

A Convex Formulation of Compliant Contact between Filaments and Rigid Bodies

Wei-Chen Li

Glen Chou

Abstract—We present a computational framework for simulating filaments interacting with rigid bodies through contact. Filaments are challenging to simulate due to their codimensionality, i.e., they are one-dimensional structures embedded in three-dimensional space. Existing methods often assume that filaments remain permanently attached to rigid bodies. Our framework unifies discrete elastic rod (DER) modeling, a pressure field patch contact model, and a convex contact formulation to accurately simulate frictional interactions between slender filaments and rigid bodies – capabilities not previously achievable. Owing to the convex formulation of contact, each time step can be solved to global optimality, guaranteeing complementarity between contact velocity and impulse. We validate the framework by assessing the accuracy of frictional forces and comparing its physical fidelity against baseline methods. Finally, we demonstrate its applicability in both soft robotics, such as a stochastic filament-based gripper, and deformable object manipulation, such as shoelace tying, providing a versatile simulator for systems involving complex filament-filament and filament-rigid body interactions.

I. INTRODUCTION

The simulation of robotic systems has proven invaluable for controller design, robot learning, data generation, and system verification. Despite widespread reliance on simulation software, few toolkits can accurately model slender deformable objects such as ropes, cables, and shoelaces with high physical fidelity. These objects are central to many applications, including knot tying, cable routing, and surgical suturing, yet they undergo large deformations, self-contact, and topological changes such as knotting, while also making and breaking contact with external bodies – factors that make them especially challenging to simulate. Existing approaches either employ simplified models, which may be physically inaccurate [1], or rely on full volumetric finite element method (FEM), which can be computationally prohibitive for the fine meshes required by slender objects, as the computational cost typically grows superlinearly with the number of mesh elements [2]. This work addresses this gap by presenting a simulation framework for slender deformable objects that combines discrete elastic rod (DER) modeling, a pressure field patch contact model, and a convex contact formulation to enable accurate simulation of contact with rigid bodies. Our efficient and accurate simulation framework aims to support both the characterization of soft robotics and the development of control strategies for deformable object manipulation [3].

We implement our framework on top of Drake [5]. Our main contributions are as follows:

The authors are with the Georgia Institute of Technology, Atlanta, GA, USA. {wli777, chou}@gatech.edu

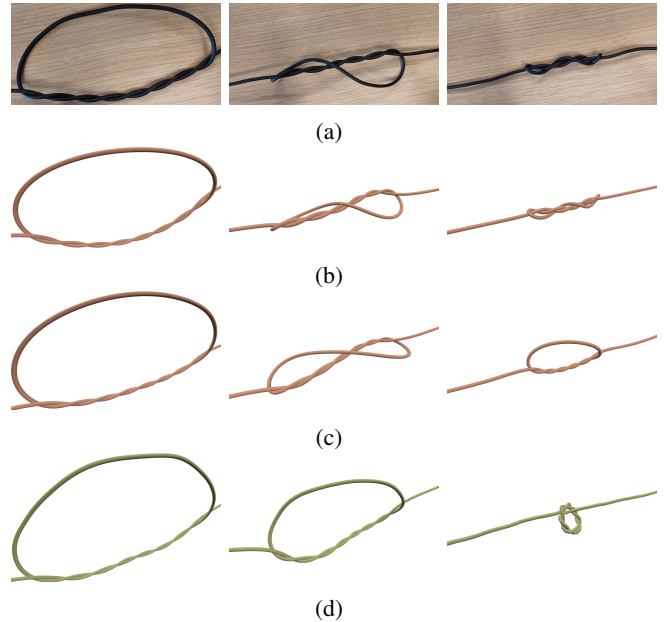


Fig. 1: Keyframes of knot tightening with an unknotting number of four, achieved by pulling on both ends of a rope. (a) Real rope. (b) Our simulation framework. (c) Existing contact energy-based method [4], which fails to prevent interpenetration; in the final keyframe, the rope undergoes tunneling, reducing the unknotting number to three. (d) A chain of rigid capsules with bending and twisting springs, which fails to reproduce realistic knot tightening behavior.

- 1) *Convex contact formulation for filaments.* We develop a convex contact solver, previously limited to rigid and volumetric deformables, that can simulate slender deformable bodies.
- 2) *Patch contact for slender deformables.* We extend a pressure field patch contact model, previously applied to rigid bodies, for slender deformables. This enables the simulation of lateral compliance and distributed contact forces that go beyond traditional point contact models.
- 3) *Demonstration in robotics applications.* We validate the framework on physics benchmarks such as knot tightening and showcase novel robotic applications including shoelace tying, which was not feasible with previous point contact models due to the highly variable pinching forces, preventing stable friction forces from being established and allowing the shoelace to slip.

Our source code is available at <https://github.com/wei-chen-li/drake-DER>.

II. RELATED WORK

The accurate simulation of filaments requires both a faithful mechanical model of slender deformables, and a robust treatment of self-contact and contact with external bodies. Prior research has approached these two aspects largely independently: studies on filament mechanics have focused on capturing bending, twisting, and stretching behaviors, while work on contact modeling has concentrated on non-penetration enforcement and frictional interactions. In the following, we review existing approaches in these two areas.

A. Models for Filaments

Rigid body simulators can be adapted to model filaments by connecting a series of rigid capsules via spherical joints, with springs used to approximate the twisting and bending stiffness of the filament. Previous work has employed simulators such as MuJoCo and PyBullet for this purpose [6], [7]. While this approach is straightforward, it requires careful tuning of the spring stiffness parameters and does not guarantee convergence to the physically correct behavior as the model resolution is increased.

Position-based dynamics (PBD) models materials as a set of particles with holonomic constraints. Future particle positions are first predicted from the current positions and velocities, and then projected onto the constraint manifold to enforce the constraints [8]. PhysX employs this approach for simulating cloth and rope, where ropes are represented using distance and angular constraints, as well as additional constraints for interactions with external objects. A major limitation of this method is that the effective stiffness depends on the integration time step, which limits its ability to accurately simulate force effects.

The full mechanics of filaments can be described by the Cosserat rod model [9], which governs the dynamics of slender bodies capable of bending, twisting, stretching, and shearing. Because shear deformation is explicitly modeled, each rod segment is associated with an $SO(3)$ frame. Elastica [10] employs this model for simulating biological muscles, where shear deformations are essential. However, the use of explicit Verlet integration restricts the simulator to small time steps, limiting simulation speed.

Neglecting shear deformation enables the use of implicit integrators, allowing simulations to advance with larger time steps. This approach corresponds to the DER model [11]. Prior work on DER resolves contact by augmenting the dynamics with a contact energy term [4], whereas our approach explicitly resolves contact using collision detection and a contact solver.

B. Models for Contact

Incremental potential contact (IPC) [12] enforces non-penetration using barrier functions. It further employs continuous collision detection within its Newton-based solver to guarantee non-penetration. While highly effective for graphics-oriented simulations, IPC can be computationally expensive and may be too slow for robotics applications.

On the other hand, simulating multibody dynamics with frictional contact typically requires solving a nonlinear complementarity problem (NCP) and can generally only be solved to suboptimal levels in practice [13]. To improve computational tractability, convex approximations of the contact problem have been formulated [14]. More recent work has introduced regularization to render the problem strictly convex and selected regularization parameters to better match physical contact phenomena [15], [16].

In the following sections, we will overview our methodological framework as follows:

- Section III-A describes the DER mechanics.
- Section III-B introduces the point and patch contact model.
- Section IV describes the convex contact solver.

While the contact solver is based on prior work, our novelty lies in the correction of the DER dynamics, the adaptation of patch contact to slender filaments, and the integration of all components.

III. MODELING OF FILAMENT AND CONTACT

We model filaments using DERs. Both the articulated rigid system and the DER are described using generalized coordinates. The full system configuration is denoted by $\mathbf{q} \in \mathbb{R}^{n_q}$, and the generalized velocity by $\mathbf{v} \in \mathbb{R}^{n_v}$, where n_q and n_v represent the dimensions of the configuration and velocity vectors, respectively. The time derivative of \mathbf{q} is related to \mathbf{v} through a kinematic map $\mathbf{N}(\mathbf{q}) : \mathbb{R}^{n_q} \rightarrow \mathbb{R}^{n_q \times n_v}$:

$$\dot{\mathbf{q}} = \mathbf{N}(\mathbf{q}) \mathbf{v}. \quad (1)$$

For the DER degrees of freedom (DoFs), the corresponding submatrix of $\mathbf{N}(\mathbf{q})$ is simply the identity. We now detail the kinematics and dynamics specific to the DER model.

A. Mechanics of DER

The centerline of a DER is composed of n_n nodes connected sequentially by $n_e = n_n - 1$ edges, as illustrated in Fig. 2. Each node i is represented by its position $\mathbf{x}_i \in \mathbb{R}^3$ in the world frame. The edge between nodes i and $i + 1$ has a unit tangent vector \mathbf{t}^i . We use subscripts to index node-related quantities and superscripts for edge-related ones. Each edge is associated with a reference frame consisting of directors $(\mathbf{d}_1^i, \mathbf{d}_2^i, \mathbf{t}^i)$ and a material frame consisting of directors $(\mathbf{m}_1^i, \mathbf{m}_2^i, \mathbf{t}^i)$. The relative rotation between the two frames is described by the angle $\gamma^i \in \mathbb{R}$. The material frame rotates with the rod, with \mathbf{m}_1^i and \mathbf{m}_2^i aligned with the principal axes of the cross-section (Fig. 2b), whereas the reference frame remains fixed in rotation about \mathbf{t}^i . The configuration vector of the DER is given by

$$\mathbf{q}_r = [\mathbf{x}_1^\top \quad \gamma^1 \quad \mathbf{x}_2^\top \quad \dots \quad \gamma^{n_e} \quad \mathbf{x}_{n_n}^\top]^\top \in \mathbb{R}^{4n_n - 1}. \quad (2)$$

The subscript r specifies that the quantities are associated with a DER, in contrast to the full system that encompasses both the rigid bodies and all DERs.

A DER experiences internal elastic forces resulting from stretching, twisting, and bending, under the assumption that shear deformation is negligible. The corresponding internal energy contributions are defined as follows:

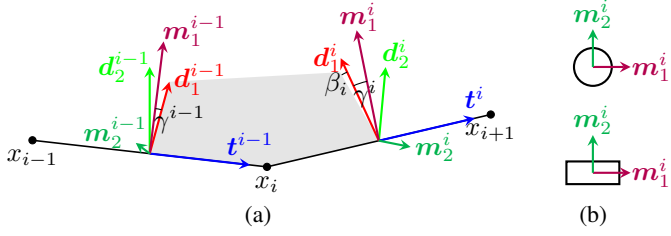


Fig. 2: Schematic of DER. (a) Centerline of a DER. Director \mathbf{d}_1^{i-1} is rotated about the axis $\mathbf{t}^{i-1} \times \mathbf{t}^i$ by the angle between \mathbf{t}^{i-1} and \mathbf{t}^i . The resulting director forms an angle β_i with \mathbf{d}_1^i . (b) Circular or rectangular cross-section of a DER.

a) *Stretching energy:*

$$E_s = \frac{1}{2} \sum_{i=1}^{n_e} EA \left(\frac{\|\mathbf{e}^i\|}{\|\bar{\mathbf{e}}^i\|} - 1 \right)^2 \|\bar{\mathbf{e}}^i\|, \quad (3)$$

where $\mathbf{e}^i = \mathbf{x}_{i+1} - \mathbf{x}_i$, and $\bar{\mathbf{e}}^i$ is the edge vector in the undeformed configuration, i.e., the vector between nodes i and $i+1$ before any deformation. Here, E is the Young's modulus and A is the cross-sectional area.

b) *Twisting energy:*

$$E_t = \frac{1}{2} \sum_{i=2}^{n_n-1} \frac{GJ}{\bar{l}_i} (\tau_i - \bar{\tau}_i)^2, \quad (4)$$

where $\tau_i = \gamma^i - \gamma^{i-1} + \beta_i$, and β_i is the signed angle between \mathbf{d}_1^{i-1} and \mathbf{d}_1^i after parallel transport to a common frame (see Fig. 2a). Here, $\bar{\tau}_i$ denotes the value of τ_i in the undeformed configuration, and the undeformed Voronoi length is defined as $\bar{l}_i = (\|\bar{\mathbf{e}}^{i-1}\| + \|\bar{\mathbf{e}}^i\|)/2$. The parameters G and J are the shear modulus and the cross-sectional polar moment of inertia, respectively.

c) *Bending energy:*

$$E_b = \frac{1}{2} \sum_{i=1}^{n_n-1} \frac{EI_1}{\bar{l}_i} (\kappa_{1,i} - \bar{\kappa}_{1,i})^2 + \frac{1}{2} \sum_{i=1}^{n_n-1} \frac{EI_2}{\bar{l}_i} (\kappa_{2,i} - \bar{\kappa}_{2,i})^2, \quad (5)$$

where the discrete curvature vector is defined as

$$\boldsymbol{\kappa}_i = \frac{2\mathbf{t}^{i-1} \times \mathbf{t}^i}{1 + \mathbf{t}^{i-1} \cdot \mathbf{t}^i},$$

and its projections define the scalar curvatures:

$$\kappa_{1,i} = \boldsymbol{\kappa}_i \cdot \frac{\mathbf{m}_2^{i-1} + \mathbf{m}_2^i}{2}, \quad \kappa_{2,i} = \boldsymbol{\kappa}_i \cdot \left(-\frac{\mathbf{m}_1^{i-1} + \mathbf{m}_1^i}{2} \right).$$

Here, $\bar{\kappa}_{1,i}$ and $\bar{\kappa}_{2,i}$ are the corresponding scalar curvatures in the undeformed configuration, and I_1 and I_2 are the second moments of area of the rod's cross-section about its two principal axes.

With the total internal elastic energy of the DER given by $E_{r,\text{int}} = E_s + E_t + E_b$, the dynamics of DER are governed by

$$\mathbf{M}_r \ddot{\mathbf{q}}_r = -\frac{\partial E_{r,\text{int}}}{\partial \mathbf{q}_r} - (\alpha \mathbf{M}_r + \beta \mathbf{K}_r) \dot{\mathbf{q}}_r + \mathbf{F}_{r,\text{ext}}, \quad (6)$$

where \mathbf{M}_r is the diagonal lumped mass matrix, $\mathbf{K}_r = \partial^2 E_{r,\text{int}} / \partial \mathbf{q}_r^2$ is the stiffness matrix, α , β are

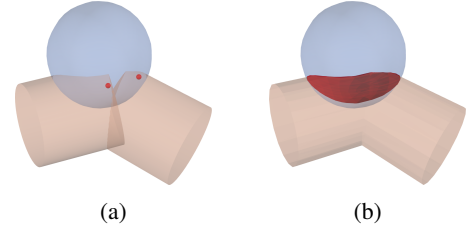


Fig. 3: Contact between a two-segment filament and a ball using different models. (a) Point contact, shown as red points. (b) Pressure field patch contact, shown as red patches.

coefficients for the Rayleigh damping model, and $\mathbf{F}_{r,\text{ext}}$ includes external forces.

In our implementation, we derived analytical expressions for the Jacobian and Hessian of $E_{r,\text{int}}$, which differ from those provided in [17, §8.5]. The corrected forms were validated using forward-mode automatic differentiation. For example, the derivative of the bending energy E_b with respect to the angle γ_i is

$$\begin{aligned} \frac{\partial E_b}{\partial \gamma_i} &= \frac{EI_1}{\bar{l}_i} (\kappa_{1,i} - \bar{\kappa}_{1,i}) \left(-\frac{1}{2} \mathbf{m}_1^i \cdot \boldsymbol{\kappa}_i \right) \\ &+ \frac{EI_2}{\bar{l}_i} (\kappa_{2,i} - \bar{\kappa}_{2,i}) \left(-\frac{1}{2} \mathbf{m}_2^i \cdot \boldsymbol{\kappa}_i \right) \\ &+ \frac{EI_1}{\bar{l}_{i+1}} (\kappa_{1,i+1} - \bar{\kappa}_{1,i+1}) \left(-\frac{1}{2} \mathbf{m}_1^i \cdot \boldsymbol{\kappa}_{i+1} \right) \\ &+ \frac{EI_2}{\bar{l}_{i+1}} (\kappa_{2,i+1} - \bar{\kappa}_{2,i+1}) \left(-\frac{1}{2} \mathbf{m}_2^i \cdot \boldsymbol{\kappa}_{i+1} \right), \end{aligned}$$

rather than zero. Since the Jacobian directly determines the internal elastic force, using the corrected expressions are essential for accurate dynamics.

B. Kinematics of Contact

A simple model for contact is to approximate the intersection between two bodies as occurring at a single point, where both normal and frictional forces are applied. While effective for rigid or stiff bodies, this assumption can be overly simplistic for soft filaments, where deformation occurs not only along the length of the filament but also laterally. In these cases, modeling contact as distributed over patches rather than points yields more physically realistic results. We present the descriptions for both point and patch contact models below.

1) *Point Contact:* Each DER segment is geometrically approximated as either a cylinder (for circular cross-sections) or a box (for rectangular cross-sections). Other rigid bodies are likewise represented using primitive geometries. A geometry engine then identifies candidate pairs of primitives that are in collision and reports them as contact pairs, excluding neighboring segments of the same filament from collision detection (see Fig. 3a).

Each contact pair i is defined by a contact point, the contact normal direction \mathbf{n}_i , and the signed distance ϕ_i where $\phi_i < 0$ indicates interpenetration. For each contact, we define a local contact frame with the \mathbf{n}_i direction aligned with the z -axis. The relative velocity of contact, $\mathbf{v}_{c,i} \in \mathbb{R}^3$, is expressed in this frame. It can be decomposed into tangential

and normal components as

$$[\mathbf{v}_{c,i}] = \begin{bmatrix} \mathbf{v}_{c,i} - v_{n,i} \mathbf{n}_i \\ v_{n,i} \end{bmatrix}, \quad \text{where } v_{n,i} = \mathbf{v}_{c,i} \cdot \mathbf{n}_i.$$

2) *Pressure Field Patch Contact*: Each DER segment is geometrically modeled as a volumetric mesh, with an associated pressure field. This pressure field is defined to be zero on its outer surface and reaches a prescribed maximum along its centerline. Similarly, each rigid body is represented as a volume mesh with a pressure field that is zero on its surface and maximal at its geometric center. These pressure fields approximate the contact forces that would arise upon interpenetration [18]. According to Newton's third law, the contact patch between two overlapping bodies corresponds to the locus where their pressure fields are equal. A geometry engine detects this contact patch and reports it as a set of polygons (see Fig. 3b).

Each polygon i within the contact patch is characterized by its normal vector \mathbf{n}_i and area A_i , as well as the pressure p_i and pressure gradient ∇p_i evaluated at its centroid. The stiffness in the direction of the normal is given by

$$k = A_i \nabla p_i \cdot (-\mathbf{n}_i).$$

Since the normal contact force is

$$\mathbf{f}_n = A_i p_i = -k \phi_i,$$

the effective signed distance can be expressed as

$$\phi_i = \frac{p_i}{\nabla p_i \cdot \mathbf{n}_i}.$$

Once the effective signed distance ϕ_i is computed, the polygon within the contact patch is treated as an equivalent point contact in subsequent calculations.

Assuming there are a total of n_c contact points, stacking the relative velocities of all contact points yields the vector $\mathbf{v}_c \in \mathbb{R}^{3n_c}$, which is related to the generalized velocity through the contact Jacobian $\mathbf{J} \in \mathbb{R}^{n_v \times 3n_c}$:

$$\mathbf{v}_c = \mathbf{J}^\top \mathbf{v}. \quad (7)$$

C. Discrete Time Stepping Scheme

To simulate the system's evolution over time, we divide time into equal intervals of duration δt , repeatedly advancing the system state from t_n to $t_{n+1} = t_n + \delta t$. To simplify notation, variables evaluated at the current time step t_n are indicated with a subscript 0, while variables without a subscript refer to values at the next time step t_{n+1} . Following the θ -method formulation in accordance with [2], [15], we define intermediate quantities as

$$\mathbf{q}^\theta \triangleq \theta \mathbf{q} + (1 - \theta) \mathbf{q}_0, \quad (8a)$$

$$\mathbf{v}^\theta \triangleq \theta \mathbf{v} + (1 - \theta) \mathbf{v}_0, \quad (8b)$$

$$\mathbf{v}^{\theta v_q} \triangleq \theta v_q \mathbf{v} + (1 - \theta v_q) \mathbf{v}_0, \quad (8c)$$

which along with

$$\mathbf{q} = \mathbf{q}_0 + \delta t \mathbf{N}(\mathbf{q}^\theta) \mathbf{v}^{\theta v_q}, \quad (9)$$

allows the representation of backwards Euler with $\theta = \theta_{v_q} = 1$, symplectic Euler with $\theta = 0$, $\theta_{v_q} = 1$, or the midpoint rule with $\theta = \theta_{v_q} = 1/2$.

Using (8), we formulate the constrained dynamics of the coupled rigid body and DER system as

$$\mathbf{M}(\mathbf{q}^\theta) (\mathbf{v} - \mathbf{v}_0) = \delta t \mathbf{k}(\mathbf{q}^\theta, \mathbf{v}^\theta) + \mathbf{J}(\mathbf{q}_0) \boldsymbol{\lambda}, \quad (10a)$$

$$\mathcal{C} \ni \boldsymbol{\lambda} \perp \mathbf{v}_c - \hat{\mathbf{v}}_c \in \mathcal{C}^*, \quad (10b)$$

where

$$\hat{\mathbf{v}}_c \triangleq -\frac{1}{\delta t} [0 \ 0 \ \phi_{1,0} \ \dots \ 0 \ 0 \ \phi_{n_c,0}]^\top \in \mathbb{R}^{3n_c},$$

with $\phi_{i,0}$ the signed distance of the i -th contact at the current time step. The convex cone \mathcal{C} is defined as $\mathcal{C} \triangleq \mathcal{C}_1 \times \dots \times \mathcal{C}_{n_c}$, where \mathcal{C}_i is the friction cone of the i -th contact. The dual cone of \mathcal{C} is written as \mathcal{C}^* . Throughout, we use the notation $\mathbf{a} \perp \mathbf{b}$ to mean $\mathbf{a}^\top \mathbf{b} = 0$.

Equation (10a) expresses the momentum balance over the time step, incorporating both rigid body and DER DoFs. The function $\mathbf{k}(\mathbf{q}^\theta, \mathbf{v}^\theta)$ collects all non-contact forces, including Coriolis terms, gravitational forces, and internal elastic forces. The contact impulse vector $\boldsymbol{\lambda} \in \mathbb{R}^{3n_c}$ captures the net impulses due to contact, including both normal and frictional components. The contact constraint (10b) requires that the contact impulse $\lambda_{n,i}$ and the gap velocity $\phi_i/\delta t \approx (\phi_{i,0} + v_{n,i} \delta t)/\delta t$ satisfy a complementarity condition, such that if the contact is separating ($\phi_i/\delta t > 0$), no impulse is applied ($\lambda_{n,i} = 0$); if the contact is maintained ($\phi_i = 0$), a nonzero impulse may be applied.

IV. DYNAMICS SOLVER

This section outlines the procedure used to solve the constrained dynamics in (10) to advance the state from one time step to the next. The solver employs a two-stage approach where in the first stage, we compute the unconstrained (i.e., free-motion) velocities that the system would have in the absence of contact. In the second stage, we solve a convex approximation of (10) to enforce contact interactions and determine the velocities and contact impulses at the next time step.

We define the momentum residual as

$$\mathbf{m}(\mathbf{v}) = \mathbf{M}(\mathbf{q}^\theta) (\mathbf{v} - \mathbf{v}_0) - \delta t \mathbf{k}(\mathbf{q}^\theta, \mathbf{v}^\theta). \quad (11)$$

In the first stage of the solver, we compute the free-motion velocity \mathbf{v}^* by solving the nonlinear equation

$$\mathbf{m}(\mathbf{v}^*) = \mathbf{0}. \quad (12)$$

This equation is solved using the Newton-Raphson method, which iteratively updates the velocity estimate until the residual norm is sufficiently small. Notably, the Hessian of the momentum residual has a fixed sparsity pattern, allowing the reuse of symbolic factorization throughout the iterations.

In the second stage of the solver, we linearize (10) around the previously-computed free-motion velocity \mathbf{v}^* . This yields

$$\mathbf{A} (\mathbf{v} - \mathbf{v}^*) = \mathbf{J} \boldsymbol{\lambda}, \quad (13a)$$

$$\mathcal{C} \ni \boldsymbol{\lambda} \perp \mathbf{v}_c - \hat{\mathbf{v}}_c \in \mathcal{C}^*. \quad (13b)$$

Here, \mathbf{A} is the matrix obtained by linearizing the momentum balance equation (10a) with respect to the generalized velocity \mathbf{v} about the nominal point $\mathbf{v} = \mathbf{v}^*$. By substituting the DER dynamics from (6) into the time-discretized system (8)–(10), the diagonal block of \mathbf{A} corresponding to the DER degrees of freedom is given by

$$\mathbf{A}_r = (1 + \alpha\theta\delta t) \mathbf{M}_r + \theta\delta t (\beta + \theta_{vq}\delta t) \mathbf{K}_r. \quad (14)$$

Because the stiffness matrix \mathbf{K}_r may be indefinite, we regularize \mathbf{A}_r to ensure positive definiteness.

Since only a subset of the total DoFs are involved in contact, the contact Jacobian \mathbf{J} contains nonzero entries only in the corresponding subset of its rows. To exploit this sparsity, we permute the DoFs so that those participating in contact appear first. Under this reordering, (13a) can be written in block form as

$$\begin{bmatrix} \mathbf{A}_{pp} & \mathbf{A}_{pn} \\ \mathbf{A}_{np} & \mathbf{A}_{nn} \end{bmatrix} \begin{bmatrix} \mathbf{v}_p - \mathbf{v}_p^* \\ \mathbf{v}_n - \mathbf{v}_n^* \end{bmatrix} = \begin{bmatrix} \mathbf{J}_p \boldsymbol{\lambda} \\ \mathbf{0} \end{bmatrix}, \quad (15)$$

where the subscript p refers to the participating (contact-related) DoFs, and n denotes the non-participating ones. The Schur complement of \mathbf{A}_{pp} is given by

$$\mathbf{S}_p = \mathbf{A}_{pp} - \mathbf{A}_{pn} \mathbf{A}_{nn}^{-1} \mathbf{A}_{np} \succ 0.$$

Using this, (15) can be equivalently expressed as

$$\mathbf{S}_p (\mathbf{v}_p - \mathbf{v}_p^*) = \mathbf{J}_p \boldsymbol{\lambda}, \quad (16a)$$

$$(\mathbf{v}_n - \mathbf{v}_n^*) = -\mathbf{A}_{nn}^{-1} \mathbf{A}_{np} (\mathbf{v}_p - \mathbf{v}_p^*). \quad (16b)$$

The equations (16a) and (13b) are exactly the Karush-Kuhn-Tucker (KKT) conditions of the following convex optimization problem:

$$\begin{aligned} & \underset{\mathbf{v}_p}{\text{minimize}} && \frac{1}{2} \|\mathbf{v}_p - \mathbf{v}_p^*\|_{\mathbf{S}_p}^2 \\ & \text{subject to} && \mathbf{v}_c - \hat{\mathbf{v}}_c \in \mathcal{C}^*, \end{aligned} \quad (17)$$

where $\|\mathbf{x}\|_{\mathbf{Q}}^2 \triangleq \mathbf{x}^\top \mathbf{Q} \mathbf{x}$. In particular, the Lagrangian of (17) is

$$\mathcal{L}(\mathbf{v}_p, \boldsymbol{\lambda}) = \frac{1}{2} \|\mathbf{v}_p - \mathbf{v}_p^*\|_{\mathbf{S}_p}^2 - \boldsymbol{\lambda}^\top (\mathbf{J}_p^\top \mathbf{v}_p - \hat{\mathbf{v}}_c), \quad (18)$$

where $\boldsymbol{\lambda} \in \mathcal{C}$ is the dual variable corresponding to the constraint $\mathbf{v}_c - \hat{\mathbf{v}}_c \in \mathcal{C}^*$. Recall from (7) that the stacked contact relative velocity satisfies $\mathbf{v}_c = \mathbf{J}_p^\top \mathbf{v}_p$.

Thus, solving (17) directly provides both the primal solution (the participating velocity \mathbf{v}_p) and the dual solution (the contact impulse $\boldsymbol{\lambda}$). To solve this problem, we employ the semi-analytic primal (SAP) solver introduced in [15], which offers guaranteed convergence for this class of convex formulations.

V. RESULTS AND DISCUSSION

We present a set of test cases to evaluate the accuracy, robustness, and performance of our method, with all simulations executed on a single thread. Table I summarizes the contact model and time step size used in each test case, along with the simulation real-time rate (ratio of simulated time to wall-clock time) and the total number of contacts. Fig. 4

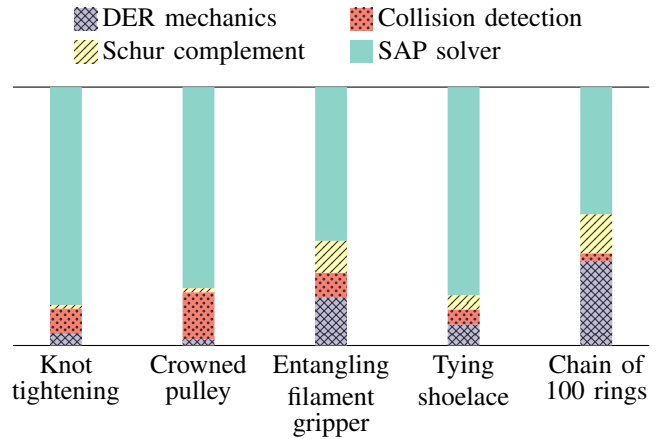


Fig. 4: Simulation runtime split among computation components.

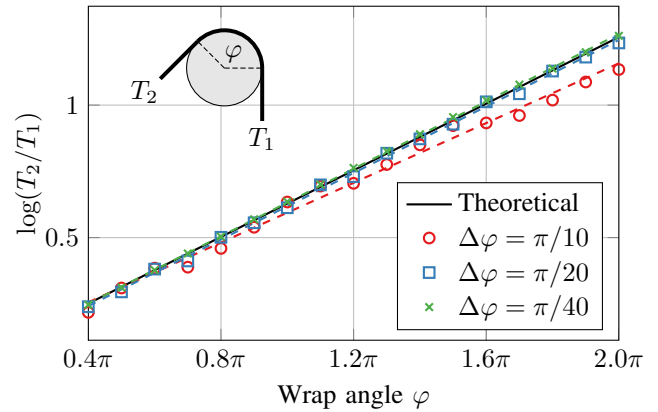


Fig. 5: Ratio of pulling forces at both ends of a rope as a function of wrap angle around a cylindrical post. Each rope segment has length $R \Delta\varphi$, where R is the radius of the cylindrical post.

further reports the distribution of computation time across the main components of the simulation pipeline. To prevent slender deformable objects from tunneling due to missed collision detections, and to ensure that the linearization in (13) remains a valid approximation, the time step size must be set on the order of 1×10^{-3} s or smaller. We also find that patch contact models significantly increase the number of effective contact points, since each contact patch consists of multiple polygons, each treated as an individual contact point. This leads to an increase in the contact resolution workload. Finally, the SAP solver consistently dominates runtime, accounting for at least 50% of the total cost across all test cases, making it the primary computational bottleneck.

A. Capstan Effect

We simulate the capstan effect, in which a rope is wrapped around a cylindrical post through a wrap angle φ and

TABLE I: Timing and scene statistics.

Example	Contact model	DoFs	Time step (s)	Real-time rate	Number of contact points average / max
Knot tightening	Point contact	1205	0.0001	0.002	58 / 164
Crowned pulley	Patch contact	202	0.003	0.037	804 / 917
Entangling filament gripper	Point contact	6283	0.001	0.013	104 / 200
Tying shoelace	Patch contact	816	0.002	0.088	164 / 591
Chain of 100 rings	Point contact	8000	0.0008	0.012	396 / 396

subjected to pulling forces T_1 and T_2 at its two ends. Due to frictional interaction between the rope and the capstan surface, the ratio of transmitted forces grows exponentially with the wrap angle. The maximum pulling force that can be sustained is given by the classical expression $T_2 = T_1 e^{\mu\varphi}$, where μ is the coefficient of friction [19, §6.8].

In our setup, the dynamic coefficient of friction μ between the cylindrical post and the rope is set to 0.2, and self-collision of the rope is disabled to isolate interaction between the rope and the cylindrical post. One end of the rope is controlled via a proportional-derivative (PD) controller to track a fixed position, while the other end is gradually pulled with increasing force. Once steady state is reached, the ratio of forces at both ends is recorded, and the experiment is repeated for multiple wrap angles φ . The results, shown in Fig. 5, indicate that the simulation accurately reproduces the expected exponential force amplification. A mesh refinement study further confirms that finer discretizations, where each rope segment spans an angle $\Delta\varphi$ around the post, converge toward the theoretical prediction, validating the accuracy of our frictional contact model.

B. Knot Tightening

We simulate the tightening of a knot with an unknotting number of four (Fig. 1), corresponding to a configuration where the rope braids around itself four times. In the simulation, the rope is pulled at both ends with a constant velocity of 0.8 m/s, while the applied force is regulated by a PD controller. Due to the high pulling speed, the time step is set to 1×10^{-4} s. The result of our framework (Fig. 1b) closely matches that of a real rope (Fig. 1a), and successfully reproduces the phenomenon of snap-through buckling: once the knot becomes sufficiently tight, it abruptly transitions from a near-circular shape to a collapsed configuration.

In contrast, the method in [4], which augments the dynamics with a contact energy term to enforce non-penetration, fails under the same material properties, pulling speed, and time step size (Fig. 1c). While the contact energy dominates when the rope is loose, it becomes comparatively small relative to the internal elastic energy as the rope tightens, resulting in tunneling.

For comparison, we also model the rope as a chain of rigid capsules with virtual springs to emulate bending and twisting stiffness (Fig. 1d). Although this model avoids tunneling, it fails to reproduce the characteristic snap-through buckling, instead producing a non-physical tightened configuration.

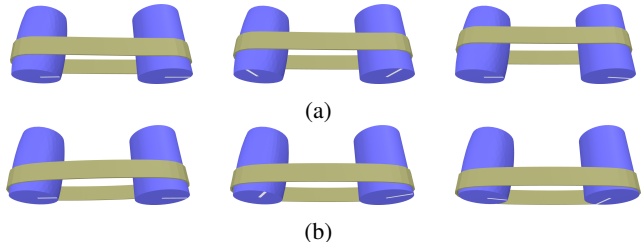


Fig. 6: Keyframes of a belt running over a pair of pulleys, with the left pulley having a crowned profile. (a) Our framework simulates the self-centering of the belt as the pulley rotates. (b) Volumetric FEM [2] fails to capture even the belt tensioning.

C. Crowned Pulley

A crowned pulley has a convex surface, with a slightly larger diameter at the center that tapers toward the edges. When a flat belt runs over such a pulley, it naturally self-centers [20, §17.1]. We simulate a belt riding on a crowned pulley, discretized so that each segment has a length equal to one-twentieth of the pulley’s circumference (Fig. 6a). The belt configuration is represented by the nodal positions x_i of its centerline together with twisting angles γ^i , and is assigned a wide rectangular cross-section. Although this representation does not capture bending across the belt’s width, it is sufficient to reproduce the self-centering phenomenon.

This behavior emerges only when both the distributed contact between the belt and pulley and the belt’s internal stretching and twisting are modeled. If any component is omitted, the self-centering effect disappears: a point contact model cannot generate the distributed forces needed to support the belt on the pulley’s curved surface, neglecting stretching prevents proper belt tensioning, and neglecting twisting prevents the segments from forming the geometry that drives self-centering.

We compare against the volumetric FEM in [2] (Fig. 6b). Even with a mesh resolution in the thickness direction equal to one-third of the belt thickness, the volumetric FEM fails to capture belt tensioning and, as a result, cannot form the geometry required for self-centering. This failure is likely due to the large element aspect ratio: the mesh is coarse along the belt’s longitudinal and width directions but fine in the thickness direction, which artificially increases bending stiffness and prevents the belt from deforming correctly. Using an even finer mesh to reduce the aspect ratio would

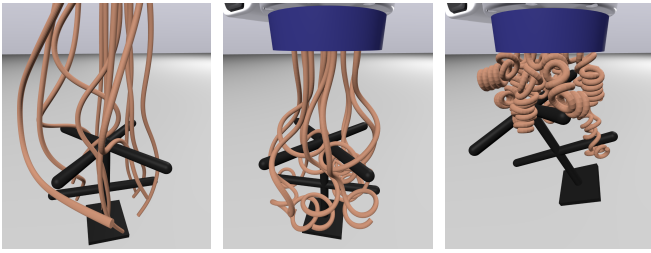


Fig. 7: Keyframes of the entangling filament gripper grasping a branched object. The gripper initially approaches the object from above, and as the filaments’ natural curvature increases, it achieves a stochastic grasp on the object.

be computationally prohibitive.

D. Entangling Filament Gripper

Another application of filament–rigid body interaction simulation lies in the study of novel soft robotic grippers [21], [22]. In particular, we simulate an entangling filament gripper [21] grasping a branched object as illustrated in Fig. 7. This type of gripper is composed of multiple slender elastomer filaments whose natural curvature can be actuated through pneumatic pressure. Grasping is achieved by exploiting the stochastic self-entanglement and cross-entanglement of the filaments, enabling the gripper to grasp topologically-complex objects. In our simulations, the directions of natural curvature are held fixed (the ratio $\bar{\kappa}_{1i}/\bar{\kappa}_{2i}$ remains constant), while the overall curvature is gradually increased (simultaneously increasing $\bar{\kappa}_{1i}$ and $\bar{\kappa}_{2i}$) to achieve grasping. The simulation robustly resolves contact interactions among filaments, between filaments and the branched object, and between filaments and the gripper base. Furthermore, owing to the use of implicit time integration, our simulations can advance with significantly larger time steps than the 1×10^{-5} s step size used in the original study [21], which employs *Elastica* [10] as the simulator.

E. Tying Shoelace

To further demonstrate frictional and compliant contact between filaments and rigid bodies, we simulate bimanual robot arms tying a shoelace. The arms follow a predefined open-loop trajectory. Using the pressure field patch contact model for both the shoelace and gripper fingers successfully simulates the full shoelace tying sequence (Fig. 8a).

In contrast, a point contact model causes the shoelace to slip from the gripper fingers, preventing task completion (Fig. 8b). The difference is evident in the finger pinching forces (Fig. 8d). With patch contact, the force smoothly increases until it balances the PD-controlled finger forces, producing stable friction. With point contact, however, the pinching force fluctuates violently, then drops as the shoelace escapes and the fingers close completely. These oscillations stem from how contact is resolved: the solver enforces non-penetration constraints (10b) at the end of each step, while the free-motion update (12) is applied in the next. The penetration depth after the free-motion update can vary

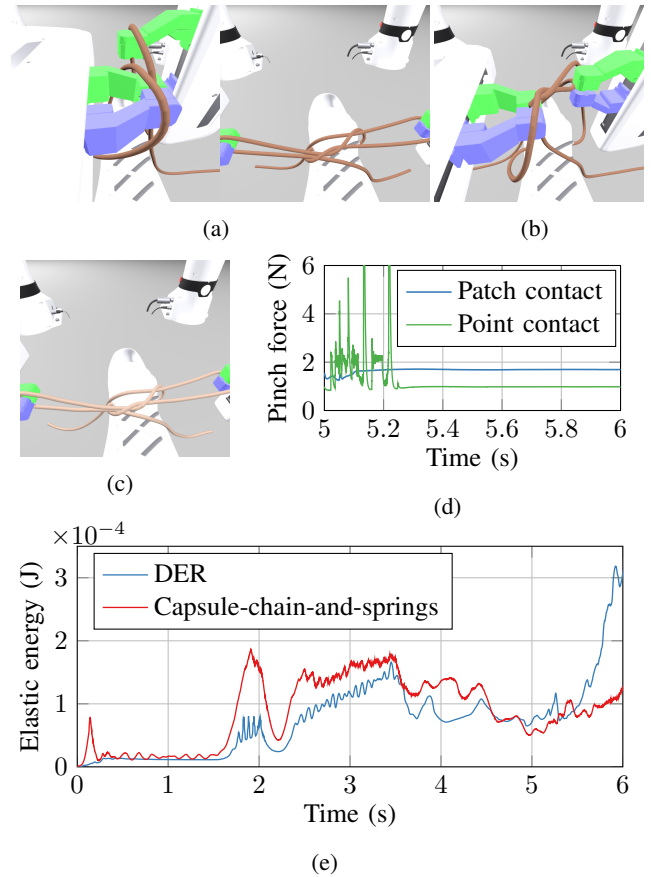


Fig. 8: Bimanual robotic arms tying a shoelace. (a) **DER with patch contact** successfully simulates the full shoelace-tying sequence. (b) **DER with point contact** fails, as the shoelace slips out of the fingers. (c) **Capsule-chain-and-springs with patch contact** produces visually convincing results. (d) Pinching forces of the gripper fingers. (e) Internal elastic energy of the shoelace with different models.

across time steps, leading to unstable force estimates and ultimately preventing steady friction.

For comparison, we also model the shoelace with a chain of rigid capsules with virtual springs to emulate bending and twisting stiffness. By extensively tuning the stiffness and damping parameters, the capsule-chain-and-springs model can produce visually similar results to the DER model (Fig. 8c). However, due to the high stiffness of the springs, this simulation requires a time step of 1×10^{-4} s, much smaller compared to the DER model. Furthermore, examining the internal elastic energy of both models (Fig. 8e) reveals that the capsule-chain-and-springs model exhibits much higher frequency fluctuations. These high-frequency fluctuations are an artifact, as such high-frequency content would naturally dissipate quickly in real-world materials.

F. Chain of Rings

A chain composed of identical rings is simulated under gravity (Fig. 9a). Each ring has a radius of 1 cm, a circular cross-section of diameter 2.5 mm, a material density of

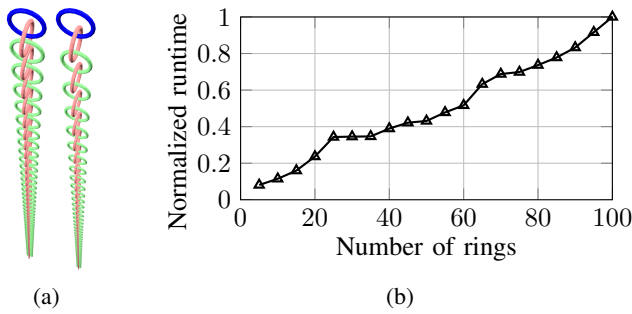


Fig. 9: A chain of rings suspended under gravity. (a) Keyframes of a 100-ring chain. (b) Runtime for varying number of rings.

500 kg/m³, and a Young’s modulus of 10 MPa. The geometry of each ring is discretized into 20 segments. A time step of 8×10^{-4} s is used, small enough to prevent missed collision detections and the resulting tunneling artifacts. To evaluate long-term stability, the simulation was continued for an additional 10 minutes of physical time after the chain had settled under gravity. The chain remained in static equilibrium throughout this period, confirming the stability of the simulation over extended durations.

To assess the scalability of our framework, we simulate chains with varying numbers of rings and measure the runtime (Fig. 9b). The runtime exhibits linear scaling with the number of rings. This behavior is largely due to the example exhibiting stable contact patterns, which allows the SAP solver to reuse the symbolic factorization in its internal Cholesky decomposition.

VI. CONCLUSION

We presented a convex formulation of frictional contact between filaments and rigid bodies, combining physically accurate DER dynamics and contact mechanics. Contact is modeled using either point contact, suited for efficiency and approximately fixed cross-sections, or pressure field patch contact, which captures lateral deformation and fine-grained friction. Both models are stateless, and contact is resolved by partitioning out non-participating DoFs and solving with the SAP solver, which guarantees global optimality.

Through a series of test cases, we validated the accuracy of frictional forces and the robustness of our framework in handling scenes with dense contacts. We also compared our approach against baseline methods, demonstrating improvements in physical fidelity in enforcing non-penetration, producing stable frictional forces, and maintaining smooth internal elastic energy.

Several limitations point to promising directions for future work. Profiling indicates that the SAP solver accounts for the majority of computational cost, motivating exploration of parallelization strategies, particularly for the Cholesky decomposition. Additionally, the requirement for small time steps remains a bottleneck for simulation speed. Adaptive time-stepping strategies could allow larger time steps in scenarios with sparse contacts.

REFERENCES

- [1] H. Yin, A. Varava, and D. Kragic, “Modeling, learning, perception, and control methods for deformable object manipulation,” *Sci. Robotics*, vol. 6, no. 54, p. eabd8803, 2021.
- [2] X. Han, J. Masterjohn, and A. Castro, “A convex formulation of frictional contact between rigid and deformable bodies,” *IEEE Robotics Autom. Lett.*, vol. 8, no. 10, pp. 6219–6226, 2023.
- [3] J. Zhu, A. Cherubini, C. Dune, D. Navarro-Alarcón, F. Alambeigi, D. Berenson, F. Ficuciello, K. Harada, J. Kober, X. Li, J. Pan, W. Yuan, and M. Gienger, “Challenges and outlook in robotic manipulation of deformable objects,” *IEEE Robotics & Autom. Mag.*, vol. 29, no. 3, p. 67–77, 2022.
- [4] A. Choi, D. Tong, M. K. Jawed, and J. Joo, “Implicit contact model for discrete elastic rods in knot tying,” *J. Appl. Mech.*, vol. 88, no. 5, 2021.
- [5] R. Tedrake and the Drake Development Team, “Drake: Model-based design and verification for robotics,” 2019. [Online]. Available: <https://drake.mit.edu>
- [6] W. Yan, A. Vangipuram, P. Abbeel, and L. Pinto, “Learning predictive representations for deformable objects using contrastive estimation,” in *Conference on Robot Learning*. PMLR, 2021, Conference Proceedings, p. 564–574.
- [7] V. Lim, H. Huang, L. Y. Chen, J. Wang, J. Ichnowski, D. Seita, M. Laskey, and K. Goldberg, “Real2sim2real: Self-supervised learning of physical single-step dynamic actions for planar robot casting,” in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 8282–8289.
- [8] M. Müller, B. Heidelberger, M. Hennix, and J. Ratcliff, “Position based dynamics,” *J. Vis. Commun. Image Represent.*, vol. 18, no. 2, pp. 109–118, 2007.
- [9] M. Gazzola, L. H. Dudte, A. G. McCormick, and L. Mahadevan, “Forward and inverse problems in the mechanics of soft filaments,” *Royal Soc. Open Sci.*, vol. 5, no. 6, p. 171628, 2018.
- [10] N. Naughton, J. Sun, A. Tekinalp, T. Parthasarathy, G. Chowdhary, and M. Gazzola, “Elastica: A compliant mechanics environment for soft robotic control,” *IEEE Robotics Autom. Lett.*, vol. 6, no. 2, p. 3389–3396, 2021.
- [11] M. Bergou, B. Audoly, E. Vouga, M. Wardetzky, and E. Grinspun, “Discrete viscous threads,” *ACM Trans. on Graph.*, vol. 29, no. 4, p. Article 116, 2010.
- [12] M. Li, Z. Ferguson, T. Schneider, T. R. Langlois, D. Zorin, D. Panozzo, C. Jiang, and D. M. Kaufman, “Incremental potential contact: Intersection- and inversion-free, large-deformation dynamics,” *ACM Trans. on Graph.*, vol. 39, no. 4, p. Article 49, 2020.
- [13] Q. L. Lidec, W. Jallet, L. Montaut, I. Laptev, C. Schmid, and J. Carpentier, “Contact models in robotics: A comparative analysis,” *IEEE Trans. on Robotics*, vol. 40, p. 3716–3733, 2024.
- [14] M. Anitescu, “Optimization-based simulation of nonsmooth rigid multibody dynamics,” *Math. Program.*, vol. 105, no. 1, p. 113–143, 2006.
- [15] A. M. Castro, F. N. Permenter, and X. Han, “An unconstrained convex formulation of compliant contact,” *IEEE Trans. on Robotics*, vol. 39, no. 2, pp. 1301–1320, 2023.
- [16] A. Castro, X. Han, and J. Masterjohn, “Irrotational contact fields,” *arXiv:2312.03908*, 2023.
- [17] M. K. Jawed, A. Novelia, and O. M. O’Reilly, *A Primer on the Kinematics of Discrete Elastic Rods*. Cham, Switzerland: Springer, 2018.
- [18] J. Masterjohn, D. Guoy, J. Shepherd, and A. Castro, “Velocity level approximation of pressure field contact patches,” *IEEE Robotics Autom. Lett.*, vol. 7, no. 4, pp. 11 593–11 600, 2022.
- [19] J. L. Meriam and L. G. Kraige, *Engineering Mechanics: Statics*, 8th ed. Hoboken, NJ: Wiley, 2015.
- [20] J. E. Shigley, C. R. Mischke, and R. G. Budynas, *Mechanical Engineering Design*, 9th ed. New York, NY: McGraw-Hill, 2011.
- [21] K. Becker, C. Teeple, N. Charles, Y. Jung, D. Baum, J. C. Weaver, L. Mahadevan, and R. Wood, “Active entanglement enables stochastic, topological grasping,” *Proc. National Acad. Sci.*, vol. 119, no. 42, p. e2209819119, 2022.
- [22] J. Eom, S. Y. Yu, W. Kim, C. Park, K. Y. Lee, and K.-J. Cho, “MOGrip: Gripper for multiobject grasping in pick-and-place tasks using translational movements of fingers,” *Sci. Robotics*, vol. 9, no. 97, p. eado3939, 2024.