The piecewise constant curvature model for the forward and inverse kinematics of continuum robot based on Lie Theory

Siyuan Lin, Yida Wang, Aoqi Li and Wenbin Chen, *Member, IEEE*

***Abstract*— Continuum robots have received much attention because of their high degree of flexibility. For modelling the kinematics of continuum robots, the piecewise constant curvature model is widely used, but it several of drawbacks. firstly, the forward kinematics calculation has a high computational complexity. Secondly, the numerical solution of the inverse kinematic Jacobian matrix is difficult to describe for multi-segments continuum robots, and thirdly, the model does not take the case of torsion into account. To address above issues, this paper proposes a piecewise constant curvature kinematic model based on Lie theory that can describe the forward and inverse kinematics of a continuum robot in a more concise manner, and some simulations are performed to validate the proposed model and the simulated results show the effectiveness of the model.**

I. INTRODUCTION

Continuum robot is a kind of robot with infinite degree of freedom with an elastic structure. Due to its enhanced dexterity and manipulability, These robots have been applied in many fields such as home services [1], navigation [2], exploration [3], minimally invasive surgeries [4]. With such a large number of application demand, it still attracts considerable critical attention in the past three decades. One of the classic problem is that how to describe the shape about the continuum robot (i.e. kinematic modeling). As presented by [5], the kinematics of continuum robot needs to build two mapping relationships of three spaces including actuation space (AS), configuration space (CS), and task space (TS). Piecewise constant curvature approach gives a perspective on geometry, it suppose that the robot is shaped by a series of arc, which makes it simple and useful in the field of continuum robot and soft robot. This model can make purely mapping between CS and TS without consideration of specific structure (the mapping between AS and CS still needs to be dependent on the design of robot).

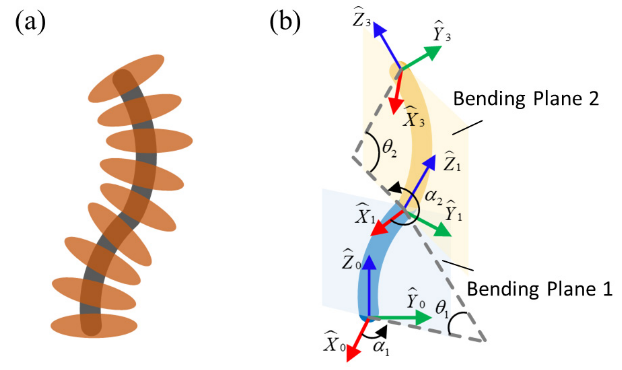

Fig. 1. (a) A schematic of continuum robot. (b) PCC model: the shape of continuum robots backbone is connected using a series of arcs

The general PCC approach can be classified in three ways. The first way is taken by multiplying a set of local transformation matrix to get the representation of global transformation matrix like arc geometry [6], D-H parameters [7], but they exists computational challenges with an algorithm of $O(n^2)$ [8]. Also, when it comes to the analysis of differential kinematics, the straight state (i.e. the bending angle equals to zero) will cause singularities when calculating the Jacobian matrix. Another way is to use the knowledge of differential geometry like FrenetSerret frames [9], integral representations [10] This method can overcome the calculational barrier that the fist method possesses because it doesn't repeat the previous calculation but make it directly by advanced geometric tool. however, both two classes of PCC approach mentioned above ignore the torsion. To get over the limit, a simplified approach formulated by the exponential coordinates based on the Lie theory can provide a fresh perspective. This method was firstly used in the rigid robots[11], and in recent years, it has been applied in the similar realm of soft robotics [12] as PCS model. While in terms of continuum robot, the motion is often broken down into a translation and a rotation in the local frame with two twist coordinates. [13]. Compared to the first method to obtain the local pose description using four multiplication, it only requires product of two exponentials to reach the same goal, cutting computational complexity in half. Even so, this description only contains the pose of some characteristic points in the shape curve like the second method. For instance, in the single backbone continuum robots, the characteristic points can be the center of each discs. This means that a complete way of the curve can't be derived.

这里是walker
那篇建模方法?

?

Motivated by the above discussion in the earlier work, we put forward a more efficient form combining the FrenetSerret frames and exponential coordinates. By using just one exponential instead of normal two exponentials to define the local pose, the computational cost of global pose can be reduced. Furthermore, the forward and inverse kinematics can be also modeled in this way.

*This work was not supported by any organization

*This work was partially supported by the National Natural Science Foundation of China (Grant No. 52075191, U1913205) and the Program for HUST Academic Frontier Youth Team.

Corresponding author: Wenbin Chen, wbchen@hust.edu.cn.

S. Y. Lin, Y. D. Wang, A. Q. Li and W. B. Chen are with the Institute of Medical Equipment Science and Engineering, State Key Lab of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China.

The outline of this work is as follows. Firstly, the general piecewise constant curvature model is reviewed in Section II, which explain the original mathematical foundations to give a intuitive sense and can be compared with our method better. In Section III, forward kinematics are analysed in the single-segment and multiple-segments continuum robots driven by tendon. Meanwhile, the complete Jacobian defined based on Lie theory is obtained to solve the inverse kinematics problem. In Section IV, simulations are shown to verify the proposed kinematic method. In Section V, we present conclusions and outlook of this work.

II. Overview of Piecewise Constant Curvature Model

Here, the constant-curvature (CC) and piecewise constant-curvature (PCC) model is briefly reviewed. The following notation is used throughout the paper: x, $\boldsymbol{x}$, and $\mathbf{x}$ denote a scalar, a vector, and a matrix, respectively.

The main assumption is as follows, with CC containing from i to iv and PCC containing from i to v:

i. Each section is connected equally spaced by discs, which are considered to be infinitely thin.

ii. Each section bends without shear effect;

iii. The backbone are inextensible.

iv. Each segment deforms as a circular arc with satisfing the constant curvature condition.

v. The adjacent segments should guarantees the continuity of shape.

Due to the constant curvature assumption, the kinematics of continuum robot can be described by only two parameters: angle of the arc θ and angle of the bending plane α.

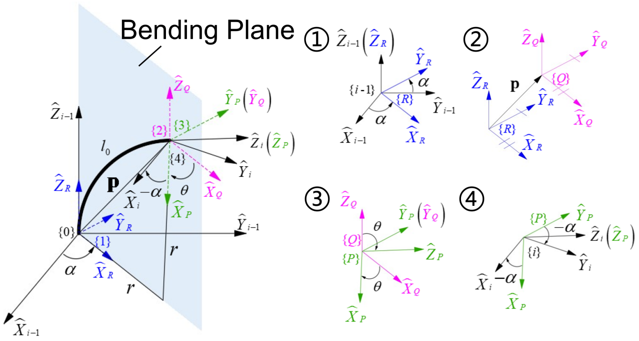

Fig. 2. Coordinate conversion steps for the conventional PCC method

As demonstrated in Fig. 2, We define three intermediate frames ($\{P\}$, $\{Q\}$ and $\{R\}$) to make the drawing clearer. The transform that defines the i th section's position and orientation w.r.t the $(i-1)$ th section's, namely ${}^{i-1}_{i}\mathbf{T}$, can be determined by four sequential transformation: three rotations and one translation: Fist, a rotation of α around the $\widehat{Z}_{i-1}$ axis makes frame $\{i-1\}$ change into frame $\{R\}$, then a translation of $\mathbf{p}$ makes frame $\{R\}$ change into $\{Q\}$. After that, a rotation of θ from frame $\{Q\}$ makes frame around the $\widehat{Y}_Q$ makes frame $\{Q\}$ change into frame $\{P\}$, and finally a reverse rotation of α around the $\widehat{Z}_P$ makes frame $\{Q\}$ change into frame $\{i\}$. Thus, the homogeneous transformation matrix ${}^{i-1}_{i}\mathbf{T}$ of the local section is given as

$$\begin{aligned}{}^{i-1}_{i}\mathbf{T} &= \mathrm{Rot}\left(\widehat{Z}_{i-1},\alpha\right)\mathrm{Trans}(\mathbf{p})\,\mathrm{Rot}\left(\widehat{Y}_{Q},\theta\right)\mathrm{Rot}\left(\widehat{Z}_{P},-\alpha\right)\\ &= \begin{bmatrix}{}^{i-1}_{i}\mathbf{R} & {}^{i-1}_{i}\mathbf{p}\\ \mathbf{0} & 1\end{bmatrix}\end{aligned} \tag{1}$$

where ${}^{i-1}_{i}\mathbf{R}$ and ${}^{i-1}_{i}\mathbf{p}$ are the rotation and position part described in local section, we can obtain

$${}^{i-1}_{i}\mathbf{R} = \begin{bmatrix}\cos\alpha^2(\cos\theta-1)+1 & \sin\alpha\cos\alpha(\cos\theta-1) & \cos\alpha\sin\theta\\ \sin\alpha\cos\alpha(\cos\theta-1) & \cos\alpha^2(1-\cos\theta)+\cos\theta & \sin\alpha\sin\theta\\ -\cos\alpha\sin\theta & -\sin\alpha\sin\theta & \cos\theta\end{bmatrix} \tag{2}$$

$${}^{i-1}_{i}\mathbf{p} = \begin{cases}\dfrac{L_0}{\theta}\left[\cos\alpha(1-\cos\theta),\sin\alpha(1-\cos\theta),\sin\theta\right]^T, & \theta\neq 0\\ (0,0,l_0)^T\,, & \theta=0\end{cases} \tag{3}$$

where l_0, the length of the constant curvature arc section, are considered to be a constant when the continuum robot bends.

Once the individual transformation matrices in local frame can be computed, they can be multiplied together to find the transformation in the global frame that relates frame $\{N\}$ to frame $\{0\}$, which N means the end

$${}^{0}_{N}\mathbf{T} = {}^{0}_{1}\mathbf{T}\cdots{}^{i-2}_{i-1}\mathbf{T}{}^{N-1}_{N}\mathbf{T} \tag{4}$$

The above is a description of the constant curvature kinematics of a single-segment continuum robot. To satisfy the application of multiple-segments continuum robot, piecewise constant-curvature (PCC) model is introduced, which can define different θ and α for various segments. It should be noted that the PCC model guarantees the continuity of shape, which means that the origin of the end frame of the former segment and the start frame of the latter segment are coincided, and their z-axis (the tangential direction of the backbone) are oriented in the same direction. A detailed exposition of other forms about PCC model can be seen in [5].

III. Methodology

In this section, we describe a simplified PCC methodology based on differential geometry and exponential mapping.

In general differential geometry, the FrenetSerret formulas are often used to describe the evolution of a differentiable curve in three-dimensional Euclidean space $\mathbb{R}^3$, which can be written in the following form

$$\begin{cases}\mathbf{t}'(s) = \kappa(s)\mathbf{n}(s)\\ \mathbf{n}'(s) = -\kappa(s)\mathbf{t}(s)+\tau(s)\mathbf{b}(s)\\ \mathbf{b}'(s) = -\tau(s)\mathbf{n}(s)\end{cases} \tag{5}$$

where s is valid for $0 \leq s \leq L$, and L means the whole length of the curve, $\mathbf{t}(s)$, $\mathbf{n}(s)$ and $\mathbf{b}(s)$ are called tangent, normal, and binormal unit vectors (TNB vectors), and $\kappa(s)$ and $\tau(s)$ are the curvature and torsion of the curve, $(\cdot)'$

indicates derivatives according to s. Since every arc segment in PCC model must be located in the bending plane, and so do the normal vectors of the points on the arc segments. Compared with the local frame, it's obvious that the angle between the $\widehat{X}(s)$ axis and the normal vector $\mathbf{n}(s)$, the $\widehat{Y}(s)$ axis and the binormal vector $\mathbf{b}(s)$ is α while the $\widehat{Z}(s)$ axis and the tangent vector are in the same direction $\mathbf{t}$. A rotation matrix $\mathbf{R} \in SO(3)$ associating the global frame and local frame can be expressed as

$$\begin{aligned} \mathbf{R}(s) &= \left[\widehat{\mathbf{X}}(s)\ \widehat{\mathbf{Y}}(\mathbf{s})\ \widehat{\mathbf{Z}}(s)\right] \\ &= \left[\mathbf{n}(s)\ \mathbf{b}(s)\ \mathbf{t}(s)\right] \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{6}$$

With the above equation, we can know the three column vectors of the rotation matrix and the TNB vectors of the FrenetSerret have the following relationship

$$\begin{cases} \mathbf{t}(s) = \widehat{\mathbf{Z}}(s) \\ \mathbf{n}(s) = \cos\alpha\widehat{\mathbf{X}}(s) + \sin\alpha\widehat{\mathbf{Y}}(s) \\ \mathbf{b}(s) = -\sin\alpha\widehat{\mathbf{X}}(s) + \cos\alpha\widehat{\mathbf{Y}}(s) \end{cases} \tag{7}$$

By uing equation (7), The derivative of equation (6) with respect to s yields

$$\begin{aligned} \mathbf{R}'(s) &= \left[\mathbf{n}(s)\ \mathbf{b}(s)\ \mathbf{t}(s)\right] \begin{bmatrix} \tau(s)\sin\alpha & -\tau(s)\cos\alpha & \kappa(s) \\ \tau(s)\cos\alpha & \tau(s)\sin\alpha & 0 \\ -\kappa(s)\cos\alpha & \kappa(s)\sin\alpha & 0 \end{bmatrix} \\ &= \left[\widehat{\mathbf{X}}(s)\ \widehat{\mathbf{Y}}(\mathbf{s})\ \widehat{\mathbf{Z}}(s)\right] \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &\quad \begin{bmatrix} \tau(s)\sin\alpha & -\tau(s)\cos\alpha & \kappa(s) \\ \tau(s)\cos\alpha & \tau(s)\sin\alpha & 0 \\ -\kappa(s)\cos\alpha & \kappa(s)\sin\alpha & 0 \end{bmatrix} \\ &= \mathbf{R}(s)\widehat{\mathbf{u}} \end{aligned} \tag{8}$$

where $\widehat{\cdot}$ represents the mapping from $\mathbb{R}^3$ to $so(3)$. This is defined in the local frame by

$$\widehat{\mathbf{u}} = \begin{bmatrix} 0 & -\tau(s) & \kappa(s)\cos\alpha \\ \tau(s) & 0 & \kappa(s)\sin\alpha \\ -\kappa(s)\cos\alpha & -\kappa(s)\sin\alpha & 0 \end{bmatrix} \tag{9}$$

where $\widehat{\mathbf{u}}$ satisfies $(\widehat{\mathbf{u}})^\vee = \mathbf{u} = \left[u_1\ u_2\ u_3\right]^T = \left[-\kappa(s)\sin\alpha\ \kappa(s)\cos\alpha\ \tau(s)\right]^T$, a vector in $\mathbb{R}^3$, which specifies bending of the curve that the first two elements gives bending about the local $\widehat{X}$ and $\widehat{Y}$ axis and the last element gives twist about the curve. Likewise the derivative of the global position $\mathbf{p}(s)$ with respect to s in the local frame is denoted by $\mathbf{v}$, that is

$$\mathbf{p}'(s) = \mathbf{R}(s)\mathbf{v} \tag{10}$$

where the 3×1 vector $\mathbf{v} = \left[v_1\ v_2\ v_3\right]^T$ specifies shear along the local $\widehat{X}$ and $\widehat{Y}$ in the first two elements $v_{1,2}$ and stretch or compression in the last elment v_3, where $v_3 = 1$ denotes neither stretch nor compression, $0 < v_3 < 1$ compression(note the case $v_3 = 0$ of infinite compression is not allowed), and $v_3 > 1$ stretch [14].

By combing equation (8) and equation (10), we can write a more compact form

$$\mathbf{H}'(s) = \mathbf{H}(s)\widehat{\mathbf{f}} \tag{11}$$

where $\mathbf{H}(s)$ is the homogeneous matrix composed of the rotation matrix $\mathbf{R}(s)$ and position vector $\mathbf{p}(s)$, $\mathbf{H}'(s)$ states the derivative according to s, and $\widehat{\mathbf{f}}$ introduced by $\widehat{\mathbf{u}}$ and $\mathbf{v}$ is referred to as a twist in $se(3)$. All the above quantities can be expressed as

$$\mathbf{H}'(s) = \begin{bmatrix} \mathbf{R}'(s) & \mathbf{p}'(s) \\ 0 & 0 \end{bmatrix} \tag{12}$$

$$\mathbf{H}(s) = \begin{bmatrix} \mathbf{R}(s) & \mathbf{p}(s) \\ 0 & 0 \end{bmatrix} \in SE(3) \tag{13}$$

$$\widehat{\mathbf{f}} = \begin{bmatrix} [\mathbf{u}] & \mathbf{v} \\ 0 & 0 \end{bmatrix} \in se(3) \tag{14}$$

and $\widehat{\mathbf{f}}$ satisfies $(\widehat{\mathbf{f}})^\vee = \mathbf{f} = \left(\mathbf{v}^T\ \mathbf{u}^T\right)^T$, a deformation vector in $\mathbb{R}^6$ that consists of the position part and rotation part caused by deformation. $[\cdot]$ gives a map from $\mathbb{R}^3$ to $so(3)$.

A. Forward Kinematics

Owing to the PCC assumption (iv), $\kappa(s)$ and $\tau(s)$ can be regarded as constants κ and τ , respectively (i.e., $\mathbf{u} = \left[-\kappa\sin\alpha\ \kappa\cos\alpha\ \tau\right]^T$), and similarly, $\mathbf{v} = \left[0\ 0\ 1\right]^T$ because of the PCC assumption (ii) and (iii), that is, $\widehat{\mathbf{f}}$ is independent of s. It leads to the equivalent solution of equation (12), which can decide the kinematic case of single-segment continuum robot

$$\mathbf{H}(s) = \mathbf{H}_0 e^{\widehat{\mathbf{f}}s} \quad s \in [0, L] \tag{15}$$

where $\mathbf{H}_0$ represents the homogenous matrix at $s = 0$ (i.e., $\mathbf{H}_0 = \mathbf{I}_{4\times4}$), $\widehat{\mathbf{f}}$, an element of $se(3)$, can be turned into an element in $SE(3)$ by an exponential map as [15].

$$e^{\widehat{\mathbf{f}}s} = \begin{cases} \begin{bmatrix} e^{[\mathbf{u}]s} & \left(\mathbf{I} - e^{[\mathbf{u}]s}\right)\left(\frac{\mathbf{u}\times\mathbf{v}}{\|\mathbf{u}\|^2}\right) + \frac{\mathbf{u}\mathbf{u}^T\mathbf{v}}{\|\mathbf{u}\|^2}s \\ \mathbf{0} & 1 \end{bmatrix}, & \mathbf{u} \neq \mathbf{0}_{3\times1} \\ \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{v}s \\ \mathbf{0} & 1 \end{bmatrix}, & \mathbf{u} = \mathbf{0}_{3\times1} \end{cases} \tag{16}$$

where

$$e^{[\mathbf{u}]s} = \mathbf{I}_{3\times3} + \frac{[\mathbf{u}]}{\|\mathbf{u}\|}\sin(\|\mathbf{u}\|s) + \frac{[\mathbf{u}]^2}{\|\mathbf{u}\|^2}(1 - \cos(\|\mathbf{u}\|s)) \tag{17}$$

denotes the exponential map surjective onto SO(3).

For the case of n-segments manipulator, equation (15) can be extended as a general form of piecewise function

$$\mathbf{H}(s) = \begin{cases} \mathbf{H}_0 e^{\widehat{\mathbf{f}}_1 (s-s_0)}, & s \in [s_0, s_1] \\ \mathbf{H}_0 e^{\widehat{\mathbf{f}}_1 (s_1-s_0)} e^{\widehat{\mathbf{f}}_2 (s-s_1)}, & s \in [s_1, s_2] \\ \quad \vdots & \\ \mathbf{H}_0 e^{\widehat{\mathbf{f}}_1 s_1} e^{\widehat{\mathbf{f}}_2 (s_2-s_1)} \dots e^{\widehat{\mathbf{f}}_n (s-s_{n-1})}, & s \in [s_{n-1}, s_n] \end{cases} \tag{18}$$

where $s_0 = 0$, $s_n = L$.

B. Inverse Kinematics

Unlike the forward kinematics, which can directly yield a unique result, the inverse kinematics of continuum robot often has multiple solutions, and due to the redundancy and nonlinearity of the structure, an analytical closed-form solution is often not achievable. Thus, numerical solutions are accepted in most cases.

The inverse Jacobian method based on iteration is used to solve inverse kinematics at the velocity level, which mathematically involves solving a system of linear equations, making it relatively simple and suitable for real-time computation. However, in the case of multi-segments continuum robots (more than three segments), the traditional PCC method using multiple matrix products makes it difficult to solve the Jacobian matrix. The use of Lie group theory can help us better understand the essence of the Jacobian matrix and thus solve the inverse kinematics.

Since inverse kinematics is concerned with how the tip position achieve the desired target, For the sake of convenience in ensuing discussions, we denote $e^{\widehat{\mathbf{f}}_n (s_n - s_{n-1})}$ as $\mathbf{H}_n$, we only consider the Euclidean groups $\mathbf{H}(L) \in SE(3)$

$$\begin{aligned} \mathbf{H}(L) &= \mathbf{H}_0 e^{\widehat{\mathbf{f}}_1 (s_1-s_0)} e^{\widehat{\mathbf{f}}_2 (s_2-s_1)} \dots e^{\widehat{\mathbf{f}}_n (s_n-s_{n-1})} \\ &= \mathbf{H_0 H_1 H_2} \cdots \mathbf{H_n} \end{aligned} \tag{19}$$

Differential kinematics can define the spatial velocity $\widehat{\mathbf{V}}_{tip}$ of the tip

$$\begin{aligned} \widehat{\mathbf{V}}_{tip} =& \dot{\mathbf{H}}(s) \mathbf{H}(s)^{-1} \\ =& \left(\frac{d}{dt} (\mathbf{H_0 H_1 H_2} \cdots \mathbf{H_n}) \right) (\mathbf{H_0 H_1 H_2} \cdots \mathbf{H_n})^{-1} \\ =& (\dot{\mathbf{H}}_\mathbf{0} \mathbf{H_1} \cdots \mathbf{H_n} + \mathbf{H_0} \dot{\mathbf{H}}_\mathbf{1} \cdots \mathbf{H_n} + \mathbf{H_0 H_1} \cdots \dot{\mathbf{H}}_\mathbf{n}) \\ & \mathbf{H_n^{-1}} \cdots \mathbf{H_2^{-1} H_1^{-1} H_0^{-1}} \\ =& \dot{\mathbf{H}}_\mathbf{0} \mathbf{H_0^{-1}} + \mathbf{H_0} \dot{\mathbf{H}}_\mathbf{1} \mathbf{H_1^{-1} H_0^{-1}} + \cdots + \\ & \mathbf{H_0 H_1} \cdots \mathbf{H_{n-1}} \dot{\mathbf{H}}_\mathbf{n} \mathbf{H_n^{-1} H_{n-1}^{-1}} \cdots \mathbf{H_1^{-1} H_0^{-1}} \end{aligned} \tag{20}$$

Here we set $\dot{\mathbf{H}}_\mathbf{k} \mathbf{H_k^{-1}}$ to be $\widehat{\mathbf{V}}_\mathbf{k} \in se(3)$, and $\widehat{\mathbf{V}}_\mathbf{0}$ is apparently a matrix of all zeros. To calculate other $\widehat{\mathbf{V}}_\mathbf{k}$, we apply the following formular [16]

$$\widehat{\mathbf{V}}_\mathbf{k} = \int_{s_{k-1}}^{s_k} e^{(s-s_{k-1})\widehat{\mathbf{f}}_k} (\dot{\widehat{\mathbf{f}}}_k) e^{-(s-s_{k-1})\widehat{\mathbf{f}}_k} ds, \; k = 1, 2, \cdots, n \tag{21}$$

A equivalent form using adjoint representation (See Appendix for a more detailed exposition) is given so that we can calculate the velocity in $\mathbb{R}^6$

$$\begin{aligned} \mathbf{V_k} &= \int_{s_{k-1}}^{s_k} \mathbf{ad}_{e^{x\widehat{\mathbf{f}}_k}} (\dot{\mathbf{f}}_k) ds \\ &= \int_{s_{k-1}}^{s_k} e^{x \mathbf{ad}_{\widehat{\mathbf{f}}_k}} \dot{\mathbf{f}}_k ds \\ &= \left(\int_{s_{k-1}}^{s_k} e^{x \mathbf{ad}_{\widehat{\mathbf{f}}_k}} ds \right) \dot{\mathbf{f}}_k \\ &= \mathbf{T}_k \dot{\mathbf{f}}_k, \quad k = 1, 2, \cdots, n \end{aligned} \tag{22}$$

The equality $\mathbf{ad}_{e^{\widehat{\mathbf{f}}}} = e^{\mathbf{ad}_{\widehat{\mathbf{f}}}}$ [17] has been used. The exponential set as $\mathbf{T}_k$ in equation (22) can be computed by [18] (To have a more compact form in the following, we define $x = s - s_{k-1}$, $x_k = s_k - s_{k-1}$, $\mathrm{Ad}_{k+1} = \mathrm{Ad}_{\mathbf{H_0} \cdots \mathbf{H_k}}$)

From equation (20) and (22), we can change the presentation of equation (20)

$$\mathbf{V}_{tip} = (\mathbf{Ad}_1 \mathbf{T}_1) \dot{\mathbf{f}}_1 + (\mathbf{Ad}_2 \mathbf{T}_2) \dot{\mathbf{f}}_2 + \cdots + (\mathbf{Ad}_n \mathbf{T}_n) \dot{\mathbf{f}}_n \tag{23}$$

where $\mathbf{V}_{tip} = \left(\dot{\mathbf{d}}^T \; \dot{\mathbf{\Theta}}^T \right)^T$ is the velocity defined in $\mathbb{R}^6$, containing the linear velocity $\dot{\mathbf{d}}^T$ and the angular velocity $\dot{\mathbf{\Theta}}^T$,$\mathbf{J} = \left[\mathbf{ad}_1 \mathbf{T}_1 \; \mathbf{ad}_2 \mathbf{T}_2 \cdots \mathbf{ad}_n \mathbf{T}_n \right] \in \mathbb{R}^{6 \times 6n}$ is the the Jacobian matrix, $\dot{\mathbf{f}} = \left(\dot{\mathbf{f}}_\mathbf{1}^T \; \dot{\mathbf{f}}_\mathbf{2}^T \cdots \dot{\mathbf{f}}_\mathbf{n}^T \right)^T \in \mathbb{R}^{6n}$ is the continuum robots joint vector, thus rewriting equation (23)

$$\mathbf{V}_{tip} = \mathbf{J} \dot{\mathbf{f}} \tag{24}$$

Here we propose to use $\dot{\mathbf{f}}$ instead of the general bending angle α and the curvature κ (the angle of arc θ) to obtain the Jacobian matrix. Since $\mathbf{v} = \left(0 \; 0 \; 1 \right)^T$, $\dot{\mathbf{v}}$ is a constant vector with three elements all zeros, thus it doesn't contribute to $\mathbf{V}_{tip}$. Considering only the effect of $\dot{\mathbf{u}}$ on $\mathbf{V}_{tip}$, equation (24) can be simplified by cutting matrix into blocks

$$\begin{aligned} \mathbf{V}_{tip} = \begin{pmatrix} \dot{\mathbf{d}} \\ \dot{\mathbf{\Theta}} \end{pmatrix} &= \left[\mathbf{J_{L1}} \vdots \mathbf{J_{R1}} \vdots \cdots \vdots \mathbf{J_{Ln}} \vdots \mathbf{J_{Rn}} \right] \begin{pmatrix} \dot{\mathbf{v}}_\mathbf{1} \\ \dot{\mathbf{u}}_\mathbf{1} \\ \vdots \\ \dot{\mathbf{v}}_\mathbf{n} \\ \dot{\mathbf{u}}_\mathbf{n} \end{pmatrix} \\ &= \left[\mathbf{J_{R1}} \vdots \cdots \vdots \mathbf{J_{Rn}} \right] \begin{pmatrix} \dot{\mathbf{u}}_\mathbf{1} \\ \vdots \\ \dot{\mathbf{u}}_\mathbf{n} \end{pmatrix} \\ &= \mathbf{J_R} \dot{\mathbf{u}} \end{aligned} \tag{25}$$

where each 6 by 6 matrix $\mathbf{J}_k$ in the Jacobian matrix $\mathbf{J}$ is broken into blocks of size 6 by 3− $\mathbf{J}_{\mathbf{L}k}$ and $\mathbf{J}_{\mathbf{R}k}$, $k = 1, 2, \cdots, n$. Similarly, we have the differential kinematics expressed in the following form

$$\mathbf{\Delta P} = \mathbf{J_R} \mathbf{\Delta u} \tag{26}$$

where we use the relationship $\mathbf{\Delta P} = \mathbf{V}_{tip} \Delta t$ and $\mathbf{\Delta u} = \dot{\mathbf{u}} \Delta t$. To get a numerical solution of the inverse kinematics, we employ the Levenberg-Marquardt Method [19] introducing a non-zero damping constant λ, which has a relatively high level of stability. It can be explained as an optimization problem

$$\underset{\Delta\mathbf{u}}{\arg\min} \|\mathbf{J_R}\Delta\mathbf{u} - \mathbf{\Delta P}\|^2 + \lambda^2\|\mathbf{\Delta u}\|^2 \tag{27}$$

that is

$$\mathbf{\Delta u} = \mathbf{J_R}^T(\mathbf{J^T J} + \lambda^2\mathbf{I})\mathbf{\Delta P} \tag{28}$$

By continuously iterating using 2, the numerical solution can eventually be obtained.

IV. SIMULATION STUDY

In Section III, we have found a simplified expression of the forward kinematics and inverse kinematics based on Lie theory in the assumption of piecewise constant curvature. Next, multi-segments continuum robot is selected as an example to verify the effectiveness of the proposed kinematic model.

A. Forward Kinematics Simulation For Multi-segments Robots

The first simulation was carried out on a three-segments continuum robot. The parameters defined in this example are given in the Table I

TABLE I

PARAMETERS OF THE 3-SEGMENT CONTINUUM ROBOT

k	L_k	α_k	κ_k
1	$0.5m$	0	$0:\pi/10:3\pi/5$
2	$0.3m$	$\pi/3$	$0:\pi/10:3\pi/5$
3	$0.3m$	$\pi/6$	$\pi/5:\pi/10:4\pi/5$

First we don't consider torsion and set the torsion of the three arc segments $\tau_1 = \tau_2 = \tau_3 = 0$, the results of the simulation are shown in Fig. 3a, this case with the conventional segmented constant curvature model can get the same results.

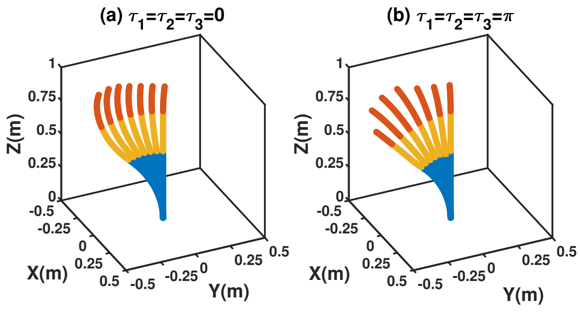

Fig. 3. Forward kinematics for multi-segments continuum robot: (a) torsion is not considered, (b) torsion is considered

Our proposed method based on the Lie theory can also captures the effect of torsion, which is not reflected in the general piecewise constant curvature model. To demonstrate the effect of torsion and to compare it with the non-torsion case, we set the value of torsion $\tau_1 = \tau_2 = \tau_3 = \pi$, and the results can be seen in Fig. 3b, where it can be seen that the shape undergoes a large deformation under the effect of torsion, which may have explanatory implications for the torsion of the continuum robots during the actual process.

B. Inverse Kinematics Simulation For Multi-segments Robots

Our simulations revolve around trajectory tracking, and straight, circular and square trajectories were tested. The parameters are chosen: the segment length is the same as in case A. The damping factor λ was set to 3.5, which better ensures that the desired accuracy is achieved using less time without falling into singular points. The maximum iterations is 1000. When the result has not converged beyond 1000, we consider the solution fails, which means the desired point is unreachable. The error tolerance of $0.001\mathrm{m}(1\mathrm{mm})$ is acceptable for practical applications in continuum robots. The initial configuration of robot is set to be straight.

The results are shown in Fig. 4. it can be seen that convergence is achieved in the end. The iteration and convergence time can be found in Table II, showing the inverse kinematic model is effective.

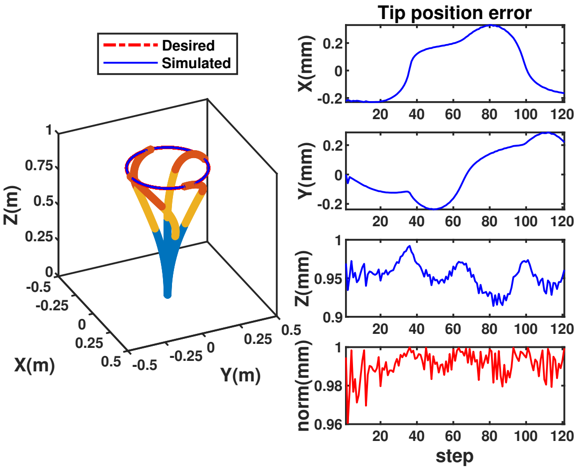

Fig. 4. Inverse kinematics for multi-segments continuum robot

TABLE II

PERFORMANCE OF INVERSE KINEMATICS SIMULATION FOR MULTI-SEGMENTS ROBOTS

Average iterations	Max iterations	Average time (s)	Max time (s)
99.1983	496	0.1106	0.5478

V. CONCLUSIONS

In this paper, we derive another form of the PCC model based on the Frenet framework in differential geometry and Lie theory. We use this theory for both forward and inverse kinematic description of a continuum robot, which is more concise than the traditional forward kinematics with multiple matrix multiplication. The inverse kinematics using Jacobi matrices can be greatly simplified. Finally, simulations of forward and inverse kinematics are carried out, where forward kinematics not only gives the same results calculated by the conventional PCC model without torsion, but also allows the calculation of the case with torsion. The inverse kinematic simulations were performed by trajectory tracking, with better results in terms of number of iterations and convergence time.

The proposed model can take the torsion into account and is more generalisable. Future work will use this model on trajectory planning and obstacle avoidance, control of continuum robots.

APPENDIX

1) The adjoint $\mathbf{ad}$ representation of $se(3)$ and the Adjoint $\mathbf{Ad}$ representation of $SE(3)$

For $\mathbf{H} = \begin{bmatrix}\mathbf{R} & \mathbf{p}\\ \mathbf{0} & 1\end{bmatrix} \in SE(3)$, $\widehat{\mathbf{f}} = \begin{bmatrix}[\mathbf{u}] & \mathbf{v}\\ \mathbf{0} & 0\end{bmatrix} \in se(3)$ and $\widehat{\mathbf{f}} = \left(\mathbf{v}^T\ \mathbf{u}^T\right) \in \mathbb{R}^6$, we have

$$\mathbf{Ad}_{\mathbf{H}} = \begin{bmatrix}\mathbf{R} & [\mathbf{p}]\mathbf{R}\\ \mathbf{0} & \mathbf{R}\end{bmatrix} \quad \mathbf{ad}_{\mathbf{f}} = \begin{bmatrix}[\mathbf{u}] & [\mathbf{v}]\\ \mathbf{0} & [\mathbf{u}]\end{bmatrix} \tag{29}$$

2) The integral of adjoint $\mathbf{ad}$ representation in equation (22), For $\mathbf{f}_k = \left(\mathbf{v}_k^T\ \mathbf{u}_k^T\right)^T$, we have

$$e^{x\mathbf{ad}_{\widehat{\mathbf{f}}_k}} = \begin{cases}\begin{pmatrix}\mathbf{I}_{6\times 6} + A_1\mathbf{ad}_{\widehat{\mathbf{f}}_k} + A_2\mathbf{ad}^2_{\widehat{\mathbf{f}}_k}\\ + A_3\mathbf{ad}^3_{\widehat{\mathbf{f}}_k} + A_4\mathbf{ad}^4_{\widehat{\mathbf{f}}_k}\end{pmatrix}, & u_k \neq 0\\ \mathbf{I}_{6\times 6} + x\mathbf{ad}_{\widehat{\mathbf{f}}_k}, & u_k = 0\end{cases} \tag{30}$$

where

$$\begin{aligned} A_1 &= \frac{3\sin(xu_k)) - xu_k\cos(xu_k)}{2u_k}\\ A_2 &= \frac{4 - 4\cos(xu_k)) - xu_k\sin(xu_k)}{2u_k^2}\\ A_3 &= \frac{\sin(xu_k)) - xu_k\cos(xu_k)}{2u_k^3}\\ A_4 &= \frac{2 - 2\cos(xu_k)) - xu_k\sin(xu_k)}{2u_k^4}\\ u_k &= \|\mathbf{u}_k\|_2 = \sqrt{\kappa_k^2 + \tau_k^2} \end{aligned} \tag{31}$$

With the definition of equation (31), we develop the integral in equation (22) as

$$\int_{s_{k-1}}^{s_k} e^{x\mathbf{ad}_{\widehat{\mathbf{f}}_k}} ds = \begin{cases}\begin{pmatrix}x_k\mathbf{I}_6 + B_1\mathbf{ad}_{\widehat{\mathbf{f}}_k} + B_2\mathbf{ad}^2_{\widehat{\mathbf{f}}_k}\\ + B_3\mathbf{ad}^3_{\widehat{\mathbf{f}}_k} + B_4\mathbf{ad}^4_{\widehat{\mathbf{f}}_k}\end{pmatrix}, & u_k \neq 0\\ x_k\mathbf{I}_{6\times 6} + \frac{1}{2}x_k^2\mathbf{ad}_{\widehat{\mathbf{f}}_k}, & u_k = 0\end{cases} \tag{32}$$

where

$$\begin{aligned} B_1 &= \frac{4 - 4\cos(x_k u_k) - x_k u_k\sin(x_k u_k)}{2u_k^2}\\ B_2 &= \frac{4x_k u_k - 5\sin(x_k u_k) + x_k u_k\cos(x_k u_k)}{2u_k^3}\\ B_3 &= \frac{2 - 2\cos(x_k u_k) - x_k u_k\sin(x_k u_k)}{2u_k^4}\\ B_4 &= \frac{2x_k u_k - 3\sin(x_k u_k) + x_k u_k\cos(x_k u_k)}{2u_k^5} \end{aligned} \tag{33}$$

REFERENCES

[1] I. D. Walker, "Continuous backbone continuum robot manipulators," *International Scholarly Research Notices*, vol. 2013, 2013.

[2] J. K. Hopkins, B. W. Spranklin, and S. K. Gupta, "A survey of snake-inspired robot designs," *Bioinspiration & biomimetics*, vol. 4, no. 2, p. 021001, 2009.

[3] J. M. M. Tur and W. Garthwaite, "Robotic devices for water main in-pipe inspection: A survey," *Journal of Field Robotics*, vol. 4, no. 27, pp. 491–508, 2010.

[4] J. Burgner-Kahrs, D. C. Rucker, and H. Choset, "Continuum robots for medical applications: A survey," *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1261–1280, 2015.

[5] R. J. Webster III and B. A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *The International Journal of Robotics Research*, vol. 29, no. 13, pp. 1661–1683, 2010.

[6] R. E. Goldman, A. Bajo, and N. Simaan, "Compliant motion control for multisegment continuum robots with actuation force sensing," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 890–902, 2014.

[7] B. A. Jones and I. D. Walker, "Kinematics for multisection continuum robots," *IEEE Transactions on Robotics*, vol. 22, no. 1, pp. 43–55, 2006.

[8] J. Barrientos-Diez, X. Dong, D. Axinte, and J. Kell, "Real-time kinematics of continuum robots: modelling and validation," *Robotics and Computer-Integrated Manufacturing*, vol. 67, p. 102019, 2021.

[9] M. W. Hannan and I. D. Walker, "Kinematics and the implementation of an elephant's trunk manipulator and other continuum style robots," *Journal of robotic systems*, vol. 20, no. 2, pp. 45–63, 2003.

[10] G. S. Chirikjian and J. W. Burdick, "A modal approach to hyper-redundant manipulator kinematics," *IEEE Transactions on Robotics and Automation*, vol. 10, no. 3, pp. 343–354, 1994.

[11] F. C. Park, "Computational aspects of the product-of-exponentials formula for robot kinematics," *IEEE Transactions on Automatic Control*, vol. 39, no. 3, pp. 643–647, 1994.

[12] F. Renda, M. Cianchetti, H. Abidi, J. Dias, and L. Seneviratne, "Screw-based modeling of soft manipulators with tendon and fluidic actuation," *Journal of Mechanisms and Robotics*, vol. 9, no. 4, 2017.

[13] J. Zhang, K. Lu, J. Yuan, J. Di, and G. He, "Kinematics modeling and motion planning of tendon driven continuum manipulators," *Journal of Artificial Intelligence and Technology*, vol. 1, no. 1, pp. 28–36, 2021.

[14] B. A. Jones, R. L. Gray, and K. Turlapati, "Three dimensional statics for continuum robotics," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2009, pp. 2659–2664.

[15] R. M. Murray, Z. Li, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 2017.

[16] F. Bullo and R. M. Murray, "Proportional derivative (pd) control on the euclidean group," vol. 2, pp. 1091–1097, 1995.

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

[18] J. M. Selig, *Geometric fundamentals of robotics*. Springer, 2005, vol. 128.

[19] C. W. Wampler, "Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 16, no. 1, pp. 93–101, 1986.