Cosserat-Rod-Based Dynamic Modeling of Soft Slender Robot Interacting With Environment

Lingxiao Xun, Gang Zheng, Alexandre Kruszewski

Cosserat-Rod Based Dynamic Modeling of Soft Slender Robot Interacting with Environment

Lingxiao Xun, Gang Zheng, *Senior Member, IEEE*, Alexandre Kruszewski

***Abstract*—Soft slender robots have attracted more and more research attentions in these years due to their continuity and compliance natures. However, mechanics modeling for soft robots interacting with environment is still an academic challenge because of the non-linearity of deformation and the non-smooth property of the contacts. In this work, starting from a piece-wise local strain field assumption, we propose a nonlinear dynamic model for soft robot via Cosserat rod theory using Newtonian mechanics which handles the frictional contact with environment and transfer them into the nonlinear complementary constraint (NCP) formulation. Moreover, we smooth both the contact and friction constraints in order to convert the inequality equations of NCP to the smooth equality equations. The proposed model allows us to compute the dynamic deformation and frictional contact force under common optimization framework in real time when the soft slender robot interacts with other rigid or soft bodies. In the end, the corresponding experiments are carried out which valid our proposed dynamic model.**

***Index Terms*—Soft slender robot, Cosserat, dynamics, frcition, contact, interaction, numerical optimization.**

I. Introduction

Soft robots possess the ability to adapt to complex external environments by utilizing material deformations and structural changes. This unique characteristic has led to an increasing interest among scholars in recent years regarding the modeling and control of soft robots [1] [2] [3]. Unlike traditional rigid structures, soft structures exhibit infinite degrees of freedom, making kinematics and dynamics analysis challenging. Additionally, in contrast to traditional continuum computational mechanics, which primarily emphasizes high simulation accuracy, modeling soft robots requires a balance between real-time performance and a certain level of accuracy to enable real-time extraction of model information during control.

The key challenge in soft robot modeling lies in dealing with its large deformation and computational complexity, which arise due to its high dimensionality. Thus, the pursuit of efficient and reliable models has become a crucial research direction in soft robotics. It is important to note that in most robotic applications, real-time interaction with the external world is essential, such as robot locomotion, manipulation tasks, and medical operations within the human body such as endoscopic surgery, aneurysm surgery, and cochlear implantation [4] [5] [6]. While in these applications, contact deformation and contact forces between objects are indispensable and important factors in the design and control of soft robots, thus the contact modeling plays a pivotal role in the overall analysis. However, while these aspects have been extensively studied in the field of rigid body robotics [7], they have not yet received sufficient attention in the realm of soft robotics. In the following review, we will discuss relevant works focusing on the mechanical and contact models of soft robots separately.

Lingxiao Xun, Gang Zheng, Alexandre Kruszewski are in Defrost team, Inria, university of Lille, Centrale Lille, CRIStAL - Centre de Recherche en Informatique Signal et Automatique de Lille - UMR 9189, France (e-mail: lingxiao.xun@inria.fr; gang.zheng@inria.fr; alexandre.kruszewski@centralelille.fr).

A. Soft slender robot modeling

The mechanical modeling of soft robots falls within the realm of continuum mechanics. When dealing with objects that possess complex geometric structures, the finite element method (FEM) is commonly employed, which has found widespread applications in engineering practice over the past few decades [8]. However, for soft robots with slender geometric structures, they can be treated as single beams or combinations of multi-body beams. As a result, the modeling of many soft and slender robots can be approximated using beam models with lower dimensions. In recent years, beam models have been extensively utilized in the modeling of various soft robots, yielding promising outcomes such as soft arms [9] or octopus robot [10].

Beam/rod theory is indeed a subclass of continuum mechanics that has found wide application in engineering mechanics simulations due to its universal structure and mathematical simplicity. The classical Euler-Bernoulli beam theory has been extensively used in the mechanical modeling of slender robot structures. For instance, in the study by Olson et al. [9], a quasi-static bending model based on the geometrically accurate Euler-Bernoulli formulation was developed, allowing for the prediction of design outcomes for various soft-arm designs. Another commonly employed beam model in soft robot modeling is the Timoshenko beam model, which takes into account the influence of shear strain [11] [12].

In many continuum models, constant curvature (CC) assumptions are frequently employed to approximate large deformations, simplifying the computational aspects of the mathematical models. This motivates the utilization of generalized elastic models, such as the Cosserat rod theory, in various applications [13] [14] [15]. Cosserat rods, which are geometrically nonlinear generalizations of Timoshenko-Reissner beams, offer the capability to simulate bending, torsion, shear, and tension in soft-body continuum robots. A detailed review on the statics, dynamics, and stability of Cosserat-based slender elastic rod continuum robots is provided in [16]. Building upon this foundation, new methods for solving Cosserat partial differential equations (PDEs) in continuous space have been proposed in [17] based on Newton-Euler dynamics.

In the context of soft slender robots, the work in [18] introduces a Cosserat-based piecewise constant strain model where the PDEs are transformed into an approximate weak form expressed as ordinary differential equations (ODEs). Another approach, presented in [19], utilizes a Cosserat discrete solution method based on strain nonlinear parameterization and Lagrangian dynamics. However, this technique can become computationally complex when modeling involves complex deformations like buckling behavior and local strain variations due to contact. To address this challenge, we will employ a piecewise local approximation of the strain field in our work.

The Cosserat model has also been applied in specific engineering scenarios. In [20], a Cosserat-based geometrically accurate dynamics model was developed for a soft manipulator actuated by cables. Another study [21] introduces a method that assembles heterogeneous, active, and passive Cosserat rods to simulate dynamic musculoskeletal structures capable of withstanding all deformation modes. These examples demonstrate the practical applications of the Cosserat model in the field of soft robotics.

B. Interaction with environment

The interaction between a soft robot and its environment is a fundamental aspect of robotics that involves how these flexible and deformable machines interact with and adapt to their surroundings [3]. Soft robots are distinct from traditional rigid robots in that they are made from flexible materials, which gives them unique advantages and challenges in their interactions with the environment [22]. In recent decades, outstanding work is being conducted in various domains. For instance, researchers are drawing inspiration from nature, particularly the octopus, to develop soft slender robots that exhibit remarkable dexterity and adaptability in manipulating objects and navigating constrained spaces [23]. Bio-hybrid soft robots, which integrate living cells or tissues with soft robotic structures, are also gaining attention for applications in tissue engineering and drug delivery [24]. Additionally, soft robot locomotion is a prominent area of research, with robots capable of crawling, slithering, or undulating like snakes, making them suitable for tasks such as search and rescue missions [25] [26]. In the medical field, soft robotic exoskeletons are being designed to provide support and assistance to individuals with mobility impairments or to enhance the physical abilities of healthy users, offering a new dimension to rehabilitation and physical augmentation [27] [28].

However, the interaction of soft robotics faces several challenges. One of the key challenges in this domain is contact modeling [29]. The flexible and deformable structures of soft robots present unique modeling complexities compared to their rigid counterparts. Unlike rigid robots, where contact interactions are well-understood and often simplified, soft robots require more intricate models to accurately predict how they interact with objects and surfaces. The challenge lies in developing contact models that can account for the intricate deformation of soft robot components upon contact, the distribution of forces, and the frictional effects. These models are crucial for applications like grasping, manipulation, and locomotion, where precise control and understanding of contact interactions are vital.

C. Frictional contact

The solution to contact problems in continuum mechanics has been a subject of considerable interest in recent decades, with a wide range of contact models being employed in engineering fields. However, solving the problem of multiple frictional contacts with large deformations remains challenging [30]. Although contact research for flexible systems has made advancements [31], the development of efficient and robust frictional contact algorithms still present open challenges.

The penalty function method is widely used in computational contact mechanics due to its simplicity and directness [32]. However, stiffness and stability issues persist despite recent progress [33]. Numerical optimization-based contact solution methods often involve linear complementary programming (LCP) or nonlinear complementary programming (NCP) formulations, which offer higher accuracy at the cost of increased computational complexity and the use of frictional approximations [34] [35]. LCP can be solved using relaxation methods like projected Gauss-Seidel (PGS) or direct methods such as Dantzig's pivoting algorithm or Lemke's algorithm. In the work of Stewart and Trinkle [36], the Coulomb friction cone is linearized, and Lemke's method is used to solve the resulting polygonal pyramid LCP, handling nonlinear forces. Subsequently, [37] proposed a method based on the Gaussian Seidel iterative inverse solution to LCP, which introduced a solution framework to the field of deformable solids [38] [39]. [40] introduced a contact model for asynchronously handling deformable solids by discretizing the contact barrier potential. For a comprehensive review of numerical methods for linear complementarity problems, we refer to [41].

In addition to LCP formulations, another common approach to solve contact problems is through Newton-type methods, which typically utilize the generalized projection operator and augmented Lagrangian to address contact constraints [42] [43] [44]. Recent works have applied these methods in various applications [45] [46] [47]. The complementarity problem in these works is addressed using non-smooth functions and solved from the perspective of a common Newton's method. This Newton-type optimization ensures quadratic convergence, although the number of iterations is typically higher compared to relaxation methods. For a comprehensive review of non-smooth methods applied to dynamics problems, we refer to [48].

D. Contributions and outline

Building on the related work and limitations discussed above, the contributions of our work in this paper are as follows:

(1). This work aims to establish a general framework for modeling the dynamics of soft slender robots via Cosserat rod theory to handle frictional contact. Precisely, we proposed a novel approach for modeling the dynamics of soft slender robots through piecewise linear interpolation for both strain field and contact field. The proposed model, profited by relative low dimension and a balance between accuracy, robustness

and computational complexity, intents to lay the foundation for real-time simulation and control works for soft slender robot under environment interaction.

(2). We proposed a new formulation for solving the dynamics with frictional contact constraint. In stead of solving LCP or NCP non-smoothly, we reformulate the contact constraint as the equality constraint using the general smooth NCP functions and extend this formulation to handle soft slender robots. This new formulation can be solved by common Newton direct method, which allows us naturally and efficiently to accommodate nonlinear friction models.

In summary, this paper is organized as follows: The problem statement is addressed in Section II. Section III provides introduces the configuration method of soft slender robots based on Lie groups and discusses the representation of contact surfaces. Section VI and VII discuss the contact dynamics of soft slender robots in both continuous and discrete forms. In Section IV, we present an overview of contact constraints in soft robotics. Subsequently, Section V introduces the reformulation of contact constraints via equality equations. In Section VIII, we propose the formulation of implicit dynamics which can be solved by common Newton method. Section X and XI present numerical simulation examples and experimental comparisons based on the proposed model, respectively. Finally, Section XII concludes this article.

II. Problem Statement

Due to the continuum nature of soft robot deriving a infinite DoF, it is difficult or impossible to get its exact dynamics handling contacts. Thus researchers usually use approximate technique to these models. Cosserat is a relative efficient way to approximate the dynamics of soft slender robot with a lower dimension than FEM on profiting the geometric nature of rod. Normally, if one can approximate the configuration of soft slender robot with a set of generalized coordinates $\boldsymbol{q} \in \mathbb{R}^n$, the dynamics without contacts could be described through the following second-order differential equations [49] [50]:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{K}\boldsymbol{q} = \boldsymbol{H}(\boldsymbol{q})\boldsymbol{\tau} + \boldsymbol{P}(\boldsymbol{q}), \qquad (1)$$

where $\boldsymbol{M}$ is the generalized mass matrix and $\boldsymbol{C}$ represents the contribution of viscosity and Coriolis acceleration. $\boldsymbol{K}$ is the stiffness matrix. $\boldsymbol{H}$ is the actuation matrix and $\boldsymbol{\tau}$ denotes the actuation force. $\boldsymbol{P}$ contains the contributions of the external force such as the gravity. When contact exists (see

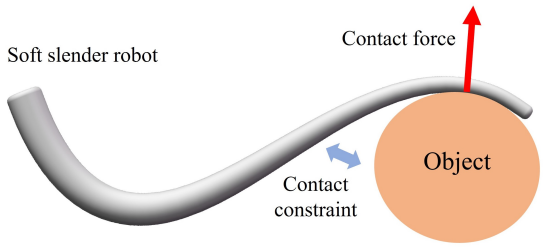


Fig. 1. Soft slender robot contacting with object.

Fig. 1), the contribution of contact constraint to the dynamic system (1) in generalized coordinates could be expressed by

TABLE I
Nomenclature and Definitions

Symbol	Unit	Definition
$[s, \beta]$	—	Coordinates of contact point.
t	s	Time.
$\boldsymbol{d}(s,\beta)$	m	Local distance between the midline of soft slender robot and the contact point.
$\boldsymbol{R}(s,t)$	—	Rotation matrix with respect to the inertail frame.
$\boldsymbol{p}(s,t)$	m	Position vector with respect to the inertail frame.
$\boldsymbol{g}(s,t)$	—	The configuration tensor of cross-section of soft slender robot
$\boldsymbol{g}_0(s,t)$	—	The transformation matrix from the soft slender robot's base frame to the inertial frame.
$\boldsymbol{g}_c(s,t)$	—	The configuration tensor of contact frame.
$\boldsymbol{g}_{bc}(s,t)$	—	the configuration tensor of contact frame with respect to body frame.
$\boldsymbol{g}_d(s,t)$	—	The configuration tensor of slave contact frame.
$\boldsymbol{g}_{cd}(s,t)$	—	the configuration tensor of slave contact frame with respect to master contact frame.
$\boldsymbol{\xi}(s,t)$	—	Strain in the body frame. i.e., $\boldsymbol{\xi} = [\boldsymbol{\kappa}^\top, \boldsymbol{\epsilon}^\top]^\top$.
$\boldsymbol{\eta}(s,t)$	—	Velocity in the body frame. i.e., $\boldsymbol{\eta} = [\boldsymbol{\omega}^\top, \boldsymbol{v}^\top]^\top$.
$\boldsymbol{\alpha}(t)$	—	Parameter vector of $\boldsymbol{g}_0$.
$\boldsymbol{\theta}(t)$	—	Parameter vector of strain field
$\widetilde{(\cdot)}$	—	Mapping from $\mathbb{R}^3$ to $so(3)$, e.g. $\widetilde{\boldsymbol{a}} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$.
$\widehat{(\cdot)}$	—	Mapping from $\mathbb{R}^6$ to $se(3)$, e.g. $\widehat{\boldsymbol{\xi}} = \begin{pmatrix} \widetilde{\boldsymbol{\kappa}} & \boldsymbol{\epsilon} \\ \boldsymbol{0} & 0 \end{pmatrix}$, $\widehat{\boldsymbol{\eta}} = \begin{pmatrix} \widetilde{\boldsymbol{\omega}} & \boldsymbol{v} \\ \boldsymbol{0} & 0 \end{pmatrix} \in se(3)$
$(\cdot)^\vee$	—	Mapping from $se(3)$ to $\mathbb{R}^6$.
ρ	kg/m^3	Density of material.
$R(s)$	m	Cross-sectional radius.
E	Pa	Young's modulus.
μ	—	Coefficient of friction.
$\mathbf{I}$	—	identity matrix.
$\boldsymbol{J}(s,t)$	—	Jacobian matrix of Cosserat kinematics.
$\mathrm{ad}_{(\cdot)}$	—	The adjoint map of the Lie algebra, e.g. $\mathrm{ad}_{\boldsymbol{\xi}} = \begin{pmatrix} \widetilde{\boldsymbol{\kappa}} & \boldsymbol{0}_{3\times3} \\ \widetilde{\boldsymbol{\epsilon}} & \widetilde{\boldsymbol{\kappa}} \end{pmatrix}$, $\mathrm{ad}_{\boldsymbol{\eta}} = \begin{pmatrix} \widetilde{\boldsymbol{\omega}} & \boldsymbol{0}_{3\times3} \\ \widetilde{\boldsymbol{v}} & \widetilde{\boldsymbol{\omega}} \end{pmatrix}$.
$\boldsymbol{\mathcal{M}}$	—	Cross-sectional mass matrix.
$\mathrm{Ad}_{\boldsymbol{g}(X)}$	—	The matrix transforming the velocity or acceleration twist from body frame to inertial frame, i.e., $\mathrm{Ad}_{\boldsymbol{g}(X)} = \begin{pmatrix} \boldsymbol{R} & \boldsymbol{0}_{3\times3} \\ \widetilde{\boldsymbol{p}}\boldsymbol{R} & \boldsymbol{R} \end{pmatrix} \in \mathbb{R}^{6\times6}$.
$\mathrm{Ad}^*_{\boldsymbol{g}(X)}$	—	The matrix transforming the wrench from contact frame to body frame, i.e., $\mathrm{Ad}^*_{\boldsymbol{g}(X)} = \begin{pmatrix} \boldsymbol{R} & \widetilde{\boldsymbol{p}}\boldsymbol{R} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{R} \end{pmatrix} \in \mathbb{R}^{6\times6}$.
n	—	Number of sections divided for strain field.
m	—	Number of sections divided for contact field.
$\delta_n(s,t)$	m	Normal gap of contact in contact frame.
$\boldsymbol{v}_t(s,t)$	m	Relative tangent velocity of contact in contact frame
$\Lambda_n(s,t)$	N/m	Normal contact load of collision in contact frame.
$\boldsymbol{\Lambda}_t(s,t)$	N/m	Tangent contact load of collision in contact frame.
$\boldsymbol{\Lambda}_c(s,t)$	N/m	Contact load of collision in contact frame.
$\boldsymbol{\Lambda}_a(s,t)$	N/m	Contact load of articulated constraint in contact frame.
$u_n(t)$	—	Slack variable of normal contact constraint.
$\boldsymbol{u}_t(t)$	—	Slack variable of tangent contact constraint.
$\boldsymbol{u}(t)$	—	Slack variable of contact constraint.
$\boldsymbol{\lambda}_c(t)$	—	Assembly vector of slack variable $\boldsymbol{u}$.
$\boldsymbol{\lambda}_f(t)$	—	Assembly vector of contact force of articulated constraints.
$\boldsymbol{\lambda}_a(t)$	—	Assembly vector of contact force of fixed constraints.
$\boldsymbol{G}_c(t)$	—	Collision contact constraints.
$\boldsymbol{G}_f(t)$	—	Fixed contact constraints.
$\boldsymbol{G}_a(t)$	—	Articulated contact constraints.

a set of Lagrange multipliers $\boldsymbol{\lambda} \in \mathbb{R}^m$ which physically represents the contact force. Note that the introduction of Lagrange multipliers introduces additional unknown variables to the system. Consequently, solving the system necessitates augmenting the equations with constraint equations that govern the contact behavior.

On account of these backgrounds, we would like to highlight the following issues at the beginning of this paper:
(1). How to derive the dynamics of soft slender robot under contact with limited degree of freedoms without losing local accuracy?
(2). How to compute the contribution of the contact force?
(3). How to efficiently deal with the contact constraints of dynamics?

Those questions will be investaged in the following sections.

III. Coordinates of Soft Slender Robot Based on Lie Group

Compared to traditional FEM, the Cosserat-based method can offer computational advantages for soft slender robot models. Due to its simplified geometric and kinematic description on Lie group, the Cosserat-based model can lead to reduced computational complexity and faster simulation, which bring advantage for robotic application. In the following we will define the configuration as well as the surface of soft slender robot by different coordinates, which will be used hereafter to deduce the contact dynamics of soft slender robot.

A. Backbone of soft slender robot

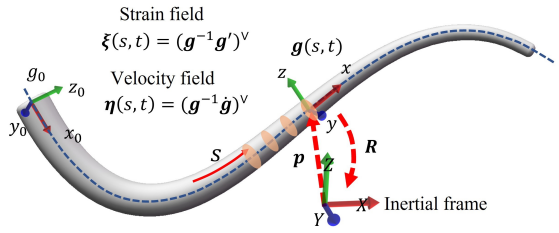


Fig. 2. Backbone of the soft slender robot.

In Cosserat rod theory, the soft slender robot is considered as a set of rigid cross-sections along its centerline, see Fig. 2. The spatial configuration of each cross-sections is defined in $SE(3)$ and is represented by the continuous configuration tensor $\boldsymbol{g}(s,t)$, which describes the position and orientation of each cross-section with respect to the inertial (global) frame. $\boldsymbol{g}(s,t)$ is defined by rotation matrix $\boldsymbol{R}(s,t) \in \mathrm{SO}(3)$ and position vector $\mathbf{p}(s,t) \in \mathbb{R}^3$, which represent the orientation and position of each cross-section of the rod with respect to global frame, as depicted in Fig. 2. For a length normalized soft slender robot, its configuration space is defined as follows:

$$\mathcal{U} = \{\boldsymbol{g} : s \in [0,1] \mapsto \boldsymbol{g}(s,t) \in \mathrm{SE}(3)\},\ \boldsymbol{g} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ \mathbf{0}^\top & 1 \end{bmatrix}$$

The space variation of the configuration can be defined with the vector fields $\boldsymbol{\xi}(s,t) = (\boldsymbol{g}^{-1}\boldsymbol{g}')^\vee$ which stand for the strain of the cross-section located at s in body frame, with $\boldsymbol{\xi}(s,t) = [\boldsymbol{\kappa}^\top\ \boldsymbol{\epsilon}^\top]^\top \in \mathbb{R}^6$, where $\boldsymbol{\kappa}(s,t)$ stands for the angular strain and $\boldsymbol{\epsilon}(s,t)$ represents the linear strain. By the above notations, the geometric model of soft slender robot is derived as below:

$$\boldsymbol{g}' = \boldsymbol{g}\widehat{\boldsymbol{\xi}} \tag{2}$$

with initial condition $\boldsymbol{g}(0) = \boldsymbol{g}_0$. As (2) indicates, if the initial configuration $\boldsymbol{g}_0$ and the overall strain space $\boldsymbol{\xi}(s)$ of the soft slender robot are known for an instant, the configuration space of the soft slender robot can be constructed by solving (2). Consequently, the configuration space $\mathcal{U}$ of the soft slender robot can be totally defined by a set of $\{\boldsymbol{g}_0, \boldsymbol{\xi}(s)\}$, defined as follows:

$$\mathcal{U} = SE(3) \times \mathbb{S} \tag{3}$$

where $SE(3)$ denotes the configuration space of $\boldsymbol{g}(0)$ and $\mathbb{S}$ denotes the strain field, with $\mathbb{S} = \{\boldsymbol{\xi} : s \in [0,1] \mapsto \boldsymbol{\xi} \in \mathbb{R}^6\}$.

As for velocity, the time variation of the configuration can be defined with the vector fields $\boldsymbol{\eta} = (\boldsymbol{g}^{-1}\dot{\boldsymbol{g}})^\vee$ which stand for the velocity twist of soft slender robot in local frame, with $\boldsymbol{\eta} = [\boldsymbol{w}^\top\ \boldsymbol{v}^\top]^\top \in \mathbb{R}^6$, where $\boldsymbol{w}$ stands for the angular velocity and $\boldsymbol{v}$ represents the linear velocity with respect to local frame.

B. Contact point on the surface of soft slender robot

In the previous subsection, we focused on the geometric representation of the soft slender robot backbone by employing the coordinate s to denote the position of each disc along its centerline. However, to describe the coordinates for every point on the surface $\partial\mathcal{C}$ of the soft slender robot, we introduce the second coordinate $\beta \in \mathbb{R}$ representing the rotation angle in radians of the contact point around the center point in the direction of x-axis. As illustrated in Fig. 3. Denoting $\boldsymbol{d}(s,\beta)$

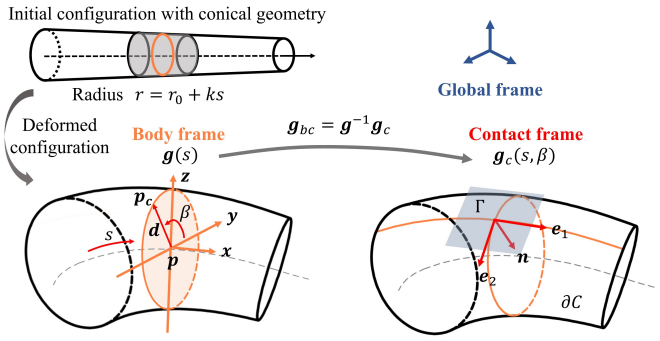


Fig. 3. Coordinates of surface of soft slender robot.

as the local distance vector defined in the body frame from the center-line of soft slender robot to the point on the surface, and using the coordinate $\boldsymbol{X} = [s,\beta]^\top$, we can precisely locate any point on the surface of the soft slender robot, which allows us to analyze the contact situation of the soft slender robot during various tasks and interactions. For any point on the surface, its position can be given by:

$$\boldsymbol{p}_c(s,\beta) = \boldsymbol{p}(s) + \boldsymbol{R}(s)\boldsymbol{d}(s,\beta) \tag{4}$$

Since we assume that the form of cross-section never changes during the deformation of soft slender robot, this vector

only depends on robot's original geometries of the soft slender robot. For example, for a conical soft slender robot with circular cross-section, as shown in Fig. 3, $\boldsymbol{d}(s,\beta) = \begin{bmatrix}0 & r\cos\beta & r\sin\beta\end{bmatrix}^\top$, where $r = r_0 + ks$ is the radius of the cross-section at length s. r_0 is the radius of the initial section and k is the radial gradient with respect to s.

C. Contact frame

Once we define the contact point on the surface of slender robot, the contact frame at this point can be represented by a contact plane which is tangent to the surface of soft slender robot, as shown in Fig. 3. This plane, noted as Γ, can be defined by a pair of covariant tangent vectors $\{\boldsymbol{\tau}_1, \boldsymbol{\tau}_2\}$, with:

$$\boldsymbol{\tau}_1 = \frac{\partial \boldsymbol{p}_c}{\partial s} = \frac{\partial \boldsymbol{p}}{\partial s} + \frac{\partial \boldsymbol{R}}{\partial s}\boldsymbol{d} + \boldsymbol{R}\frac{\partial \boldsymbol{d}}{\partial s} = \boldsymbol{R}(\boldsymbol{\epsilon} + \tilde{\boldsymbol{\kappa}}\boldsymbol{d} + \frac{\partial \boldsymbol{d}}{\partial s})$$

$$\boldsymbol{\tau}_2 = \frac{\partial \boldsymbol{p}_s}{\partial \beta} = \frac{\partial \boldsymbol{d}}{\partial \beta}$$

The unit normal vector of this tangent surface is then given by:

$$\boldsymbol{n} = \frac{\boldsymbol{\tau}_1 \times \boldsymbol{\tau}_2}{\|\boldsymbol{\tau}_1 \times \boldsymbol{\tau}_2\|} \tag{5}$$

Note that the basis $\{\boldsymbol{\tau}_1, \boldsymbol{\tau}_2, \boldsymbol{n}\}$ is not necessarily orthogonal in the deformed configuration, i.e., $\boldsymbol{\tau}_1^\top \boldsymbol{\tau}_2 \neq 0$, thus a new orthogonal basis can be defined from $\{\boldsymbol{\tau}_1, \boldsymbol{\tau}_2, \boldsymbol{n}\}$ by replacing $\boldsymbol{\tau}_1$ and $\boldsymbol{\tau}_2$ with:

$$\boldsymbol{e}_1 = \frac{\boldsymbol{\tau}_1}{\|\boldsymbol{\tau}_1\|}\ ,\ \boldsymbol{e}_2 = \frac{\boldsymbol{n} \times \boldsymbol{\tau}_1}{\boldsymbol{n} \times \|\boldsymbol{\tau}_1\|}$$

We can finally use the orthogonal basis $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{n}\}$ to define the contact frame. Note that these three unit vectors are all defined in the global frame, thus can be regarded as a rotation matrix noted as $\boldsymbol{R}_c$ which transfers the contact frame to global frame:

$$\boldsymbol{R}_c = \begin{bmatrix}\boldsymbol{n} & \boldsymbol{e}_1 & \boldsymbol{e}_2\end{bmatrix}$$

Based on this matrix, we can then define a configuration tensor $\boldsymbol{g}_c \in SE(3)$ which represents the position and orientation of the contact frame with respect to the global frame.

$$\boldsymbol{g}_c = \begin{bmatrix}\boldsymbol{R}_c & \boldsymbol{p}_c \\ \boldsymbol{0}^\top & 1\end{bmatrix}$$

IV. Contact Force and Contact Constraint

As depicted in the Fig. 4, contact constraints can arise in soft robotics, including fixed constraints, articulated constraints, and collision constraints. Among these, fixed constraints and articulated constraints can be represented using equalities and fall into the category of bilateral constraints. On the other hand, collision constraints are typically expressed using inequalities and fall into the category of unilateral constraints [51].

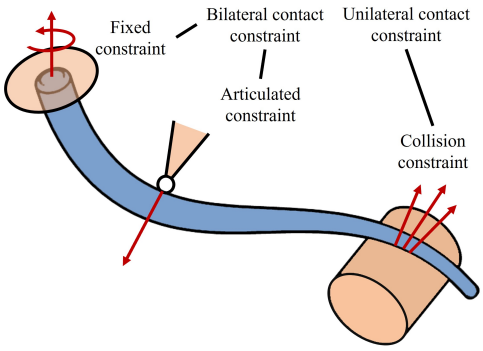


Fig. 4. Different contact constraints of soft slender robot.

A. Contact force

All contact constraints give rise to contact forces at the contact points. For any contact frame, we denote the contact force expressed in this contact frame as $\boldsymbol{\Lambda}_c \in \mathbb{R}^6$. Then, we denote $\boldsymbol{\Lambda}_c^*$ as the equivalent wrench with respect of body frame, which is generated by translating the distributed contact force onto the center of cross-section. There exists the following relationship between them:

$$\boldsymbol{\Lambda}_c^* = \mathrm{Ad}_{\boldsymbol{g}_{bc}}^* \boldsymbol{\Lambda}_c \tag{6}$$

where the coadjoint representation of the Lie group Ad^* allows changing the wrench from contact frame to body frame, with $\boldsymbol{g}_{bc} = \boldsymbol{g}^{-1}\boldsymbol{g}_c$, the configuration tensor of contact frame with respect to body frame.

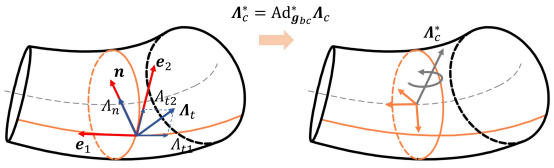


Fig. 5. Transformation of contact load from contact point to center line.

B. Bilateral contact constrain

Bilateral constraints are mathematical representations of kinematic pairs, such as spherical, or revolving joints. These constraints can be expressed as algebraic equations that impose restrictions on the relative positions of two bodies.

1) Fixed constraint: In engineering applications or some simulation scenarios, some parts of the soft slender robot will be totally fixed by the external environment. For example, the head section of the soft slender robot is fixed on a mobile base or some of its parts are hinged to external objects like truss system [52]. These persistent constraints can be regarded as bilateral constraints. Specifically, the satisfaction of bilateral constraints can be realized by adding Lagrangian multipliers in the system. For any points $s \in [0,1]$ along the centerline of the soft slender robot, if we want to fix some sections of soft slender robot to the desired configuration, that is, given a constrained configuration matrix $\boldsymbol{g}_f(s)$, then the bilateral constraint can be expressed as:

$$\boldsymbol{G}_f(s) \in \mathbb{R}^6 = \left(\log(\boldsymbol{g}_f^{-1}\boldsymbol{g})\right)^\vee = \boldsymbol{0}$$

which is equivalent to the following bilateral constriant $\boldsymbol{g}_f(s) - \boldsymbol{g}(s) = \mathbf{0}$, stated in the matrix form.

2) Articulated constraint: Articulatd constraints in soft slender robots impose limitations on the translation of contact points while permitting rotational freedom at the cross-sections of these points. Consequently, only the three translational degrees of freedom are constrained by articulated constraints. For $s \in [0, 1]$, denoting the desired position of the joint as $\boldsymbol{p}_a(s)$ with respect to the global frame, the articulated constraint is given by the following equation:

$$\mathbb{R}^3 \ni \boldsymbol{G}_a(s) = \boldsymbol{p}_a - \mathbf{U}\boldsymbol{g}_c(s, \beta)\mathbf{A} = \mathbf{0}$$

where $\mathbf{U} = [1\ 1\ 1\ 0]$ and $\mathbf{A} = [0\ 0\ 0\ 1]^\top$. Note that for the articulated constraint, the joint can only generate the force but not the moment, which means that $\boldsymbol{\Lambda}_c$ is not full filled. Denoting the generated force as $\boldsymbol{\Lambda}_a \in \mathbb{R}^3$, (6) should be rewritten as:

$$\boldsymbol{\Lambda}_c^* = \mathrm{Ad}^*_{\boldsymbol{g}_{bc}} \mathbf{B}^\top \boldsymbol{\Lambda}_a$$

where $\mathbf{B} = [\mathbf{0}_{3\times3}\ , \mathbf{I}_{3\times3}]$.

C. Unilateral contact constrain

In the field of contact mechanics, the unilateral constraint refers to a mechanical constraint that prohibits any form of penetration or interpenetration between two bodies, whether they are rigid or flexible. This constraint ensures that the tangential contact forces between the bodies adhere to the contact law, which governs the interaction between contacting surfaces.

For collision constraint, the generated force is composed by the normal contact force and tangent friction force, i.e., $\boldsymbol{\Lambda}_c = [\mathbf{0}_{1\times3}, \Lambda_n, \Lambda_{t1}, \Lambda_{t2}]^\top$, where Λ_n, Λ_{t1} and Λ_{t2} denote respectively the normal contact force in the direction of $\boldsymbol{n}$, the friction force in the direction of $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$. Thus (6) should be rewritten as:

$$\boldsymbol{\Lambda}_c^* = \mathrm{Ad}^*_{\boldsymbol{g}_{bc}}(\mathbf{B}_n^\top \Lambda_n + \mathbf{B}_t^\top \boldsymbol{\Lambda}_t) \tag{7}$$

where $\mathbf{B}_n = [\mathbf{0}_{1\times3}, 1, \mathbf{0}_{1\times2}]$, $\mathbf{B}_t = [\mathbf{0}_{2\times4}, \mathbf{I}_{2\times2}]$ and $\boldsymbol{\Lambda}_t = [\Lambda_{t1}, \Lambda_{t2}]^\top$.

In accordance with the problem statement, the collision constraints encompass a set of inequalities that involve contact gaps and contact sliding velocities. Hence, our initial objective is to elucidate how to define contact gaps and contact sliding velocities within the framework of Cosserat rod theory.

As shown in Fig. 6, we designate the contact frame of the soft slender robot as $\boldsymbol{g}_c$, serving as the master contact frame. We also denote a known slave contact frame $\boldsymbol{g}_d$, located on the surface $\partial\mathcal{D}$ of another body $\mathcal{D}$. These two frames constitute a contact pair. In order to represent the relative position and orientation of slave contact frame on $\partial\mathcal{D}$ with respect to contact frame $\boldsymbol{g}_c$ on $\partial\mathcal{C}$, we employ a tensor denoted as $\boldsymbol{g}_{cd} = \boldsymbol{g}_c^{-1}\boldsymbol{g}_d$.

The normal gap is the relative position vector projected on the opposite direction of normal vector $\boldsymbol{n}$ and can be obtained from the relative configuration tensor of contact via the following relation:.

$$\delta_n = -\mathbf{P}\boldsymbol{g}_{cd}\mathbf{A} \tag{8}$$

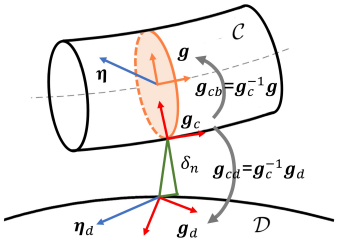


Fig. 6. Geometries of the contact pair.

with $\mathbf{P} = [1\ 0\ 0\ 0]$ and $\mathbf{A} = [0\ 0\ 0\ 1]^\top$.

The normal force and normal gap should always be non-negative, i.e., $\delta_n \geq 0$, $\Lambda_n \geq 0$. This principle imposes a constraint on the occurrence of mutual penetration among objects in contact, thereby prohibiting their ability to traverse another one. The magnitude of the normal contact force is invariably non-negative, and contact forces are solely generated upon contact establishment.

The friction force is constrained within a convex set $\mathcal{C}_f$ corresponding to the friction cone, such that $\boldsymbol{\Lambda}_t \in \mathcal{C}_f$. In our work we use the Coulomb's law to model the friction. The section of the Coulomb's friction cone, i.e., the disk $\mathcal{D}(\mu\Lambda_n)$ is defined by

$$\mathcal{D}(\mu\Lambda_n) = \{\Lambda_t | \mu\Lambda_n - \Lambda_t \geq 0\} \tag{9}$$

with μ being the coefficient of friction.

Denoting the velocity of backbone of soft slender robot with respect to body frame as $\boldsymbol{\eta}$ and supposing the velocity of the slave contact point on object $\mathcal{D}$ with respect to the slave contact frame $\boldsymbol{g}_d$ is $\boldsymbol{\eta}_d$. The slip velocity is defined as the projection of the relative contact velocity onto the tangential plane at the contact point. Thus, the relative velocity twist of contact point with respect to contact frame $\boldsymbol{g}_c$ of $\partial\mathcal{C}$ can be then given by:

$$\boldsymbol{\eta}_c = \mathrm{Ad}_{\boldsymbol{g}_{cb}}\boldsymbol{\eta} - \mathrm{Ad}_{\boldsymbol{g}_{cd}}\boldsymbol{\eta}_d \tag{10}$$

where $\boldsymbol{g}_{cb} = \boldsymbol{g}_c^{-1}\boldsymbol{g}$ and the adjoint representation of the Lie group $\mathrm{Ad}_{\boldsymbol{g}_i^{-1}\boldsymbol{g}_j}$ allows changing the velocity twist from frame j to frame i. Consequently the slip tangent velocity $\boldsymbol{v}_t \in \mathbb{R}^2$ on $\partial\mathcal{C}$ with respect to contact frame equals to:

$$\boldsymbol{v}_t = \mathbf{B}_t\boldsymbol{\eta}_c \tag{11}$$

where $\mathbf{B}_t$ was defined in (7).

The Coulomb's friction cone contains two different states (stick $v_t = 0$ or slide $v_t \neq 0$) of a contact point which is given by:

$$\begin{cases} \text{Stick}: \mu\Lambda_n - \|\boldsymbol{\Lambda}_t\| > 0\ ,\ \boldsymbol{v}_t = \mathbf{0} & \text{(12a)} \\ \text{Silde}: \boldsymbol{\Lambda}_t = -\mu\Lambda_n\mathbf{T}\ ,\ \boldsymbol{v}_t \neq \mathbf{0} & \text{(12b)} \end{cases}$$

Here $\mathbf{T}$ is an identity vector which indicates the tangential sliding direction with $\mathbf{T} = \frac{\boldsymbol{v}_t}{\|\boldsymbol{v}_t\|}$.

There are two main approaches to model the above unilateral constraints. The first is based on the regularization of contact constraint such as penalty methods, while the second is based on non-smooth contact dynamics, modeling the contact constraint as the exact inequalities.

Within penalty methods, the constraint is characterized as soft, allowing for the possibility of penetration between two contacting bodies. In the event of geometric penetration between these bodies, a penalty potential energy term is incorporated into the studied system, wherein its magnitude is directly proportional to the degree of penetration. The penalty potential will generate the resisted forces in normal and tangential directions. Penalty formulations in constraint-based simulations offer the advantage of being unconstrained and relatively straightforward to simulate and differentiate. They are particularly useful when it comes to stability, as implicit integration methods allow for stable simulations across a wide range of stiffness values. However, one limitation of penalty methods is that they allow for small constraint violations. While minor violations of normal constraints may not be visually apparent, softening the static friction constraint can lead to undesirable artifacts in certain scenarios. For example, when simulating a heavy object resting on an inclined surface under static friction, if the simulation runs for a long duration, the introduction of tangential slipping due to softened stick constraints can eventually become visually noticeable. Therefore, in our work, we did not utilize this method, but instead, we consistently treated contact as a hard constraint throughout, which will be introduced in the next subsection.

D. NCP formulation

As discussed in the previous subsection, the precise approach involves formulating the contact as a hard constraint. In the case of unilateral constraints, this formulation is commonly expressed as nonlinear complementary conditions.

1) Normal contact constraint: the NCP of normal contact is given by the Signorini's condition [53]:

$$0 \leq \delta_n \perp \Lambda_n \geq 0 \tag{13}$$

where the symbol "$\perp$" denotes that the product of δ_n and Λ_n equals to zero.

2) Tangential contact constraint: The tangential contact constraint is about friction. We first denote $\boldsymbol{\Lambda}_t \in \mathbb{R}^2$ as the friction force of any contact point in contact frame on the surface $\partial\mathcal{C}$. $\boldsymbol{v}_t \in \mathbb{R}^2$ stands for the slip velocity. Then according to the maximal dissipation principle [54], the instantaneous power of friction force at this contact point given by $\boldsymbol{\Lambda}_t^\top \boldsymbol{v}_t$ should always takes the minimum within the limit of friction cone, i.e., $\boldsymbol{\Lambda}_t$ should satisfy the following minimization:

$$\begin{aligned} \boldsymbol{\Lambda}_t = \operatorname{argmin} \quad & \boldsymbol{\Lambda}_t^\top \boldsymbol{v}_t \\ \text{subject to} \quad & \mu\Lambda_n - \|\boldsymbol{\Lambda_t}\| \geq 0 \end{aligned}$$

where μ is the coefficient of friction. The first order Karush-Kuhn-Tucker conditions of above minimization is then given by:

$$\boldsymbol{v}_t + \lambda_{sl} \nabla \|\boldsymbol{\Lambda}_t\| = \mathbf{0} \tag{14}$$

$$0 \leq \lambda_{sl} \perp \mu\Lambda_n - \|\boldsymbol{\Lambda}_t\| \geq 0 \tag{15}$$

with λ_{sl} the Lagrange multiplier. In fact, the above two equations can be further simplified. First, one can derive $\nabla\|\boldsymbol{\Lambda}_t\| = \boldsymbol{\Lambda}_t/\|\boldsymbol{\Lambda}_t\|$. Subsequently, taking this equation into (14) we can get $\boldsymbol{v}_t = -\lambda_{sl}\boldsymbol{\Lambda}_t/\|\boldsymbol{\Lambda}_t\|$. Then taking the norm of both two sides derives $\lambda_{sl} = \|\boldsymbol{v}_t\|$. Finally, (14) and (15) are transferred to the following complementary conditions where the Lagrange multiplier is eliminated:

$$\|\boldsymbol{\Lambda}_t\|\boldsymbol{v}_t + \|\boldsymbol{v}_t\|\boldsymbol{\Lambda}_t = \mathbf{0} \tag{16}$$

$$0 \leq \|\boldsymbol{v}_t\| \perp \mu\Lambda_n - \|\boldsymbol{\Lambda}_t\| \geq 0 \tag{17}$$

However, it is clear that the coupling between normal and frictional complementarity is non-convex and non-smooth. In the next section, we propose a smoothing method to solve this issue.

V. Reformulation of the contact constraint

Mathematically, the nonlinear complementary constraint (13) can be linearized to form a linear complementary problem (LCP). However, the tangential constraints (16) and (17) cannot be directly formulated as an LCP due to the nonlinearity of the friction cone. A common approach is to approximate the Coulomb's friction cone with a pyramid. This allows for the utilization of LCP solvers such as Lemke's algorithm [36] and Gauss-Seidel method [38]. Despite some advancements made in these methods specifically for simulating contact dynamics, the computational efficiency remains a challenge. In order to ensure both accuracy and efficiency, we propose a novel approach to reformulate the contact constraint as a smooth equality constraint. This new formulation aims to strike a balance between computational accuracy and efficiency.

A. NCP-function and slack variable

The idea is to convert the NCP into a nonlinear function (NCP-function) in order to formulate the NCP as the equality constraint. Since the work by Mangasarian [55] it has been well known that by means of a suitable function $\phi : \mathbb{R}^2 \to \mathbb{R}$, the complementary condition $a \geq 0, b \geq 0, ab = 0$ can be transferred to an equivalent nonlinear equation: $\phi(a, b) = 0$. This technique has drawn interest for solving complementary programming in the field of mathematics in these years. Our work is mainly based on this idea, using a slack variable to define both a and b.

Denoting $u \in \mathbb{R}$ as the slack variable and $D(u)$ as the Heaviside function with $D(u) = (\operatorname{sgn}(u) + 1)/2$. The complementary condition of a, b is equivalent to the following definition:

$$a = D(u)u \ , \ b = D^*(u)u \tag{18}$$

with $D^*(u) = -D(-u)$. It is evident that with the aforementioned definitions, the complementary condition holds for any $u \in \mathbb{R}$. Consequently, the nonlinear complementary constraint is now transformed into a manifold defined by (18) with respect to the slack variable u.

B. Reformulation of collision constraint

1) Normal contact: Returning to the topic of normal contact, we can utilize a variable $u_n \in \mathbb{R}$ to represent both the normal contact force Λ_n and the normal gap δ_n at each contact point as the following definitions:

$$\Lambda_n = D(u_n)u_n \ , \ \delta_n = D^*(u_n)u_n \tag{19}$$

2) Friction contact: The equality constraint (16) means that $\boldsymbol{v}_t$ and $\boldsymbol{\Lambda}_t$ are always parallel and opposite, it is possible to define them by the parametric equation. At this aim, we first represent them in the polar coordinates, i.e., (16) equals to the following representation:

$$\boldsymbol{v}_t = \rho_1 \begin{bmatrix} \sin\theta \\ \cos\theta \end{bmatrix} \ , \ \boldsymbol{\Lambda}_t = -\rho_2 \begin{bmatrix} \sin\theta \\ \cos\theta \end{bmatrix} \tag{20}$$

where $\rho_1 \geq 0$ and $\rho_2 \geq 0$. Therefore, we can rewrite the complemantarity (17) as:

$$0 \leq \rho_1 \perp \mu\Lambda_n - \rho_2 \geq 0$$

which is equivalent to: $\forall u_t > 0$,

$$\rho_1 = f(u_t - \mu\Lambda_n) \ , \ \rho_2 = \mu\Lambda_n - f(\mu\Lambda_n - u_t) \tag{21}$$

where function $f(x)$ denotes $D(x)x$. Then by taking the above representation into (20) one can get: $\forall u_t > 0$,

$$\boldsymbol{v}_t = f(1 - \frac{\mu\Lambda_n}{u_t}) \begin{bmatrix} u_t \sin\theta \\ u_t \cos\theta \end{bmatrix} \tag{22}$$

$$\boldsymbol{\Lambda}_t = \left(-\frac{\mu\Lambda_n}{u_t} + f(\frac{\mu\Lambda_n}{u_t} - 1) \right) \begin{bmatrix} u_t \sin\theta \\ u_t \cos\theta \end{bmatrix} \tag{23}$$

Note that the vector part $\begin{bmatrix} u_t \sin\theta & u_t \cos\theta \end{bmatrix}^\top$ represents an arbitrary vector defined in the polar coordinates, thus it can be rewritten in the Cartesian coordinates as $\boldsymbol{u}_t \in \mathbb{R}^2$ with $u_t = \|\boldsymbol{u}_t\|$. By doing so, we can describe (22) and (23) again in the Cartesian coordinates:

$$\boldsymbol{\Lambda}_t = W(\boldsymbol{u}_t)\boldsymbol{u}_t \ , \ \boldsymbol{v}_t = W^*(\boldsymbol{u}_t)\boldsymbol{u}_t \tag{24}$$

where

$$W(\boldsymbol{u}_t) = -\frac{\mu\Lambda_n}{u_t} + f(\frac{\mu\Lambda_n}{u_t} - 1) \ , \ W^*(\boldsymbol{u}_t) = f(1 - \frac{\mu\Lambda_n}{u_t})$$

Remark 1: Note that (24) has no definition when $\boldsymbol{u}_t = \mathbf{0}_{2\times 1}$, thus we define $W(0,0) = W^*(0,0) = \mathbf{0}_{2\times 1}$ and $\nabla W(0,0) = \frac{\mu\Lambda_n}{l}\mathbf{I}_{2\times 2}$, $\nabla W^*(0,0) = \mathbf{0}_{2\times 2}$.

For convenience purposes, we define the slack vector $\boldsymbol{u} = [u_n \ \boldsymbol{u}_t^\top]^\top$ to cover both normal constraint and friction constraint. Based on the previous expression of contact forces, we can establish the relationship between the relaxation vector and contact forces using the following equation.

$$\boldsymbol{\Lambda}_c = (\mathbf{B}_n^\top D\mathbf{C}_n + \mathbf{B}_t^\top W\mathbf{C}_t)\boldsymbol{u} \tag{25}$$

where $\mathbf{C}_n = [1 \ \mathbf{0}_{1\times 2}]$ and $\mathbf{C}_t = [\mathbf{0}_{2\times 1} \ \mathbf{I}_{2\times 2}]$. $\mathbf{B}_n$ and $\mathbf{B}_t$ was defined in (7). Similarly, the constraints of normal gap and the slip tangent velocity are written as:

$$\delta_n = D^*\mathbf{C}_n\boldsymbol{u} \ , \ \boldsymbol{v}_t = W^*\mathbf{C}_t\boldsymbol{u} \tag{26}$$

C. Smooth reformulation

Nevertheless, the aforementioned formulation includes the Heaviside function, which is known to be discontinuous. Additionally, the function $f(x)$ in (21) possesses C^0 continuity, meaning that it is only continuous but not necessarily to be smooth. As a consequence, during the numerical solution process of dynamics, the presence of these functions leads to sudden changes, requiring additional iterations to achieve a solution. To address this concern, we propose a smooth approximation in our work, whereby the Heaviside function in (18) is replaced by some well-known smooth functions [56], which will be discussed hereafter.

1) Sigmoid function: The first smooth function we proposed is based on the sigmoid function defined as follows:

$$D(x) = \frac{1}{1 + e^{-cx}} \ , \ c > 0 \tag{27}$$

The parameter c plays a crucial role in determining the level of smoothing in (27). As c approaches infinity, (27) progressively converges to the Heaviside function, as depicted in Fig. 7. It is important to note that this approximation of complementarity is not perfectly accurate in the vicinity of the critical state. In situations where c is chosen to be extremely small, significant errors may occur when x approaches zero, potentially leading to penetrations of contact. In addition, the friction behavior no longer adheres strictly to Coulomb's friction law. In practical simulation computations, it is essential to meticulously select the parameter c to strike a balance between accuracy and the numerical robustness of the calculations.

2) Trigonometric function: The second smooth function we proposed is based on trigonometric function as follows:

$$D(x) = \begin{cases} 0 \ , \ x < 0 \\ \dfrac{1 - \cos(wx)}{2} \ , 0 \leq \ x < \dfrac{\pi}{w} \\ 1 \ , \ x \geq \dfrac{\pi}{w} \end{cases} \tag{28}$$

where w is a positive parameter.

In contrast to the sigmoid function, the system described by (28) rigorously fulfills the complementarity condition. This is attributed to their positive definiteness in the positive half-axis of x, while being identically zero in the negative half-axis of x. In this case, the parameter w solely affects the robustness of numerical optimization but does not impact the dynamics of the soft robot.

By utilizing the aforementioned function as a substitute for the impact function, we obtain the smoothed contact constraints, as depicted in Fig. 8. At the critical state between sticking and sliding, a sudden change in gradient can be observed in the left image of Fig. 8 which is non smooth, whereas the gradient in the right image of Fig. 8 exhibits a continuous transition using trigonometric function.

VI. Cosserat dynamics with contact

After addressing all types of contact constraints, this section will primarily focus on the kinematics and dynamics of soft robots under contact constraints.

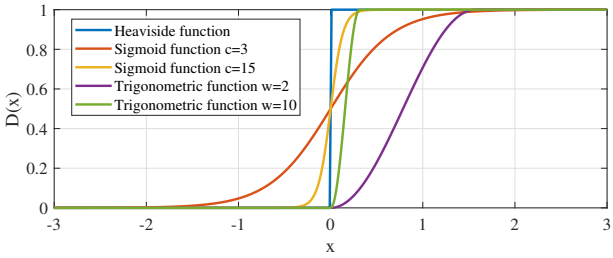


Fig. 7. Proposed function for the reformulation of complementary constraint.

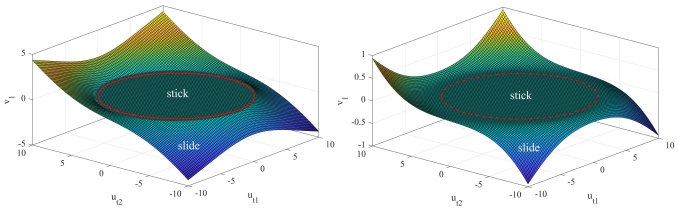


Fig. 8. The manifold of tangent velocity with respect of slack variable. The figure illustrates a comparison between the non-smooth case utilizing the Heaviside function (on the left) and the smooth case employing a trigonometric function (on the right).

A. Continuous kinematics

It was derived in [18] that the continuous kinematics model is represented by the following two differential equations:

$$\boldsymbol{g}' = \boldsymbol{g}\widehat{\boldsymbol{\xi}} \tag{29}$$

$$\boldsymbol{\eta}' = \dot{\boldsymbol{\xi}} - \mathrm{ad}_{\boldsymbol{\xi}}\,\boldsymbol{\eta} \tag{30}$$

The significance of the above kinematic equations lies in the fact that the complete velocity field of a soft slender robot can be determined from the strain field and the rate of change of the strain field over time. These equations establish a fundamental relationship between the deformation of the soft slender robot and its corresponding motion.

B. Strong form of dynamics

1) PDE: The corresponding dynamic model written in the body frame was derived in [57] and [19] as the following partial differential equations (PDEs):

$$\begin{aligned}\boldsymbol{\mathcal{M}}\dot{\boldsymbol{\eta}} - ad_{\boldsymbol{\eta}}^{\top}\boldsymbol{\mathcal{M}}\boldsymbol{\eta} = \boldsymbol{\Lambda}_i' - \mathrm{ad}_{\boldsymbol{\xi}}^{\top}\,\boldsymbol{\Lambda}_i + \boldsymbol{\Lambda}_e \\ + \mathrm{Ad}^{*}_{\boldsymbol{g}_{bc}}\boldsymbol{\Lambda}_c + \boldsymbol{\Lambda}_f + \mathrm{Ad}^{*}_{\boldsymbol{g}_{bc}}\mathbf{B}^{\top}\boldsymbol{\Lambda}_a\end{aligned} \tag{31}$$

satisfying the following boundary conditions:

$$\begin{aligned}\boldsymbol{\Lambda}_i(0,t) = -\boldsymbol{\Lambda}_0(t)\ ,\ \text{or}\ \boldsymbol{g}(0,t) = \boldsymbol{g}_0(t) \\ \boldsymbol{\Lambda}_i(1,t) = \boldsymbol{\Lambda}_1(t)\ ,\ \text{or}\ \boldsymbol{g}(1,t) = \boldsymbol{g}_1(t)\end{aligned} \tag{32}$$

where $\boldsymbol{\mathcal{M}} \in \mathbb{R}^{6\times 6}$ is the tensor of mass linear density along central axis. $\boldsymbol{\Lambda}_i \in \mathbb{R}^6$ is the elastic internal wrench. $\boldsymbol{\Lambda}_c \in \mathbb{R}^6$ is the contact force produced by collision. $\boldsymbol{\Lambda}_f \in \mathbb{R}^6$ is the contact force produced by fixed constraints and $\boldsymbol{\Lambda}_a \in \mathbb{R}^3$ is the contact force produced by articulated constraints. $\boldsymbol{\Lambda}_e \in \mathbb{R}^6$ are the distributed external wrench applied on the soft slender robot. One contribution of $\boldsymbol{\Lambda}_e$ comes from the gravity which can be given by $\boldsymbol{\Lambda}_e = \boldsymbol{\mathcal{M}}\mathrm{Ad}_{\boldsymbol{g}}^{-1}\boldsymbol{\mathcal{G}}$, where the inverse of adjoint representation of the Lie group is used to transform twists from the global frame to the body frame. $\boldsymbol{\mathcal{G}}$ is the gravity acceleration twist w.r.t. the global frame $\boldsymbol{\mathcal{G}} = [\mathbf{0}_{1\times 5}\ -9.81]^{\top}$.

2) Constitutive equation: The internal wrench of the soft slender robot is determined by strain and time derivative of strain. Its constitutive equation based on Kelvin-Voigt model is as follows:

$$\boldsymbol{\Lambda}_i = \mathcal{K}(\boldsymbol{\xi} - \boldsymbol{\xi}_0) + \mathcal{D}\dot{\boldsymbol{\xi}} \tag{33}$$

where $\mathcal{K}$ and $\mathcal{D}$ are screw stiffness matrix and viscosity matrices [50], determined by the dimension of soft slender robot and its material. $\boldsymbol{\xi}_0 = [0\ 0\ 0\ 1\ 0\ 0]^{\top}$. The viscosity (damping) matrix of a soft robot will influence the robot's dynamic performance. Greater damping leads to a shorter stabilization time for the robot.

3) Constraints: If all contact constraints introduced in section IV are considered, the dynamics must also satisfy the following constraint conditions:

$$\boldsymbol{G}_f = \mathbf{0} \tag{34}$$

$$\boldsymbol{G}_a = \mathbf{0} \tag{35}$$

$$\boldsymbol{\Lambda}_c - (\mathbf{B}_n^{\top} D\mathbf{C}_n + \mathbf{B}_t^{\top} W\mathbf{C}_t)\boldsymbol{u} = \mathbf{0} \tag{36}$$

$$\mathbf{P}\boldsymbol{g}_{cd}\mathbf{A} + D^{*}\mathbf{C}_n\boldsymbol{u} = 0 \tag{37}$$

$$\mathrm{Ad}_{\boldsymbol{g}_{cb}}\boldsymbol{\eta} - \mathrm{Ad}_{\boldsymbol{g}_{cd}}\boldsymbol{\eta}_d - W^{*}\mathbf{C}_t\boldsymbol{u} = \mathbf{0} \tag{38}$$

(34) and (35) contains all bilateral constraints and (36)-(38) denote the reformulation of the unilateral constraint based on the slack variable. (36) represents the constraint of contact force. (37) and (38) describe the constraint on normal contact gap and the constraint on tangential sliding respectively. The unknown variables in the partial differential-algebraic system formed by the dynamics and constraints discussed above (29)-(38) encompass the initial pose tensor $\boldsymbol{g}_0(t)$ at the starting end, the strain field $\boldsymbol{\xi}(s,t)$, the contact force $\boldsymbol{\Lambda}_f(s,t)$, $\boldsymbol{\Lambda}_a(s,t)$ and the contact field represented by slack variables $\boldsymbol{u}(s,t)$. In order to facilitate the solution of this partial differential-algebraic system, we aim to derive its corresponding weak form. By introducing suitable test functions $\boldsymbol{\Phi}_{\boldsymbol{\xi}}(s)$ and $\boldsymbol{\Phi}_u(s)$, the weak form of (31) can be expressed as the integration through the backbone of soft slender robot:

$$\begin{aligned}\int_0^1 \boldsymbol{\Phi}_{\boldsymbol{\xi}}^{\top}(\boldsymbol{\mathcal{M}}\dot{\boldsymbol{\eta}} - ad_{\boldsymbol{\eta}}^{\top}\boldsymbol{\mathcal{M}}\boldsymbol{\eta} - \boldsymbol{\Lambda}_i' + \mathrm{ad}_{\boldsymbol{\xi}}^{\top}\,\boldsymbol{\Lambda}_i - \boldsymbol{\Lambda}_e \\ - \mathrm{Ad}^{*}_{\boldsymbol{g}_{bc}}\boldsymbol{\Lambda}_c - \boldsymbol{\Lambda}_f - \mathrm{Ad}^{*}_{\boldsymbol{g}_{bc}}\mathbf{B}^{\top}\boldsymbol{\Lambda}_a)\mathrm{d}s = 0\end{aligned} \tag{39}$$

and the weak form of constraints (37) and (38) are given by:

$$\int_0^1 \boldsymbol{\Phi}_{\boldsymbol{u}}^{\top}\begin{bmatrix}\mathbf{P}\boldsymbol{g}_{cd}\mathbf{A} - D^{*}\mathbf{C}_n\boldsymbol{u} \\ \mathrm{Ad}_{\boldsymbol{g}_{cb}}\boldsymbol{\eta} - \mathrm{Ad}_{\boldsymbol{g}_{cd}}\boldsymbol{\eta}_d - W^{*}\mathbf{C}_t\boldsymbol{u}\end{bmatrix}\mathrm{d}s = 0 \tag{40}$$

As for the bilateral constrain (34) and (35), their weak forms retain the same representation as the strong form since they are normally discretely located on the soft slender robot. Applying discretization techniques, the above integral equations of weak forms are further approximated. This discretization process involves dividing the spatial space of robot's backbone into discrete nodes, thereby transforming the weak form into a more tractable differential-algebraic system.

VII. Spatial discritization of strain and contact

A. Spatial interpolation of soft slender robot

In the subsequent discussion, we propose a redefinition of the robot's configuration space by parameterizing $\boldsymbol{g}_0(t)$ and $\boldsymbol{\xi}(s,t)$. $\boldsymbol{g}_0$ is composed of a rotation matrix $\boldsymbol{R}_0$ and a position vector $\boldsymbol{p}_0$. In our approach, we use the exponential map to define the rotation matrix $\boldsymbol{R}_0$ by a vector $\boldsymbol{\phi} \in \mathbb{R}^3$:

$$\boldsymbol{R}_0(t) = \exp \tilde{\boldsymbol{\phi}}(t) \ , \ \boldsymbol{\phi}(t) \in \mathbb{R}^3 \tag{41}$$

Thus the configuration matrix $\boldsymbol{g}_0(t)$ is parameterized by the vector $\boldsymbol{\phi}(t)$ and $\boldsymbol{p}_0(t)$. We use vector $\boldsymbol{\alpha}(t) \in \mathbb{R}^6$ to denote their combination, i.e., $\boldsymbol{\alpha}(t) = [\boldsymbol{\phi}^\top(t) \ , \ \boldsymbol{p}_0^\top(t)]^\top$.

For parameterizing the strain field, we adopt the piecewise linear strain (PLS) assumption to interpolate the strain field. The details of PLS method can be found in our previous modeling work [50]. Dividing the soft slender robot into p sections along the s direction in the form of $[0, s_1]$, $[s_1, s_2]$, $\ldots$, $[s_{p-1}, 1]$, the entire strain field is reformulated by a set of linear interpolation basis functions $\boldsymbol{\Phi}(s) \in \mathbb{R}^{6\times 6(p+1)}$ and a discrete set of strain vectors $\boldsymbol{\theta} \in \mathbb{R}^{6(p+1)}$:

$$\boldsymbol{\xi}(s,t) = \boldsymbol{\xi}_0 + \boldsymbol{\Phi}(s)\boldsymbol{\theta}(t) \tag{42}$$

with $\boldsymbol{\theta}(t) = [\boldsymbol{\xi}^\top(0,t), \boldsymbol{\xi}^\top(s_1,t), \ldots, \boldsymbol{\xi}^\top(1,t)]^\top$. By employing the aforementioned parameterization, the entire configuration space of soft slender robot is now reconstructed in the following form:

$$\mathcal{U}_d = SE(3) \times \mathbb{R}^{6(p+1)} \tag{43}$$

where $SE(3)$ stands for the configuration $\boldsymbol{g}_0$ parameterized via (41). Therefore, by numerically integrating (29) in $SE(3)$, the geometric model of the slender robot can be expressed by the following equation:
$\forall s \in [s_{k-1}, s_k]$,

$$\boldsymbol{g}(s,t) = \boldsymbol{g}_0(t) \left(\prod_{i=1}^{k-1} \exp\widehat{\boldsymbol{\xi}}(s_i,t)\Delta \mathrm{s}) \right) \exp\widehat{\boldsymbol{\xi}}(s - s_k, t) \tag{44}$$

where Δs is the integrate segment along length of slender robot. Building upon the foundation of piecewise linear strain, we have broadened the modeling scope to encompass any freely moving, rod-like, flexible robotic structure, such as a soft slender robot thrown into the air or a robot snake crawling on a surface. Consequently, our work aims to transition from focusing solely on manipulation to encompassing locomotion as well.

B. Discrete kinematics

As a common way in robotics community, Jacobian matrix is defined to map the generalized coordinates $\boldsymbol{q}(t) = [\boldsymbol{\alpha}^\top(t), \boldsymbol{\theta}^\top(t)]^\top \in \mathbb{R}^{6(p+2)}$ to velocity field $\boldsymbol{\eta}(s,t) \in \mathbb{R}^6$ via the following form:

$$\boldsymbol{\eta}(s,t) = \underbrace{\begin{bmatrix} \boldsymbol{J}_{\boldsymbol{\alpha}}(s,t) & \boldsymbol{J}_{\boldsymbol{\theta}}(s,t) \end{bmatrix}}_{\boldsymbol{J}(s,t) \in \mathbb{R}^{6\times 6(p+2)}} \dot{\boldsymbol{q}}(t) \tag{45}$$

where $\boldsymbol{J}_{\boldsymbol{\alpha}}(s,t) \in \mathbb{R}^{6\times 6}$ and $\boldsymbol{J}_{\boldsymbol{\theta}}(s,t) \in \mathbb{R}^{6\times 6(p+1)}$ are two Jacobian matrices respectively related with $\boldsymbol{\alpha}(t)$ and $\boldsymbol{\theta}(t)$. Then, by substituting (45) into (30), one can get:

$$\boldsymbol{J}'_{\boldsymbol{\alpha}}\dot{\boldsymbol{\alpha}} + \boldsymbol{J}'_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}} = \boldsymbol{\Phi}\dot{\boldsymbol{\theta}} - \mathrm{ad}_{\boldsymbol{\xi}}(\boldsymbol{J}_{\boldsymbol{\alpha}}\dot{\boldsymbol{\alpha}} + \boldsymbol{J}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}})$$

Note that the above equation holds for any $\dot{\boldsymbol{\alpha}}$ and $\dot{\boldsymbol{\theta}}$, thus the following two equations stand:

$$\boldsymbol{J}'_{\boldsymbol{\theta}} = -\mathrm{ad}_{\boldsymbol{\xi}}\boldsymbol{J}_{\boldsymbol{\theta}} + \boldsymbol{\Phi} \tag{46}$$

$$\boldsymbol{J}'_{\boldsymbol{\alpha}} = -\mathrm{ad}_{\boldsymbol{\xi}}\boldsymbol{J}_{\boldsymbol{\alpha}} \tag{47}$$

Taking derivative of the two equations above with respect to time one can get:

$$\dot{\boldsymbol{J}}'_{\boldsymbol{\theta}} = -\mathrm{ad}_{\dot{\boldsymbol{\xi}}}\boldsymbol{J}_{\boldsymbol{\theta}} - \mathrm{ad}_{\boldsymbol{\xi}}\dot{\boldsymbol{J}}_{\boldsymbol{\theta}} \tag{48}$$

$$\dot{\boldsymbol{J}}'_{\boldsymbol{\alpha}} = -\mathrm{ad}_{\dot{\boldsymbol{\xi}}}\boldsymbol{J}_{\boldsymbol{\alpha}} - \mathrm{ad}_{\boldsymbol{\xi}}\dot{\boldsymbol{J}}_{\boldsymbol{\alpha}} \tag{49}$$

Then, the Jacobian matrices and their time derivatives can be computed by numerical integration along arc space $s \in [0,1]$ through (46)-(49) respectively starting from the initial value of $\boldsymbol{J}_{\boldsymbol{\alpha}}(0,t)$ and $\boldsymbol{J}_{\boldsymbol{\theta}}(0,t)$. It is obvious that for $s = 0$ the velocity $\boldsymbol{\eta}(0,t)$ only depends on the base parameter $\boldsymbol{\alpha}(t)$, i.e., $\boldsymbol{\eta}(0,t) = \boldsymbol{J}_{\boldsymbol{\alpha}}(0,t)\dot{\boldsymbol{\alpha}}$, thus we can directly deduce $\boldsymbol{J}_{\boldsymbol{\theta}}(0,t) = \boldsymbol{0}$ and $\dot{\boldsymbol{J}}_{\boldsymbol{\theta}}(0,t) = \boldsymbol{0}$. For the Jacobian with respect to $\boldsymbol{\alpha}(t)$ at the initial position, i.e., $\boldsymbol{J}_{\boldsymbol{\alpha}}(0,t)$, it can be derived from the derivation of exponential map.

For the exponential map of rotation matrix, the following differential relationship stands:

$$\dot{\boldsymbol{R}}_0(\boldsymbol{\phi})\boldsymbol{R}_0^\top(\boldsymbol{\phi}) = \widetilde{\left(\boldsymbol{J}_l(\boldsymbol{\phi})\dot{\boldsymbol{\phi}}(t)\right)} \tag{50}$$

where $\tilde{(\cdot)}$ denotes the mapping from $\mathbb{R}^3$ to $SO(3)$. $\boldsymbol{J}_l$ is the left Jacobian of group $SO(3)$, with the definition as below:

$$\boldsymbol{J}_l(\boldsymbol{\phi}) = \mathbf{I} + \frac{1-\cos\phi}{\phi^2}\tilde{\boldsymbol{\phi}} + \frac{\phi - \sin\phi}{\phi^3}\tilde{\boldsymbol{\phi}}^2$$

Using (50), the angular velocity of initial position with respect to body frame can be deduced:

$$\tilde{\boldsymbol{w}}(0,t) = \boldsymbol{R}_0^\top\dot{\boldsymbol{R}}_0 = \boldsymbol{R}_0^\top\widetilde{(\boldsymbol{J}_l\dot{\boldsymbol{\phi}})}\boldsymbol{R}_0$$

Notice that for any vector $\boldsymbol{x} \in \mathbb{R}^3$ and matrix $\boldsymbol{R} \in SO(3)$, one holds: $\boldsymbol{R}^\top\tilde{\boldsymbol{x}}\boldsymbol{R} = \widetilde{(\boldsymbol{R}^\top\boldsymbol{x})}$. Thus one can get

$$\boldsymbol{w}(0,t) = \boldsymbol{R}_0^\top(\boldsymbol{\phi})\boldsymbol{J}_l(\boldsymbol{\phi})\dot{\boldsymbol{\phi}} \tag{51}$$

The linear velocity of initial position with respect to body frame is

$$\boldsymbol{v}(0,t) = \boldsymbol{R}_0^\top(\boldsymbol{\phi})\dot{\boldsymbol{p}}_0$$

We can finally deduce the map from time derivation of vector $\boldsymbol{\alpha}$ to velocity twist of initial position with respect to body frame:

$$\boldsymbol{\eta}(0,t) = \begin{bmatrix} \boldsymbol{R}_0^\top(\boldsymbol{\phi})\boldsymbol{J}_l(\boldsymbol{\phi}) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R}_0^\top(\boldsymbol{\phi}) \end{bmatrix} \dot{\boldsymbol{\alpha}}(t) \tag{52}$$

where the matrix mapping $\dot{\boldsymbol{\alpha}}(t)$ to $\boldsymbol{\eta}(0,t)$ stands for the Jacobian $\boldsymbol{J}_{\boldsymbol{\alpha}}(0,t)$.

C. Spatial interpolation of contact field

It is clear that the distribution of contact forces depends on the arc length parameter s, thus similar to the interpolation of the strain field, we employ the concept of piecewise linear interpolation to approximate the distribution of contact forces through interpolating the corresponding slack variable along the length of the center line. By dividing the soft slender robot into m elements along the s direction, denoted as $[0, s_1]$, $[s_1, s_2]$, $\ldots$, $[s_{m-1}, 1]$, the entire contact field $\boldsymbol{\Lambda}_c(s)$ can be expressed as a linear combination of interpolation basis functions $\boldsymbol{\Psi}(s)$ and a discrete set of slack variable $\boldsymbol{\lambda}_c$:

$$\boldsymbol{\Lambda}_c(s) = (\mathbf{B}_n^\top D\mathbf{C}_n + \mathbf{B}_t^\top W\mathbf{C}_t)\boldsymbol{\Psi}(s)\boldsymbol{\lambda}_c \tag{53}$$

where $\boldsymbol{\lambda}_c = [u_0 \ u_1 \ \ldots \ u_m]^\top \in \mathbb{R}^{m+1}$ contains the slack variables of each node along centerline. The interpolation basis functions $\boldsymbol{\Psi}(s)$ capture the linear variation of contact forces within each section of the soft slender robot, which are defined as follows:

$$\boldsymbol{\Psi}(s) = \begin{bmatrix} \Psi_0\mathbf{I}_{3\times3} & \Psi_1\mathbf{I}_{3\times3} & \ldots & \Psi_m\mathbf{I}_{3\times3} \end{bmatrix} \in \mathbb{R}^{3\times3(m+1)}$$

where for $i \in \{0, 1, \ldots, m\}$

$$\Psi_i(s) = \begin{cases} ms - i, & \text{for } s \in [\frac{i-1}{m}, \frac{i}{m}) \\ i + 1 - ms, & \text{for } s \in [\frac{i}{m}, \frac{i+1}{m}] \\ 0, & \text{for } s \notin [\frac{i-1}{m}, \frac{i+1}{m}] \end{cases}$$

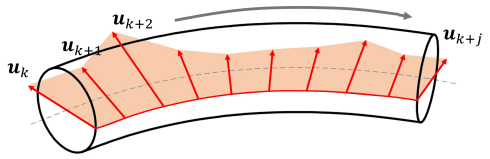


Fig. 9. Linear interpolation of contact load along arc length.

As for the contact force produced by fixed constraints and articulated constraints, supposing that they are discretely located on the soft slender robot with quantities of m_f and m_a respectively, we use two vectors to cover all of them:

$$\boldsymbol{\lambda}_f = [\boldsymbol{\Lambda}_{f,1}^\top \ \boldsymbol{\Lambda}_{f,2}^\top \ \ldots \ \boldsymbol{\Lambda}_{f,m_f}^\top]^\top \in \mathbb{R}^{6m_f}$$

$$\boldsymbol{\lambda}_a = [\boldsymbol{\Lambda}_{a,1}^\top \ \boldsymbol{\Lambda}_{a,2}^\top \ \ldots \ \boldsymbol{\Lambda}_{a,m_a}^\top]^\top \in \mathbb{R}^{6m_a}$$

It should be noted that when the contact area is notably narrow, the rate of change in the contact load is significant, or the force is concentrated, the linearized contact force often requires a finer level of discretization.

D. Weak form of dynamics

For $s \in [0, 1]$, by introducing the test function $\boldsymbol{\Phi}_{\boldsymbol{\xi}}(s)$ as virtual displacement $\boldsymbol{\delta}(s) = \boldsymbol{J}\delta\boldsymbol{q}$, the weak form (39) is given by:

$$\begin{aligned}\delta\boldsymbol{q}^\top \int_0^1 \boldsymbol{J}^\top (\boldsymbol{\mathcal{M}}\dot{\boldsymbol{\eta}} - ad_{\boldsymbol{\eta}}^\top \boldsymbol{\mathcal{M}}\boldsymbol{\eta} - \boldsymbol{\Lambda}_i' + \mathrm{ad}_{\boldsymbol{\xi}}^\top \boldsymbol{\Lambda}_i - \boldsymbol{\Lambda}_e \\ - \mathrm{Ad}_{\boldsymbol{g}_{bc}}^* \boldsymbol{\Lambda}_c - \boldsymbol{\Lambda}_f - \mathrm{Ad}_{\boldsymbol{g}_{bc}}^* \mathbf{B}^\top \boldsymbol{\Lambda}_a)\mathrm{ds} = 0\end{aligned} \tag{54}$$

By introducing the test function $\boldsymbol{\Phi}_{\boldsymbol{u}}(s)$ as $\delta\boldsymbol{u} = \boldsymbol{\Psi}\delta\boldsymbol{\lambda}$ for contact constrain (37) and (38) respectively, their weak form (40) is deduced as follows:

$$\delta\boldsymbol{\lambda}_c^\top \int_0^1 \boldsymbol{\Psi}^\top \begin{bmatrix} \mathbf{P}\boldsymbol{g}_{cd}\mathbf{A} + D^*\mathbf{C}_n\boldsymbol{\Psi}\boldsymbol{\lambda}_c \\ \mathrm{Ad}_{\boldsymbol{g}_{cb}}\boldsymbol{\eta} - \mathrm{Ad}_{\boldsymbol{g}_{cd}}\boldsymbol{\eta}_d - W^*\mathbf{C}_t\boldsymbol{\Psi}\boldsymbol{\lambda}_c \end{bmatrix} \mathrm{ds} = 0 \tag{55}$$

E. Discrete dynamics equation

By reformulating the contact force using $\boldsymbol{\lambda}_n$, $\boldsymbol{\lambda}_t$, $\boldsymbol{\lambda}_{fc}$ and $\boldsymbol{\lambda}_{ac}$, along with the aforementioned definitions, we have successfully transformed the dynamics from the NCP form into a system of DAEs (Differential-Algebraic Equations) solely composed of equality constraints:

$$\boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{C}\dot{\boldsymbol{q}} - \boldsymbol{K}\boldsymbol{q} = \boldsymbol{P} + \boldsymbol{H}_c\boldsymbol{\lambda}_c + \boldsymbol{H}_f\boldsymbol{\lambda}_f + \boldsymbol{H}_a\boldsymbol{\lambda}_a \tag{56}$$

$$\boldsymbol{K}_{BC}\boldsymbol{q} - \boldsymbol{\Lambda}_{BC} = \mathbf{0} \tag{57}$$

$$\boldsymbol{G}_c - \boldsymbol{E}_c\boldsymbol{\lambda}_c = \mathbf{0} \tag{58}$$

$$\boldsymbol{G}_f = \mathbf{0} \tag{59}$$

$$\boldsymbol{G}_a = \mathbf{0} \tag{60}$$

For ease of description, we define the following operators Vec to assemble discrete matrices together:

$$\underset{i=1}{\overset{N}{\mathrm{Vec}}}(\boldsymbol{X}_i) = [\boldsymbol{X}_1^\top \ \boldsymbol{X}_2^\top \ \ldots \ \boldsymbol{X}_N^\top]^\top$$

All the matrices of (56)-(60) are given by the following definitions:

- $\boldsymbol{M}(\boldsymbol{q}) = \int_0^1 \boldsymbol{J}^\top \boldsymbol{\mathcal{M}}\boldsymbol{J}ds \in \mathbb{R}^{6(n+1)\times6(n+1)}$, the mass matrix;
- $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \int_0^1 [\boldsymbol{J}^\top(\boldsymbol{\mathcal{M}}\dot{\boldsymbol{J}} - ad_{\boldsymbol{J}\dot{\boldsymbol{q}}}^\top \boldsymbol{\mathcal{M}}\boldsymbol{J}) - \boldsymbol{\Phi}^\top\mathcal{D}\boldsymbol{\Phi}]ds \in \mathbb{R}^{6(n+1)\times6(n+1)}$, the Coriolis and damping matrix;
- $\boldsymbol{K} = \int_0^1 \boldsymbol{\Phi}^\top\mathcal{K}\boldsymbol{\Phi}ds \in \mathbb{R}^{6(n+1)\times6(n+1)}$, the stiffness matrix;
- $\boldsymbol{H}_c(\boldsymbol{q}) = \int_0^1 \boldsymbol{J}^\top \mathrm{Ad}_{\boldsymbol{g}_{bc}}^*(\mathbf{B}_n^\top D\mathbf{C}_n + \mathbf{B}_t^\top W\mathbf{C}_t)\boldsymbol{\Psi}\mathrm{ds} \in \mathbb{R}^{6(n+1)\times3(m+1)}$, the collision contact force matrix;
- $\boldsymbol{H}_f(\boldsymbol{q}) = \underset{i=1}{\overset{m_f}{\mathrm{Vec}}}(\boldsymbol{J}_i) \in \mathbb{R}^{6(n+1)\times6m_f}$, the force matrix of fixed constraints;
- $\boldsymbol{H}_a(\boldsymbol{q}) = \underset{i=1}{\overset{m_a}{\mathrm{Vec}}}(\boldsymbol{J}_i^\top \mathrm{Ad}_{\boldsymbol{g}_{bc,i}}^* \mathbf{B}_i^\top) \in \mathbb{R}^{6(n+1)\times3m_a}$, the force matrix of articulated constraints;
- $\boldsymbol{P}(\boldsymbol{q}) = \int_0^1 \boldsymbol{J}^\top \boldsymbol{\Lambda}_e\mathrm{ds} + \int_0^1 \boldsymbol{J}^\top \boldsymbol{\mathcal{M}}\mathrm{Ad}_{\boldsymbol{g}}^{-1}\mathrm{ds}\boldsymbol{\mathcal{G}} \in \mathbb{R}^{6(n+1)}$, the contribution of concentrated external load and gravity.
- $\boldsymbol{K}_{BC} = \mathrm{diag}\big(\mathcal{K}(0)\boldsymbol{\Phi}(0), \mathcal{K}(1)\boldsymbol{\Phi}(1)\big) \in \mathbb{R}^{12\times12}$, the stiffness matrix of boundary condition;
- $\boldsymbol{\Lambda}_{BC} = [-\boldsymbol{\Lambda}_0^\top \ \boldsymbol{\Lambda}_1^\top]^\top \in \mathbb{R}^{12}$, the external force of boundary;
- $\boldsymbol{G}_c(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \int_0^1 \boldsymbol{\Psi}^\top \begin{bmatrix} \mathbf{P}\boldsymbol{g}_{cd}\mathbf{A} \\ \mathrm{Ad}_{\boldsymbol{g}_{cb}}\boldsymbol{\eta} - \mathrm{Ad}_{\boldsymbol{g}_{cd}}\boldsymbol{\eta}_d \end{bmatrix} \mathrm{ds} \in \mathbb{R}^{3(m+1)}$, the tangent contact velocity matrix;
- $\boldsymbol{G}_f(\boldsymbol{q}) = \underset{i=1}{\overset{N}{\mathrm{Vec}}} \left(\log(\boldsymbol{g}_{fc,i}^{-1}\boldsymbol{g}_i)\right)^\vee \in \mathbb{R}^{6m_f}$, the fixed constraint vector;
- $\boldsymbol{G}_a(\boldsymbol{q}) = \underset{i=1}{\overset{N}{\mathrm{Vec}}}(\boldsymbol{p}_{a,i} - \mathbf{U}\boldsymbol{g}_{c,i}\mathbf{A}) \in \mathbb{R}^{6m_a}$, the articulated constraint vector;
- $\boldsymbol{E}_c(\boldsymbol{\lambda}_c) = \int_0^1 \boldsymbol{\Psi}^\top \begin{bmatrix} D^*\mathbf{C}_n \\ -W^*\mathbf{C}_t \end{bmatrix} \boldsymbol{\Psi}\mathrm{ds} \in \mathbb{R}^{3(m+1)\times3(m+1)}$, the collision contact constraint matrix;

VIII. Time discretization

Time-stepping is a prevalent technique employed for the time discretization of dynamic systems. It has gained significant popularity in the field of robotics for simulation purpose. Considering a time interval $[t_{k-1}, t_k]$ and denoting $h = t_k - t_{k-1}$ as the time step, for the explicit representation, the discritization of this time interval is given by:

$$\boldsymbol{q}_k = \boldsymbol{q}_{k-1} + h\dot{\boldsymbol{q}}_{k-1} \ , \ \dot{\boldsymbol{q}}_k = \dot{\boldsymbol{q}}_{k-1} + h\ddot{\boldsymbol{q}}_{k-1}$$

For the implicit scheme, the discritization of this time interval is given by:

$$\boldsymbol{q}_k = \boldsymbol{q}_{k-1} + h\dot{\boldsymbol{q}}_k \ , \ \dot{\boldsymbol{q}}_k = \dot{\boldsymbol{q}}_{k-1} + h\ddot{\boldsymbol{q}}_k$$

In our work, we use the implicit scheme to ensure convergence of the solution. By taking the implicit equations to (56) and using the abbreviation $\widehat{\boldsymbol{M}} = \boldsymbol{M} + h\boldsymbol{C}$, one can get:

$$\mathscr{L}(\dot{\boldsymbol{q}}_k, \boldsymbol{q}_k) = \widehat{\boldsymbol{M}}\dot{\boldsymbol{q}}_k - \boldsymbol{M}\dot{\boldsymbol{q}}_{k-1} - h\boldsymbol{K}\boldsymbol{q}_k - h\boldsymbol{P}$$

Denoting

$$\mathscr{F}(\boldsymbol{q}_k, \boldsymbol{\lambda}_c, \boldsymbol{\lambda}_f, \boldsymbol{\lambda}_a) = \boldsymbol{H}_c\boldsymbol{\lambda}_c + \boldsymbol{H}_f\boldsymbol{\lambda}_f + \boldsymbol{H}_a\boldsymbol{\lambda}_a$$

the implicit time discretization of (56)-(60) is replaced by the following nonlinear algebraic equations:

$$\begin{cases} \mathscr{L}(\dot{\boldsymbol{q}}_k, \boldsymbol{q}_k) - h\mathscr{F}(\boldsymbol{q}_k, \boldsymbol{\lambda}_c, \boldsymbol{\lambda}_f, \boldsymbol{\lambda}_a) = \boldsymbol{0} & \text{(61a)} \\ \boldsymbol{q}_k - h\dot{\boldsymbol{q}}_k - \boldsymbol{q}_{k-1} = \boldsymbol{0} & \text{(61b)} \\ \boldsymbol{K}_{BC}\boldsymbol{q}_k - \boldsymbol{\Lambda}_{BC} = \boldsymbol{0} & \text{(61c)} \\ \boldsymbol{G}_c(\boldsymbol{q}_k, \dot{\boldsymbol{q}}_k) - \boldsymbol{E}_c\boldsymbol{\lambda}_c = \boldsymbol{0} & \text{(61d)} \\ \boldsymbol{G}_f(\boldsymbol{q}_k) = \boldsymbol{0} & \text{(61e)} \\ \boldsymbol{G}_a(\boldsymbol{q}_k) = \boldsymbol{0} & \text{(61f)} \end{cases}$$

The unknown variables in the algebraic equations (61) consist of $\boldsymbol{q}_k$, $\dot{\boldsymbol{q}}_k$, $\boldsymbol{\lambda}_c$, $\boldsymbol{\lambda}_f$, and $\boldsymbol{\lambda}_a$. Notably, we directly define the normal contact force and friction force on the manifold of $\boldsymbol{\lambda}_n$ and $\boldsymbol{\lambda}_t$. An essential advantage of our method lies in the provision of a smooth system of nonlinear algebraic equations. Consequently, these equations can be solved using widely used techniques such as the gradient method or the Newton-Raphson method.

IX. Collision detection

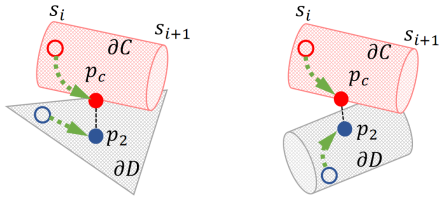
In our simulation, we set the Cosserat rod as the master contact object, with distributed master contact points $\boldsymbol{p}_c$ on it. Corresponding to each master contact point is a slave contact point $\boldsymbol{p}_2$ located on the slave contact object. $\boldsymbol{p}_c$ and $\boldsymbol{p}_2$ construct a contact pair. If the soft slender rod is divided into m contact elements, to ensure the continuity of contact within the contact elements, we posit the existence of a sliding contact point therein. For a given contact element, with $s \in [s_i, s_{i+1}], \forall i \in [1, m]$, its surface $\partial\mathcal{C}$ are defined by two parameters s and β, thus the contact point in the contact element can be represented by the following formula (see as well eq. (4)):

$$\boldsymbol{p}_c(s, \beta) = \boldsymbol{p}(s) + \boldsymbol{R}(s)\boldsymbol{d}(s, \beta), \ s \in [s_i, s_{i+1}], \ \forall i \in [1, m] \quad \text{(62)}$$

At each point $\boldsymbol{p}_c$ on the surface, there corresponds a local coordinate system with $\boldsymbol{p}$ as the origin. The vector corresponding to $\boldsymbol{p}_c$ in this local coordinate system is denoted as $\boldsymbol{d}$. The rotation matrix from the local coordinate system to the global coordinate system is represented by $\boldsymbol{R}$.

For the slave surface $\partial\mathcal{D}$, we suppose that its function is parameterized by τ_1, τ_2, i.e., $\boldsymbol{p}_2 \in \partial\mathcal{D}(\tau_1, \tau_2)$. Here, we discuss two types, the contact between contact between a Cosserat rod and a plane and the contact between a Cosserat rod and another Cosserat rod, as shown in Fig. 10:

- For the former, the slave surface can be defined using the parametric equations of the plane, i.e., $\boldsymbol{p}_2 = \begin{bmatrix} \tau_1 & \tau_2 & f(\tau_1, \tau_2) \end{bmatrix}^\top$, where $\tau_1 = x$, $\tau_2 = y$, and $f(x, y) = z$ explicitly defines the slave surface.
- For the latter, the definition of the slave surface can be chosen as the same as (62), i.e, $\boldsymbol{p}_2(\tau_1, \tau_2) = \boldsymbol{p}(\tau_1) + \boldsymbol{R}(\tau_1)\boldsymbol{d}(\tau_1, \tau_2)$ with $\tau_1 = s$, $\tau_2 = \beta$.



(a) Contact with plane (b) Contact with rod

Fig. 10. Searching the contact pair from initial guess.

When two objects are on the verge of contact, optimizing computational efficiency necessitates an initial assessment of potential contact occurrences among contact elements. This process begins with the calculation of distances between the centers of all contact elements and the slave surface. If these distances fall below a predefined threshold, the respective contact elements are deemed activated. Subsequently, for these activated elements, the following algorithm is employed to identify their corresponding contact pairs. When the master contact point moves, the corresponding slave contact point is defined as the contact point closest to the master contact point, i.e., the slave contact point is obtained by minimizing the contact distance. As shown in Fig. 10, denoting $\boldsymbol{d}_{2c} = \boldsymbol{p}_c - \boldsymbol{p}_2$, the contact pair can be searched by solving the minimization as follows:

$$\begin{aligned} &\arg\min_{s,\beta,\tau_1,\tau_2} \mathscr{Y} = \tfrac{1}{2}\boldsymbol{d}_{2c}^\top\boldsymbol{d}_{2c} \\ &\text{s. t. } \ \boldsymbol{p}_c \in \partial\mathcal{C}(s, \beta), \boldsymbol{p}_2 \in \partial\mathcal{D}(\tau_1, \tau_2), s_i \le s \le s_{i+1} \end{aligned} \quad \text{(63)}$$

When contact is established on rod, and as changes occur in the contact configuration, this method allows for a smooth transition of contact points from one contact element to another element.

X. Numerical simulation

In this section, we have configured various simulation scenarios with the objective of comparing our method to others and analyzing the influence of different model parameters on simulation outcomes. All simulations were conducted within the Matlab environment with CPU Intel(R) Core(TM) i7-7820HQ @2.90GHz.

A. Influence of friction

1) Internal contact: In this test, we insert a soft rod (blue) of length $\mathrm{L} = 26\mathrm{mm}$ inside a rigid tube (red) along axis x. The diameter of soft rod is $0.3\mathrm{mm}$ at tip and $0.2\mathrm{mm}$ at end. The radius of curvature of the tube is $3\mathrm{mm}$. The Young modulus of soft rod is $55\mathrm{MPa}$ and Poisson ratio is 0.45. Fig. 11 shows the final state of insertion with different friction coefficients. The figure on the right side shows that the soft rod fails to be inserted and sticks inside the tube due to the large friction $\mu = 0.5$, while the figure on left side shows the successful insertion with $\mu = 0.2$. Fig. 12 shows the evolution of the norm of the constrained force $\mathbf{\Lambda}_b$ at tip with respect of the insertion displacement. The total insertion time is 70 seconds, while the

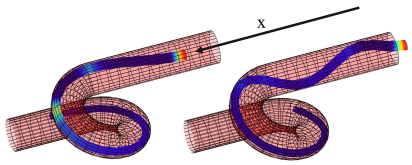


Fig. 11. Fictional contact in a curved tube. Successful case(left) and buckling case(right). Implicit time step of simulation $\mathrm{dt} = 0.05s$. Number of sections: $n = m = 26$. The color shows the contact of rods, which gradually increases from blue to red.

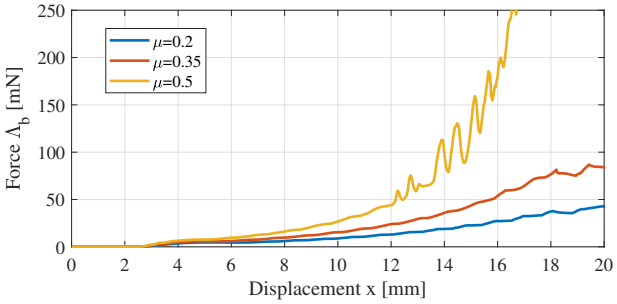


Fig. 12. The evolution of insertion force with different coefficient of friction.

simulation time is 156 seconds. We repeated this experiment in the Finite Element Method (FEM) software SOFA, with a consistent time step, resulting in a computational time of 722 seconds. Compared to the FEM algorithm, our method significantly accelerates computation speed due to its lower system dimensionality.

2) External contact: In this test, we throw a soft rod (blue) towards a rigid obstacle (red). The soft rod is initially positioned horizontally above the obstacle. The length of the soft rod is $30\mathrm{cm}$ and its diameter of two side of are $0.3\mathrm{cm}$ and $0.2\mathrm{cm}$ respectively. The Young modulus of soft rod is $55\mathrm{MPa}$ and Poisson ratio is 0.45. As shown in Fig. 13, we test the impact with different coefficient of friction. When the coefficient of friction is small, the soft rod cannot stay on the rigid rod after the collision occurs because there is not enough friction to resist the asymmetrical gravity. Meanwhile, if we increase the coefficient of friction, the soft robot will not slide down.

B. Comparison with different modeling methods

In this experiment, a soft slender rod comes into contact with external obstacles and naturally sags under the influence of gravity. The soft slender rod is 60cm long with Young's modulus of 3×10^5Pa. We investigate the impact of different existing approaches in the literature on contact behavior, comparing with our proposed method. We compare three discretization strategies: Piecewise Constant Strain (PCS) [18], Global Variable Strain (GVS) [19], and the Piecewise Linear Strain (PLS) [50] that we are using in this work. Subsequently, we utilize Comsol, a general finite element method (FEM) software, to determine the equilibrium position of the soft slender rod, which is used as the reference. The spatial discretization in the Finite Element Method (FEM) employs quadrilateral mesh elements, dividing the rod into 820 elements along the arc axis. The contact algorithm in FEM is implemented using the augmented Lagrangian method. The FEM simulation result is used as the position reference and PLS (divided into 30 sections) is used as the strain reference, comparing with the outcomes of PLS (divided into 5 sections), PCS (divided into 6 sections), and GVS (5th-order polynomial). Except of FEM, all simulations were conducted using the same set of physical parameters and solver.

Table II presents the errors of different discretization strategies compared to the reference (PLS with 30 sections). These errors encompass both strain (curvature along Y-axis) and position discrepancies along the arc length, defined as follows:

$$e_{\kappa_Y} = \frac{1}{L}\int_0^L |\kappa_Y(s) - \kappa_{Yref}(s)|\mathrm{ds}$$

$$e_{\boldsymbol{p}} = \frac{1}{L}\int_0^L \|\boldsymbol{p}(s) - \boldsymbol{p}_{ref}(s)\|^2\mathrm{ds}$$

TABLE II
ERROR OF DIFFERENT DISCRETIZATION STRATEGIES

Modeling method	DOF	Strain error	Position error
FEM (800 elements)	2460	-	-
PCS (6 sections)	36	0.027 cm^{-2}	0.721 cm
GVS (5 orders)	36	0.026 cm^{-2}	0.941 cm
PLS (5 sections)	36	0.014 cm^{-2}	0.262 cm
PLS (30 sections)	186	-	0.047 cm

The result distinctly reveals that, with an equal degree of freedom, PLS achieves the highest accuracy. This is attributed to the superior local interpolation capabilities of linear interpolation.

Fig. 14 showcases the deformation of the soft rod. Due to its constant strain limitation, PCS necessitates fine discretization for accurately capturing local deformation during contact. Although polynomial interpolation offers continuous differentiability in strain, achieving precise fitting of the contact area's deformation demands an increase in polynomial order. However, high-order polynomials are susceptible to the Runge phenomenon, leading to localized strain distortion and oscillation, thereby challenging the robustness of simulations. In regions with significant local deformation and substantial

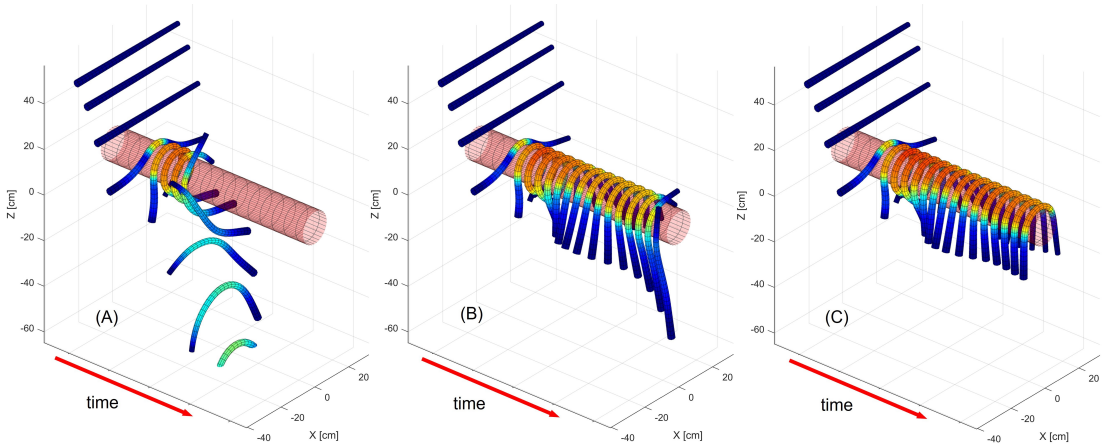


Fig. 13. Fictional contact under gravity. The configuration of soft rod is plotted every $0.15s$. Number of sections: $n = m = 30$. (A). friction coefficient $\mu = 0$; (B). friction coefficient $\mu = 0.1$; (C). friction coefficient $\mu = 0.2$. Implicit time step of simulation $\mathrm{dt} = 0.005s$. The color shows the internal force of rods, which gradually increases from blue to red.

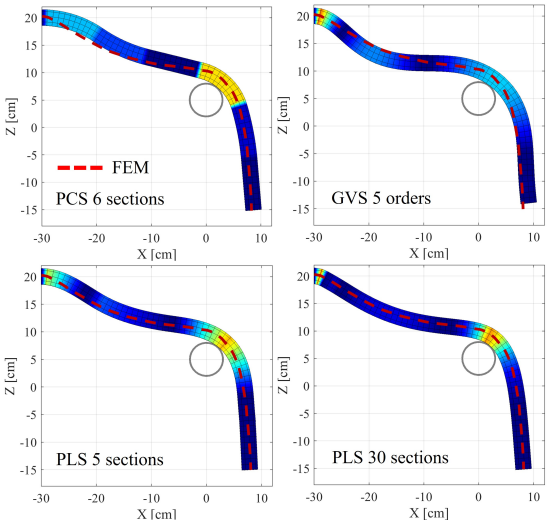


Fig. 14. A soft rod naturally droops under the action of gravity and contacts with external obstacles.The figure shows the static results obtained using different modeling methods: FEM, PCS, GVS and PLS. The color shows the norm of strain of rods, which gradually increases from blue to red.

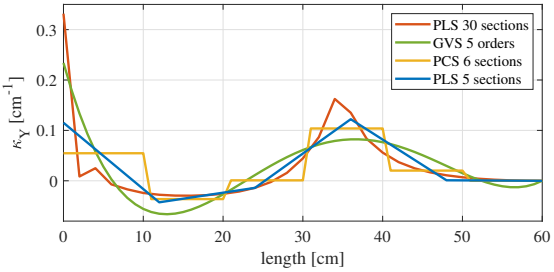


Fig. 15. The evolution of strain along arc length of soft rod.

strain variations, such as the left end of the soft rod and areas in contact with external objects, PLS exhibits superior fitting capabilities. Moreover, The computational results of PLS with 30 sections are closer to the FEM reference compared to the PLS with 5 sections. This observation highlights that, with increasingly precise discretization, computational results converge toward the true values.

Fig. 15 illustrates the evolution of strain along the arc length of the soft rod for each test.

In summary, in contact scenarios, PLS demonstrates superior accuracy when compared to PCS and GVS with an equivalent degree of freedom. Significantly, as the segmentation of PLS becomes finer, computational results tend to converge, closely aligning with FEM outcomes.

C. Discretization scale and convergence

In this subsection, we compare and analyze the impact of different discretization strategies on the computational results of our model. Specifically, we investigate the case of two intertwined rods initially positioned in a crossed configuration. The rods are then simultaneously twisted, resulting in mutual entanglement until a total twist angle of 540 degrees is reached.

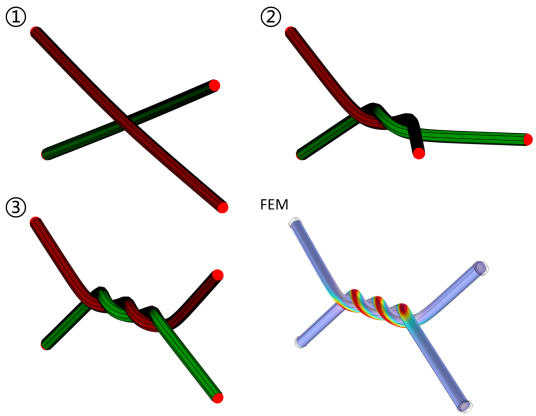


Fig. 16. The screenshots of the simulation process of the Cosserat rods, corresponding to rotations of 0 degrees, 270 degrees, and 540 degrees, respectively.The simulation result of the FEM is shown in the fourth image.

As shown in Fig. 16, two rods rotate around a central horizontal axis at a constant speed of rotation, up to 540

degrees. The numbers 1, 2, and 3 in the figure are screenshots of the simulation process of the Cosserat rods, corresponding to rotations of 0 degrees, 270 degrees, and 540 degrees, respectively. The increment of rotation angle per time step ($\mathrm{dt} = 0.01\mathrm{s}$) is set as 0.1 degree and adopted more stringent convergence criteria, in order to reach a more precise solution. Denote the residual error r_e as the norm of the left part of (61), the convergence criteria are set as $r_e < 10^{-4}$. To verify the accuracy of our algorithm, we conducted the same simulation in the finite element software Comsol to serve as a benchmark for comparison with our algorithm's results. In Comsol, tetrahedral elements were used to mesh the two rods, totaling 10,681 elements. The contact method employed was the penalty function approach. The simulation result of the FEM is shown in the fourth image of Fig. 16.

For the method we proposed, We separately investigated the effects of discretizing the strain space and discretizing the contact area on the computational results.

1) Simulation result:

- Varying the number of strain elements n: In these tests,

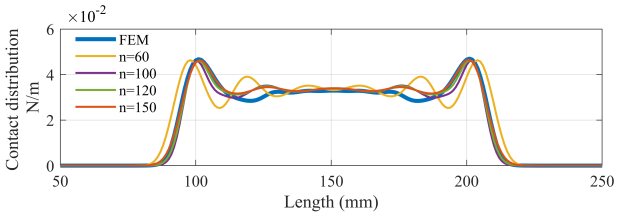


Fig. 17. The distribution of contact forces between the two rods after a twist of 540 degrees

we set the number of contact element as 300 (i.e., $m = 300$). On this basis, we adopted different strategies for the discretization of the strain space, dividing the strain space along the length of the rod into 60, 100, 120, and 150 elements, respectively. The simulation results are shown in Fig. 17.

- Varying the number of contact elements m: In these tests,

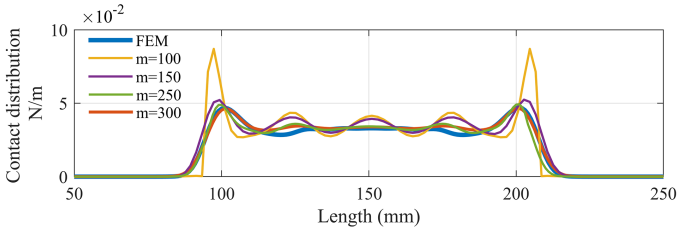


Fig. 18. The distribution of contact forces between the two rods after a twist of 540 degrees

we set the number of strain elements as 150 (i.e., $n = 150$) and divide the contact space along the length of the rod into 100, 150, 250 and 300 elements, respectively. The simulation results are shown in Fig. 18.

From the simulation results in Fig. 17 and Fig. 18, the distribution of contact forces presented by the method we proposed is close to the computational results of Comsol. As the refinement of the division of the strain elements and contact elements increase, the computational results tend to converge, and the distribution of contact forces also tends to be smoother.

2) Numerical iteration of simulation: For the case of 150 strain elements and 300 contact elements, within each time step, the number of numerical computation iterations in order to reach the convergence criteria is shown in Fig.19: It can

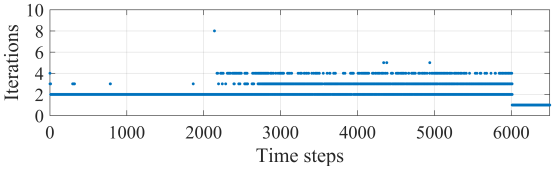


Fig. 19. The number of numerical computation iterations in order to reach the convergence criteria ($r_e < 10^{-4}$) of each time step.

be seens from Fig. 19. that most of time it is needed only 2 iterations to solve the nonlinear equation (61) via Newton-Rapson method since the initial condition is chosen as the states of the previous time step.

3) Convergence rate analysis: Regarding the distribution of contact force, we define the computational results from Comsol as the exact solution. The 2-norm residual between the solution from our method and this exact solution is defined as follows:

$$e = \frac{1}{L}\int_0^L \| \|\mathbf{\Lambda}_c(s)\|^2 - \|\mathbf{\Lambda}_{\mathrm{FEM}}(s)\|^2 \|^2 \mathrm{ds}$$

where $\mathbf{\Lambda}_c$ denotes the contact load of Cosserat rod, $\mathbf{\Lambda}_{\mathrm{FEM},i}$ denotes the contact load of FEM rod. The change in the 2-norm residual with respect to the strain and the contact elements are respectively shown in Fig. 20 and Fig. 21. From

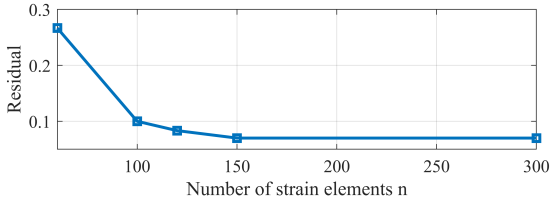


Fig. 20. The change in the 2-norm residual with the number of strain space division elements n when the number of contact elements $m = 300$.

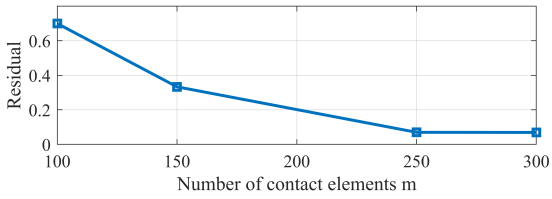


Fig. 21. The change in the 2-norm residual with the number of strain elements m when the number of contact elements $n = 150$.

the figures, it can be seen that the settlement results converge as the number of strain or contact elements increases.

D. Choice of smooth function

In Section V-C, we mentioned that the Heaviside function used to construct the complementarity conditions can be approximated or equivalently represented by other continuous or smooth functions. Through simulations, we will explore the influence of different approximation functions on the solution results. Fig. 22 illustrates the case of a homogeneous tapered slender robot placed on a plane. In the initial state, the left end of the slender robot is suspended and fixed by a constraint force, while the right end contacts the ground under the influence of gravity. The slender robot has a length of $30\mathrm{cm}$, density of $3\times10^{3}\mathrm{kg/m}^{3}$, Young's modulus of $1\mathrm{Mpa}$ and friction coefficient of 0.5. After the initial state, the left end of the slender robot is released and starts to fall until it collides with the ground, reaching a final equilibrium state. Fig.22 shows the steady states obtained using the impact function and different smoothing functions. In all three test groups, the strain field of the slender robot is divided into 30 sections, as well as the contact field. From the distribution of contact loads shown

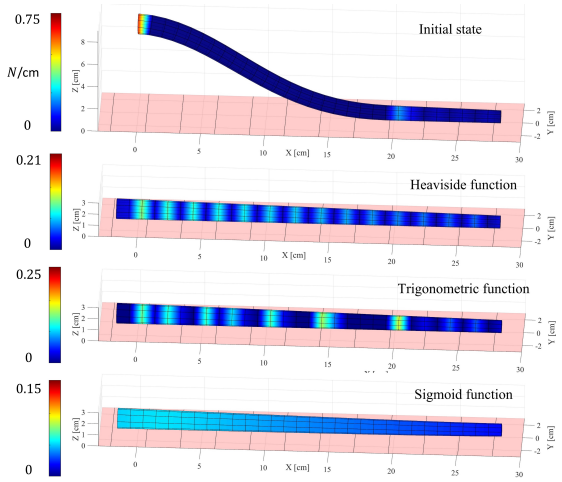


Fig. 22. Final steady state calculated using different functions. The colors in the plot represent the contact load distribution.

in Fig. 22, it can be observed that for the Heaviside function and trigonometric function, which satisfy the complementarity constraints exactly, although the slender robot is in a force-balanced state and remains stationary, not all contact points are activated but some contact points are in a critical state (virtual contact). In this case, due to the discretization of the system, the solution of the mechanical equation system is not unique. For the Heaviside function, as shown in fig.23, the contact force undergoes small jumps during the iteration process, which is also due to the presence of virtual contacts. Since the slack variables associated with virtual contacts are zero, the Heaviside function is discontinuous, resulting in numerical jumps during the iteration. However, this does not occur for the trigonometric function, which remains smooth and continuous. Among the three simulations, the sigmoid function yields results closest to the real solution. As shown in Fig. 22, the contact load is smoothly distributed and remains stable during the iteration process. This is because the sigmoid function is an approximation of the complementarity constraint. In this case, there are no virtual contacts where both the contact gap and contact force are zero, and therefore, all contact points are activated, yielding a unique solution.

Based on these findings, we conclude that approximating the impact function with trigonometric functions can address the issue of numerical jumps in critical states. However, in some cases, non-uniqueness of solutions may arise. On the other hand, using the sigmoid function, although it cannot precisely satisfy the complementarity constraint in the vicinity of critical states, offers better robustness.

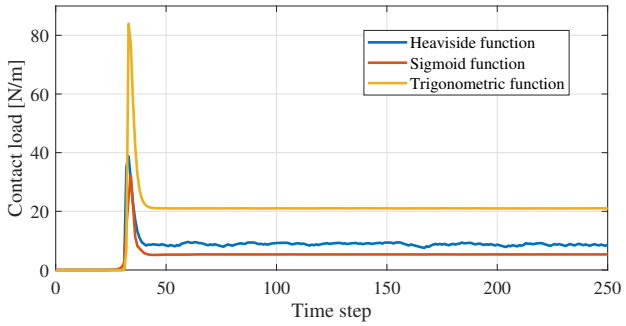


Fig. 23. Evolution of contact loads over time step iterations. The point of contact is 6 cm from the left end. The time step is set as $0.01s$.

XI. Experiment for Model Validation

A. Experiment setup

As shown in fig.24, our platform contains the soft silicone robot whose base is fixed with the rigid manipulator (Universal robot ur3). There is a steel string passing through the center line of the soft silicone robot in order to increase its stiffness. We installed force sensor (Onrobot) at the end of the rigid manipulator to measure the contact force of the soft silicone robot. Specific image processing program has been developed by us to track the red plastic markers on the soft silicone robot to interpolate the shape of polymer actuator and compute the corresponding curvatures at each control points, which will be detailed in the next subsection. TABLE III shows the measured physic parameters of the soft silicone robot and steel string. In general, soft silicone robots can be equipped with various actuators, such as cable-driven, tendon-driven, and magnetic-driven mechanisms. Our study introduces a contact dynamics framework that is suitable for analyzing soft slender robots with these actuators. However, in our research, we intentionally focused on validating the deformation and contact force magnitude of the soft slender robot during interactions with the external environment. Consequently, we designed the robot under investigation without any actuation structure and only passive mode (i.e. no actuation) is used to validate the proposed modeling framework. The investigation of contact issues in actuated soft slender robots will be addressed in our future work.

B. Strain observer design

To compare the strain of a real robot with our proposed model, we initially develop a strain observer that estimates the strain of the real robot based on the positions of markers.

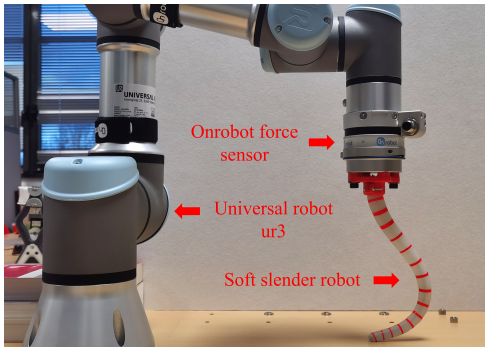


Fig. 24. Experiment setup

TABLE III
PHYSIC PARAMETERS OF SILICONE ROBOT AND STEEL STRING

	Silicone robot	Steel string
Length L	25 cm	25 cm
Radius R_0	0.85 cm	0.15 cm
Radius R_1	0.5 cm	0.15 cm
Poisson ratio γ	0.45	0.3
Young's modulus E	2.56 $\times 10^5$Pa	1.2 $\times 10^9$Pa
Mass density ρ	1.41$\times 10^3$ kg/m^3	7.8$\times 10^3$ kg/m^3

By analyzing the images captured by a rapid camera, we can extract the shape of the actuator, enabling us to compute the strain along the soft robot. We employ a combination of image binarization, inter-frame difference [58], and center point extraction algorithms in Matlab to obtain the marker positions.

We can conceptualize the presence of a virtual soft robot with zero inertia that consistently mimics the deformation of the real soft robot. This can be viewed as the virtual soft robot passing through the real robot while being constrained by the measured positions of the markers using the bilateral constraint discussed in Section IV-B. In this context, the real soft robot and the virtual robot correspond to two contact objects, while the markers serve as the fixed contact constraints. Thus the design of observer can be formulated as a contact problem. Specifically, according to the principle of minimum potential energy, the total potential energy of the virtual robot should hold the minimum under the contact constraints of markers, which leads to a minimization problem with constraints:

$$\underset{\hat{\boldsymbol{q}}}{\operatorname{argmin}} \ \frac{1}{2}\hat{\boldsymbol{q}}^\top \boldsymbol{K}\hat{\boldsymbol{q}} + V_g(\hat{\boldsymbol{q}}) \tag{64}$$

$$\text{subject to } \boldsymbol{G}_f(\hat{\boldsymbol{q}}) = \boldsymbol{0} \tag{65}$$

where $\hat{\boldsymbol{q}}$ is the estimated states, representing the strain field of the virtual robot. $\boldsymbol{K}$ stands for the positive diagonal stiffness matrix and $V_g(\hat{\boldsymbol{q}})$ stands for the gravitational potential energy of the virtual robot. $\boldsymbol{G}_f(\hat{\boldsymbol{q}})$ contents the constraints which are defined in (59) by setting the configurations of markers as the fixed contact constraints. Then, we can find the solution of this minimization problem by solving its Karush-Kuhn-Tucker Conditions:

$$\begin{cases} \boldsymbol{K}\hat{\boldsymbol{q}} + \boldsymbol{P}(\hat{\boldsymbol{q}}) - \boldsymbol{H}_f(\hat{\boldsymbol{q}})\boldsymbol{\Lambda}_f = \boldsymbol{0} & \text{(66a)} \\ \boldsymbol{G}_f(\hat{\boldsymbol{q}}) = \boldsymbol{0} & \text{(66b)} \end{cases}$$

where $\boldsymbol{P}(\hat{\boldsymbol{q}})$ is defined in (56), meaning the gradient of $V_g(\hat{\boldsymbol{q}})$. $\boldsymbol{H}_f(\hat{\boldsymbol{q}})$ is defined in (56), meaning the transpose of the Jacobian of $\boldsymbol{G}_f(\hat{\boldsymbol{q}})$. $\boldsymbol{\Lambda}_f$ is the Lagrange multiplier, meaning the contact force.

C. Sticking test

We first valid our proposed model through comparing the deformation and contact force both obtained from experiment and simulation under the stick state. As shown in Fig. 25, the base of the soft robot is pressed slowly by the rigid manipulator while the tip of the soft robot is always at stick contact state with the planar of table due to the friction during the total process. We compare the deformation of experiment and simulation through the observed strain of real soft robot and the strain of simulation. To determine the coefficient of friction between the silicone and the tabletop surface, we conducted experiments using silicone block made from the same material as the robot. The block was placed on the tabletop, and a pushing force was applied to initiate movement. By measuring the ratio of the applied force to the weight of the silicone gel blocks, we were able to calculate the coefficient of friction. After performing several experiments, we obtained a measured coefficient of friction of 0.83 between the silicone block and the tabletop surface, which will be used in the simulation. Fig. 26 shows the evolution of strain at control points. Fig. 27 shows the evolution of the total normal contact force of soft robot. The test is repeated 5 times and the average error percentages of both strain and force from tests are less than 10%.

D. Sliding test

Our second test aims to valid the proposed model in aspect of dynamic evolution from sticking state to sliding state. In this test, for the sake of a obvious conversion phenomena, we decrease the coefficient of friction by lubricating some lube on the surface of table. The coefficient of friction is measured as 0.48. As shown in Fig. 28(a), the soft robot is first pre-curved by the rigid manipulator and is at stick state. Then the rigid manipulator moves the base of soft robot away from the sticking contact point following the planned trajectory. The change of configuration of soft robot will lead to the transition of contact between soft robot and table from stick to slide. Fig. 28(b)-(d) shows the dynamic evolution of this phenomena during the experiment on comparing with the simulation result. Fig. 29 shows the evolution of strain at control points. Fig. 30 shows the evolution of the slide distance of tip. The test is repeated 5 times and the average error percentages of both strain and sliding distance from texts are less than 15%.

XII. CONCLUSION

In this article, we propose a contact dynamics model for soft robots based on the Cosserat theory. The contact dynamics

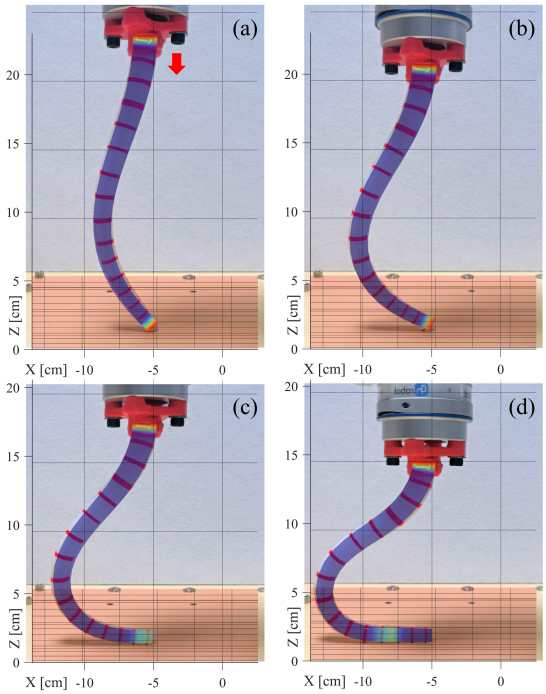


Fig. 25. Configurations of sticking test and simulation.

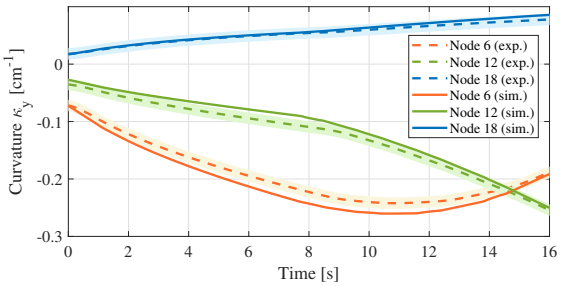


Fig. 26. Evolution of strain of sticking test and simulation. Shaded area represents measurement errors.

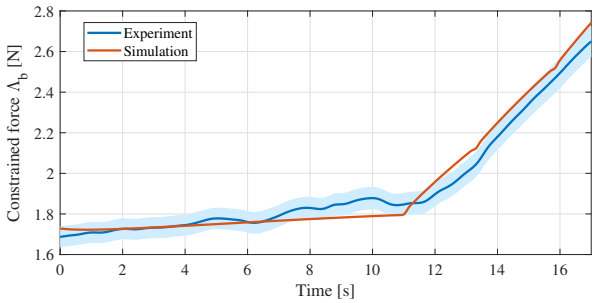


Fig. 27. Evolution of contact force of sticking test and simulation. Shaded area represents measurement errors.

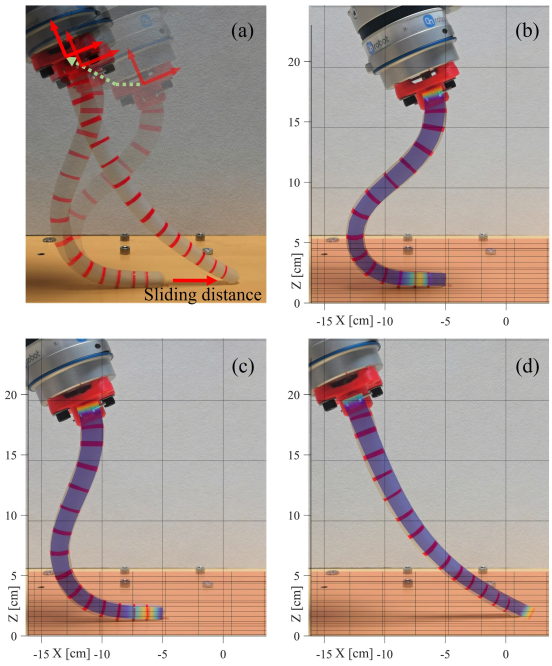


Fig. 28. Configurations of sliding test and simulation.

formulation considers the geometric relationships, forces, and moments at the contact points, allowing for the analysis of contact forces, frictional forces, and their effects on the robot's dynamics.

By incorporating the Cosserat theory into the contact dynamics model, we achieve a more comprehensive and accurate representation of the contact interactions in soft robots. This enables us to study and analyze the behavior and performance of soft robots under different contact conditions, facilitating the design and control of soft slender robot systems for various applications. In our future endeavors, we will expand upon the contact dynamics framework we have proposed and shift our focus towards modeling and analyzing soft slender robots that incorporate actuators, such as cable-driven, tendon-driven, and magnetic-driven mechanisms. Specifically, we will tackle the control challenges that arise when soft slender robots interact with the external environment. By incorporating these drive systems, our aim is to enhance the understanding and capabilities of soft slender robot control in a wide range of real-world applications.

Overall, the proposed contact dynamics model based on the Cosserat theory provides a valuable framework for understanding and simulating the contact behavior of soft robots, advancing the field of soft slender robotics and its applications.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (Grant No. 62073081), by the ANR roject ROBOCOP [ANR-19-CE19], and by the ANR project COSSEROOTS [ANR-20-CE33].

References

[1] C. Laschi, B. Mazzolai, and M. Cianchetti, "Soft robotics: Technologies and systems pushing the boundaries of robot abilities," *Science robotics*, vol. 1, no. 1, p. eaah3690, 2016.

[2] D. Trivedi, C. D. Rahn, W. M. Kier, and I. D. Walker, "Soft robotics: Biological inspiration, state of the art, and future research," *Applied bionics and biomechanics*, vol. 5, no. 3, pp. 99–117, 2008.

[3] D. Rus and M. T. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, no. 7553, pp. 467–475, 2015.

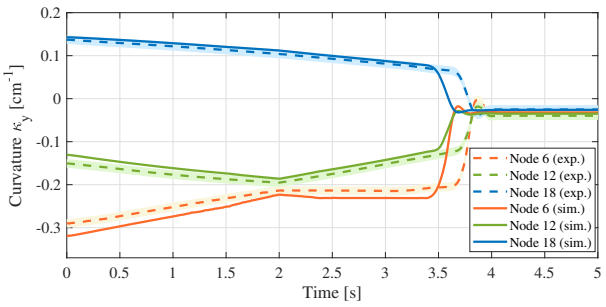


Fig. 29. Evolution of strain of sliding test and simulation. Shaded area represents measurement errors.

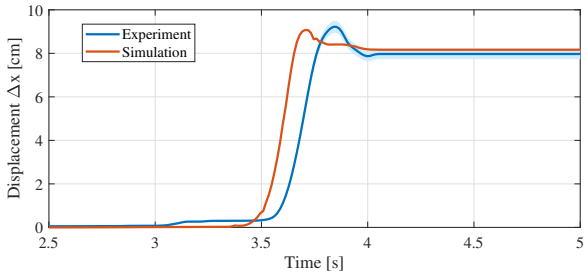


Fig. 30. Evolution of sliding displacement of sliding test and simulation. Shaded area represents measurement errors.

[4] G. Dogangil, B. Davies, and F. Rodriguez y Baena, "A review of medical robotics for minimally invasive soft tissue surgery," *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, vol. 224, no. 5, pp. 653–679, 2010.

[5] M. Calisti, G. Picardi, and C. Laschi, "Fundamentals of soft robot locomotion," *Journal of The Royal Society Interface*, vol. 14, no. 130, p. 20170101, 2017.

[6] M. Cianchetti, C. Laschi, A. Menciassi, and P. Dario, "Biomedical applications of soft robotics," *Nature Reviews Materials*, vol. 3, no. 6, pp. 143–153, 2018.

[7] Y. Gonthier, "Contact dynamics modelling for robotic task simulation," 2007.

[8] T. M. Bieze, F. Largilliere, A. Kruszewski, Z. Zhang, R. Merzouki, and C. Duriez, "Finite element method-based kinematics and closed-loop control of soft, continuum manipulators," *Soft robotics*, vol. 5, no. 3, pp. 348–364, 2018.

[9] G. Olson, R. L. Hatton, J. A. Adams, and Y. Mengüç, "An euler–bernoulli beam model for soft robot arms bent through self-stress and external loads," *International Journal of Solids and Structures*, vol. 207, pp. 113–131, 2020.

[10] F. Renda, M. Cianchetti, M. Giorelli, A. Arienti, and C. Laschi, "A 3d steady-state model of a tendon-driven continuum soft manipulator inspired by the octopus arm," *Bioinspiration & biomimetics*, vol. 7, no. 2, p. 025006, 2012.

[11] L. Lindenroth, J. Back, A. Schoisengeier, Y. Noh, H. Würdemann, K. Althoefer, and H. Liu, "Stiffness-based modelling of a hydraulically-actuated soft robotics manipulator," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 2458–2463.

[12] H. Godaba, F. Putzu, T. Abrar, J. Konstantinova, and K. Althoefer, "Payload capabilities and operational limits of eversion robots," in *Towards Autonomous Robotic Systems: 20th Annual Conference, TAROS 2019, London, UK, July 3–5, 2019, Proceedings, Part II 20*. Springer, 2019, pp. 383–394.

[13] F. Boyer and F. Renda, "Poincare's equations for cosserat media: Application to shells," *Journal of Nonlinear Science*, vol. 27, pp. 1–44, 2017.

[14] D.-Q. Cao and R. W. Tucker, "Nonlinear dynamics of elastic rods using the cosserat theory: Modelling and simulation," *International Journal of Solids and Structures*, vol. 45, no. 2, pp. 460–477, 2008.

[15] Y. Haibin, K. Cheng, L. Junfeng, and Y. Guilin, "Modeling of grasping force for a soft robotic gripper with variable stiffness," *Mechanism and Machine Theory*, vol. 128, pp. 254–274, 2018.

[16] J. D. Till, "On the statics, dynamics, and stability of continuum robots: Model formulations and efficient computational schemes," 2019.

[17] J. Till, V. Aloi, and C. Rucker, "Real-time dynamics of soft and continuum robots based on cosserat rod models," *The International Journal of Robotics Research*, vol. 38, no. 6, pp. 723–746, 2019.

[18] F. Renda, F. Boyer, J. Dias, and L. Seneviratne, "Discrete cosserat approach for multisection soft manipulator dynamics," *IEEE Transactions on Robotics*, vol. 34, no. 6, pp. 1518–1533, 2018.

[19] F. Boyer, V. Lebastard, F. Candelier, and F. Renda, "Dynamics of continuum and soft robots: A strain parameterization based approach," *IEEE Transactions on Robotics*, vol. 37, no. 3, pp. 847–863, 2020.

[20] F. Renda, M. Giorelli, M. Calisti, M. Cianchetti, and C. Laschi, "Dynamic model of a multibending soft robot arm driven by cables," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1109–1122, 2014.

[21] X. Zhang, F. K. Chan, T. Parthasarathy, and M. Gazzola, "Modeling and simulation of complex dynamic musculoskeletal architectures," *Nature communications*, vol. 10, no. 1, p. 4825, 2019.

[22] H. Wang, M. Totaro, and L. Beccai, "Toward perceptive soft robots: Progress and challenges," *Advanced Science*, vol. 5, no. 9, p. 1800541, 2018.

[23] C. Laschi, M. Cianchetti, B. Mazzolai, L. Margheri, M. Follador, and P. Dario, "Soft robot arm inspired by the octopus," *Advanced robotics*, vol. 26, no. 7, pp. 709–727, 2012.

[24] T. Patino, R. Mestre, and S. Sanchez, "Miniaturized soft bio-hybrid robotics: a step forward into healthcare applications," *Lab on a Chip*, vol. 16, no. 19, pp. 3626–3630, 2016.

[25] M. Luo, M. Agheli, and C. D. Onal, "Theoretical modeling and experimental analysis of a pressure-operated soft robotic snake," *Soft Robotics*, vol. 1, no. 2, pp. 136–146, 2014.

[26] X. Zhang, N. Naughton, T. Parthasarathy, and M. Gazzola, "Friction modulation in limbless, three-dimensional gaits and heterogeneous terrains," *Nature communications*, vol. 12, no. 1, p. 6076, 2021.

[27] T. Shahid, D. Gouwanda, S. G. Nurzaman, and A. A. Gopalai, "Moving toward soft robotics: A decade review of the design of hand exoskeletons," *Biomimetics*, vol. 3, no. 3, p. 17, 2018.

[28] M. Xiloyannis, R. Alicea, A.-M. Georgarakis, F. L. Haufe, P. Wolf, L. Masia, and R. Riener, "Soft robotic suits: State of the art, core technologies, and open challenges," *IEEE Transactions on Robotics*, vol. 38, no. 3, pp. 1343–1362, 2021.

[29] P. Schegg and C. Duriez, "Review on generic methods for mechanical modeling, simulation and control of soft robots," *Plos one*, vol. 17, no. 1, p. e0251059, 2022.

[30] D. E. Stewart, "Rigid-body dynamics with friction and impact," *SIAM review*, vol. 42, no. 1, pp. 3–39, 2000.

[31] T. M. Wasfy and A. K. Noor, "Computational strategies for flexible multibody systems," *Appl. Mech. Rev.*, vol. 56, no. 6, pp. 553–613, 2003.

[32] A. Munjiza and K. Andrews, "Penalty function method for combined finite–discrete element systems comprising large number of separate bodies," *International Journal for numerical methods in engineering*, vol. 49, no. 11, pp. 1377–1396, 2000.

[33] J. Spillmann, M. Becker, and M. Teschner, "Non-iterative computation of contact forces for deformable objects," 2007.

[34] V. J. Milenkovic and H. Schmidl, "Optimization-based animation," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001, pp. 37–46.

[35] D. M. Kaufman, T. Edmunds, and D. K. Pai, "Fast frictional dynamics for rigid bodies," in *ACM SIGGRAPH 2005 Papers*, 2005, pp. 946–956.

[36] D. E. Stewart and J. C. Trinkle, "An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction," *International Journal for Numerical Methods in Engineering*, vol. 39, no. 15, pp. 2673–2691, 1996.

[37] K. Erleben, "Velocity-based shock propagation for multibody dynamics animation," *ACM Transactions on Graphics (TOG)*, vol. 26, no. 2, pp. 12–es, 2007.

[38] C. Duriez, F. Dubois, A. Kheddar, and C. Andriot, "Realistic haptic rendering of interacting deformable objects in virtual environments," *IEEE transactions on visualization and computer graphics*, vol. 12, no. 1, pp. 36–47, 2005.

[39] M. A. Otaduy, R. Tamstorf, D. Steinemann, and M. Gross, "Implicit contact handling for deformable objects," in *Computer Graphics Forum*, vol. 28, no. 2. Wiley Online Library, 2009, pp. 559–568.

[40] D. Harmon, E. Vouga, B. Smith, R. Tamstorf, and E. Grinspun, "Asynchronous contact mechanics," in *ACM SIGGRAPH 2009 papers*, 2009, pp. 1–12.

[41] S. Niebe and K. Erleben, "Numerical methods for linear complementarity problems in physics-based animation," *Synthesis Lectures on Computer Graphics and Animation*, vol. 7, no. 1, pp. 1–159, 2015.

[42] P. Alart and A. Curnier, "A mixed formulation for frictional contact problems prone to newton like solution methods," *Computer methods in applied mechanics and engineering*, vol. 92, no. 3, pp. 353–375, 1991.

[43] A. Curnier and P. Alart, "A generalized newton method for contact problems with friction," *Journal de mécanique théorique et appliquée*, 1988.

[44] S. P. Dirkse and M. C. Ferris, "The path solver: a nommonotone stabilization scheme for mixed complementarity problems," *Optimization methods and software*, vol. 5, no. 2, pp. 123–156, 1995.

[45] F. Bertails-Descoubes, F. Cadoux, G. Daviet, and V. Acary, "A non-smooth newton solver for capturing exact coulomb friction in fiber assemblies," *ACM Transactions on Graphics (TOG)*, vol. 30, no. 1, pp. 1–14, 2011.

[46] G. Daviet, F. Bertails-Descoubes, and L. Boissieux, "A hybrid iterative solver for robustly capturing coulomb friction in hair dynamics," in *Proceedings of the 2011 SIGGRAPH Asia Conference*, 2011, pp. 1–12.

[47] D. M. Kaufman, R. Tamstorf, B. Smith, J.-M. Aubry, and E. Grinspun, "Adaptive nonlinearity for collisions in complex rod assemblies," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, pp. 1–12, 2014.

[48] V. Acary and B. Brogliato, *Numerical methods for nonsmooth dynamical systems: applications in mechanics and electronics*. Springer Science & Business Media, 2008.

[49] C. Armanini, F. Boyer, A. T. Mathew, C. Duriez, and F. Renda, "Soft robots modeling: A structured overview," *IEEE Transactions on Robotics*, 2023.

[50] H. Li, L. Xun, and G. Zheng, "Piecewise linear strain cosserat model for soft slender manipulator," *IEEE Transactions on Robotics*, pp. 1–18, 2023.

[51] K. L. Johnson and K. L. Johnson, *Contact mechanics*. Cambridge university press, 1987.

[52] C. Armanini, I. Hussain, M. Z. Iqbal, D. Gan, D. Prattichizzo, and F. Renda, "Discrete cosserat approach for closed-chain soft robots: Application to the fin-ray finger," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 2083–2098, 2021.

[53] P. Wriggers and T. A. Laursen, *Computational contact mechanics*. Springer, 2006, vol. 2.

[54] E. Drumwright and D. A. Shell, "Modeling contact friction and joint friction in dynamic robotic simulation using the principle of maximum dissipation," in *Algorithmic Foundations of Robotics IX: Selected Contributions of the Ninth International Workshop on the Algorithmic Foundations of Robotics*. Springer, 2011, pp. 249–266.

[55] O. L. Mangasarian, "Equivalence of the complementarity problem to a system of nonlinear equations," *SIAM Journal on Applied Mathematics*, vol. 31, no. 1, pp. 89–92, 1976.

[56] O. Mangasarian and C. Chen, "A class of smoothing functions for nonlinear and mixed complementarity problems," Tech. Rep., 1994.

[57] C. B. Black, J. Till, and D. C. Rucker, "Parallel continuum robots: Modeling, analysis, and actuation-based force sensing," *IEEE Transactions on Robotics*, vol. 34, no. 1, pp. 29–47, 2017.

[58] Y. H. Cheng and J. Wang, "A motion image detection method based on the inter-frame difference method," in *Applied Mechanics and Materials*, vol. 490. Trans Tech Publ, 2014, pp. 1283–1286.

[59] F. Boyer, V. Lebastard, F. Candelier, and F. Renda, "Dynamics of continuum and soft robots: A strain parameterization based approach," *IEEE Transactions on Robotics*, vol. 37, no. 3, pp. 847–863, 2021.

[60] Z. Zhu and K. Zhang, "A superlinearly convergent sqp algorithm for mathematical programs with linear complementarity constraints," *Applied mathematics and computation*, vol. 172, no. 1, pp. 222–244, 2006.

[61] F. Facchinei and J. Soares, "A new merit function for nonlinear complementarity problems and a related algorithm," *SIAM Journal on Optimization*, vol. 7, no. 1, pp. 225–247, 1997.

[62] D. G. Luenberger, Y. Ye *et al.*, *Linear and nonlinear programming*. Springer, 1984, vol. 2.

[63] E. Todorov, "Implicit nonlinear complementarity: A new approach to contact dynamics," in *2010 IEEE international conference on robotics and automation*. IEEE, 2010, pp. 2322–2329.

[64] T. S. Munson, F. Facchinei, M. C. Ferris, A. Fischer, and C. Kanzow, "The semismooth algorithm for large scale complementarity problems," *INFORMS Journal on Computing*, vol. 13, no. 4, pp. 294–311, 2001.

[65] W. Rossmann, *Lie groups: an introduction through linear groups*. Oxford University Press on Demand, 2006, vol. 5.

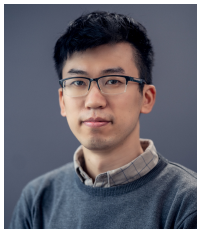
[66] A. Novelia, *Discrete Elastic Rods for Simulating Soft Robot Limbs*. University of California, Berkeley, 2018.

[67] S. Huang, D. Meng, X. Wang, B. Liang, and W. Lu, "A 3d static modeling method and experimental verification of continuum robots based on pseudo-rigid body theory," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 4672–4677.

[68] D. C. Rucker and R. J. Webster III, "Statics and dynamics of continuum robots with general tendon routing and external loading," *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1033–1044, 2011.

[69] D. M. Kaufman, S. Sueda, D. L. James, and D. K. Pai, "Staggered projections for frictional contact in multibody systems," in *ACM SIGGRAPH Asia 2008 papers*, 2008, pp. 1–11.

[70] R. Fletcher, *Practical methods of optimization*. John Wiley & Sons, 2013.



Lingxiao Xun received the B.E degree in mechanical engineering from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2017 and a M.E degree in Mechatronics system from ENSAM, France, in 2019. He is now a Ph.D. candidate in robotics at INRIA, Lille, France. His research interests include mechatronics, automation and robotics.

Gang Zheng received the B.E. and M.E. degrees in Communication and systems from Wuhan University, China, in 2001 and 2004, respectively, and the Ph.D. degree in automatic control from ENSEA, Cergy-Pontoise, France, in 2006. Since 2007, he has held postdoctoral positions at INRIA Grenoble, at the Laboratoire Jean Kuntzmann, and at ENSEA. He joined INRIA Lille as a permanent researcher from September 2009.

Alexandre Kruszewski (Full Professor, first-time participant to FET H2020) is a full professor at Centrale Lille Institutes. He is a member of the joint team (CRIStAL UMR 9189 – Inria) Defrost. He has a strong experience (both theoretical and practical) in the robust control design based on numerical optimization. Since 2015, his main research topic is the control design of soft robots based on Finite Element Methods.