_{\mathrm{cap}}(\boldsymbol{z}_{\mathrm{init}})$ as an optimal kinodynamic motion planning problem P of which the goal set is the complement set of the capture set ($\mathcal{Z}_{\mathrm{goal}} = \mathcal{Z}_{\mathrm{cap}}^c(\boldsymbol{z}_{\mathrm{init}})$). We employ the AO-$x$ meta algorithms [24], which are asymptotically optimal motion planners that decompose P into feasible motion planning problems in the state-cost space. They iter-

atively find an upper bound approaching the minimum effort of escape by running as subroutines feasible kinodynamic planner x, such as Expansive Space Tree (EST) [25] or Rapidly-exploring Random Tree (RRT) [26] algorithms.

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

$$C(\sigma) = \int_0^1 Q(\sigma(t), \boldsymbol{u}(t))dt = \int_0^1 |\dot{W}_{\mathrm{ext}}(\sigma(t))|dt. \qquad (9)$$

Here, Q denotes the incremental cost (terminal cost is 0) following Eq. (3). $\mathcal{U}_b$ is the set of control constraints and $\boldsymbol{u} \in \mathcal{U}_b \subset \mathcal{U}$. $\mathcal{Z}_b$ refers to the set of kinematic constraints and $\boldsymbol{z} \in \mathcal{Z}_b \subset \mathcal{Z}_{\mathrm{free}}$. The dynamics is subjected to $\boldsymbol{z}' = G(\boldsymbol{z}, \boldsymbol{u})$. The control input $\boldsymbol{u} = \begin{bmatrix} \boldsymbol{f}^\top & \boldsymbol{\tau}^\top \end{bmatrix}^\top$ refers to a wrench (force $\boldsymbol{f}$ applied at its CoM and torque $\boldsymbol{\tau}$) applied on the object, mimicking an external perturbation.

Given the optimal motion planning problem P, a feasible motion planning problem $P_{\bar{c}} = (\mathcal{Z}_{\mathrm{free}} \times \mathbb{R}^+, \mathcal{U}, (\boldsymbol{z}_{\mathrm{init}}, 0), \hat{\mathcal{Z}}_{\mathrm{goal}}^{\bar{c}}, \mathcal{Z}_b \times \mathbb{R}^+, \mathcal{U}_b, \hat{G})$ 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 $\boldsymbol{z}_{\mathrm{init}}$, i.e. cost-to-come. Here, $\hat{\mathcal{Z}}_{\mathrm{goal}}^{\bar{c}} = \{(\boldsymbol{z}, c) : \boldsymbol{z} \in \mathcal{Z}_{\mathrm{goal}}, c \in [0, \bar{c}]\}$ denotes the augmented goal set with the range of cost space upper-bounded by $\bar{c}$, and $\hat{G}$ refers to the augmented dynamics given by

$$\hat{\boldsymbol{z}}' = \begin{bmatrix} \boldsymbol{z}' \\ c' \end{bmatrix} = \begin{bmatrix} G(\boldsymbol{z}, \boldsymbol{u}) \\ Q(\boldsymbol{z}, \boldsymbol{u}) \end{bmatrix}. \qquad (10)$$

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). A feasible planner is called well-behaved if there is a nonzero chance that the planner does not produce the worst-possible path in finite time (if one exists).

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 σ_0 by growing a tree rooted at $\boldsymbol{z}_{\mathrm{init}}$ (Algo. 1, Line 2). The tree expands a new node $\boldsymbol{z}'$ by enforcing a randomized control $\boldsymbol{u}$ based on algorithm-specific strategies from an existing node $\boldsymbol{z}$, 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 $\boldsymbol{z}'$ abides by the augmented kinematic constraints $(\boldsymbol{z}', c(\boldsymbol{z}')) \in \mathcal{Z}_b \times \mathbb{R}^+$. If a path does not exist, it indicates the effort of escape is infinite $\Omega_{\mathrm{esc}}(\boldsymbol{z}_{\mathrm{init}}) = \infty$ (Line 3-5). Once we found a path σ_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 ∞ 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 σ_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

Algorithm 2: Compute Capture Scores

---

1 Initialize, P_∞, $\mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}})$, $\mathcal{Z}_{\text{suc}}$, M
2 ▷ Obtain energy cost field by growing a tree
3 $\{(\boldsymbol{z}^0, c^0), ..., (\boldsymbol{z}^M, c^M)\} \leftarrow \text{EST}(P_\infty, M)$
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 $\quad\lfloor\ L(\boldsymbol{z}^m) \leftarrow \frac{1}{S} e^{-\lambda(c^m - c_{\min})}$ ▷ EQ. (5).

9 ▷ Compute scores
10 $\Omega_{\text{cap}}(\boldsymbol{z}_{\text{init}}) \leftarrow \sum_{m=0}^{M} \delta(\boldsymbol{z}^m \in \mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}})) \cdot L(\boldsymbol{z}^m)$ ▷ EQ. (6).
11 $\Omega_{\text{suc}}(\boldsymbol{z}_{\text{init}}) \leftarrow \sum_{m=0}^{M} \delta(\boldsymbol{z}^m \in \mathcal{Z}_{\text{suc}}) \cdot L(\boldsymbol{z}^m)$ ▷ EQ. (7).
12 **Return** $\Omega_{\text{cap}}(\boldsymbol{z}_{\text{init}})$, $\Omega_{\text{suc}}(\boldsymbol{z}_{\text{init}})$

---

infinity. Therefore, we can approximately upper-bound the minimum effort of escape $\Omega_{\text{esc}}(\boldsymbol{z}_{\text{init}})$ in finite (n) loops.

In practice, we consider EST or RRT as the subroutine algorithms A. RRT attempts to randomly sample a state in the state space and select an extension from its nearest neighbor node already in the tree towards it. EST takes the strategy of prioritizing extensions in the area with less density of existing nodes in the tree.

C. Compute 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 $\boldsymbol{z}_{\text{init}}$ 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 Algo. 2, we consider the cost-augmented unbounded feasible motion planning problem P_∞ (Line 2-3) with the same settings as in Section V-B except an infinitely far away goal set $\mathcal{Z}_{\text{goal}} = \mathcal{Z}_\infty \subset \mathcal{Z}$ with unconstrained kinematics $\mathcal{Z}_b = \mathcal{Z}_{\text{free}}$. The algorithm terminates after growing M nodes in the tree. We thus obtain an energy cost field of M state-cost pairs, $\{(\boldsymbol{z}^0, c^0), ..., (\boldsymbol{z}^M, c^M)\}$. We thereafter approximate the probabilistic distribution (Line 4-8) and compute the scores $\Omega_{\text{cap}}(\boldsymbol{z}_{\text{init}})$ and $\Omega_{\text{suc}}(\boldsymbol{z}_{\text{init}})$ (Line 9-11) following the procedure in Section IV-B.

VI. EVALUATION

In this section, we aim to validate the practicality and effectiveness of our metrics and algorithms by conducting thorough simulation and real-world experiments. Our primary 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 conducted ablation studies to verify
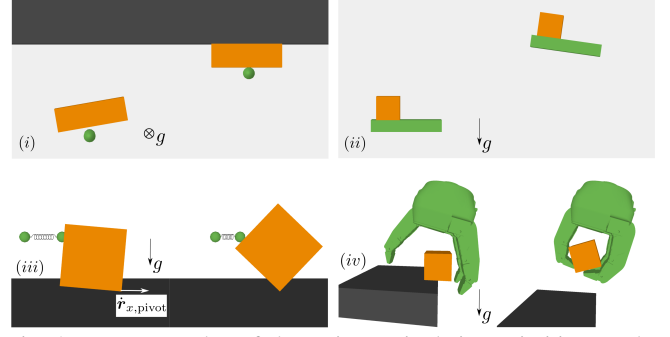

Fig. 2: Four examples of dynamic manipulation primitives under gravity — (i) planar pushing, (ii) balanced transport, (iii) toppling and (iv) grasping from a table, where the task objectives are all eventually achieved. Objects, end-effectors and obstacles are shown in orange, green and black, respectively.

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 [27], [28], we provide 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 $\boldsymbol{z} \in \mathcal{Z}_{\text{suc}}$ lies in the task success set $\mathcal{Z}_{\text{suc}} = \{\boldsymbol{z} \in \mathcal{Z} : \boldsymbol{r}_x \in \mathcal{R}_{\text{suc}}\}$, 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 $\mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}}) = \{\boldsymbol{z} \in \mathcal{Z} : \boldsymbol{r}_x \in \mathcal{R}_{\text{cap}}(\boldsymbol{z}_{\text{init}})\}$ includes state $\boldsymbol{z} \in \mathcal{Z}$ such that the CoM position of the object $\boldsymbol{r}_x$ lies inside a subset of the workspace $\mathcal{R}_{\text{cap}}(\boldsymbol{z}_{\text{init}}) \subset \mathbb{R}^2$. $\mathcal{R}_{\text{cap}}(\boldsymbol{z}_{\text{init}})$ indicates the circular sector region in the workspace that the end-effector will sweep over given the current instantaneous rotation center of the end-effector (Fig. 1, right).

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 $\mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}})$ and the task success set $\mathcal{Z}_{\text{suc}}$ above, we add constraints on the object CoM position $\boldsymbol{r}_x$ such that the capture condition (maintaining on the support surface, $\mathcal{R}_{\text{cap}}(\boldsymbol{z}_{\text{init}})$) 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 tabletop using a spring-like fingertip manipulator (Fig. 2-iii). The task is accomplished when the box rotates by $\pi/2$ rad pivoting on one of the corners in contact with the table, i.e. $\mathcal{Z}_{\text{suc}} = \{\boldsymbol{z} \in \mathcal{Z} : p(\boldsymbol{q}_x) = \pi/2\}$, where $p(\boldsymbol{q}_x)$ 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 $\mathcal{Z}_{\text{cap}}(\boldsymbol{z}_{\text{init}}) = \{\boldsymbol{z} \in \mathcal{Z} : \|\dot{\boldsymbol{r}}_{x,\text{pivot}}\| \le 0.1\}$, 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
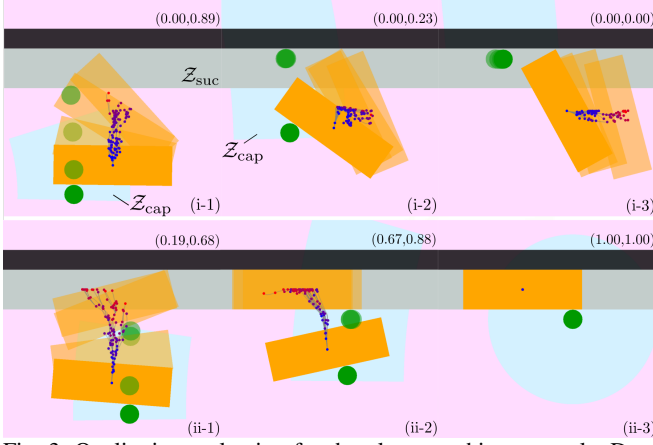
Fig. 3: Qualitative evaluation for the planar pushing example. Data in the subfigures indicate $(\Omega_{\text{suc}}, \Omega_{\text{cap}})$. (i) and (ii) refer to a failed and a successful trajectory towards a task objective of fixturing a rectangular box (yellow) on the wall (black), respectively. Three screenshots are taken along each trajectory, for which we run Algo. 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).

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 $\mathcal{Z}_{\text{suc}}$.

B. Baseline

We compare our approach with a manipulation quality score from [6], which is designed to measure the robustness of contact modes (sticking, sliding, disengaged, etc) against disturbance forces (e.g. avoid slipping for sticking contacts). This characterizes manipulation robustness against failure modes similar to the goal in this paper. However, our energy margin approach quantifies manipulation robustness more globally and captures contact mode transitions and loss of contact to a certain degree. To make a fair comparison, we employ a hybrid force-based score Ω_{force} with a weighted sum of the engaging score Ω_{engage}, the sticking score Ω_{stick} and a kinematic heuristic Ω_{dist} - the minimum distance between any two points on the object and the end-effector. Specifically, Ω_{engage} uses the magnitude of the contact normal force $\lambda_{\perp}$ to evaluate how much normal disturbance forces a contact can withstand, $\Omega_{\text{engage}} = \lambda_{\perp}$. Ω_{stick} refers to the minimal amount of disturbance force required to transition 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):

$$\Omega_{\text{stick}} = (\mu\lambda_{\perp} - |\lambda_{/\!/}|)\cos(\arctan\mu), \qquad (11)$$

where μ 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) 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 $z_k, k \in \{1, ..., K\}$ 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 μ, the contact normal force $\lambda_{\perp}$ and lateral force $\lambda_{/\!/}$ in each frame. A trajectory is labeled 1 (successful, e.g. Fig. 3 bottom) if the task objective is accomplished in the last recorded frame, i.e. $z_K \in \mathcal{Z}_{\text{suc}}$, otherwise 0 (Fig. 3 top); 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 $k' \in \{k, ..., k + \hat{k}\}$.

2) 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 z_k in each trajectory of the examples. 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 Algo. 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 first $\bar{k}$ 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 both the robustness predictions (Ω_{cap}, Ω_{esc} and Ω_{force}, Table I left) and the success predictions (Ω_{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 (Ω_{cap}, Ω_{esc}) consistently exceed 0.9 across all examples, demonstrating strong discriminative power over the baseline Ω_{force}. Here, Ω_{esc} is computed by employing both RRT and EST as subroutine algorithms A for Algo. 1. The values for the success predictions given $\bar{k} = 5$ indicate that we can reliably predict successful trajectories with high precision using the

Task	Ω_{cap}	Ω_{esc} (**EST**)	Ω_{esc} (**RRT**)	Ω_{force} (**Baseline**)	Ω_{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

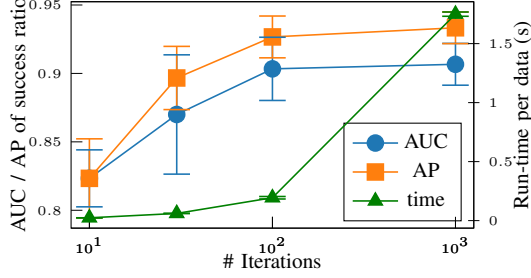TABLE I: Energy margins evaluation in simulation (AUC/AP).

Fig. 4: Time efficiency and performance (AUC/AP values for $\Omega_{\text{suc}}(\boldsymbol{z}, \bar{k})$) of Algo. 2 in a planar pushing task.
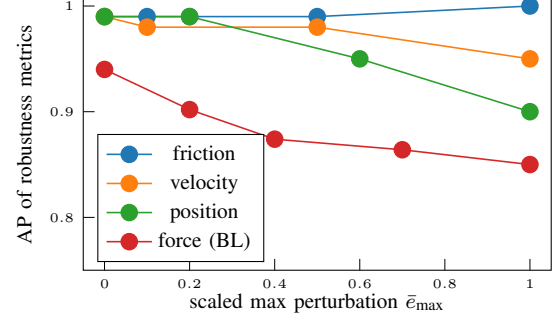

Fig. 5: Robustness evaluation of Algo. 2 under various modeling errors: This plot illustrates the algorithm performance against different levels of estimation errors of various types, including friction coefficient (μ), velocity ($\dot{\boldsymbol{x}}, \dot{\boldsymbol{y}}$), relative position ($\boldsymbol{r}_x,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

score Ω_{suc} we proposed.

2) Algorithm Efficiency: We examined the time efficiency of Algo. 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 the caging-in-motion algorithms given different unknown 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 $\lambda_{/\!/}, \lambda_{\perp}$, the friction coefficient μ, etc. For instance, we simulated perturbed friction coefficient $\hat{\mu}$ by adding a perturbation μ_e randomly, uniformly sampled from the interval $[0, e_{\text{max}}]$ on the true values μ, 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 $e_{\text{max}} \in \{0, e^1_{\text{max}}, ..., e^s_{\text{max}}\}$ for each trajectory $\boldsymbol{z}$. We then ran Algo. 2 and obtained the capture score $\Omega_{\text{cap}}(\boldsymbol{z}_k)$ and the AP values with respect to scaled maximal thresholds $\bar{e}_{\text{max}} \in \{0, e^1_{\text{max}}/e^s_{\text{max}}, ..., 1\}$ given the ground-truth labels (Fig. 5). Similarly, we obtained the (simulated) perturbed positions $\hat{\boldsymbol{r}}_x, \hat{\boldsymbol{r}}_y$ and the perturbed velocities $\hat{\dot{\boldsymbol{x}}}, \hat{\dot{\boldsymbol{y}}}$ for each state-level data point $\boldsymbol{z}_k$ and the corresponding AP curves of velocity and position. The perturbed contact forces $\hat{\lambda}_{/\!/}, \hat{\lambda}_{\perp}$ are generated for the force-based score $\Omega_{\text{force}}(\boldsymbol{z}_k)$ with the AP curve of force. Note that the maximal thresholds e^s_{max} are of varying values for different types of error but exhibit comparable intensity of disturbance empirically so that the curves under the scaled thresholds $\bar{e}_{\text{max}}$ are comparable. It can be seen that higher noises affect the capability of positivity identification of our algorithm. However, our methods (the friction, velocity, and position curves) demonstrate better robustness against model parameter errors than the baseline (the force curve).

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 have ten combinations of objects and end-effectors. For each combination, we collected 12 trajectories, with 100 recorded states for each trajectory. The velocities were solved given the poses and the frequency of the recording (33 Hz). A human observer provided success labels of the task objective. Two failed (Fig. 6-iv) and successful (Fig. 6-v) trajectories are also illustrated as examples. Given the recorded states and system kinematic and dynamic parameters (mass, moment of inertia, friction coefficients, geometric information, etc.), the trajectories were replicated in the simulated environment (Fig. 6-iii). We utilize the simulation for computing energy margins offline.

2) Geometry Effect Evaluation: Statistical analysis results of the AUC/AP values for the scores $\Omega_{\text{cap}}(\boldsymbol{z}_k)$ and $\Omega_{\text{suc}}(\boldsymbol{z}, \bar{k})$, 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 Ω_{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 $\Omega_{\text{suc}}(\boldsymbol{z})$ of 3
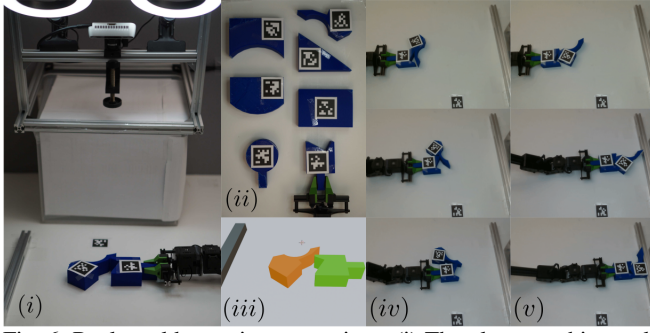
Fig. 6: Real-world experiments settings. (i) The planar pushing task using an Interbotix WidowX-200 robot arm with a top-view camera. (ii) 5 3D-printed objects and 2 end-effectors. (iii) Replication of the scene in (i) in simulation for computing the energy margins. (iv,v) Screenshots of a failed/successful trajectory from the camera view.

of the 10 combinations of geometries (the jaw end-effector with the triangle, concave and irregular objects) is below the average though. It could be a reason for the failure that the relatively complicated geometric shapes and the shape complementarity in these three cases added uncertainty to the planar pushing. It is thereby barely reasonable to predict whether the task objective will be achieved given a partial observation of such a trajectory.

Object	End-effector	Robust Ω_{cap}	Success Ω_{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

TABLE II: Real-world experiments result (AUC/AP).

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 the simulation, which requires accurate system modeling and causes computational costs. Additionally, our analysis is confined to single 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 robust motion planning for general dexterous manipulations.

REFERENCES

[1] A. Bhatt, A. Sieler, S. Puhlmann, and O. Brock, "Surprisingly robust in-hand manipulation: An empirical study," *arXiv preprint arXiv:2201.11503*, 2022.

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

[3] R. Tedrake *et al.*, "Lqr-trees: Feedback motion planning on sparse randomized trees." in *Robotics: Science and Systems*, vol. 2009, 2009.

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

[5] M. A. Roa and R. Suárez, "Grasp quality measures: review and performance," *Autonomous 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. Int. Conf. Intell. Robot. Syst.* IEEE, 2019, pp. 6555–6561.

[7] S. Makita and W. Wan, "A survey of robotic caging and its applications," *Advanced Robotics*, 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," *Proc. Int. Conf. Robot. Automat.*, 2024.

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

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

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

[12] 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.

[13] 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, 2016.

[14] 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, 2018.

[15] R. R. Ma, W. G. Bircher, and A. M. Dollar, "Modeling and evaluation of robust whole-hand caging manipulation," *IEEE Trans. Robot.*, vol. 35, no. 3, pp. 549–563, 2019.

[16] N. S. Pollard, "Synthesizing grasps from generalized prototypes," in *Proc. Int. Conf. Robot. Automat.*, vol. 3. IEEE, 1996, pp. 2124–2130.

[17] A. T. Miller and P. K. Allen, "Examples of 3d grasp quality computations," in *Proc. Int. Conf. Robot. Automat.*, vol. 2. IEEE, 1999, pp. 1240–1246.

[18] C. Ferrari, J. F. Canny, *et al.*, "Planning optimal grasps." in *Proc. Int. Conf. Robot. Automat.*, vol. 3, no. 4, 1992, p. 6.

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

[20] 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. Int. Conf. Robot. Automat.* IEEE, 2020, pp. 1546–1552.

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

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

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

[24] 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, 2016.

[25] D. Hsu, J.-C. Latombe, and R. Motwani, "Path planning in expansive configuration spaces," in *Proc. Int. Conf. Robot. Automat.*, vol. 3. IEEE, 1997, pp. 2719–2726.

[26] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," *Research Report 9811*, 1998.

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

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