Discrete Cosserat Static Model-Based Control of Soft Manipulator

Haihong Li, Lingxiao Xun, Gang Zheng, Federico Renda

Discrete Cosserat Static Model-Based Control of Soft Manipulator

Haihong Li, Lingxiao Xun, Gang Zheng, and Federico Renda

***Abstract*—In this paper, the continuous Cosserat static model is re-formulated by employing the piecewise linear strain (PLS) approach, which can obtain the analytic formula of the model to facilitate the controller design. The robust closed-loop control architecture based on the PLS Cosserat static model for the soft arm around its workspace is then established. To validate the performance of the proposed model-based control scheme, the point-to-point and time-varying trajectory tracking experiments under external disturbances have been conducted on the soft manipulator actuated by cables.**

***Index Terms*—Soft manipulator, PLS Cosserat static model, model-based control.**

I. Introduction

SOFT robots make full use of natural compliance through continuously deformable structure to simplify a wide range of well-known complex tasks. Compared to the traditional rigid robots, they are able to emulate numerous biological systems and result in an infinite number of degrees of freedom (DoFs), which can greatly reduce complexity in interactions with the unstructured environments. Due to the characteristics of being flexible and compliant, soft robots have been increasingly used in various fields such as simple gripping system [**?**], biomedical devices [**?**], surgical procedures [**?**], and so on. However, these advantages come at the cost of increased complexities in existing modeling and control frameworks.

The continuum mechanics of soft robots is more complicated owing to highly geometric nonlinearities. In comparison with the rigid-link robots, the controller design via the exact model of a soft robot is not only difficult, but also faced with implementation issues. One straightforward approach to model and control the soft robots is to obtain the simplified approximate models of the robotic motion [**?**]. Currently, most works on modeling still tend to utilize the piecewise constant curvature (PCC) which regards the beam as consisting of a sequence of continuous sections with constant curvature in space but variable with time [**?**], [**?**]. Despite the significant achievements, the constant curvature assumption is applied to the specific designs and actuators, therefore, it cannot be considered as a general method [**?**]. As an extension of the PCC modeling framework, the piecewise constant strain (PCS) Cosserat model [**?**] aims to reduce the dimension of the configuration space by discretizing the arc length into several sections with piecewise constant strain. Although PCS assumption can provide high precision, it essentially depends on the fine discretization which will in turn produce high-dimension system and take much time for numerical simulation [**?**]. As a result, PCS model will be an issue for designing model-based controllers. [**?**] proposed a variable strain (VS) Cosserat method to globally approximate the strain field. This approach can provide comparable precision when the number of basis function is high enough, however, the deduced models are no longer analytic compared to PCS, and will be not friendly for control design. In [**?**], a geometric variable strain (GVS) method for static modeling of soft manipulators with tendon and fluidic actuation has been presented, which extends the PCS approach to the case of variable strain. Finite element method (FEM) [**?**], providing a systematic and physics-based modeling approach, is another preferred numerical tool to model the kinematics and dynamics of soft robots. In contrast to the Cosserat method, the model of a soft robot established based on FEM is of very high dimension, and the control design using these models of infinite dimensions are difficult to achieve. Last but not least, [**?**] unified the advantages of PCS and VS to put forward a piecewise linear strain (PLS) Cosserat model with lower dimension and comparable accuracy, which is able to obtain the analytic solution of the model and facilitate the control design.

Another type of modeling method different from the above mentioned is to establish models directly using the relationship between input and output information [**?**], [**?**], [**?**], yet this technique fails to use physics-based principles, and the accuracy of these models are limited to the high data requirement.

In recent years, model-based controllers have been the most widely used and developed strategies for control of soft continuum robots. Although many researchers have proposed in-depth investigations of various control techniques based on different modeling methods, model-based feedback control schemes for soft robotics have remained an open challenge. The strategies for inverse kinematics-based control [**?**] directly use FEM models. [**?**] used asynchronous FEM and quadratic programming (QP) algorithm to obtain the inverse solution which is utilized to control the actuators. Nevertheless, the high-dimensional FEM makes the design of real-time controllers tough, which motivates the development of FEM model reduction methods well suitable for controller design [**?**]. In [**?**], the authors proposed a novel approach for model-based optimal control of soft robots via order-reduction FEM without loss in modeling accuracy. Under the PCC assumption,

Haihong Li, Lingxiao Xun and Gang Zheng are in Defrost team, Inria, university of Lille, Centrale Lille, CRIStAL - Centre de Recherche en Informatique Signal et Automatique de Lille - UMR 9189, France.

Federico Renda is with Khalifa University Center for Autonomous Robotic Systems, Khalifa University, Abu Dhabi.

the contributors of [**?**] presented a closed-loop control approach which enabled soft robots to track trajectories in three-dimensional space. [**?**] investigated an adaptive image-based visual servoing controller to track the reference image shape features based on PCC kinematic model. Some representative achievements about Cosserat-rod-based controllers have also been made. In [**?**], the approach for performing stiffness control on continuum robots modeled by the Cosserat rod was first proposed. Although the designed controller provided good dynamic performance and stability, it did not provide any theoretical convergence proof. [**?**] first presented a coupled Cosserat rod model for parallel continuum robots to realize real-time kinematic control, nevertheless, no control experiments were conducted. In [**?**], the Cosserat rod model was used to design robust controller for continuum robots for the first time. The authors of [**?**] coupled the Cosserat kinematic model with integrated sensing and the efficient Jacobian matrix to provide a practically feasible, closed-loop control system for application to the soft continuum manipulators. [**?**] presented a PCS model-based reinforcement learning algorithm for the global closed-loop predictive control of the soft manipulator. Since controllers employing second-order dynamic models tend to be computationally expensive, [**?**] proposed to reduce the PCS dynamic model under a first-order approximation assumption and developed closed-loop controllers by the use of the simplification of the planning and sensor feedback. For the work of [**?**], the GVS approach is used for Jacobian-based inverse kinematic control of general routing manipulator.

Until now, to our best of knowledge, there has been few controller design via the discrete static Cosserat models. In this work, we design a robust controller based on the PLS Cosserat static model to control the end-effector position of cable-driven soft manipulator in real-time. Compared to those existing results of model-based controllers for soft manipulators, the main contributions by this paper are as follows:

- The PLS Cosserat static model suitable for the real-time control is presented;
- A closed-loop control approach is proposed, and the rigorously theoretical convergence proof for the designed controller is provided;
- A series of physical experiments (i.e., point to point, time-varying trajectory tracking, and robustness analysis under external disturbances) on the soft manipulator made of silicone and actuated by 4 cables are conducted to evaluate the effectiveness and robustness of the proposed static model-based control scheme.

II. Problem statement

In this paper, we want to design a static model-based controller to achieve end-effector position control of the cable-driven soft manipulator. In general, it is difficult to derive the exact kinematic or static models for soft robots. However, one can make an attempt to approximate to the exact models by using the most common modeling techniques such as PCC, PCS and FEM, and then design the approximated model-based controller. It has be pointed out that there exists some drawbacks for these modeling methods [**?**], [**?**] such that these model-based controllers might be invalid for the real system. Here we are devoted to using the PLS Cosserat modeling method to approximate the input-output relationship of the soft arm, based on which robust model-based controller will be designed to control the end-effector position of the soft manipulator to reach the targeted areas.

III. PLS Cosserat static model

In this section, we present the piecewise linear strain (PLS) static model framework based on classical Cosserat rod theory for soft manipulator actuated by cables.

From the continuous Cosserat model, the space-time variations of the position field can be separately represented by the deformation field $\boldsymbol{\xi}(X)$ and velocity field $\boldsymbol{\eta}(X)$. The 6×1 strain twist $\boldsymbol{\xi}(X) = \left(\boldsymbol{K}^\top, \boldsymbol{Q}^\top\right)^\top$ includes the linear and angular strains, and the 6×1 velocity twist $\boldsymbol{\eta}(X) = \left(\boldsymbol{\Omega}^\top, \boldsymbol{V}^\top\right)^\top$ consists of the linear and angular velocities.

The PLS is a spatial discretization technique for solving the differential equations based on the continuous Cosserat model. This method divides the slender rod into several sections composed of a number of infinitesimal segments whose states are represented by constant strain twists, while the linearly varying ones are used to describe the states of the sections (i.e., the linear interpolation is introduced in PLS). For more details of this modeling method, one can refer to [**?**]. Under the PLS assumption, defining the generalized joint position vector $\boldsymbol{q} = [\bar{\boldsymbol{\xi}}_0^\top, \bar{\boldsymbol{\xi}}_1^\top, \bar{\boldsymbol{\xi}}_2^\top, \cdots, \bar{\boldsymbol{\xi}}_N^\top]^\top \in \mathbb{R}^{6(N+1)}$ composed of strain twists of all interpolation nodes (N being the total number of sections), the configuration and velocity of any cross section of the soft manipulator can be separately formulated as

$$\boldsymbol{g}(\boldsymbol{q}, X) = \begin{bmatrix} \boldsymbol{R}(\boldsymbol{q}, X) & \boldsymbol{u}(\boldsymbol{q}, X) \\ \boldsymbol{0} & 1 \end{bmatrix} \tag{1}$$

$$\boldsymbol{\eta}(\boldsymbol{q}, X) = \boldsymbol{J}(\boldsymbol{q}, X)\dot{\boldsymbol{q}} \tag{2}$$

where $\boldsymbol{R} \in SO(3)$ stands for the orientation matrix of the local frame w.r.t. the global frame, $\boldsymbol{u} \in \mathbb{R}^3$ is the position vector, and $\boldsymbol{J}(\boldsymbol{q}, X) \in \mathbb{R}^{6\times6(N+1)}$ represents the geometric Jacobian given in Appendix A, which relates the time derivative of the joint position vector to the velocity twist.

For the PLS Cosserat model, the strain field can be reformulated as

$$\boldsymbol{\xi}(X) = \boldsymbol{\xi}_0 + \boldsymbol{\Phi}(X)\boldsymbol{q}(t)$$

where $\boldsymbol{\Phi}(X) \in \mathbb{R}^{6\times6(N+1)}$ is comprised of coefficient matrices of the strain interpolation nodes, $\boldsymbol{\xi}_0$ represents the undeformed reference shape the slender rod takes.

To obtain the discrete Cosserat statics, the principle of virtual work is used, and the weak form of the continuous static system can be then obtained. The linear elastic constitutive law relating the internal wrench to strain twists is introduced to fully constrain the ODE system, and the force as well as torque exerted by cables on the midline of the rod are calculated by using the method of [**?**]. Thus, the PLS Cosserat static model yields [**?**]:

$$\boldsymbol{K}\boldsymbol{q} + \boldsymbol{G}(\boldsymbol{q})\mathrm{Ad}_{g_r}^{-1}\boldsymbol{\mathcal{G}} = -\boldsymbol{H}\boldsymbol{T} \tag{3}$$

with

$$\boldsymbol{K} = \overline{\boldsymbol{\mathcal{P}}}^\top \int_0^{L_N} \boldsymbol{\Phi}^\top \boldsymbol{\mathcal{H}}(X)\boldsymbol{\Phi}\mathrm{d}X$$

$$\boldsymbol{G}(\boldsymbol{q}) = -\overline{\boldsymbol{\mathcal{P}}}^{\top} \int_0^{L_N} \boldsymbol{J}^{\top} \boldsymbol{\mathcal{M}}(X) \mathrm{Ad}_{\boldsymbol{g}(X)}^{-1} \mathrm{d}X$$

and

$$\boldsymbol{H} = \overline{\boldsymbol{\mathcal{P}}}^{\top} \int_0^{L_N} \boldsymbol{\Phi}^{\top} \boldsymbol{\Lambda}(X) \mathrm{d}X$$

where $\boldsymbol{K} \in \mathbb{R}^{6N \times 6(N+1)}$ is the stiffness matrix, $\boldsymbol{H} \in \mathbb{R}^{6N \times s}$ represents the actuation matrix, which are all constant matrices independent of $\boldsymbol{q}$; $\boldsymbol{G}(\boldsymbol{q}) \in \mathbb{R}^{6N \times 6}$ is the gravitational matrix, Ad is the Adjoint operator of the Lie group transformation, $\boldsymbol{g}_r$ is the transformation matrix between the inertial frame and the manipulator base frame, $\boldsymbol{\mathcal{G}} \in \mathbb{R}^6$ denotes the gravity acceleration twist, and $\boldsymbol{T} \in \mathbb{R}^s$ consists of tension magnitude of s cables acting on the manipulator; $\overline{\boldsymbol{\mathcal{P}}} = \begin{bmatrix} \mathbf{I}_{6N \times 6N}^{\top} & \mathbf{0}_{6 \times 6N}^{\top} \end{bmatrix}^{\top}$ is the selection matrix of the nodal strain twists for the purpose of making the tip strain twist $\overline{\boldsymbol{\xi}}_N$ not limited by the relation between the virtual displacement and the state vector, but only constrained by the boundary condition of the end cross section; $\mathrm{ad}_{(\cdot)}$ denotes the adjoint operator of Lie algebra expressed in Appendix B; $\boldsymbol{\mathcal{H}}(X) \in \mathbb{R}^{6 \times 6}$ is the stiffness matrix for each cross section which could vary with arc length X, $\boldsymbol{\mathcal{M}}(X) \in \mathbb{R}^{6 \times 6}$ is the mass matrix for each cross section, and $\boldsymbol{\Lambda}(X) \in \mathbb{R}^{6 \times s}$ stands for a matrix function whose column vectors transform the cables' tension from the cable frame to the midline of the arm.

Finally, we should take into account both the boundary condition of the end cross section and the constitutive law to complement the system (3), which are as follows:

$$\begin{aligned} \boldsymbol{\mathcal{F}}_{ie}(L_N) + \boldsymbol{\mathcal{F}}_{ia}(L_N) &= \boldsymbol{\mathcal{F}}_e(L_N) \\ \boldsymbol{\mathcal{F}}_{ie} &= \boldsymbol{\mathcal{H}}(X)(\boldsymbol{\xi}(X) - \boldsymbol{\xi}_0) \end{aligned}$$

with $\boldsymbol{\mathcal{F}}_{ia} = \boldsymbol{\Lambda}(X)\boldsymbol{T}$, where $\boldsymbol{\mathcal{F}}_{ie} \in \mathbb{R}^6$ is the internally elastic wrench, $\boldsymbol{\mathcal{F}}_{ia} \in \mathbb{R}^6$ denotes the internal wrench produced by the cable actuation, $\boldsymbol{\mathcal{F}}_e(L_N) \in \mathbb{R}^6$ is the tip external wrench at $X = L_N$.

Consequently, the generalized PLS Cosserat static model can be then written as the following nonlinear algebraic equation

$$\underline{\boldsymbol{K}}\boldsymbol{q} + \underline{\boldsymbol{G}}(\boldsymbol{q}) = \underline{\boldsymbol{H}}\boldsymbol{T} \tag{4}$$

with

$$\underline{\boldsymbol{K}} = \begin{bmatrix} \boldsymbol{K} \\ \boldsymbol{\mathcal{H}}(L_N)\boldsymbol{\Phi}(L_N) \end{bmatrix}, \ \underline{\boldsymbol{G}}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{G}(\boldsymbol{q})\mathrm{Ad}_{\boldsymbol{g}_r}^{-1}\boldsymbol{\mathcal{G}} \\ -\boldsymbol{\mathcal{F}}_e(L_N) \end{bmatrix}, \ \underline{\boldsymbol{H}} = - \begin{bmatrix} \boldsymbol{H} \\ \boldsymbol{\Lambda}(L_N) \end{bmatrix}.$$

Remark 1: It should be emphasized that, due to PLS assumption, the accurate modeling result can be achieved with less DoFs compared to the common PCS Cosserat modeling technique even for complicated deformations. By means of the PLS Cosserat method, the static system with the boundary condition is converted into an equivalent but simplified nonlinear algebraic equation (4) with a finite number of DoFs, which greatly facilitates the analysis and design of control laws.

Remark 2: For the soft slender manipulator modeled by PLS Cosserat, the strain-stress relation is linear, with constant Young's modulus. Therefore, the stiffness matrix in the constitutive law is independent of the 6×1 strain twist. In the PLS framework, all types of deformation, including compression/elongation, can be modeled. It is true that the linear strain-stress assumption will introduce certain approximation error in the static PLS model, especially when large deformation occurs. Such an approximation error might be eliminated or attenuated by closed-loop control.

IV. Robust controller design

The aim of this section is to present an approach to design closed-loop controller for the soft manipulator via the PLS Cosserat static model. Note that we are interested in the end-effector position control of the manipulator, and the problem can be considered as an output regulation of the following nonlinear Cosserat static model

$$\underline{\boldsymbol{F}}(\boldsymbol{q}) = \underline{\boldsymbol{H}}\boldsymbol{T} \tag{5a}$$

$$\boldsymbol{u}(\boldsymbol{q}) = \mathbf{P}\, \boldsymbol{g}(\boldsymbol{q}, L)\, \mathbf{A} \tag{5b}$$

with $\underline{\boldsymbol{F}}(\boldsymbol{q}) = \underline{\boldsymbol{K}}\boldsymbol{q} + \underline{\boldsymbol{G}}(\boldsymbol{q})$, $\mathbf{P} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} \end{bmatrix}$, $\mathbf{A} = \begin{bmatrix} \mathbf{0}_3 & 1 \end{bmatrix}^{\top}$, where $\boldsymbol{g}(\boldsymbol{q}, L)$ stands for the position and orientation of the end-effector, $\boldsymbol{u}(\boldsymbol{q})$ implies the measurable end-effector position which needs to be controlled with well pre-defined matrices $\mathbf{P}$ and $\mathbf{A}$.

Assumption 1: For the investigated soft manipulator, it is assumed that

1. the workspace of the soft manipulator is bounded, i.e., $\boldsymbol{u}(\boldsymbol{q}) \in \boldsymbol{\Xi}$, where $\boldsymbol{\Xi}$ stands for the workspace boundedness.
2. the control input $\boldsymbol{T}$ is physically bounded, i.e., $\boldsymbol{T} \in \mathcal{T}$. where $\mathcal{T} = [\underline{T}_1, \overline{T}_1] \times [\underline{T}_2, \overline{T}_2] \times \cdots \times [\underline{T}_s, \overline{T}_s]$ represents the boundedness of magnitude of each cable's tension, and $\underline{\boldsymbol{T}} = [\underline{T}_1, \underline{T}_2, \cdots, \underline{T}_s]^{\top}$ as well as $\overline{\boldsymbol{T}} = [\overline{T}_1, \overline{T}_2, \cdots, \overline{T}_s]^{\top}$ denote the minimum and maximum bound vectors composed of each cable's tension, respectively.
3. the control input $\boldsymbol{T}$ and the output $\boldsymbol{u}(\boldsymbol{q})$ are differentiable, i.e., $\boldsymbol{u}(\boldsymbol{q}), \boldsymbol{T} \in \mathcal{C}^{\infty}$.

Based on the robust control design scheme (5) for the PLS Cosserat static model, the relation between the position of the end-effector $\boldsymbol{u}(\boldsymbol{q})$ and the cables' tension $\boldsymbol{T}$ can be deduced as follows:

$$\dot{\boldsymbol{u}}(\boldsymbol{q}) = \boldsymbol{\Gamma}(\boldsymbol{q})\dot{\boldsymbol{T}} \tag{6}$$

where $\boldsymbol{\Gamma}(\boldsymbol{q}) \in \mathbb{R}^{3 \times s}$ takes the partial derivative of $\boldsymbol{u}$ with respect to $\boldsymbol{T}$, which describes the mechanical coupling between end-effector of the robot and its actuators. In what follows, the calculation of $\boldsymbol{\Gamma}(\boldsymbol{q})$ is exhaustively deduced.

A. Analytic Solution of $\boldsymbol{\Gamma}(\boldsymbol{q})$

According to its definition, $\boldsymbol{\Gamma}(\boldsymbol{q})$ can be formulated as

$$\boldsymbol{\Gamma}(\boldsymbol{q}) = \frac{\partial \boldsymbol{u}(\boldsymbol{q})}{\partial \boldsymbol{T}} = \frac{\partial \boldsymbol{u}(\boldsymbol{q})}{\partial \boldsymbol{q}} \frac{\partial \boldsymbol{q}}{\partial \boldsymbol{T}} = \boldsymbol{u}_{\boldsymbol{q}}(\boldsymbol{q})\boldsymbol{q}_{\boldsymbol{T}} \tag{7}$$

where $\boldsymbol{u}_{\boldsymbol{q}}(\boldsymbol{q}) = \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{q}}$ and $\boldsymbol{q}_{\boldsymbol{T}} = \frac{\partial \boldsymbol{q}}{\partial \boldsymbol{T}}$.

In the light of the matrix representation of the velocity twist $\widehat{\boldsymbol{\eta}}(\boldsymbol{q}, L)$ of the end-effector, the linear velocity $\boldsymbol{V}(\boldsymbol{q}, L)$ in the local frame can be derived

$$\boldsymbol{V}(\boldsymbol{q}, L) = \boldsymbol{R}^{-1}(\boldsymbol{q}, L)\dot{\boldsymbol{u}}(\boldsymbol{q}) = \mathbf{P}\, \widehat{\boldsymbol{\eta}}(\boldsymbol{q}, L)\, \mathbf{A}$$

with $\widehat{\boldsymbol{\eta}}(\boldsymbol{q},L) = \boldsymbol{g}^{-1}(\boldsymbol{q},L)\ \dot{\boldsymbol{g}}(\boldsymbol{q},L)$, where $\widehat{(\cdot)}$ denotes the standard isomorphic mapping from $\mathbb{R}^6$ to $se(3)$. Using the definition of the configuration matrix $\boldsymbol{g}(\boldsymbol{q},L)$, the term $\mathbf{P}\widehat{\boldsymbol{\eta}}(\boldsymbol{q},L)\mathbf{A}$ can be then re-formulated, and it yields

$$\begin{aligned}\mathbf{P}\widehat{\boldsymbol{\eta}}(\boldsymbol{q},L)\ \mathbf{A} &= \boldsymbol{P}\boldsymbol{g}^{-1}(\boldsymbol{q},L)\ \frac{\partial \boldsymbol{g}(\boldsymbol{q},L)}{\partial \boldsymbol{q}}\ (\mathbf{I}_4 \otimes \dot{\boldsymbol{q}})\ \mathbf{A} \\ &= \begin{bmatrix}\boldsymbol{R}^{-1}(\boldsymbol{q},L) & -\boldsymbol{R}^{-1}(\boldsymbol{q},L)\boldsymbol{u}\end{bmatrix} \begin{bmatrix}\frac{\partial \boldsymbol{R}(\boldsymbol{q},L)}{\partial \boldsymbol{q}} & \frac{\partial \boldsymbol{u}(\boldsymbol{q})}{\partial \boldsymbol{q}} \\ \mathbf{0} & \mathbf{0}\end{bmatrix} \begin{bmatrix}\mathbf{0} \\ \dot{\boldsymbol{q}}\end{bmatrix} \\ &= \boldsymbol{R}^{-1}(\boldsymbol{q},L)\frac{\partial \boldsymbol{u}(\boldsymbol{q})}{\partial \boldsymbol{q}}\dot{\boldsymbol{q}}\end{aligned}$$

where the operator $\otimes$ is the Kronecker tensor product.

Substituting the above equation of $\mathbf{P}\widehat{\boldsymbol{\eta}}(\boldsymbol{q},L)\mathbf{A}$ into the linear velocity in the global frame, the following expression of $\dot{\boldsymbol{u}}(\boldsymbol{q})$ can be obtained

$$\dot{\boldsymbol{u}}(\boldsymbol{q}) = \frac{\partial \boldsymbol{u}(\boldsymbol{q})}{\partial \boldsymbol{q}}\ \dot{\boldsymbol{q}} \tag{8}$$

Combining the definition of velocity twist with (2), the relation between velocity twist $\boldsymbol{\eta}(\boldsymbol{q},L)$ and geometric Jacobian $\boldsymbol{J}(\boldsymbol{q},L)$ is given by

$$\boldsymbol{\eta}(\boldsymbol{q},L) = \begin{bmatrix}\boldsymbol{\Omega}^\top(\boldsymbol{q}) & (\boldsymbol{R}^{-1}(\boldsymbol{q},L)\dot{\boldsymbol{u}}(\boldsymbol{q}))^\top\end{bmatrix}^\top = \boldsymbol{J}(\boldsymbol{q},L)\ \dot{\boldsymbol{q}}$$

which arrives at

$$\dot{\boldsymbol{u}}(\boldsymbol{q}) = \boldsymbol{R}(\boldsymbol{q},L)[\mathbf{0}_3,\ \mathbf{I}_3]\ \boldsymbol{J}(\boldsymbol{q},L)\ \dot{\boldsymbol{q}} \tag{9}$$

Using the expressions (8) and (9), the following equation can be deduced

$$\frac{\partial \boldsymbol{u}(\boldsymbol{q})}{\partial \boldsymbol{q}}\ \dot{\boldsymbol{q}} = \boldsymbol{R}(\boldsymbol{q},L)\ [\mathbf{0}_3,\ \mathbf{I}_3]\ \boldsymbol{J}(\boldsymbol{q},L)\ \dot{\boldsymbol{q}}$$

Accordingly, the partial derivative of $\boldsymbol{u}(\boldsymbol{q})$ with respect to $\boldsymbol{q}$ is given by

$$\begin{aligned}\boldsymbol{u}_{\boldsymbol{q}}(\boldsymbol{q}) = \frac{\partial \boldsymbol{u}(\boldsymbol{q})}{\partial \boldsymbol{q}} &= \boldsymbol{R}(\boldsymbol{q},L)\ [\mathbf{0}_3,\ \mathbf{I}_3]\boldsymbol{J}(\boldsymbol{q},L) \\ &= [\mathbf{0}_3, \boldsymbol{R}(\boldsymbol{q},L)]\boldsymbol{J}(\boldsymbol{q},L)\end{aligned}$$

Using the principle of variable separation, the above equation is equivalent to

$$\frac{\partial[\underline{\boldsymbol{F}}(\boldsymbol{q})]}{\partial \boldsymbol{q}}\frac{\partial \boldsymbol{q}}{\partial \boldsymbol{T}} = \underline{\boldsymbol{H}}$$

From above equation, we can obtain

$$\left(\underline{\boldsymbol{K}} + \frac{\partial[\underline{\boldsymbol{G}}(\boldsymbol{q})]}{\partial \boldsymbol{q}}\right)\frac{\partial \boldsymbol{q}}{\partial \boldsymbol{T}} = \underline{\boldsymbol{H}}$$

It is then supposed that the cables are installed in a manner such that the matrix $\left(\underline{\boldsymbol{K}} + \frac{\partial[\underline{\boldsymbol{G}}(\boldsymbol{q})]}{\partial \boldsymbol{q}}\right)$ deduced is invertible (if it is not, we then use pseudo-inverse), then the partial derivative of $\boldsymbol{q}$ with respect to $\boldsymbol{T}$ can be written as

$$\frac{\partial \boldsymbol{q}}{\partial \boldsymbol{T}} = \left(\underline{\boldsymbol{K}} + \frac{\partial[\underline{\boldsymbol{G}}(\boldsymbol{q})]}{\partial \boldsymbol{q}}\right)^\dagger \underline{\boldsymbol{H}}$$

where $(\cdot)^\dagger$ implies the pseudo-inverse of $(\cdot)$. Finally, the analytic solution of $\boldsymbol{\Gamma}(\boldsymbol{q})$ yields

$$\boldsymbol{\Gamma}(\boldsymbol{q}) = \boldsymbol{u}_{\boldsymbol{q}}(\boldsymbol{q})\boldsymbol{q}_{\boldsymbol{T}} = [\mathbf{0}_3\ \boldsymbol{R}(\boldsymbol{q},L)]\ \boldsymbol{J}(\boldsymbol{q},L)\left(\underline{\boldsymbol{K}} + \frac{\partial[\underline{\boldsymbol{G}}(\boldsymbol{q})]}{\partial \boldsymbol{q}}\right)^\dagger \underline{\boldsymbol{H}}$$

Remark 3: It is worth noting that, for any desired position $\boldsymbol{u}_r \in \boldsymbol{\Xi}$, there always exists $\boldsymbol{T} \in \mathcal{T}$ such that the end-effector position $\boldsymbol{u}$ can be driven to $\boldsymbol{u}_r$. Consequently, this implies that $\boldsymbol{\Gamma}(\boldsymbol{q})$ defined in (7) is right invertible, i.e., $\exists\ \boldsymbol{\Gamma}_R^{-1} \in \mathbb{R}^{s\times 3}$ such that $\boldsymbol{\Gamma}\boldsymbol{\Gamma}_R^{-1} = \mathbf{I}_3$. Compared to other methods (such as FEM, PCC, PCS, etc.), the PLS Cosserat model can obtain analytic input-output relationship of the soft slender robots accurately with low dimension, which is favorable to realize control objective in real time.

B. Control Law and System Stability Proof

Given a known end-effector position $\boldsymbol{u}^* \in \boldsymbol{\Xi}$ associated to a pair $(\boldsymbol{q}^*, \boldsymbol{T}^*)$ satisfying (3), we denote its equilibrium neighborhood as $\mathcal{N}(\boldsymbol{q}^*, \boldsymbol{T}^*) \in \boldsymbol{\Xi}$. Then, for any arbitrary end-effector position $\boldsymbol{u}(t) \in \boldsymbol{\Xi}$ and a given desired end-effector position $\boldsymbol{u}_r \in \mathcal{N}$, the objective of our work is to design a robust controller $\boldsymbol{T}(t)$ such that $\boldsymbol{u}(t) \to \boldsymbol{u}_r$ as $t \to \infty$.

When designing such a controller, the main difficulty is that the generalized joint vector $\boldsymbol{q}$ is generally not measurable in practice, thus conventional kinematic model-based control methods which require the knowledge of $\boldsymbol{\Gamma}(\boldsymbol{q})$ in real time, such as pseudo-inverse, cannot be directly applied in this study. To overcome this difficulty, the following proposes to approximate $\boldsymbol{\Gamma}(\boldsymbol{q})$ around its equilibrium $(\boldsymbol{q}^*, \boldsymbol{T}^*)$ by a constant matrix $\boldsymbol{\Gamma}_0(\boldsymbol{q}^*)$ which can be analytically computed due to the knowledge of the pair $(\boldsymbol{q}^*, \boldsymbol{T}^*)$ in advance, and then design a robust controller based on $\boldsymbol{\Gamma}_0(\boldsymbol{q}^*)$.

Concretely, according to the deduced input-output relation (6), the following robust controller is designed:

$$\dot{\boldsymbol{T}} = -\lambda\boldsymbol{\Gamma}_{0R}^{-1}\boldsymbol{e} \tag{10}$$

where $\lambda > 0$, and $\boldsymbol{e} = \boldsymbol{u}(t) - \boldsymbol{u}_r$. Then, the following result theoretically ensures the convergence of $\boldsymbol{u}(t)$ to the desired constant position $\boldsymbol{u}_r$.

Theorem 1: For any $\boldsymbol{u}(t) \in \mathcal{N}$ and a desired $\boldsymbol{u}_r \in \mathcal{N}$ satisfying (5), if the following inequality is satisfied:

$$\|\Delta\boldsymbol{\Gamma}\boldsymbol{\Gamma}_{0R}^{-1}\|_2 \le \varepsilon < 1 \tag{11}$$

where $\Delta\boldsymbol{\Gamma}(\boldsymbol{q}) = \boldsymbol{\Gamma}(\boldsymbol{q}) - \boldsymbol{\Gamma}_0(\boldsymbol{q}^*)$, then with the proposed controller (10), $\boldsymbol{u}(t) \in \boldsymbol{\Xi}$ can asymptotically converge to $\boldsymbol{u}_r \in \boldsymbol{\Xi}$, i.e., $\lim_{t\to\infty}\|\boldsymbol{u}(t) - \boldsymbol{u}_r\|_2 = 0$.

Proof 1: Note that we want to prove $\boldsymbol{e}(t) \to \mathbf{0}$ when $t \to \infty$, and this is equivalent to prove $V(\boldsymbol{e}) \to 0$ when $t \to \infty$, where $V(\boldsymbol{e})$ is a Lyapunov function defined as $V(\boldsymbol{e}) = \boldsymbol{e}^\top\boldsymbol{e}/2$. Then, the derivative of $V(\boldsymbol{e})$ w.r.t. time t yields

$$\begin{aligned}\frac{\partial V}{\partial t} = \dot{V} = \boldsymbol{e}^\top\dot{\boldsymbol{e}} &= \boldsymbol{e}^\top\dot{\boldsymbol{u}} = \boldsymbol{e}^\top\boldsymbol{\Gamma}\dot{\boldsymbol{T}} \\ &= -\lambda\boldsymbol{e}^\top\left(\boldsymbol{\Gamma}_0 + \Delta\boldsymbol{\Gamma}\right)\boldsymbol{\Gamma}_{0R}^{-1}\boldsymbol{e} \\ &= -2\lambda V - \lambda\boldsymbol{e}^\top\Delta\boldsymbol{\Gamma}\boldsymbol{\Gamma}_{0R}^{-1}\boldsymbol{e}\end{aligned}$$

Consequently, if (11) holds, we can come to a conclusion that

$$\dot{V} \le -2\lambda V + \lambda\boldsymbol{e}^\top\varepsilon\boldsymbol{e} = -2\lambda\left(1-\varepsilon\right)V = -2\lambda\beta V < 0$$

with $\lambda > 0$ and $\beta = 1 - \varepsilon > 0$, which guarantees that V exponentially converges to 0. Thus, we can derive that $\boldsymbol{e}(t)$ exponentially converges to $\mathbf{0}$.■

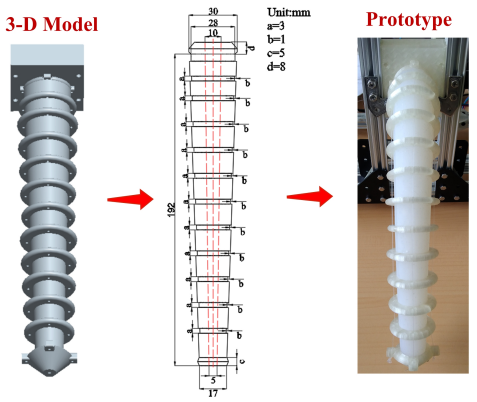

Fig. 1. Soft manipulator prototype actuated by 4 cables.

Remark 4: In Theorem 1, the inequality of (11) implies that the approximation of $\boldsymbol{\Gamma}(\boldsymbol{q})$ by $\boldsymbol{\Gamma}_0(\boldsymbol{q}^*)$ should be close in the norm sense. For the trajectory tracking problem, the detailed proof of the convergence by using the same controller (10) can refer to [**?**]. Furthermore, we would like to emphasize that it is possible to replace $\dot{\boldsymbol{T}} = -\lambda \boldsymbol{\Gamma}_{0R}^{-1} \boldsymbol{e}$ by $\dot{\boldsymbol{T}} = -\lambda \boldsymbol{\Gamma}_0^{\top} \boldsymbol{e}$ in order to avoid the inverse calculation of $\boldsymbol{\Gamma}_0$. For this, by the same idea of V, we can prove as well the convergence of $\boldsymbol{e}$ to $\mathbf{0}$.

V. Closed-loop Experiment on the Prototype

A trunk-like soft manipulator depicted in Fig. 1 is selected as the controlled plant. The primary geometric parameters of the investigated manipulator are as follows: total length of the arm $L = 0.192$ m, radius of the fixed end is 1.4×10^{-2} m, radius of the free end is 8.5×10^{-3} m. In addition, the material parameters include Young's modulus $E = 2.563 \times 10^5$ Pa, shear modulus $G = 8.543 \times 10^4$ Pa, and density of material $\rho = 1.41 \times 10^3$ kg/m^3, which have been identified in our prior work [**?**]. A position sensor is equipped on the tip of the manipulator through the long hole displayed by red dot-and-dash line in Fig. 1, and several rigid rings are installed along the arm to minimize friction between the cables and isotropic silicone body. Finally, 4 symmetrically distributed cables, remotely actuated by 4 independent stepping motors, are attached to free end of the arm and pass through the small holes of all rings.

The experimental setup shown at the top of Fig. 2 includes the soft manipulator (see Fig. 1), the MATLAB environment on the laptop, the Polhemus magnetic position sensor attached to the robot's end-effector, the micro controller, and the actuation system. The LIBERTY 240/16 base system supports four sensors, but is scalable to support up to 16 sensors, as shown at the bottom of Fig. 2. To provide the detailed experimental information, the specifications of the employed sensor in the experiment is summarized by Table I.

Several different experiments were conducted to demonstrate the capabilities of the proposed controller, which mainly consist of point-to-point test, time-varying trajectory tracking tests, and robustness analysis under the temporary and permanent disturbances.

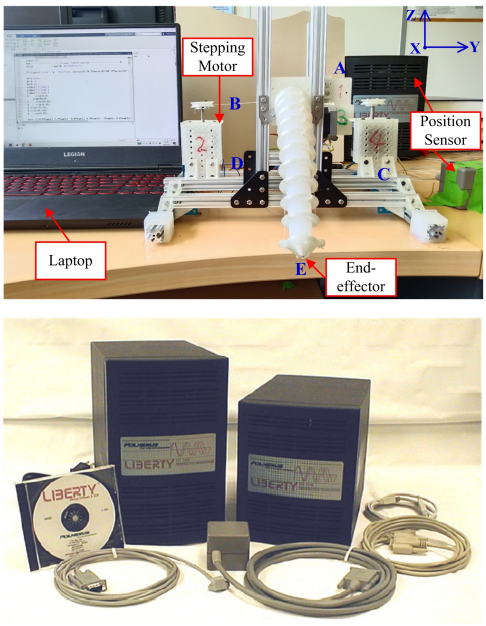

Fig. 2. Experimental setup for the manipulator control and LIBERTY 240/16.

TABLE I
The Specifications of the Sensor

Specifications	Polhemus Liberty-240/16
Resolution (mm)	0.038
Update rate (HZ)	240 (per sensor)
Angular range (°)	All
Time response (ms)	3.5
Operating voltage (V)	$85 - 264$
Transmitter range (m)	$1.8(\pm 0.9)$
Operating temperature (°C)	$10 - 40$

A. Point-to-Point Test

In this test, 4 different equilibrium points (see Table II) from their bounded workspace ($\boldsymbol{u}_r \in \boldsymbol{\Xi}$) were selected as the desired points to follow for the soft manipulator, and then the manipulator returns to the initial equilibrium point E. The experimental results have been depicted in Fig. 3. It can be clearly seen that the end-effector position of the arm converges to its desired points with a comparable tracking precision through the proposed control scheme. When following the paths (A-B-C, and D-E), despite the presence of small overshoot in the X-axis direction, the soft manipulator is always able to rapidly converge to the targeted points within 2 s. However, the soft arm converges very slowly in the X-axis direction of the segment D, and takes about 8 s to stabilize from point C to point D. For this phenomenon, we think it occurs since there exists the difference between the real Jacobian matrix and the matrix $\boldsymbol{\Gamma}_0$ obtained from the local linearization of the PLS Cosserat just at the point E.

B. Trajectory Tracking Tests

In this experiment, the manipulator was expected to track the circular and star-shaped trajectories by using the proposed controller with a fixed $\boldsymbol{\Gamma}_0$ obtained from the PLS Cosserat static model. The results are displayed in Fig. 4, Fig. 5 and

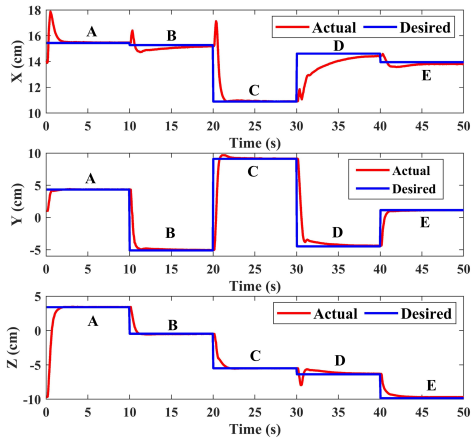

Fig. 3. The results of point-to-point test by using the proposed controller.

TABLE II
EQUILIBRIUM POINTS SELECTED FROM ITS WORKSPACE

Equilibrium points	A	B	C	D	E
X(cm)	15.438	15.272	10.888	14.615	13.951
Y(cm)	4.335	-5.114	9.106	-4.486	0.000
Z(cm)	3.399	-0.463	-5.492	-6.375	-9.842

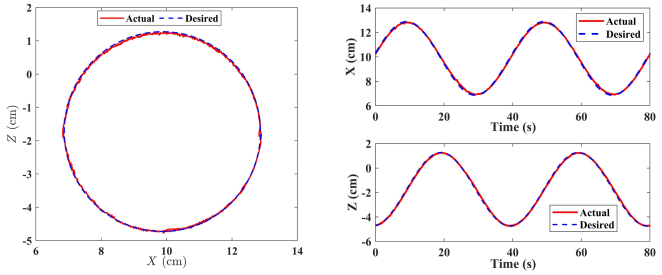

Fig. 4. Time-varying circular trajectory tracking in X-Z plane by employing the proposed controller.

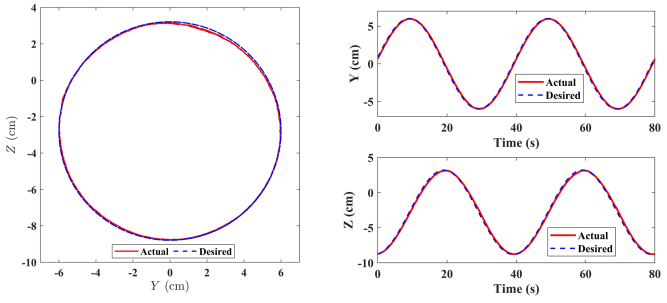

Fig. 5. Time-varying circular trajectory tracking in Y-Z plane by using the proposed controller.

Fig. 6, which indicate that the end-effector of the manipulator was capable of rapidly reacting to the different slowly time-varying trajectories.

To intuitively present the tracking effects of the proposed controller, the absolute error analysis of the actual and the desired values for tracking the different trajectories is displayed

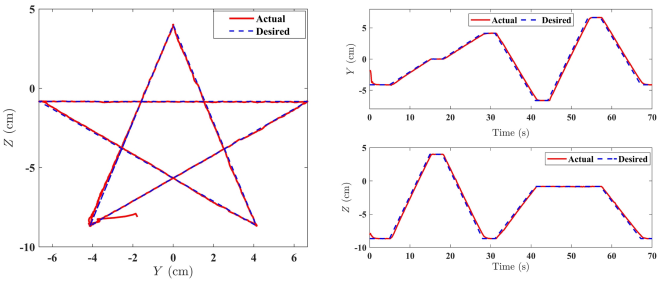

Fig. 6. Time-varying star-shaped trajectory tracking by applying the proposed controller.

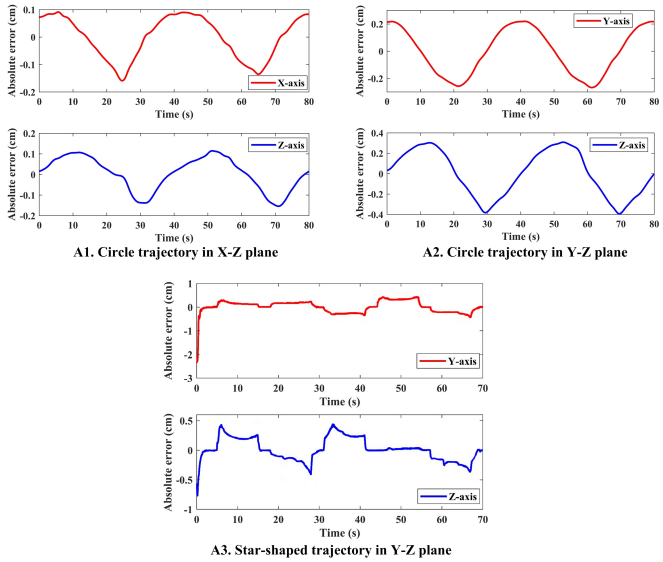

Fig. 7. Absolute error between the actual and the desired trajectories for different trajectory tracking of the soft manipulator.

in Fig. 7. By comparison, there exists a maximum absolute error within ± 0.4 cm for the circle trajectory tracking, while the absolute error between the actual and desired paths for tracking the star-shaped trajectory is within ± 0.5 cm except for the large absolute error at the beginning of the system. To put it another way, the proposed controller is of high feasibility and effectiveness for the arm moving around its workspace.

C. Robustness Analysis

As we all know, the need for a closed-loop control strategy is more important in the presence of external disturbances. To prove that our proposed model-based controller is robust to external disturbances, we formulated the following tracking tasks with external disturbances.

Permanent disturbance: In this test, the manipulator carries a Teenage Mutant Ninja Turtles (TMNT) (0.032 kg) which can be regarded as an externally permanent disturbance, and is suppose to follow a slowly time-varying circular trajectory, as shown on the left of Fig. 8. In view of the results from Fig. 9, the end-effector of the soft manipulator steadily tracks the trajectory, which demonstrates that the proposed closed-loop control scheme via the PLS Cosserat static model can resist the externally permanent disturbance.

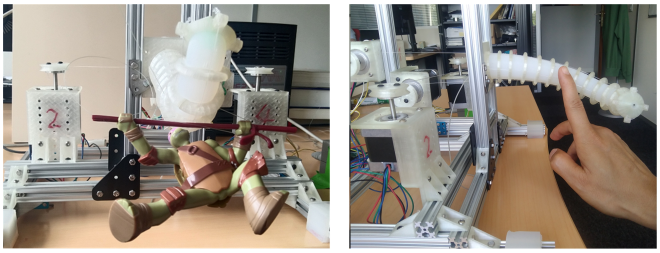

Fig. 8. From the left, the permanent disturbance by adding a TMNT under the soft manipulator, followed by the temporary disturbance by touching the soft arm with hand.

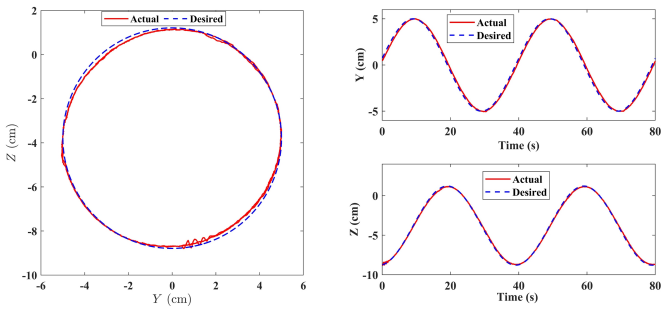

Fig. 9. The behavior of the manipulator for tracking time-varying circular trajectory in Y–Z plane under permanent disturbance.

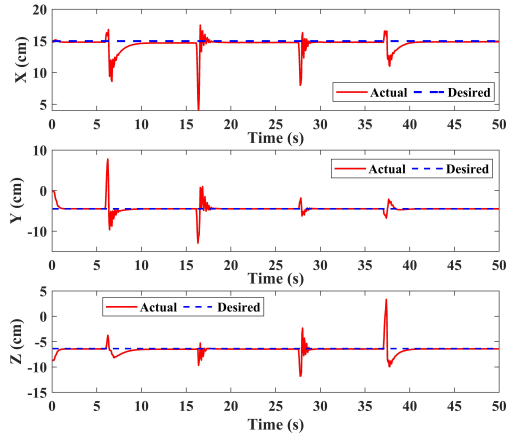

Fig. 10. The behavior of the manipulator for reaching the targeted point in X–Y–Z axis with temporary disturbances.

Temporary disturbance: The manipulator reaches the point D from the initial equilibrium position E. The externally temporary disturbances from different directions are exerted on the manipulator with hand displayed on the right of Fig. 8, and then we observe whether the manipulator can return to D. From the results of Fig. 10, the end-effector of the manipulator quickly converged to the point D, which shows great robustness of the proposed controller against the temporary disturbances.

Remark 5: It can be seen from the above experiments that the studied soft manipulator can only track the slowly time-varying trajectories well by using the proposed controller due to the ignorant of velocity and acceleration of the system. To address this issue, the dynamic model-based controller will be designed in the future. Compared to the other model-based control schemes (such as FEM-based controller [**?**]), there exist two remarkable advantages for the PLS Cosserat static-based controller: the analytical solution of the model can be obtained, which significantly reduces the model computation time and achieves real-time control; the static model via the PLS assumption has a good accuracy and much less DoFs. To the authors' knowledge, this is the first time that such a robust static model-based controller is proposed, theoretically proved and experimentally validated on the studied soft manipulator.

VI. Conclusion

In this letter, the PLS Cosserat static model of the soft manipulator was developed, which accounts for the large deformations due to extension, shear, bending and torsion, and builds on top of the Cosserat rod theory whose differential equations are formulated on the Lie group. Based on the static model, the robust controller was then designed. The control system stability was proven by constructing a Lyapunov function, and the result indicated that the control scheme can always guarantee the exponential convergence of the end-effector position of soft robots towards a desired reference point or trajectory. Finally, the experimental validation was implemented by controlling the soft manipulator within its available workspace. The robustness of this model-based controller was corroborated by tracking slowly time-varying trajectories in the presence of the externally permanent and temporary disturbances. In conclusion, the proposed closed-loop control scheme can be used in specific unstructured environments for the soft manipulators to perform some positioning tasks with high precision. In the future, we will design a PLS static model-based controller to globally control the end-effector position of the studied soft manipulator in its whole workspace, and validate feasibility as well as robustness of the proposed control scheme through the experiments. Also, a robust controller based on PLS Cosserat dynamic model will be investigated as well.

Appendix

A. Geometric Jacobian Matrix

The analytic formula of the geometric Jacobian matrix of PLS Cosserat model is obtained in (12), in which $\mathrm{T}_{\boldsymbol{g}_{n1}(X)}$ and $\mathrm{T}_{\boldsymbol{g}_{n2}(X)}$ are coefficient matrices of nodal strain twists of the section n, respectively, and their detailed derivation can be found in [**?**].

B. Lie Group Notations

The adjoint representation of the Lie algebra is given by

$$\mathrm{ad}_{\boldsymbol{\xi}} = \begin{pmatrix} \widetilde{\boldsymbol{K}} & \mathbf{0}_{3\times3} \\ \widetilde{\boldsymbol{Q}} & \widetilde{\boldsymbol{K}} \end{pmatrix} \in \mathbb{R}^{6\times6}, \ \mathrm{ad}_{\boldsymbol{\eta}} = \begin{pmatrix} \widetilde{\boldsymbol{\Omega}} & \mathbf{0}_{3\times3} \\ \widetilde{\boldsymbol{V}} & \widetilde{\boldsymbol{\Omega}} \end{pmatrix} \in \mathbb{R}^{6\times6}$$

where the operator $\widetilde{(\cdot)}$ represents a conversion from a 3-dimensional vector to its skew-symmetric matrix.

References

$$\boldsymbol{J}(\boldsymbol{q},X)=\begin{cases}0 \xrightarrow{\qquad\qquad n\qquad\qquad} N & \\ \mathrm{T}_{\boldsymbol{g}_{11}(X)} \quad \mathrm{T}_{\boldsymbol{g}_{12}(X)} \quad \mathbf{0}_6 \quad \cdots \quad \mathbf{0}_6 & X\in(0,L_1] \\ \mathrm{Ad}^{-1}_{\boldsymbol{g}_2(X)}\mathrm{T}_{\boldsymbol{g}_{11}(L_1)} \quad \mathrm{Ad}^{-1}_{\boldsymbol{g}_2(X)}\mathrm{T}_{\boldsymbol{g}_{12}(L_1)}+\mathrm{T}_{\boldsymbol{g}_{21}(X)} \quad \mathrm{T}_{\boldsymbol{g}_{22}(X)} \quad \mathbf{0}_6 \ \cdots \ \mathbf{0}_6 & X\in(L_1,L_2] \\ \mathrm{Ad}^{-1}_{\boldsymbol{g}_3(X)}\mathrm{Ad}^{-1}_{\boldsymbol{g}_2(L_2)}\mathrm{T}_{\boldsymbol{g}_{11}(L_1)} \quad \mathrm{Ad}^{-1}_{\boldsymbol{g}_3(X)}\left[\mathrm{Ad}^{-1}_{\boldsymbol{g}_2(L_2)}\mathrm{T}_{\boldsymbol{g}_{12}(L_1)}+\mathrm{T}_{\boldsymbol{g}_{21}(L_2)}\right] \ \cdots \ \mathrm{T}_{\boldsymbol{g}_{32}(X)} \quad \mathbf{0}_6 \ \cdots \ \mathbf{0}_6 & X\in(L_2,L_3] \\ \quad\vdots \qquad\qquad\qquad \vdots \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \vdots & \\ \prod_{i=2}^{N}\mathrm{Ad}^{-1}_{\boldsymbol{g}_i\min(L_i,X)}\mathrm{T}_{\boldsymbol{g}_{11}(L_1)} \qquad \cdots \qquad \mathrm{Ad}^{-1}_{\boldsymbol{g}_N(X)}\mathrm{T}_{\boldsymbol{g}_{(N-1)2}(X)}+\mathrm{T}_{\boldsymbol{g}_{N1}(X)} \quad \mathrm{T}_{\boldsymbol{g}_{N2}(X)} & X\in(L_{N-1},L_N]\end{cases} \tag{12}$$