Variable Curvature Modelling Method of Continuum Robots with Constraints

Peng Chen

Shenyang Institute of Automation Chinese Academy of Sciences

Yi Yu

Shenyang Institute of Automation Chinese Academy of Sciences

Yuwang Liu (✉ liuyuwang@sia.cn)

Shenyang Institute of Automation, Chinese Academy of Sciences <https://orcid.org/0000-0003-2870-5865>

Research Article

Keywords: Continuum robots, Variable curvature modelling, Boundary conditions, Nonlinear mechanics

Posted Date: November 22nd, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-1021986/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Variable Curvature Modelling Method of Continuum Robots with Constraints

Authors: Peng Chen^{1,2,3}, Yi Yu^{1,2,3}, and Yuwang Liu^{1,2,3*}

Affiliations:

1. State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China
2. Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Science, Shenyang 110016, China
3. University of Chinese Academy of Sciences, Beijing 100000, China

Corresponding author:

Yuwang Liu

State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences
114 Nanta Street, Shenyang City, Liaoning Province, 110016, China

Phone: +86-15840445933

Email: liuyuwang@sia.cn

Abstract: The inherent compliance of continuum robots holds great promise in the fields of soft manipulation and safe human-robot interaction. This compliance reduces the risk of damage to the manipulated object and the surroundings. However, continuum robots have theoretically infinite degrees of freedom, and this high flexibility usually leads to complex deformations with external forces and positional constraints. How to describe this complex deformation is the main challenge for modelling continuum robots. In this study, we investigated a novel variable curvature modeling method for continuum robots, considering external forces and positional constraints. The robot configuration curve is described by the developed mechanics model, and then the robot is fitted to the curve. To validate the model, a 10-section continuum robot prototype with a length of 1 m was developed. The ability of the robot to reach the target points and track complex trajectories with load verified the feasibility and accuracy of the model. The ratio of the average position error of the robot endpoint to the robot length was less than 2.38%. This work may serve a new perspective for design analysis and motion control of continuum robots.

Keywords: Continuum robots; Variable curvature modelling; Boundary conditions; Nonlinear mechanics

1. Introduction

Owing to intrinsic compliance, environmental adaptability, and operational safety, continuum robots have become an area of great interest in robotics [1]. A large number of researchers have conducted related research [2], [3], and many new continuum robots have been developed. Renda et al. [4] and Xu et al. [5] designed cable-driven continuum robots, Marchese et al. [6] and Tutcu et al. [7] developed pneumatic continuum robots. Gu et al. [8], [9] developed dielectric elastomer-driven continuum robots.

These continuum robots show great potential in a wide range of applications, such as medical equipment, unstructured environment exploration and soft manipulation [10]. However, continuum robots have a theoretically infinite number of degrees of freedom (DOF), and this high flexibility allows complex deformations of the robot in response to external forces and positional constraints. Therefore, the accurate and efficient modelling of continuum robots with external force remains challenging [11].

Different from the traditional rigid rod robot, the continuum robots achieved movement by deforming themselves. Therefore, the kinematics of continuum robots can be replaced by mechanical analysis. The continuum robots are usually discretized into a series of points when solving the model. A spatial curve can be determined by these points position mathematically. The continuum robots fitted to these spatial curves as closely as possible [12], [13]. The calculation methods of spatial curves mainly include constant curvature approach and variable curvature approach.

The constant curvature approach is a simplified approach to modelling continuum robots, which assumes that the curvature of curve is same between discrete points. The advantages of closed kinematics and easily solution make this method widely used of continuum robots modelling. In order to improve the accuracy and dexterity, Jones and Walker [14] developed a geometrical approach for modeling a constant curvature continuum robot, which was used to obtain a closed-form model. Freixedes et al. [15] established an optimization framework for continuum robots based on the assumption of constant curvature and deduced optimized structural parameters. They proposed the kinematic model to describe the deflection characteristics of the contact-assisted continuum robot, and carried out the experimental verification. Webster et al. [16] unified the kinematics and differential kinematics results of single-segment and multi-segment continuous robots with constant curvature into the common coordinate system and symbol setting. Della Santina et al. [17] analyzed the flaws of constant models, considering an alternative state representation to solve these issues. Simulation cases are used to support the theoretical analysis. In order to simulate contact with the environment, Schiller et al. [18] extended the constant curvature method based on the energy minimization technique, and the proposed model can perform robot kinematics well.

Due to the constant curvature approaches does not completely match all continuum robot characteristics, some variable curvature approaches have been developed. Based on the Lagrangian polynomial series solution method, Ritz and Ritz-Galerkin methods, Hadi Sadati et al. [19] minimized the continuum robot configuration to the geometric positions of a few physical points. Based on Pythagorean hodograph (PH) curves, Singh et al. [20] proposed a quantitative modeling method to obtain three-dimensional reconstructions of the configurations of a continuum robot with a variable curvature. Gonthina et al [21] proposed a cross-sectional modeling method for variable curvature continuum robots based on Eulerian spiral curves. They compared the simulation results with the constant curvature method, proving that the proposed method is a significantly better match for various configurations of robot hardware. The modeling approach described above provides a good description of the forward and inverse kinematics of a soft continuous robot with variable curvature, but focuses mainly on the geometric description of the robot, without considering the effect of external forces. Therefore, additional visual or displacement sensors are needed to measure the robot configuration in real time. Some scholars have considered the influence of external forces and proposed some meaningful modeling methods. Godage et al. [22] simulates the transient and steady-state dynamics of the continuum robot prototype based on the lumped parameter model. Renda et al. [23], [24] established a mechanical model of a short-thick continuum robots under the action of external forces based on the Cosserat theory. The experiment verified the most typical movements of the octopus: bending, stretching and grasping. Based on the

The backbone is discrete into n elements, and N_i represents the i_{th} nodes. There are m constraint disks, where the segment between $disk_{i-1}$ and $disk_i$ is called Seg_i , which length is Δs . A static analysis was performed on the micro element Seg_i of the robot in the inertial coordinate system $O-\xi\eta\zeta$, as shown in Fig. 1(b). The spindle coordinate system of N_i is $p_i-x_iy_iz_i$. The vector of node N_i with respect to O are r_i . The internal force at N_i are F_i . The gravity distribution force is f_g . The relative rotation angle between the sections of adjacent nodes is $\Delta\phi$. Evidently, there are a ($a = n/m$) nodes in Seg_i . The length of PAM_j in Seg_i is $l_{i,j}$. The external load force at the endpoint of the robot is F_{load} .

Suppose the projection vector of the z -axis unit vector in $O-\xi\eta\zeta$ is T . The backbone configuration can be obtained from (1),

$$r(s) = \int_0^s T(\delta) d\delta \quad (1)$$

where δ denotes the integral variable.

Suppose the rate of section angular displacement ϕ with respect to arc coordinate s is ω ,

$$\omega = \lim_{\Delta s \rightarrow 0} \frac{\Delta\phi}{\Delta s} \quad (2)$$

Using the infinitesimal rotation theory of a rigid body, the relationship between the ω and quaternions (q_1, q_2, q_3, q_4) can be derived:

$$\begin{cases} \omega_x = 2(-q_2 \frac{dq_1}{ds} + q_1 \frac{dq_2}{ds} + q_4 \frac{dq_3}{ds} - q_3 \frac{dq_4}{ds}) \\ \omega_y = 2(-q_3 \frac{dq_1}{ds} - q_4 \frac{dq_2}{ds} + q_1 \frac{dq_3}{ds} + q_2 \frac{dq_4}{ds}) \\ \omega_z = 2(-q_4 \frac{dq_1}{ds} + q_3 \frac{dq_2}{ds} - q_2 \frac{dq_3}{ds} + q_1 \frac{dq_4}{ds}) \end{cases} \quad (3)$$

Writing Eq. (3) as matrix form,

$$\omega = \Gamma q' \quad (4)$$

where $q' = \frac{dq}{ds}$, and Γ is denoted as

$$\Gamma = 2 \begin{bmatrix} -q_2 & q_1 & q_4 & -q_3 \\ -q_3 & -q_4 & q_1 & q_2 \\ -q_4 & q_3 & -q_2 & q_1 \end{bmatrix} \quad (5)$$

The total energy E_t of the backbone includes elastic strain energy E_e and external force potential energy E_p . When the robot is in balance,

$$\delta E_t = \delta E_e + \delta E_p = 0 \quad (6)$$

The elastic strain energy E_e of the backbone is

$$E_e = \frac{1}{2} \int_0^L \left[k_x (\omega_x - \omega_x^0)^2 + k_y (\omega_y - \omega_y^0)^2 + k_z (\omega_z - \omega_z^0)^2 \right] ds \quad (7)$$

where L denotes the length of the robot, and (k_x, k_y, k_z) represents the bending stiffness around the coordinate axis, and $(\omega_x^0, \omega_y^0, \omega_z^0)$ represents the initial state of ω .

$$\begin{cases} k_x = E \frac{\pi d^4}{64} \\ k_y = E \frac{\pi d^4}{64} \\ k_z = G \frac{\pi d^4}{32} \end{cases} \quad (8)$$

where d is the diameter of backbone, and E and G are the Young's modulus and shear modulus, respectively.

The variation of Eq. (7) is

$$\delta E_e = \frac{1}{2} \int_0^L \left[k_x (\omega_x - \omega_x^0) \delta \omega_x + k_y (\omega_y - \omega_y^0) \delta \omega_y + k_z (\omega_z - \omega_z^0) \delta \omega_z \right] ds \quad (9)$$

Writing Eq. (9) as matrix form,

$$\delta E_e = \int_0^L \left\{ \left[\mathbf{K}(\boldsymbol{\omega} - \boldsymbol{\omega}^0) \right]^T \delta \boldsymbol{\omega} \right\} ds \quad (10)$$

where $\boldsymbol{\omega}^0 = (\omega_x^0 \ \omega_y^0 \ \omega_z^0)^T$, $\mathbf{K} = \text{diag}(k_x \ k_y \ k_z)$.

The variation of Eq. (4) is

$$\delta \boldsymbol{\omega} = \boldsymbol{\Gamma} \delta \mathbf{q}' - \boldsymbol{\Gamma}' \delta \mathbf{q} \quad (11)$$

Substituting Eq. (11) into Eq. (10), we can obtain

$$\delta E_e = \int_0^L \left\{ \left[\mathbf{K}(\boldsymbol{\omega} - \boldsymbol{\omega}^0) \right]^T (\boldsymbol{\Gamma} \delta \mathbf{q}' - \boldsymbol{\Gamma}' \delta \mathbf{q}) \right\} ds \quad (12)$$

The external force potential energy E_p can be expressed by the internal force $\mathbf{F}$ as

$$\delta E_p = -\delta \int_0^L \mathbf{F} \cdot \mathbf{T} ds \quad (13)$$

In the $O - \xi \eta \zeta$, the force balance of Seg_i is

$$\mathbf{F}_i - \mathbf{F}_{i-1} + \mathbf{f}_g \Delta s = \mathbf{0} \quad (14)$$

Dividing the sides of Eq. (14) by Δs , and considering the condition $\Delta s \rightarrow 0$, we can obtain,

$$\frac{d\mathbf{F}}{ds} + \mathbf{f}_g = \mathbf{0} \quad (15)$$

In the spindle coordinate system $p - xyz$, Eq. (15) needs to be rewritten as

$$\frac{d\mathbf{F}}{ds} + \boldsymbol{\omega} \times \mathbf{F} + \mathbf{f}_g = \mathbf{0} \quad (16)$$

2.2 Discretization of the equations

The quaternions at node N_i is denoted as $\mathbf{q}_i (k=1,2,3,4)$, $\mathbf{q}_i = [q_{1,i} \ q_{2,i} \ q_{3,i} \ q_{4,i}]^T$. Performing linear interpolation in Seg_i , and the quaternions and its derivative in Seg_i are

$$\begin{cases} \bar{q}_{k,i} = \frac{q_{k,i-1} + q_{k,i}}{2} \\ q'_{k,i} = \frac{q_{k,i} - q_{k,i-1}}{\Delta s} \end{cases} \quad (k=1,2,3,4; i=1,2,\dots,n) \quad (17)$$

The quaternions at nodes N_{i-1} and N_i are combined into an 8-order array, which is denoted as

$$\mathbf{q}_{i-1,i} = (\mathbf{q}_{i-1}^T \ \mathbf{q}_i^T)^T \quad (18)$$

Then, Eq. (17) can be expressed as

$$\begin{cases} \bar{\mathbf{q}}_i = \boldsymbol{\Psi}_4 \mathbf{q}_{i-1,i} \\ \mathbf{q}'_i = \boldsymbol{\Phi}_4 \mathbf{q}_{i-1,i} \end{cases} \quad (19)$$

where $\boldsymbol{\Psi}_4$ and $\boldsymbol{\Phi}_4$ are 4×8 matrices composed of a 4-order unit matrix $\mathbf{E}_4$,

$$\boldsymbol{\Psi}_4 = \frac{1}{2} (\mathbf{E}_4 \ \mathbf{E}_4) \quad \boldsymbol{\Phi}_4 = \frac{1}{\Delta s} (-\mathbf{E}_4 \ \mathbf{E}_4) \quad (20)$$

The average and derivative of $\boldsymbol{\Gamma}$ in Seg_i are denoted as $\bar{\boldsymbol{\Gamma}}_i$ and $\boldsymbol{\Gamma}'_i$, respectively.

$$\begin{cases} \bar{\boldsymbol{\Gamma}}_i(\bar{\mathbf{q}}_i) = \frac{1}{2} [\boldsymbol{\Gamma}_{i-1}(\mathbf{q}_{i-1}) + \boldsymbol{\Gamma}_i(\mathbf{q}_i)] \\ \boldsymbol{\Gamma}'_i(\mathbf{q}'_i) = \frac{1}{\Delta s} [\boldsymbol{\Gamma}_{i-1}(\mathbf{q}_{i-1}) - \boldsymbol{\Gamma}_i(\mathbf{q}_i)] \end{cases} \quad (21)$$

where

$$\boldsymbol{\Gamma}_i(\mathbf{q}_i) = 2 \begin{bmatrix} -q_{2,i} & q_{1,i} & q_{4,i} & -q_{3,i} \\ -q_{3,i} & -q_{4,i} & q_{1,i} & q_{2,i} \\ -q_{4,i} & q_{3,i} & -q_{2,i} & q_{1,i} \end{bmatrix} \quad (22)$$

The average and derivative of $\boldsymbol{\omega}$ in Seg_i are denoted as $\bar{\boldsymbol{\omega}}_i$ and $\boldsymbol{\omega}'_i$, respectively.

$$\begin{cases} \bar{\boldsymbol{\omega}}_i = \bar{\Gamma}_i(\bar{\mathbf{q}}_i)\mathbf{q}'_i \\ \delta\bar{\boldsymbol{\omega}}_i = \bar{\Gamma}'_i(\bar{\mathbf{q}}_i)\delta\mathbf{q}'_i - \Gamma'(\mathbf{q}'_i)\delta\bar{\mathbf{q}}_i \end{cases} \quad (23)$$

Substituting (19) into (23), we can obtain

$$\begin{cases} \bar{\boldsymbol{\omega}}_i = \bar{\Gamma}_i\bar{\boldsymbol{\Phi}}_4\mathbf{q}_{i-1,i} \\ \delta\bar{\boldsymbol{\omega}}_i = (\bar{\Gamma}_i\bar{\boldsymbol{\Phi}}_4 - \Gamma_i\boldsymbol{\Psi}_4)\delta\mathbf{q}_{i-1,i} \end{cases} \quad (24)$$

Eq. (12) can be discretized as

$$\delta E_e = \sum_{i=1}^n \left[(\mathbf{q}_{i-1,i}^T \mathbf{K} \boldsymbol{\Phi}_4^T \bar{\Gamma}_i^T - \boldsymbol{\omega}_i^{OT}) (\bar{\Gamma}_i \bar{\boldsymbol{\Phi}}_4 - \Gamma_i \boldsymbol{\Psi}_4) \right] \delta \mathbf{q}_{i-1,i} \quad (25)$$

Eq. (25) can be simplified as

$$\delta E_e = \sum_{i=1}^n (\mathbf{q}_{i-1,i}^T \mathbf{A}_i - \mathbf{B}_i) \delta \mathbf{q}_{i-1,i} \quad (26)$$

where

$$\begin{cases} \mathbf{A}_i = \mathbf{K} \boldsymbol{\Phi}_4^T \bar{\Gamma}_i^T (\bar{\Gamma}_i \bar{\boldsymbol{\Phi}}_4 - \Gamma_i \boldsymbol{\Psi}_4) \\ \mathbf{B}_i = \boldsymbol{\omega}_i^{OT} (\bar{\Gamma}_i \bar{\boldsymbol{\Phi}}_4 - \Gamma_i \boldsymbol{\Psi}_4) \end{cases} \quad (27)$$

$\mathbf{A}_i$ is arranged diagonally, and each matrix is moved 4 rows and 4 columns to the upper left, then overlapping elements are added, and matrix $\mathbf{A}$ is formed. In the same way, $\mathbf{B}_i$ is arranged horizontally, and each matrix is moved four columns to the left, then overlapping elements are added, and the matrix $\mathbf{B}$ is formed. Eq. (24) can be expressed as

$$\delta E_e = (\mathbf{q}^T \mathbf{A} - \mathbf{B}) \delta \mathbf{q} \quad (28)$$

where $\mathbf{q} = [\mathbf{q}_0^T \quad \mathbf{q}_1^T \quad \dots \quad \mathbf{q}_n^T]^T$.

The projection of $\mathbf{T}$ in $O - \xi\eta\zeta$ can be expressed by a quaternions as

$$\mathbf{T} = \begin{bmatrix} 2(q_2q_4 + q_1q_3) & 2(q_3q_4 - q_1q_2) & q_1^2 - q_2^2 - q_3^2 + q_4^2 \end{bmatrix}^T \quad (29)$$

$\mathbf{T}_i$ can be discretized as

$$\mathbf{T}_i = \frac{1}{3} \begin{bmatrix} (2q_{2,i-1} + q_{2,i})q_{4,i-1} + (2q_{2,i} + q_{2,i-1})q_{4,i} + (2q_{1,i-1} + q_{1,i})q_{3,i-1} + (2q_{1,i} + q_{1,i-1})q_{3,i} \\ (2q_{3,i-1} + q_{3,i})q_{4,i-1} + (2q_{3,i} + q_{3,i-1})q_{4,i} - (2q_{1,i-1} + q_{1,i})q_{2,i-1} - (2q_{1,i} + q_{1,i-1})q_{2,i} \\ (q_{1,i-1}^2 + q_{1,i-1}q_{1,i} + q_{1,i}^2 - q_{2,i-1}^2 - q_{2,i-1}q_{2,i} - q_{2,i}^2 - q_{3,i-1}^2 - q_{3,i-1}q_{3,i} - q_{3,i}^2 + q_{4,i-1}^2 + q_{4,i-1}q_{4,i} + q_{4,i}^2) \end{bmatrix} \quad (30)$$

Then, $\mathbf{T}_i$ can be expressed by $\mathbf{S}_i$ as follows:

$$\mathbf{T}_i = \frac{1}{3} \mathbf{S}_i \mathbf{q}_{i-1,i} \quad (31)$$

where

$$\begin{aligned} \mathbf{S}_i^{(1)} &= \begin{bmatrix} 2q_{3,i-1} & 2q_{4,i-1} & q_{1,i} & q_{2,i} \\ -2q_{2,i-1} & -q_{1,i} & 2q_{4,i-1} & q_{3,i} \\ q_{1,i-1} + q_{1,i} & -(q_{2,i-1} + q_{2,i}) & -(q_{3,i-1} + q_{3,i}) & q_{4,i-1} + q_{4,i} \end{bmatrix} \\ \mathbf{S}_i^{(2)} &= \begin{bmatrix} 2q_{3,i} & 2q_{4,i} & q_{1,i-1} & q_{2,i-1} \\ -2q_{2,i} & -q_{1,i-1} & 2q_{4,i} & q_{3,i-1} \\ q_{1,i} & q_{2,i} & q_{3,i} & q_{4,i} \end{bmatrix} \\ \mathbf{S}_i &= [\mathbf{S}_i^{(1)} \quad \mathbf{S}_i^{(2)}] \end{aligned} \quad (32)$$

The average and derivative of $\mathbf{F}$ in Seg_i are denoted as $\bar{\mathbf{F}}_i$ and $\mathbf{F}'_i$, respectively.

$$\bar{\mathbf{F}}_i = \frac{\mathbf{F}_{i-1} + \mathbf{F}_i}{2} \quad \mathbf{F}'_i = \frac{\mathbf{F}_i - \mathbf{F}_{i-1}}{\Delta s} \quad (33)$$

where $\mathbf{F}_i = [F_{x,i} \quad F_{y,i} \quad F_{z,i}]^T$.

The internal force $\mathbf{F}_{i-1}$ and $\mathbf{F}_i$ at nodes N_{i-1} and N_i are combined into an 6-order array, which is denoted as

$$\mathbf{F}_{i-1,i} = (\mathbf{F}_{i-1}^T \quad \mathbf{F}_i^T)^T \quad (34)$$

Then, Eq. (33) can be expressed as

$$\begin{cases} \bar{\mathbf{F}}_i = \mathbf{\Psi}_3 \mathbf{F}_{i-1,i} \\ \mathbf{F}'_i = \mathbf{\Phi}_3 \mathbf{F}_{i-1,i} \end{cases} \quad (35)$$

where $\mathbf{\Psi}_3$ and $\mathbf{\Phi}_3$ are 3×6 matrices composed of a 3-order unit matrix $\mathbf{E}_3$,

$$\begin{cases} \mathbf{\Psi}_3 = \frac{1}{2}(\mathbf{E}_3 & \mathbf{E}_3) \\ \mathbf{\Phi}_3 = \frac{1}{\Delta s}(-\mathbf{E}_3 & \mathbf{E}_3) \end{cases} \quad (36)$$

Eq. (13) can be discretized as

$$\delta E_p = -\frac{1}{3} \sum_{i=1}^n \mathbf{F}_{i-1,i}^T \mathbf{\Psi}_3^T \mathbf{S}_i \delta \mathbf{q}_{i-1,i} \quad (37)$$

Let

$$\mathbf{U}_i = \mathbf{\Psi}_3^T \mathbf{S}_i \quad (38)$$

Matrices $\mathbf{U}_i$ are arranged horizontally, where each matrix is moved by four columns to the left to form matrix $\mathbf{U}$. Then, Eq. (33) can be abbreviated as follows:

$$\delta E_p = -\mathbf{F}^T \mathbf{U} \delta \mathbf{q} \quad (39)$$

where $\mathbf{F} = [\mathbf{F}_0^T \quad \mathbf{F}_1^T \quad \dots \quad \mathbf{F}_n^T]^T$.

Substituting Eq. (28) and (39) into Eq. (6), the variation of E_t is as follows:

$$\delta E_t = (\mathbf{q}^T \mathbf{A} - \mathbf{B} - \mathbf{F}^T \mathbf{U}) \delta \mathbf{q} \quad (40)$$

Eq. (41) can be obtained from Eq. (6):

$$\mathbf{q}^T \mathbf{A} - \mathbf{B} - \mathbf{F}^T \mathbf{U} = \mathbf{0} \quad (41)$$

Eq. (16) can be discretized as

$$\begin{cases} \frac{F_{x,i} - F_{x,i-1}}{\Delta s} + \omega_{y,i} F_{z,i} - \omega_{z,i} F_{y,i} = 0 \\ \frac{F_{y,i} - F_{y,i-1}}{\Delta s} + \omega_{z,i} F_{x,i} - \omega_{x,i} F_{z,i} = 0 \\ \frac{F_{z,i} - F_{z,i-1}}{\Delta s} + \omega_{x,i} F_{y,i} - \omega_{y,i} F_{x,i} + f_g = 0 \end{cases} \quad \forall i = 1 \sim n \quad (42)$$

Eqs. (41) and (42) are the discrete balance equations.

2.3 Boundary conditions

According to Eq. (1), the coordinate of any point on the backbone is

$$\mathbf{r}(s) = \begin{bmatrix} \xi(s) \\ \eta(s) \\ \zeta(s) \end{bmatrix} = \begin{bmatrix} 2 \int_0^s (q_2(\sigma) q_4(\sigma) + q_1(\sigma) q_3(\sigma)) d\sigma \\ 2 \int_0^s (q_3(\sigma) q_4(\sigma) - q_1(\sigma) q_2(\sigma)) d\sigma \\ \int_0^s (q_1^2(\sigma) - q_2^2(\sigma) - q_3^2(\sigma) + q_4^2(\sigma)) d\sigma \end{bmatrix} \quad (43)$$

where σ denotes the integral variable.

Eq. (43) can be discretized by Eq. (30) as

$$\begin{cases} \xi_i = \frac{\Delta s}{3} ((2q_{2,i-1} + q_{2,i})q_{4,i-1} + (2q_{2,i} + q_{2,i-1})q_{4,i} \\ \quad + (2q_{1,i-1} + q_{1,i})q_{3,i-1} + (2q_{1,i} + q_{1,i-1})q_{3,i}) + \xi_{i-1} \\ \eta_i = \frac{\Delta s}{3} ((2q_{3,i-1} + q_{3,i})q_{4,i-1} + (2q_{3,i} + q_{3,i-1})q_{4,i} \\ \quad - (2q_{1,i-1} + q_{1,i})q_{2,i-1} - (2q_{1,i} + q_{1,i-1})q_{2,i}) + \eta_{i-1} \\ \zeta_i = \frac{\Delta s}{3} (q_{1,i-1}^2 + q_{1,i-1}q_{1,i} + q_{1,i}^2 - q_{2,i-1}^2 - q_{2,i-1}q_{2,i} - q_{2,i}^2 \\ \quad - q_{3,i-1}^2 - q_{3,i-1}q_{3,i} - q_{3,i}^2 + q_{4,i-1}^2 + q_{4,i-1}q_{4,i} + q_{4,i}^2) + \zeta_{i-1} \end{cases} \quad (44)$$

When continuum robot operating, the endpoint usually needs to be controlled to move to a specified position, so the position of the endpoint needs to be limited. Let the include point of the robot be at the origin O , we obtain:

$$\begin{cases} \xi_n - P_{n,\xi} = 0 \\ \eta_n - P_{n,\eta} = 0 \\ \zeta_n - P_{n,\zeta} = 0 \end{cases} \quad (45)$$

where $(P_{n,\xi}, P_{n,\eta}, P_{n,\zeta})$ is the desired coordinate of the endpoint.

In addition, the pose of the robot at the initial and end point also needs to be constrained:

$$\begin{cases} q_{1,0} - Q_{1,0} = 0 \\ q_{2,0} - Q_{2,0} = 0 \\ q_{3,0} - Q_{3,0} = 0 \\ q_{4,0} - Q_{4,0} = 0 \end{cases} \quad \begin{cases} q_{1,n} - Q_{1,n} = 0 \\ q_{2,n} - Q_{2,n} = 0 \\ q_{3,n} - Q_{3,n} = 0 \\ q_{4,n} - Q_{4,n} = 0 \end{cases} \quad (46)$$

where $(Q_{1,0}, Q_{2,0}, Q_{3,0}, Q_{4,0})$ and $(Q_{1,n}, Q_{2,n}, Q_{3,n}, Q_{4,n})$ represent the desired quaternions of the initial and end point, respectively.

When the robot has a load, it is also necessary to add an endpoint force boundary condition,

$$\mathbf{F}_n - \mathbf{F}_{load} = \mathbf{0} \quad (47)$$

Eqs. (45), (46) and (47) constitute the boundary conditions of the continuous robot model.

2.4 Solving algorithm

The continuum robot model composed of Eqs. (39), (40), (43), (44) and (45) can be expressed as:

$$\begin{cases} \mathbf{h}_1 = \mathbf{q}^T \mathbf{A} - \mathbf{B} - \mathbf{F}^T \mathbf{U} = \mathbf{0} \\ \mathbf{h}_2 = \Phi_3 \mathbf{F}_{i-1,i} + \mathbf{f}_g = \mathbf{0} \quad \forall i \in 1 \sim n \\ \mathbf{h}_3 = \mathbf{r}_n - \mathbf{P}_n = \mathbf{0} \\ \mathbf{h}_4 = \mathbf{q}_0 - \mathbf{Q}_0 = \mathbf{0} \\ \mathbf{h}_5 = \mathbf{q}_n - \mathbf{Q}_n = \mathbf{0} \\ \mathbf{h}_6 = \mathbf{F}_n - \mathbf{F}_{load} = \mathbf{0} \end{cases} \quad (46)$$

By using the least square method, Eq. (46) can be transformed into an optimization problem:

$$\begin{aligned} & \min H(\mathbf{X}) \\ & H = \frac{1}{2} \sum_{i=1}^u h_i^2(\mathbf{X}) \end{aligned} \quad (47)$$

where u is the dimension of $\mathbf{h}$, $\mathbf{X} = [\mathbf{q}_1 \quad \mathbf{q}_2 \quad \mathbf{q}_3 \quad \mathbf{q}_4 \quad \mathbf{F}_x \quad \mathbf{F}_y \quad \mathbf{F}_z]^T$.

The particle swarm optimization (PSO) algorithm and Levenberg-Marquardt (LM) algorithm are used to solve (36). The PSO algorithm has strong global but weak local optimization ability, and LM algorithm is the opposite. Therefore, we first use PSO algorithm to find the appropriate iteration initial value, and then bring it into LM algorithm to optimize the solution. When finding the iteration initial value, there are two steps. In the first step the PSO algorithm is used, and in the second step the gradient descent method is used to modify the particle velocity term in PSO algorithm. In this way, a better initial iteration value can be obtained. The process is shown in Algorithm 1.

3 Results

3.1 Experimental preparation

A 10 section prototype of a continuum robot driven by PAMs was developed, as shown in Fig. 3(a). The structure of the prototype is shown in Fig. 3(b). The backbone is an elastic rod, and each section is driven by three PAMs. By changing the length of the PAMs, which can be achieved by inflating and deflating [27], [28], the robot can be controlled to bend in a three-dimensional space. The control system

Algorithm1

```
 $t = 1$ ; Generated random initial population  $\mathbf{x}^1, \mathbf{v}^1$ .  
 $\mathbf{X} \leftarrow \text{InitValue}(\mathbf{x}^1, \mathbf{v}^1)$   
 $\mathbf{X} \leftarrow \text{Optimization}(\mathbf{X})$   
for  $t = 1 : t_{\max}$   
     $\mathbf{w}^{t+1} = |\sigma c(\mathbf{X}) - \mathbf{w}^t|$   
    Update  $\mathbf{f}(\mathbf{X}), F(\mathbf{X})$   
     $\mathbf{X} \leftarrow \text{Optimization}(\mathbf{X})$   
end for  
function  $\mathbf{X} \leftarrow \text{InitValue}(\mathbf{x}^1, \mathbf{v}^1)$   
    for  $k = 1 : \text{Ite}_1 + \text{Ite}_2$   
         $\mathbf{v}^k_i = \chi(\lambda^k \mathbf{v}^k_i + c_1 r_{i1}^k (\mathbf{p}^k_i - \mathbf{x}^k_i) + c_2 r_{i2}^k (\mathbf{g}^k - \mathbf{x}^k_i))$   
         $\mathbf{x}^{k+1}_i = \mathbf{x}^k_i + \mathbf{v}^{k+1}_i$   
         $\mathbf{p}^k_i \leftarrow$  the optimal solution of the  $i_{th}$  particle  
        if  $k \in [1, \text{Ite}_1]$   
             $\mathbf{g}^k \leftarrow$  the optimal solution in all particles  
        end if  
        if  $k \in [\text{Ite}_1 + 1, \text{Ite}_1 + \text{Ite}_2]$   
             $\mathbf{g}^k \leftarrow \text{SearchStep}(\mathbf{g}^k) + \mathbf{g}^k$   
        end if  
         $\mathbf{X} \leftarrow \mathbf{g}^k$   
    end for  
end function  
function  $\mathbf{d}(\mathbf{X}) \leftarrow \text{SearchStep}(\mathbf{X})$   
     $\mathbf{d}(\mathbf{X}) = -[\mathbf{J}(\mathbf{X})^T \mathbf{J}(\mathbf{X}) + \mu \mathbf{I}]^{-1} \nabla H(\mathbf{X})$   
    while  $H(\mathbf{X} + \beta^m \mathbf{d}(\mathbf{X})) < H(\mathbf{X}) + \alpha \beta^m \nabla H(\mathbf{X}) \mathbf{d}(\mathbf{X})$   
         $m = m + 1$   
    end  
     $\mathbf{d}(\mathbf{X}) = \mathbf{d}(\mathbf{X}) + \beta^m \mathbf{d}(\mathbf{X})$   
function  $\mathbf{X} \leftarrow \text{Optimization}(\mathbf{X})$   
    while  $H(\mathbf{X}) > \varepsilon_1 \ \&\& \ |\nabla H(\mathbf{X})| > \varepsilon_2$   
         $\mathbf{d}(\mathbf{X}) \leftarrow \text{SearchStep}(\mathbf{X})$   
         $\rho = \frac{H(\mathbf{X} + \mathbf{d}(\mathbf{X})) - H(\mathbf{X})}{\nabla H(\mathbf{X}) \mathbf{d}(\mathbf{X}) + \frac{1}{2} \mathbf{d}(\mathbf{X})^T \nabla^2 H(\mathbf{X}) \mathbf{d}(\mathbf{X})}$   
        if  $\rho < 0.45$   $\mu = \mu q$  end if  
        if  $\rho > 0.75$   $\mu = \mu / q$  end if  
        if  $0.45 \leq \rho \leq 0.75$   $\mu = \mu$  end if  
    end while  
end function
```

is composed of upper computer (PC), lower computer (STM 32), relays and solenoid valves. The communication between the upper computer and the lower computer via Bluetooth. The command is sent from the upper computer to the lower computer, and the opening and closing time of the solenoid valve is controlled by the relay to control the duration of inflation and deflation, thereby adjusting the length of the PAMs. The parameters of the prototype are shown in Table I.

Fig. 3. (a) Experimental platform. (b) Schematic diagram of the prototype. (c) The spaces and mappings between them.

Table 1. Robot parameters

symbol	meaning	value
L	Robot length	1000mm
d	Backbone diameter	2.5mm
E	Young's modulus	6.2MPa
G	Shear modulus	2.4Mpa
f_g	Robot gravity	0.004N/mm

The analysis of the robot is performed in four main spaces: actuator space, joint space, configuration space and task space. For the pneumatic continuum robot, actuator space refers to the air pressure of the PAMs, joint space refers to the length of the PAMs, configuration space is determined by the discrete point quaternions, and task space refers to the coordinates of the robot endpoint. The mapping from joint space to task space is forward kinematics (f_{for}) and vice versa is inverse kinematics (f_{inv}), as shown in Fig. 3(c). The length of the PAMs is proportional to the the internal air pressure, but the air pressure is difficult to control precisely. When the inflation velocity is certain, the length of the PAMs is proportional to the inflation/deflation time. Therefore, we control the length of the PAMs by controlling the inflation/deflation time, which can be achieved by controlling the switching time of the solenoid valves, and the relationship between them is shown in Fig. 4.

Fig. 4 The relationship between the length of PAMs and the time of inflation/deflation.

Fig. 5 The relationship between the number of discrete elements and the calculate error and calculate time.

Different from traditional link robots, the model of continuum robot can usually obtain numerical solutions rather than analytical solutions. When solving continuum robot model, the difference format and optimization algorithm will inevitably introduce calculation errors, and the number of discrete elements usually has a greater impact on the calculation errors. We compared the calculation errors of different discrete elements and found that when the number of elements increases from 5 to 20, the calculation errors gradually decrease, and when it increases from 20 to 50, the calculation error is basically stable. At the same time, the calculation time is directly proportional to the number of elements. When the number of elements is 20, the calculation time is 5ms, as shown in Fig. 5. Therefore, 20 is chosen as the appropriate number of discrete elements.

3.2 Arriving at the target points

Continuum robots are often required to operate at certain specific locations. Based on the proposed model, experiments to reach the target points with a 50g load on the endpoint are performed first. Five target points in space are defined, and based on the calculations of the model, the robot is controlled to reach the target point, and the ratio of the Euclidean distance between the ideal and the actual position of the robot endpoint to the robot length is defined as the error.

Fig. 6 shows the comparison between theoretical and experimental results. The actual configuration of the robot is close to the theoretical. The average position error of the robot endpoint is 2.38%.

3.3 Complex trajectory following

In practical application scenarios, soft continuum robots often need to move along certain trajectories. Experiments to control the movement of the robot along a complex trajectory in space with a 50g load are carried out to verify the adaptability of the proposed model. A trajectory can be decomposed into a collection of discrete points, and only need to control the robot to move to these discrete points in sequence. A circular trajectory is defined, and it is discretized into 8 discrete nodes. The functional of the trajectory is $x^2 + y^2 + (z + 75)^2 = 30^2$. Based on the calculation results of the model, the robot is controlled to move to these discrete nodes in turn.

Fig. 7 shows the comparison of theoretical and experimental results. The robot can be well controlled to move along the preset trajectory, and the theoretical shape and actual shape can be well matched. The average position error of the end point of the robot is 2.37%.

Fig. 6. (a) Simulation results of robot configuration at different target points. (b) Comparison of simulation and experimental results of robot end positions. (c) Errors of the robot endpoint position. Note: Because the robot is free at position 3 without bending deformation, so the error is 0. Position 3 is not included in the calculation of the average error. (d) Experimental photos.

4 Conclusion

Based on the principle of virtual work and vector mechanics, a continuum robot modeling method is proposed, and the external payload and position constraints are considered. The finite difference method are used to discretize the equations, and the PSO and LM optimization algorithm is used to numerically solve the model.

Two sets of experiments are used to verify the model. The experimental results show that based on the proposed model, the configuration can be predicted well and the robot can be controlled to move following the desired trajectory. The average error of the endpoints of the robot does not exceed 2.38%. This work could provide a new perspective for the design, kinematic analysis and motion planning of th continuum robots.

Acknowledgements

The work was supported by National Natural Science Foundation of China (51975566, U1908214), National Key R&D Program of China (2018YFB1304600), and CAS Interdisciplinary Innovation Team (JCTD-2018-11).

Data Availability Statement

The datasets supporting the conclusions of this article are included within the article.

Fig. 7 (a) Simulation results of the configuration when the robot passes through different trajectory points. (b) Comparison of simulation and experimental results of robot endpoint positions. (c) Errors of the robot endpoint. (d) Experimental photos. The circular trajectory of the endpoint of the robot. Red arrows indicate points the robot is passing, dark blue arrows indicate points it has already passed, and light blue arrows indicate points it has not yet passed.

Declarations

Conflict of interest

The authors declare that they have no conflict of interest.

References

- [1] Rus, D., Tolley, M.T.: Design, fabrication and control of soft robots. *Nature*. 521(7553), 467-75 (2015)
- [2] Franco, E., Ayatullah, T., Sugiharto, A.: Nonlinear energy-based control of soft continuum pneumatic manipulators. *Nonlinear Dynamics*. 106(1), 229-253 (2021)
- [3] Shabana, A.A., Eldeeb, A.E.: Motion and shape control of soft robots and materials. *Nonlinear Dynamics*. 104(1), 165-189 (2021)

- [4] Renda, F., Giorelli, M., Calisti, M.: Dynamic model of a multibending soft robot arm driven by cables. *IEEE Transactions on Robotics*. 30(5), 1109-1122 (2014)
- [5] Xu, F., Wang, K., Au, K.W.S.: Underwater dynamic modeling for a cable-driven soft robot arm. *IEEE/ASME Transactions on Mechatronics*. 23(6), 7-25 (2018)
- [6] Marchese, A.D., Katzschmann, R.K., Rud, D.: A recipe for soft fluidic elastomer robots. *Soft Robotics*. 2(1), 7-25 (2015)
- [7] Tutcu, C., Baydere, B.A., Talas, S.K.: Quasi-static modeling of a novel growing soft-continuum robot. *International Journal of Robotics Research*. 40(1), 86-98 (2021)
- [8] Gu, G., Zou, J., Zhao, X.: Soft wall-climbing robots. *Science Robotics*. 3(25), 2874 (2018)
- [9] Gu, G., Gupta, U., Zhu, J.: Modeling of viscoelastic electromechanical behavior in a soft dielectric elastomer actuator. *IEEE Transactions on Robotics*. 33(5), 1263-1271 (2017)
- [10] Majidi, C.: Soft robotics: A perspective current trends and prospects for the future. *Soft Robotics*. 1(1), 5-11 (2013)
- [11] Webster III, R.J., Jones, B.A.: Design and kinematic modeling of constant curvature continuum robots: A review. *International Journal of Robotics Research*. 29(13), 1661-1683 (2010)
- [12] Andersson, S.B.: Discretization of a continuous curve. *IEEE Transactions on Robotics*. 24(2), 456-461 (2008)
- [13] Burgner-Kahrs, J., Rucker, D.C., Choset, H.: Continuum robots for medical applications: A survey. *IEEE Transactions on Robotics*. 31(6), 1261-1280 (2015)
- [14] Jones, B.A., Walker, I.D.: Practical kinematics for real-time implementation of continuum robots. *IEEE Transactions on Robotics*. 22(6), 1087-1099 (2006)
- [15] Freixedes, L.R., Gao, A., Liu, N.: Design optimization of a contact-aided continuum robot for endobronchial interventions based on anatomical constraints. *International Journal of Computer Assisted Radiology and Surgery*. 14(3), 1137-1146 (2019)
- [16] Webster, R.J., Jones, B.A.: Design and kinematic modeling of constant curvature continuum robots: A review. *International Journal of Robotics Research*. 29(13), 1661-1683 (2010)
- [17] Della Santina, C., Bicchi, A., Rus, D.: On an improved state parametrization for soft robots with piecewise constant curvature and its use in model based control. *IEEE Robotics and Automation Letters*. 5(2), 1001-1008 (2020)
- [18] Schiller, L., Seibel, A., Schlattmann, J.: A Lightweight Simulation Model for Soft Robot's Locomotion and its Application to Trajectory Optimization. *IEEE Robotics and Automation Letters*. 5(2), 1199-1206 (2020)
- [19] Hadi Sadati, S.M., Naghibi, S.E., Walker, I.D.: Control Space Reduction and Real-Time Accurate Modeling of Continuum Manipulators Using Ritz and Ritz–Galerkin Methods. *IEEE Robotics and Automation Letters*. 3(1), 1-7 (2017)
- [20] Singh, I., Amara, Y., Melingui, A.: Modeling of continuum manipulators using pythagorean hodograph curves. *Soft Robotics*. 5(4), 425-442 (2018)
- [21] Gonthina, P.S., Kapadia, A.D., Godage, I.S.: Modeling variable curvature parallel continuum robots using euler curves. 2019 International Conference on Robotics and Automation (ICRA). 1679-1685 (2019)
- [22] Godage, I.S., Wirz, R., Walker, I.D.: Accurate and efficient dynamics for variable-length continuum arms: A center of gravity approach. *Soft Robotics*. 2(3), 96-106 (2015)
- [23] Renda, F., Giorelli, M., Calisti, M.: Dynamic model of a multibending soft robot arm driven by cables. *IEEE Transactions on Robotics*. 30(5), 1-14 (2014)

- [24] Randa, F., Boyer, F., Dias, J.: Discrete Cosserat approach for multisection soft manipulator dynamics. *IEEE Transactions on Robotics*. 34(6), 1518-1533 (2018)
- [25] Bieze, T.M., Largilliere, F., Kruszewski, A.: Finite element method-based kinematics and closed-loop control of soft, continuum manipulators. *Soft Robotics*. 5(3), 348-364 (2018)
- [26] Black, B.C., Till, J., Rucker D.C.: Parallel continuum robots: modeling, analysis, and actuation-based force sensing. *IEEE Transactions on Robotics*. 34(1), 29-47 (2017)
- [27] Li, S., Vogt, D.M., Rus, D.: Fluid-driven origami-inspired artificial muscles. *Proceedings of the National Academy of Sciences of the United States of America*. 13132-13137 (2017)
- [28] Lee, J.G., Rodrigue, H.: Origami-Based Vacuum Pneumatic Artificial Muscles with Large Contraction Ratios. *Soft Robotics*. 6(1), 109-117 (2019)