

Predefined-Time Convergent Motion Control for Heterogeneous Continuum Robots

Ning Tan^{†*}, Peng Yu[†], and Kai Huang

School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China

[†] Equal Contribution.

* Corresponding Author. Email: tann5@mail.sysu.edu.cn

Abstract—As research into continuum robots flourishes, there are more and more types of continuum robots, which require researchers to tirelessly design robot-specific motion control algorithms. Besides, the convergence time of control systems for continuum robots has received very little attention. In this paper, we propose a novel predefined-time convergent zeroing dynamics (PTCZD) model, which ensures that the associated error-monitoring function converges to zero in predefined-time. Based on the PTCZD model, we design an inverse kinematics solver and a state estimator for continuum robots, thereby obtaining a generic predefined-time convergent control method for heterogeneous continuum robots for the first time. Simulations and experiments based on cable-driven continuum robots and concentric tube continuum robots are performed to verify the efficacy, robustness and adaptability of the proposed control method. In addition, comparative studies are carried out to demonstrate its advantages against existing control methods for continuum robots.

I. INTRODUCTION

Biologically inspired continuum robots have received considerable scholarly attention in recent years because of their dexterity, interaction safety and easy miniaturization. Unlike conventional rigid-link robots, this class of robots are usually made of soft or elastic materials, empowering them to work in unstructured or constraint space compliantly. As a result of these benefits, continuum robots are becoming new apparatuses covering a lot of applications, such as medical treatment [18], search and rescue operations in congested environments [26] and grasp of soft objects. Along with the prevalence of continuum robots, however, there is increasing concern over their motion control problem.

Several challenges have been faced by researchers on continuum robots. To begin with, as a result of their structural and material peculiarities, continuum robots usually deform irregularly due to their own and environmental influence. The irregular deformation undoubtedly brings great difficulties to the modeling and control of continuum robots. Besides, many heterogeneous continuum robots have been designed and fabricated in the past decades, including but not limited to cable-driven continuum robots [8], pneumatic continuum robots [10], concentric tube continuum robots and parallel continuum robots [2]. The diversity of continuum robots gives rise to the demand for the development of robot-specific control algorithms, which needs much effort. Furthermore, the convergence of control systems, which is interpreted as the

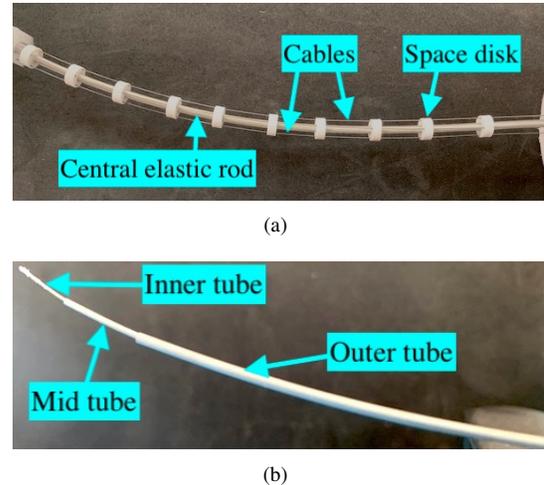


Fig. 1. Illustration of two heterogeneous continuum robots. (a) Cable-driven continuum robot (CDR). (b) Concentric tube continuum robot (CTR).

process of task-space error approaching a steady state, has always been a concern for the control of conventional rigid-link robots. Accordingly, how to ensure that the control system for continuum robots converges as expected is also a problem worth studying.

The motion control problem of continuum robots and conventional rigid-link robots mainly involves the convergence of position error (in point-to-point tasks) or trajectory tracking error (in tracking control tasks). Through years of unremitting efforts by researchers, different types of convergence have been achieved for the motion control of rigid-link robots, such as exponential convergence, super-exponential convergence, finite-time convergence, fixed-time convergence and predefined-time convergence [20]. In comparison, the convergence of control systems for continuum robots has received much less attention. Control systems with exponential and finite-time convergence have been proposed for the motion control of continuum robots [1, 24]. However, these studies have suffered from notable shortcomings. It has been proven that it takes infinitely long time for exponentially convergent controllers to drive the error function to zero [15], whereas the task duration is usually limited in practice. To make steady-state residual error arbitrarily small, related design parameters

must be set large enough or infinitely large, which is not feasible in practice because the design parameters may exceed available processing power [23]. As for the control system with finite-time convergence property, it only expresses the convergence time as a function of control gains and other factors, and it may not be easy to determine the parameters of control systems based on desired convergence time. However, the intuition is that it would be preferable to allow us to set the convergence time of control systems directly, rather than tediously tuning the control gains to achieve desired convergence. These issues have led to a better convergence property, namely predefined-time convergence, which requires that the parameters of control systems should be explicitly designed with user-defined time constraint, thereby ensuring that end-effector tasks are fulfilled by continuum robots in predefined time. However, to the best of the authors' knowledge, predefined-time convergent motion control for continuum robots has not yet been studied in the available literature.

This paper seeks to develop a generic motion control method for heterogeneous continuum robots with predefined-time convergence. Specifically, the present study makes the following noteworthy contributions to the research on continuum robots:

- 1) A novel predefined-time convergent zeroing dynamics (PTCZD) model is designed. Theoretically, we rigorously prove that the proposed PTCZD model is capable of driving the indefinite error function associated with the model to converge to zero in predefined time.
- 2) By utilizing two PTCZD models, we design an inverse kinematics solver and a state estimator for continuum robots, respectively, thereby achieving the predefined-time convergent motion control of continuum robots for the first time.
- 3) A cable-driven continuum robot (CDR) and a concentric tube continuum robot (CTR), as shown in Fig. 1, are adopted to verify the effectiveness and adaptability of proposed control method for heterogeneous continuum robots.

II. RELATED WORK

In recent years, there has been a growing number of publications focusing on the motion control of continuum robots. Typically, the piecewise constant curvature (PCC) model [27], which approximates continuum robots as multiple circular arcs of constant curvature, has been widely applied to the motion control of continuum robots [3, 7, 11]. To avoid the issues of standard parametrization for the PCC model, such as discontinuities and singularities, Della Santina et al. [4] proposed an improved state parametrization for soft continuum robots. Due to the effect of load or environment, continuum robots are often unable to maintain a constant curvature shape, which leads to nonconstant or variable curvature models [9, 17, 21].

Considering the modeling complexity of continuum robots, control systems that are independent of analytic models of continuum robots have been devised [13, 19, 28, 32]. In particular, Yip and Camarillo [31] firstly developed a model-less control method for a tendon-driven continuum robot,

which utilizes the quadratic programming method to solve the inverse kinematics problem and to estimate the Jacobian matrix of the robot respectively. There are also learning-based motion control methods for continuum robots. For instance, Lee et al. [12] reported a nonparametric online learning control method for effective endoscopic navigation. Most of the above mentioned motion control methods were only applied to specific types of continuum robots. In comparison, we make the breakthrough towards the development of a generic motion control method for heterogeneous continuum robots in this paper.

Despite the preliminary advance in the motion control of continuum robots, the convergence time of control systems has received very little attention. Tan et al. [25] proposed a neurodynamic approach for the pose tracking control of continuum robots, which can achieve exponential convergence. Ayala-Carrillo et al. [1] presented a cascade control method for robust tracking of pneumatic continuum robots with finite-time convergence. Predefined-time attitude stabilization control has been investigated for continuum robots [5]. However, as far as we know, predefined-time convergent motion control (e.g., trajectory tracking control) has never been reported in the field of soft continuum robots. The present study fills the gap in the literature by developing a motion control system with predefined-time convergence for continuum robots.

Zeroing dynamics models have been proposed and widely studied for the solution of a variety of time-variant problems since the early years of this century [33, 34]. By utilizing different activation functions, zeroing dynamics models exhibit different types of convergence. Typically, when using a linear activation function, the zeroing dynamics model ensures that the associated error-monitoring function converges to zero exponentially. Many nonlinear activation functions have been designed for zeroing dynamics models to improve the convergence [30]. Xiao et al. [29] designed a finite-time zeroing dynamics model by devising a nonlinear activation function. Likewise, Lv et al. [16] proposed a zeroing dynamics model activated by weighted sign-bi-power function to achieve finite-time convergence. Furthermore, Li [14] designed several nonlinear activation functions to obtain predefined-time convergent zeroing dynamics models. However, these zeroing dynamics models were only applied to time-variant matrix inversion, time-variant equation solving and the motion control of rigid-link robots. In this paper, we propose a novel PTCZD model and firstly achieve the predefined-time convergent motion control of continuum robots based on the model.

III. METHOD

In this section, we first present the proposed PTCZD model and prove its predefined-time convergence theoretically. Subsequently, we introduce how to design a state estimator and an inverse kinematics solver for continuum robots with the PTCZD model.

A. PTCZD Model

In the design procedure of zeroing dynamics, an indefinite error-monitoring function $\xi(t) \in \mathbb{R}$ is first defined at the time t according to the specific objective. To drive the error-monitoring function to converge to zero, the design formula of zeroing dynamics is usually given as follows

$$\dot{\xi}(t) = -\nu \mathcal{H}(\xi(t)) \quad (1)$$

where $\dot{\xi}(t)$ denotes the first time derivative of $\xi(t)$, ν is a positive constant dominating the convergence speed of the model, $\mathcal{H}(\cdot)$ could be a linear or nonlinear activation function. To achieve predefined-time convergence, we design the following PTCZD model

$$\begin{cases} \dot{\xi}(t) = -\nu \mathcal{H}(\xi(t)) \\ \mathcal{H}(z) = \frac{\exp(|z|^{2-2p} + |z|^{2+2p}) |z|^{2p-1} \text{sign}(z)}{2(1-p + (1+p)|z|^{4p})} \\ \nu = \frac{1 - \exp(-|\xi(0)|^{2-2p} - |\xi(0)|^{2+2p})}{t_c} \end{cases} \quad (2)$$

where $p \in (0.5, 1)$ is a parameter of the model, $\text{sign}(\cdot)$ is the sign function, $\xi(0)$ denotes the initial value of the error-monitoring function and t_c is the desired convergence time defined by users. The choice of p can ensure that the activation function $\mathcal{H}(z)$ is monotonically increasing provided that the input z is in a reasonable range. Finally, by expanding the zeroing dynamics model with user-defined error-monitoring function $\xi(t)$, we can obtain a differential-equation-based dynamic system to solve specific problems. Regarding the proposed PTCZD model, the following theorem is provided to ensure its predefined-time convergence.

Theorem 1: Starting from initial error $\xi(0) \neq 0$, the PTCZD model (2) ensures that the error-monitoring function $\xi(t)$ converges to zero in predefined time t_c .

Proof: To begin with, the following function is defined

$$L(t) = \xi^2(t) \geq 0. \quad (3)$$

By deriving the above function with respect to time t , the following result is attained

$$\dot{L}(t) = 2\xi(t)\dot{\xi}(t). \quad (4)$$

By utilizing (2) and replacing the variable z in $\mathcal{H}(z)$ with ξ , we can further obtain

$$\begin{aligned} \dot{L}(t) &= -2\nu\xi(t)\mathcal{H}(\xi(t)) \\ &= \frac{-\nu|\xi(t)| \exp(|\xi(t)|^{2-2p} + |\xi(t)|^{2+2p}) |\xi(t)|^{2p-1}}{1-p + (1+p)|\xi(t)|^{4p}} \\ &= \frac{-\nu \exp(|\xi(t)|^{2-2p} + |\xi(t)|^{2+2p})}{(1-p)|\xi(t)|^{-2p} + (1+p)|\xi(t)|^{2p}}. \end{aligned} \quad (5)$$

The above equation can be rewritten as

$$\dot{L}(t) = \frac{dL}{dt} = \frac{-\nu \exp(L^{1-p}(t) + L^{1+p}(t))}{(1-p)L^{-p}(t) + (1+p)L^p(t)}, \quad (6)$$

which yields

$$-\frac{(1-p)L^{-p}(t) + (1+p)L^p(t)}{\exp(L^{1-p}(t) + L^{1+p}(t))} dL = \nu dt. \quad (7)$$

Assume that the error-monitoring function $\xi(t)$ converges to zero at the time t_s , which means the function $L(t)$ converges to zeros at t_s . By integrating both sides of (7) from $t = 0$ to t_s , the following result is obtained

$$\begin{aligned} \int_{L(0)}^0 -\frac{(1-p)L^{-p}(t) + (1+p)L^p(t)}{\exp(L^{1-p}(t) + L^{1+p}(t))} dL &= \int_0^{t_s} \nu dt, \\ \exp(-L^{1-p}(t) - L^{1+p}(t)) \Big|_{L(0)}^0 &= \nu t \Big|_0^{t_s}, \\ 1 - \exp(-L^{1-p}(0) - L^{1+p}(0)) &= \nu t_s. \end{aligned} \quad (8)$$

Considering the definition of ν in (2), we have

$$\begin{aligned} t_s &= \frac{1 - \exp(-L^{1-p}(0) - L^{1+p}(0))}{\nu} \\ &= \frac{1 - \exp(-L^{1-p}(0) - L^{1+p}(0))}{1 - \exp(-|\xi(0)|^{2-2p} - |\xi(0)|^{2+2p})} t_c \\ &= t_c. \end{aligned} \quad (9)$$

The above result indicates that the convergence time of the error-monitoring function $\xi(t)$ is the predefined time t_c . The proof is thus completed. \blacksquare

B. State Estimator

For model-based control methods, there are two major problems: 1) The modeling of continuum robots is of great difficulty and complexity. 2) It is tedious to design specific models and control algorithms for heterogeneous continuum robots. Therefore, we are aimed at developing a generic model-free control system, which is independent of the robot model, to achieve the motion control of heterogeneous continuum robots.

Typically, the forward kinematics of a continuum robot can be simply described by

$$\mathbf{r}_a(t) = \mathcal{F}(\boldsymbol{\phi}(t)) \quad (10)$$

where $\mathbf{r}_a(t) \in \mathbb{R}^m$ denotes the position of robot end-effector in the m -dimension task space, $\boldsymbol{\phi}(t) \in \mathbb{R}^n$ denotes the state of the n actuators of the continuum robot and $\mathcal{F}(\cdot)$ denotes a nonlinear kinematic mapping function. Accordingly, the differential kinematics of the continuum robot is as follows

$$\dot{\mathbf{r}}_a(t) = \mathbf{J}(t)\dot{\boldsymbol{\phi}}(t) \quad (11)$$

where $\dot{\mathbf{r}}_a(t)$ denotes the first time derivative of $\mathbf{r}_a(t)$ and $\mathbf{J}(t) = \partial \mathcal{F}(\boldsymbol{\phi}(t))/\partial \boldsymbol{\phi}(t) \in \mathbb{R}^{m \times n}$ is the Jacobian matrix of the continuum robot. Without building a model for the robot, the mapping function and the Jacobian matrix can not be analytically calculated. To solve this problem, we design a Jacobian-based state estimator for the continuum robot.

First, inspired by (11), the following vector-valued error-monitoring function is defined

$$\boldsymbol{\epsilon}(t) = \dot{\mathbf{r}}_a(t) - \mathbf{J}_A(t)\dot{\boldsymbol{\phi}}(t) \in \mathbb{R}^m \quad (12)$$

where $\mathbf{J}_A(t)$ is the unknown Jacobian matrix to be estimated for the continuum robot. Then, modifying the proposed PTCZD model to vector-valued version and replacing $\xi(t)$ with $\epsilon(t)$ lead to

$$\dot{\epsilon}(t) = -\nu_s \mathcal{H}(\epsilon(t)) \quad (13)$$

where $\mathcal{H}(\cdot) \in \mathbb{R}^m$ is an activation function array with each element being $\mathcal{H}(z)$ and ν_s is a constant designed according to (2). Next, by expanding (13) with (12), the following dynamic equation is derived

$$\ddot{\mathbf{r}}_a(t) - \dot{\mathbf{J}}_A(t)\dot{\phi}(t) - \mathbf{J}_A(t)\ddot{\phi}(t) = -\nu_s \mathcal{H}(\dot{\mathbf{r}}_a(t) - \mathbf{J}_A(t)\dot{\phi}(t)) \quad (14)$$

where $\ddot{\mathbf{r}}_a(t)$ denotes the second time derivative of $\mathbf{r}_a(t)$. Finally, the evolution of the estimated Jacobian matrix follows the dynamic equation below

$$\dot{\mathbf{J}}_A(t) = (\ddot{\mathbf{r}}_a(t) - \mathbf{J}_A(t)\ddot{\phi}(t) + \nu_s \mathcal{H}(\dot{\mathbf{r}}_a(t) - \mathbf{J}_A(t)\dot{\phi}(t)))\dot{\phi}^\dagger(t) \quad (15)$$

where $\dot{\phi}^\dagger(t)$ denotes the pseudo-inverse of $\dot{\phi}(t)$.

C. Predefined-Time Convergent Inverse Kinematics Solver

Similarly, to achieve the predefined-time convergent motion control of the continuum robot, the following vector-valued error-monitoring function is defined

$$\mathbf{e}(t) = \mathbf{r}_d(t) - \mathbf{r}_a(t) \in \mathbb{R}^m. \quad (16)$$

In a tracking control task, $\mathbf{r}_d(t)$ denotes the reference path that needs to be tracked by robot end-effector. In a point-to-point task, we can simply set $\mathbf{r}_d(t)$ as the reference point, which implies the point-to-point task is a special case of the tracking control task. Our objective is to drive the error-monitoring function $\mathbf{e}(t)$ to zero in predefined time. By adopting the PTCZD model (2) again, we have

$$\dot{\mathbf{e}}(t) = -\nu_c \mathcal{H}(\mathbf{e}(t)) \quad (17)$$

where ν_c is a constant designed according to (2). Then, expanding (17) with (16) yields

$$\dot{\mathbf{r}}_d(t) - \mathbf{J}(t)\dot{\phi}(t) = -\nu_c \mathcal{H}(\mathbf{r}_d(t) - \mathbf{r}_a(t)). \quad (18)$$

Therefore, the actuation signal can be determined by the following dynamic equation

$$\dot{\phi}(t) = \mathbf{J}^\dagger(t) (\dot{\mathbf{r}}_d(t) + \nu_c \mathcal{H}(\mathbf{r}_d(t) - \mathbf{r}_a(t))). \quad (19)$$

However, the analytic Jacobian $\mathbf{J}(t)$ can not be directly used because the kinematic model of the continuum robot is unknown. By replacing the analytic Jacobian $\mathbf{J}(t)$ with the estimated Jacobian $\mathbf{J}_A(t)$, the adaptation of the actuation signal follows the dynamic equation below

$$\dot{\phi}(t) = \mathbf{J}_A^\dagger(t) (\dot{\mathbf{r}}_d(t) + \nu_c \mathcal{H}(\mathbf{r}_d(t) - \mathbf{r}_a(t))). \quad (20)$$

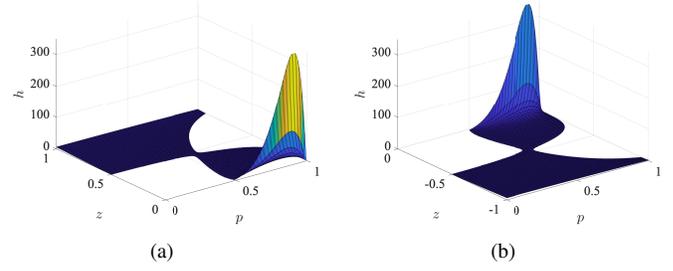


Fig. 2. Graphic analysis of $h(z, p)$. (a) $z \in (0, 1)$. (b) $z \in (-1, 0)$.

D. Control System for Heterogeneous Continuum Robots

Finally, a control system based on the state estimator (15) and the inverse kinematics solver (20) is obtained for the model-free motion control of continuum robots. There are several facts worth pointing out regarding the proposed control system. First, taking advantage of the proposed PTCZD model, the proposed control system can drive the tracking error (in tracking tasks) or position error (in point-to-point tasks) to zero in predefined time. Second, it is not difficult to find that the proposed control method is independent of the analytic model and structure of continuum robots, which means it is applicable to heterogeneous continuum robots. Third, the selection of the convergence speed parameter in the PTCZD model is a key step to ensure the convergence of the error-monitoring function. Generally the convergence speed ν is set according to (2). Specifically, in the inverse kinematics solver (20), the convergence speed is set as

$$\nu_c = \frac{1 - \exp(-|e_{\max}(0)|^{2-2p} - |e_{\max}(0)|^{2+2p})}{t_c} \quad (21)$$

where $e_{\max}(0)$ denotes the element in $\mathbf{e}(0)$ with the largest absolute value. In the state estimator (15), the convergence speed parameter is set to satisfy $\nu_s \geq 1/t_c$ for simplicity. Besides, it can be seen from the dynamic equations (15) and (20) that the end-effector position, velocity and acceleration are required to solve the equations. Unfortunately, the velocity and acceleration are not easy to be directly measured sometimes, which could be a limitation of the proposed method. In this case, one can utilize differentiators or filters to estimate the velocity and acceleration [6, 22]. Moreover, the dynamic equations (15) and (20) could be solved by numerical tools (e.g., ode45 in MATLAB) as long as the initial values are provided. The initial value of the actuation signal could be set to any reasonable value, whereas the initial value of the estimated Jacobian should follow specific procedures. Generally, one can increase the value of the i -th ($i = 1, 2, \dots, n$) actuation signal by a sufficiently small amount $\Delta\phi_i \in \mathbb{R}^+$ and measure the end-effector displacement $\Delta\mathbf{r}_i \in \mathbb{R}^m$ of the continuum robot. By repeating the procedure for each actuator, one can initialize the estimated Jacobian as $\mathbf{J}_A(0) = [\Delta\mathbf{r}_1/\Delta\phi_1, \Delta\mathbf{r}_2/\Delta\phi_2, \dots, \Delta\mathbf{r}_n/\Delta\phi_n]$.

E. Stability Analysis

In this part, we analyze the stability of the proposed control system. Firstly, based on the error-monitoring functions, the following positive semi-definite Lyapunov candidate is defined

$$V = \frac{1}{2} (\mathbf{e}^T(t)\mathbf{\Lambda}\mathbf{e}(t) + \boldsymbol{\epsilon}^T(t)\mathbf{\Lambda}\boldsymbol{\epsilon}(t)) \geq 0 \quad (22)$$

where $(\cdot)^T$ is the transpose operation and $\mathbf{\Lambda} \in \mathbb{R}^{n \times n}$ is an identity matrix. Considering (13) and (17), the first time-derivative of the candidate is as follows

$$\begin{aligned} \dot{V} &= \mathbf{e}^T(t)\mathbf{\Lambda}\dot{\mathbf{e}}(t) + \boldsymbol{\epsilon}^T(t)\mathbf{\Lambda}\dot{\boldsymbol{\epsilon}}(t) \\ &= -\nu_c \mathbf{e}^T(t)\mathcal{H}(\mathbf{e}(t)) - \nu_s \boldsymbol{\epsilon}^T(t)\mathcal{H}(\boldsymbol{\epsilon}(t)) \\ &= -\nu_c \sum_{i=1}^n e_i(t)\mathcal{H}(e_i(t)) - \nu_s \sum_{i=1}^n \epsilon_i(t)\mathcal{H}(\epsilon_i(t)) \end{aligned} \quad (23)$$

where the subscript $(\cdot)_i$ indicates the i -th element of the vector. Next, we analyze the property of the activation function $\mathcal{H}(z)$ in (2). According to the design philosophy of zeroing dynamics, the activation function should be monotonically increasing. When $z \neq 0$, we have

$$\frac{d\mathcal{H}(z)}{dz} = \frac{2 \exp(|z|^{2-2p} + |z|^{2+2p}) h(z, p)}{4((1-p)|z|^{1-2p} + (1+p)|z|^{1+2p})^2} \quad (24)$$

where

$$\begin{aligned} h(z, p) &= 2((1-p)|z|^{1-2p} + (1+p)|z|^{1+2p})^2 \\ &\quad - (1-p)(1-2p)|z|^{-2p} - (1+p)(1+2p)|z|^{2p}. \end{aligned} \quad (25)$$

Next, our objective is to find the parameter value of p to ensure $h(z, p)$ is positive. It is worth noting that $h(z, p)$ is non-convex, which implies there may be more than one parameter range that meets the requirement and we only need to determine an appropriate parameter range. Considering that the workspace of continuum robots is usually limited, it is reasonable to assume that the initial value of the error-monitoring function stays within a limited range $(-\delta, \delta)$, e.g., $\delta = 0.5$. For those continuum robots with larger workspace, which may cause that the initial absolute value of the error-monitoring function exceeds δ , we can decompose the task into several sub-tasks, where the initial absolute value of the error-monitoring function of each sub-task is less than δ . Then, the graphical illustration of $h(z, p)$ is depicted in Fig. 2. Provided that the value of z satisfies the above assumption, it can be seen that the value of $h(z, p)$ could be positive when p is properly selected from $(0.5, 1)$. It is worth pointing out that $h(z, p) \rightarrow +\infty$ when $z \rightarrow 0$ and $p \in (0.5, 1)$. After determining an appropriate value for p , we have $h(z, p) > 0$, which means

$$\frac{d\mathcal{H}(z)}{dz} > 0 \quad (26)$$

where $z \in (-\delta, 0) \cup (0, \delta)$. Therefore, we can further get the following result

$$\mathcal{H}(z) \begin{cases} > 0, & 0 < z < \delta \\ = 0, & z = 0 \\ < 0, & -\delta < z < 0 \end{cases} \quad (27)$$

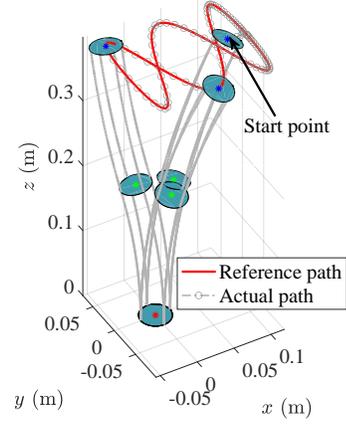


Fig. 3. Some configurations of the CDR tracking the reference path in simulations.

Considering (26) and (27), it is not difficult to find that $\mathcal{H}(z)$ is monotonically increasing when $z \in (-\delta, \delta)$, which implies $z\mathcal{H}(z) \geq 0$. According to the definition of ν_c and ν_s , both of them are positive. Then, we can get $\dot{V} \leq 0$, and $\dot{V} = 0$ if and only if both the error-monitoring functions $\mathbf{e}(t)$ and $\boldsymbol{\epsilon}(t)$ are zero. Finally, when $\mathbf{e}(t) \neq \mathbf{0}$ and $\boldsymbol{\epsilon}(t) \neq \mathbf{0}$, we have $V > 0$ and $\dot{V} < 0$. According to the Lyapunov's second method, we can conclude that the proposed control system is asymptotic stable.

IV. SIMULATIONS AND EXPERIMENTS

In this part, simulations and experiments based on heterogeneous continuum robots are performed to validate the proposed predefined-time convergent control method.

A. Simulations

1) *Settings*: In simulations, a cable-driven continuum robot (CDR) and a concentric tube continuum robot (CTR) are adopted to validate the proposed control system. Specifically, the CDR is controlled by the proposed method to track a Lissajous path and the CTR is controlled to track a circular path. Apart from the tracking tasks, a point-to-point task is also adopted for the CTR to verify the effectiveness of the proposed method. We assume that the reference paths and points are within the workspace of the robots. The task duration of the tracking tasks is set as 10 s while the predefined convergence time is set to $t_c = 1$ s. For the point-to-point task, the predefined convergence time is set to $t_c = 5$ s. In addition, the parameter of the PTCZD model is set as $p = 0.9$ throughout this paper.

2) *Cable-Driven Continuum Robot*: Some configurations of the CDR for tracking the Lissajous path are illustrated in Fig. 3. The CDR is composed of two connected segments. We can find that the actual path of the robot end-effector is almost the same as the reference path. The profiles of tracking error and control signal of the CDR are depicted in Fig. 4. It can be seen that the tracking error of the continuum robot almost converges to zero before $t = 1$ s, which satisfies the

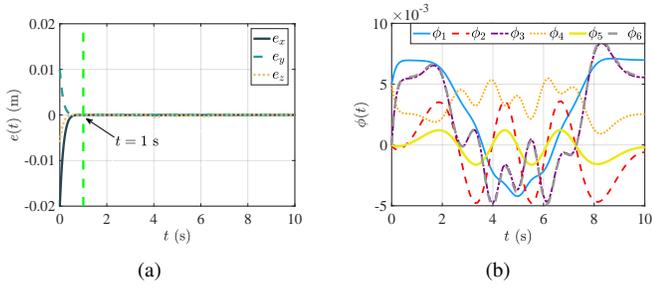


Fig. 4. Simulation results of the CDR for tracking the reference path. (a) Profiles of tracking error. (b) Profiles of control signal.

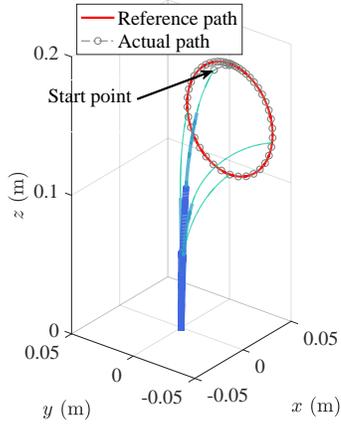


Fig. 5. Some configurations of the CTR tracking the reference path in simulations.

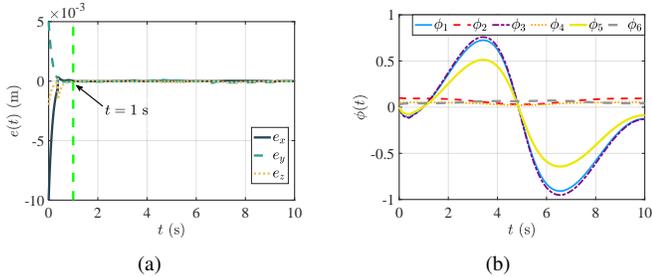


Fig. 6. Simulation results of the CTR for tracking the reference path. (a) Profiles of tracking error. (b) Profiles of control signal.

requirement ($t_c = 1$ s). The steady state tracking error of the CDR is less than 1×10^{-3} m during the tracking process.

3) *Concentric Tube Continuum Robot*: Some configurations of the CTR when tracking the circular path are illustrated in Fig. 5. The CTR consists of inner tube, mid tube and outer tube. It can be found that the actual path of the robot end-effector is almost the same as the reference path. The profiles of related tracking error and control signal of the CTR are illustrated in Fig 6. We can see that the tracking error of the CTR almost converges to zero before $t = 1$ s, which also satisfies the requirement ($t_c = 1$ s). Besides, a point-to-point task is also finished by the CTR with the aid of the proposed method. As shown in Fig. 7, the robot end-effector moves from the start point to the reference point as expected. The profiles

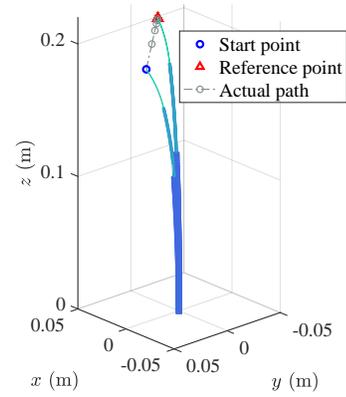


Fig. 7. Some configurations of the CTR finishing the point-to-point task in simulations.

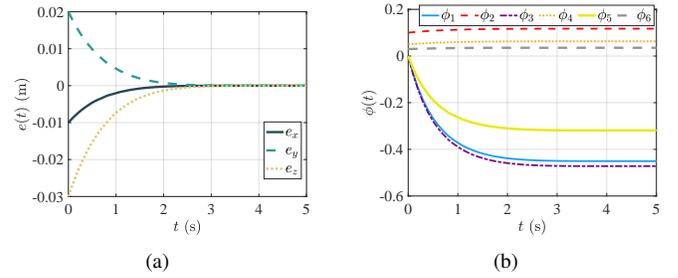


Fig. 8. Simulation results of the CTR for the point-to-point task. (a) Profiles of position error. (b) Profiles of control signal.

TABLE I
QUANTITATIVE COMPARISON OF STEADY-STATE TRACKING ERROR WITH EXISTING TRACKING CONTROLLERS.

Method	Noise-free		Noise-polluted	
	e_{RMSE} (m)	e_{max} (m)	e_{RMSE} (m)	e_{max} (m)
AKF	1.2×10^{-3}	2.4×10^{-3}	4.8×10^{-3}	4.9×10^{-3}
OZND	1.5×10^{-3}	4.4×10^{-3}	2.3×10^{-3}	5.0×10^{-3}
Ours	1.0×10^{-4}	5.1×10^{-4}	5.0×10^{-4}	8.5×10^{-4}

of position error and control signal of the CTR are illustrated in Fig 8. One can find that the point-to-point task is fulfilled before $t = 5$ s, which satisfies the requirement ($t_c = 5$ s).

4) *Comparison*: For the sake of revealing the merits of the proposed method, two existing model-free methods are adopted as a comparison in this part. The first method is the adaptive-Kalman-filter-based (AKF) method proposed in [13], which resorts to a Kalman filter to estimate the state of the continuum robot and control the robot with a Jacobian pseudo-inverse method. The second method is the zeroing neurodynamic approach proposed in [25], which utilizes the original zeroing neurodynamic model (OZND) to achieve the tracking control of continuum robots. All the three methods are implemented to control the CDR to finish the same tracking task. As depicted in Fig. 9, these methods are compared from two perspectives, i.e., tracking error and Jacobian estimation. It can be seen from Fig. 9(a) and Fig. 9(b) that the tracking

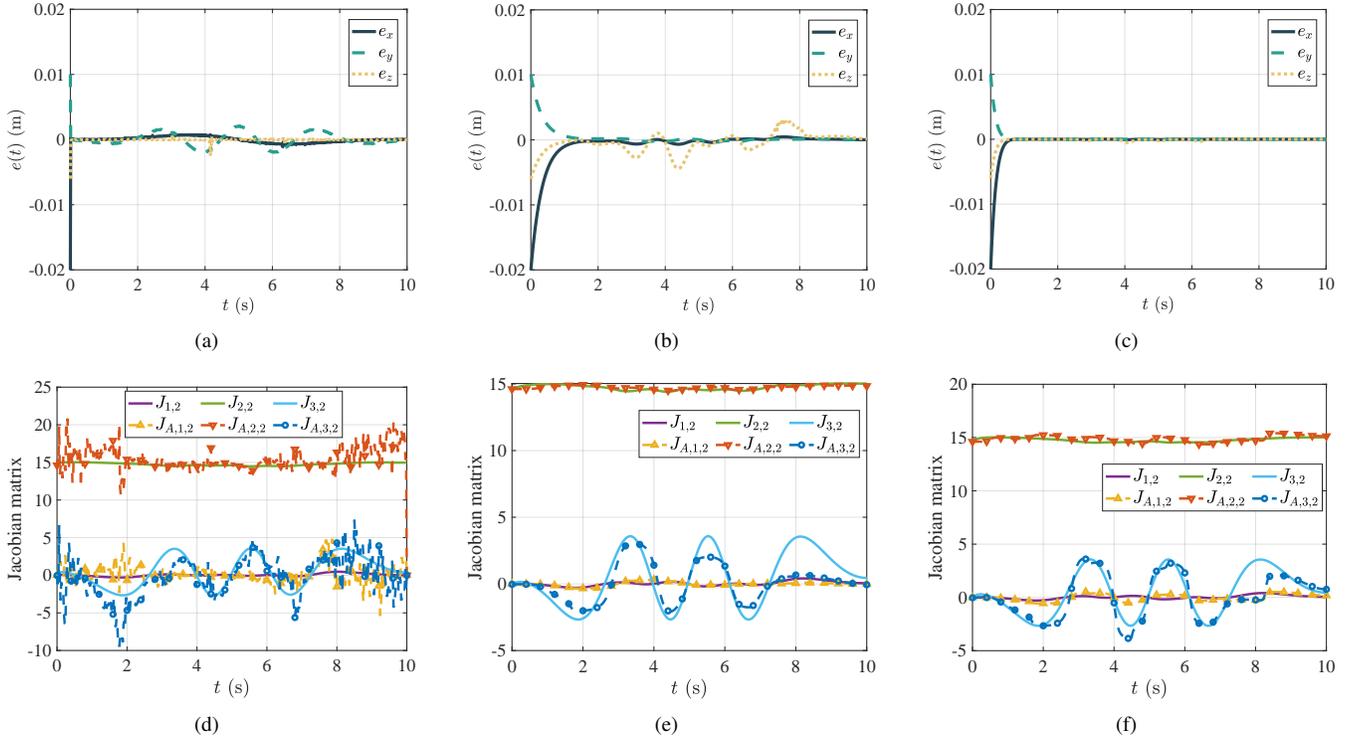


Fig. 9. Simulation results of comparative studies in the tracking task of the CDR. (a) Tracking error of the AFK method. (b) Tracking error of the OZND method. (c) Tracking error of the proposed control method. (d) Profiles of Jacobian matrices in the AFK method. (e) Profiles of Jacobian matrices in the OZND method. (f) Profiles of Jacobian matrices in the proposed method.

error of the AFK method converges to zero rapidly and the tracking error of the OZND method converges to zero exponentially. However, there are noticeable residual steady state errors when adopting these two methods. In comparison, the tracking error generated by the proposed method converges to zero in predefined time and the residual steady state error is almost zero as illustrated in Fig 9(c). For quantitative comparison, we summarize the root mean squared tracking error e_{RMSE} and maximum error e_{max} of these three methods at the steady state in Table I. Especially, we also compare the performance of these methods in the presence of time-varying noise. The quantitative results indicate that the errors achieved by our method are one order of magnitude smaller than those achieved by existing two methods in both the noise-free and the noise-polluted cases, which reveals the accuracy of our method. The comparison between the actual Jacobian matrices and the estimated Jacobian matrices by different methods is illustrated in Fig. 9(d)-9(f). For clarity and brevity, we only present the changes in some of the elements of the Jacobian matrices. We can find that the Jacobian matrix estimated by the AFK method oscillates around the actual Jacobian matrix. The OZND method has much better performance than the AFK method in terms of Jacobian estimation while the performance of the proposed method is slightly better than the OZND method. Moreover, the AFK method and the OZND were only validated on a single type of continuum robot and the OZND method was only validated in simulations. In comparison, our

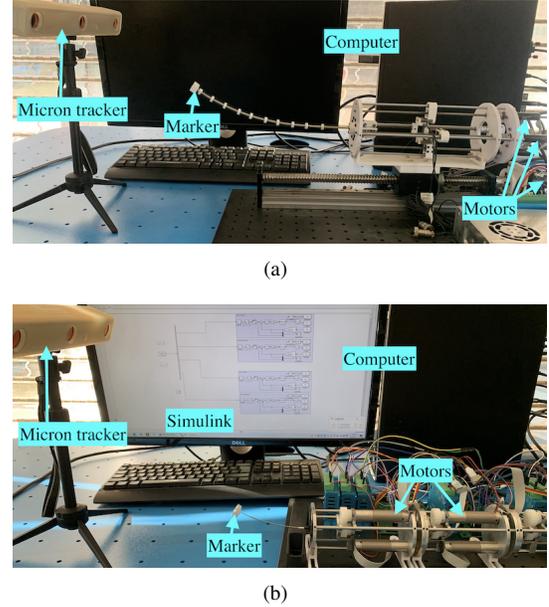


Fig. 10. Overview of the experiment platforms. (a) Cable-driven continuum robot (CDR). (b) Concentric tube continuum robot (CTR).

control system is proposed for and validated on heterogeneous continuum robots.

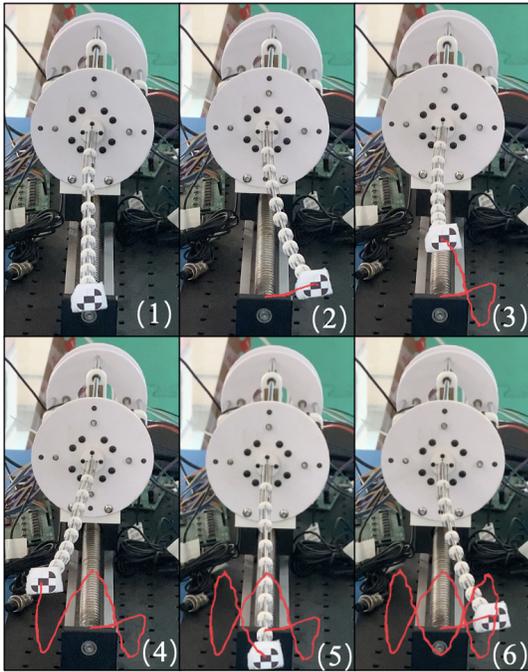


Fig. 11. Experiment snapshots of the CDR in the tracking task.

B. Experiments

1) *Settings*: Our experiment platforms are illustrated in Fig. 10, including the robots, the computer and the Micron tracker. The Micron tracker is utilized to measure the position of the marker attached to robot end-effector. The end-effector velocity and acceleration are estimated by low-pass filters. The robots are driven by motors, which are directly controlled by proportional-integral-differential (PID) controllers. The proposed control system is implemented via Simulink, which sends actuation signal to the PID controllers, thereby actuating the motors. The cable-driven continuum robot (CDR) is controlled by the proposed method to track a Lissajous path and the concentric tube continuum robot (CTR) is controlled to finish a point-to-point task. The task duration of the tracking task is set as 60 s while the predefined convergence time is set to $t_c = 3$ s. For the point-to-point task, the predefined convergence time is set to $t_c = 5$ s.

2) *Cable-Driven Continuum Robot*: Experiment snapshots of the CDR tracking the Lissajous path are shown in Fig. 11, where the actual path of the end-effector is highlighted with red. The specific experiment results of the tracking task are illustrated in Fig. 14. The figures show that the tracking error of the CDR almost converges to zero before $t = 3$ s, which satisfies the requirement ($t_c = 3$ s). Then, the end-effector moves along the reference path with the steady state error less than 2×10^{-3} m.

3) *Concentric Tube Continuum Robot*: The initial configuration and final configuration of the CTR in the point-to-point task are depicted in Fig. 12. The specific experiment results of the task are illustrated in Fig. 15. It can be found that the point-to-point task is successfully achieved by the CTR. The

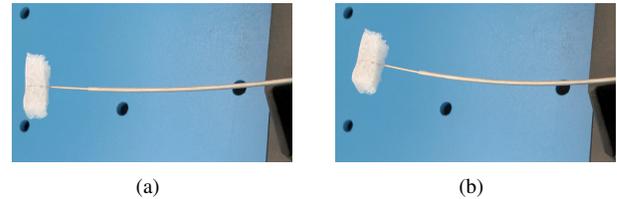


Fig. 12. Snapshots of the CTR in the point-to-point task. (a) Initial configuration. (b) Final configuration.

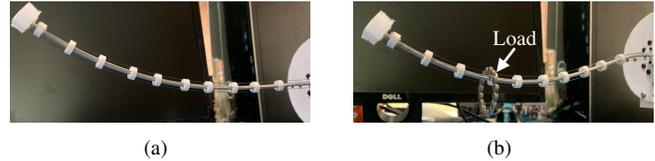


Fig. 13. Configurations of CDR. (a) Without loads. (b) With a load.

robot end-effector moves from the start point to the reference point as expected. More importantly, one can find that the position error converges to zero at $t = 5$ s, which satisfies the requirement ($t_c = 5$ s).

4) *Robustness Test*: In this part, we test the robustness of the proposed control system when imposing loads and disturbance to the CDR. The CDR is required to track a circular path, with predefined convergence time being $t_c = 5$ s and the control frequency being 10 Hz. First, we consider a load along the robot. As depicted in Fig. 13, the load is heavy enough to cause changes to the configuration of the robot. The robot is required to finish the tracking task with the load attached to it. The related experiment results is shown in Fig. 16. According to Fig. 16(b) and Fig. 16(c), we can find that the end-effector of the robot can still converge to the desired path in predefined time, which reveals the robustness of the proposed controller in the presence of a load along the robot. Then, we consider to impose an external disturbance to the robot when it is performing the tracking task. As illustrated in Fig. 17, we use a bottle to push the body of CDR when $t \in (25, 30)$ s, which leads to the deviation between the desired path and the robot as shown in Fig. 17(b). Nevertheless, Fig. 17(c) illustrates that the tracking errors still converge to zero after the disturbance, which further reveals the robustness of the proposed controller.

V. CONCLUSION

In this paper, a novel PTCZD model has been proposed to ensure the predefined-time convergence of the associated error-monitoring function. Based on the PTCZD model, a state estimator and an inverse kinematics solver have been designed for the motion control of heterogeneous continuum robots. Finally, extensive simulations and experiments have been performed on cable-driven continuum robots and concentric tube continuum robots to reveal the efficacy and merits of the proposed control system. Based on the results, we have the following findings: 1) The proposed control system is applicable to heterogeneous continuum robots, including CDRs and CTRs. 2) The proposed

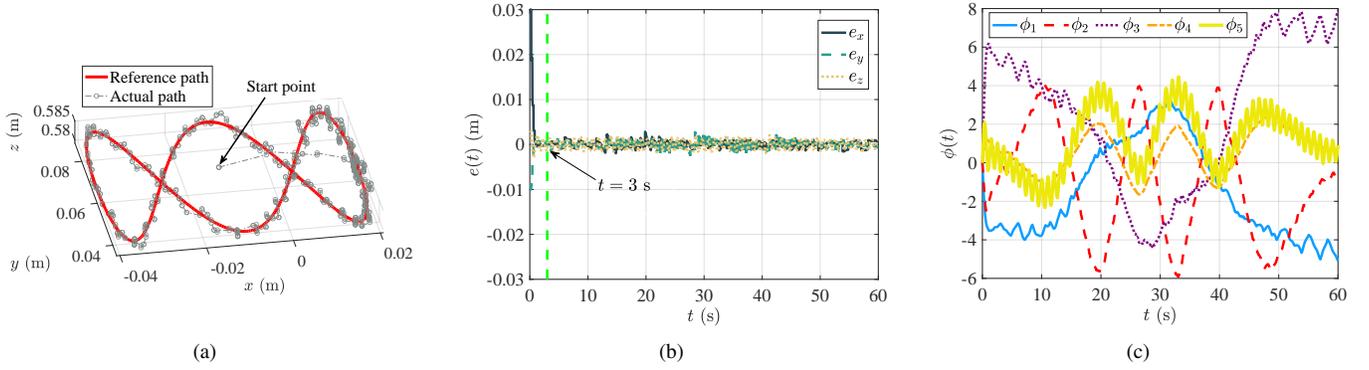


Fig. 14. Experiment results of the CDR for tracking the reference path. (a) Profiles of the reference path and the actual path. (b) Profiles of tracking error. (c) Profiles of control signal.

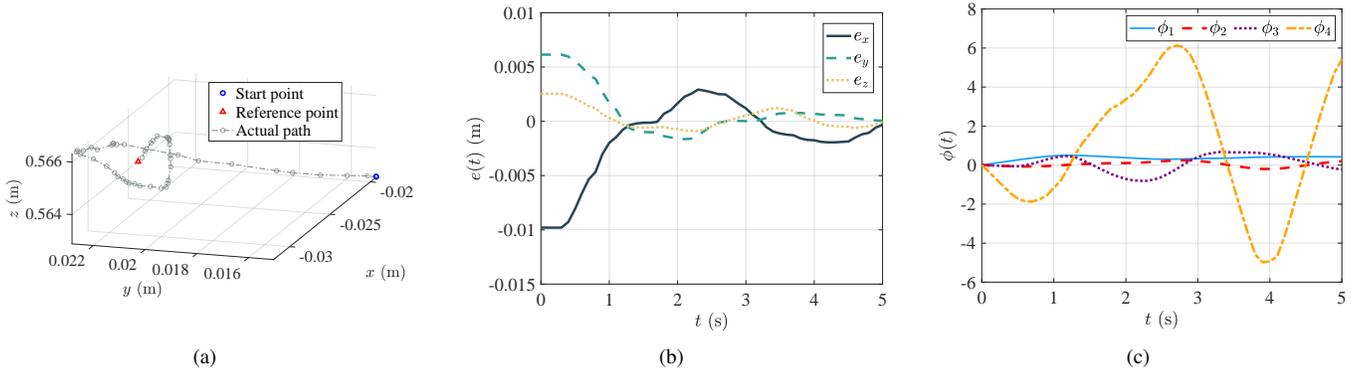


Fig. 15. Experiment results of the CTR in the point-to-point task. (a) Profiles of the reference point and the actual path. (b) Profiles of position error. (c) Profiles of control signal.

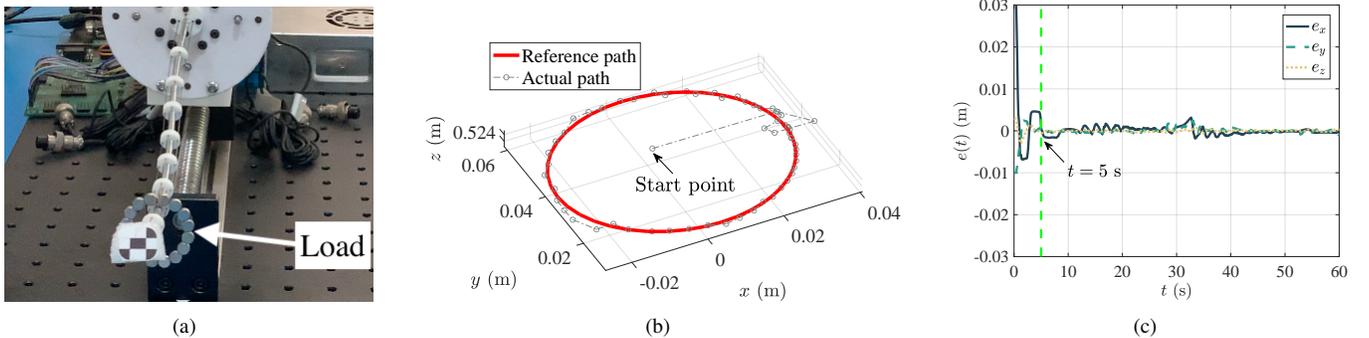


Fig. 16. Experiment results of the CDR with an unknown load for tracking a circular path. (a) A load is attached to the CDR. (b) Profiles of the reference path and the actual path. (c) Profiles of tracking error.

control system can ensure that the tracking error (in tracking tasks) and position error (in point-to-point tasks) converge to zero in predefined time. 3) The proposed control system has better performance than the AFK method and the OZND method in terms of tracking error and Jacobian estimation. Nevertheless, there are also some limitations of this work. For example, the proposed control method has been validated by only two types of heterogeneous continuum robots. The efficacy and merits of the proposed control method could be further revealed by applying the method to other types of

continuum robots in the future, e.g., pneumatic continuum robots.

ACKNOWLEDGMENTS

This research was supported by the National Natural Science Foundation of China (62173352), the Guangdong Basic and Applied Basic Research Foundation (2021A1515012314) and the Key-Area Research and Development Program of Guangzhou (202007030004).

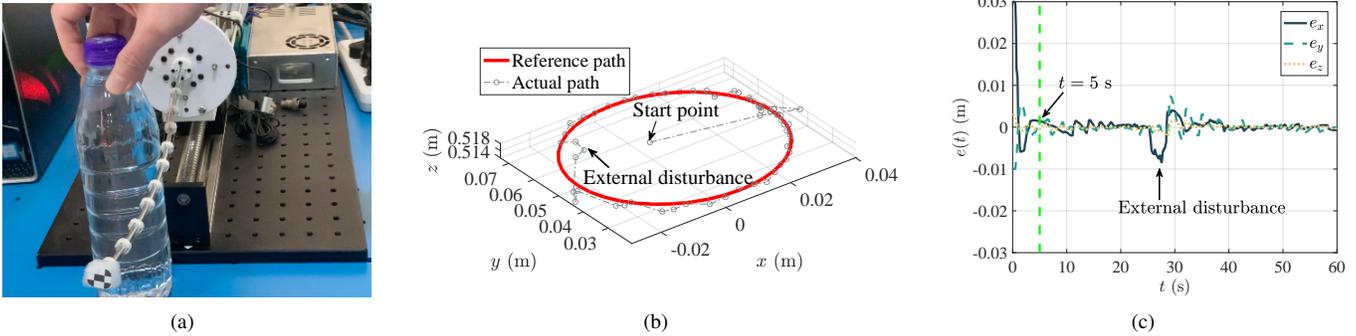


Fig. 17. Experiment results of the CDR for tracking a circular path in the presence of external disturbance. (a) Imposing external disturbance during the tracking task. (b) Profiles of the reference path and the actual path. (c) Profiles of tracking error.

REFERENCES

- [1] Jorge E. Ayala-Carrillo, Vicente Parra-Vega, Ernesto Olguín-Díaz, and Christian A. Trejo-Ramos. Cascade control for robust tracking of continuum soft robots with finite-time convergence of pneumatic system. *IEEE Control Systems Letters*, 7:577–582, 2023. doi: 10.1109/LCSYS.2022.3206211.
- [2] Caroline E Bryson and D. Caleb Rucker. Toward parallel continuum manipulators. In *IEEE International Conference on Robotics and Automation*, pages 778–785, 2014. doi: 10.1109/ICRA.2014.6906943.
- [3] Salvador Cobos-Guzman, David Palmer, and Dragos Axinte. Kinematic model to control the end-effector of a continuum robot for multi-axis processing. *Robotica*, 35(1):224–240, 2017. doi: 10.1017/S0263574715000946.
- [4] Cosimo Della Santina, Antonio Bicchi, and Daniela Rus. 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. doi: 10.1109/LRA.2020.2967269.
- [5] Meng Ding, Hailei Wu, Xianjie Zheng, and Yu Guo. Adaptive predefined-time attitude stabilization control of space continuum robot. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2022. doi: 10.1109/TCSII.2022.3204840.
- [6] Takanori Emaru, Kazuo Imagawa, Yohei Hoshino, and Yukinori Kobayashi. Estimation of velocity and acceleration by nonlinear filter using sliding mode and application to control systems. In *Asia International Symposium on Mechatronics*, pages 212–217, 2008. doi: 10.1299/kikaic.75.2547.
- [7] Valentin Falkenhahn, Frank A. Bender, Alexander Hildebrandt, Rudiger Neumann, and Oliver Sawodny. Online TCP trajectory planning for redundant continuum manipulators using quadratic programming. In *International Conference on Advanced Intelligent Mechatronics*, pages 1163–1168, 2016. doi: 10.1109/AIM.2016.7576927.
- [8] Michele Giorelli, Federico Renda, Marcello Calisti, Andrea Arienti, Gabriele Ferri, and Cecilia Laschi. Neural network and Jacobian method for solving the inverse statistics of a cable-driven soft arm with nonconstant curvature. *IEEE Transactions on Robotics*, 31(4):823–834, 2015. doi: 10.1109/TRO.2015.2428511.
- [9] Xinjia Huang, Jiang Zou, and Guoying Gu. Kinematic modeling and control of variable curvature soft continuum robots. *IEEE/ASME Transactions on Mechatronics*, 26(6):3175–3185, 2021. doi: 10.1109/TMECH.2021.3055339.
- [10] Hao Jiang, Xinghua Liu, Xiaotong Chen, Zhanchi Wang, Yusong Jin, and Xiaoping Chen. Design and simulation analysis of a soft manipulator based on honeycomb pneumatic networks. In *IEEE International Conference on Robotics and Biomimetics*, pages 350–356, 2016. doi: 10.1109/ROBIO.2016.7866347.
- [11] Bryan A. Jones and Ian D. Walker. Kinematics for multisection continuum robots. *IEEE Transactions on Robotics*, 22(1):43–55, 2006. doi: 10.1109/TRO.2005.861458.
- [12] Kit Hang Lee, Denny K.C. Fu, Martin C.W. Leong, Marco Chow, Hing Choi Fu, Kaspar Althoefer, Kam Yim Sze, Chung Kwong Yeung, and Ka Wai Kwok. Non-parametric online learning control for soft continuum robot: An enabling technique for effective endoscopic navigation. *Soft Robotics*, 4(4):324–337, 2017. doi: 10.1089/soro.2016.0065.
- [13] Minhan Li, Rongjie Kang, David T. Branson, and Jian S. Dai. Model-free control for continuum robots based on an adaptive Kalman filter. *IEEE/ASME Transactions on Mechatronics*, 23(1):286–297, 2018. doi: 10.1109/TMECH.2017.2775663.
- [14] Weibing Li. Predefined-time convergent neural solution to cyclical motion planning of redundant robots under physical constraints. *IEEE Transactions on Industrial Electronics*, 67(12):10732–10743, 2020. doi: 10.1109/TIE.2019.2960754.
- [15] Hang Liu, Tie Wang, and Dongsheng Guo. Design and validation of zeroing neural network to solve time-varying algebraic Riccati equation. *IEEE Access*, 8:211315–211323, 2020. doi: 10.1109/ACCESS.2020.3039253.
- [16] Xuanjiao Lv, Lin Xiao, Zhiguo Tan, and Zhi Yang.

- Wsbp function activated Zhang dynamic with finite-time convergence applied to Lyapunov equation. *Neurocomputing*, 314:310–315, 2018. doi: 10.1016/j.neucom.2018.06.057.
- [17] Tobias Mahl, Alexander Hildebrandt, and Oliver Sawodny. A variable curvature continuum kinematics for kinematic control of the bionic handling assistant. *IEEE Transactions on Robotics*, 30(4):935–949, 2014. doi: 10.1109/TRO.2014.2314777.
- [18] Milad S Malekzadeh, Sylvain Calinon, Danilo Bruno, and Darwin G Caldwell. Learning by imitation with the STIFF-FLOP surgical robot: A biomimetic approach inspired by octopus movements. *Robotics and Biomimetics*, 1:13, 2014. doi: 10.1186/preaccept-1725134953134311.
- [19] Achille Melingui, Othman Lakhal, Boubaker Daachi, Jean Bosco Mbede, and Rochdi Merzouki. Adaptive neural network control of a compact bionic handling arm. *IEEE/ASME Transactions on Mechatronics*, 20(6):2862–2875, 2015. doi: 10.1109/TMECH.2015.2396114.
- [20] Jonathan Obregon-Flores, Gustavo Arechavaleta, Hector M. Becerra, and America Morales-Diaz. Predefined-time robust hierarchical inverse dynamics on torque-controlled redundant manipulators. *IEEE Transactions on Robotics*, 37(3):962–978, 2021. doi: 10.1109/TRO.2020.3042054.
- [21] Andrew L. Orekhov, Elan Z. Ahronovich, and Nabil Simaan. Lie group formulation and sensitivity analysis for shape sensing of variable curvature continuum robots with general string encoder routing. *IEEE Transactions on Robotics*, 2023. doi: 10.1109/TRO.2022.3232273.
- [22] Richard Seeber, Hernan Haimovich, and Martin Horn. Robust exact differentiators with predefined convergence time. *Automatica*, 134:109858, 2021. doi: 10.1016/j.automatica.2021.109858.
- [23] Theodore E. Simos, Vasilios N. Katsikis, Spyridon D. Mourtas, and Predrag S. Stanimirović. Finite-time convergent zeroing neural network for solving time-varying algebraic Riccati equations. *Journal of the Franklin Institute*, 359(18):10867–10883, 2022. doi: 10.1016/j.jfranklin.2022.05.021.
- [24] Ning Tan, Peng Yu, Fenglei Ni, and Zhenglong Sun. Trajectory tracking of soft continuum robots with unknown models based on varying parameter recurrent neural networks. In *IEEE International Conference on Systems, Man and Cybernetics*, pages 1035–1041, 2021. doi: 10.1109/SMC52423.2021.9659281.
- [25] Ning Tan, Peng Yu, Xinyu Zhang, and Tao Wang. Model-free motion control of continuum robots based on a zeroing neurodynamic approach. *Neural Networks*, 133: 21–31, 2021. doi: 10.1016/j.neunet.2020.10.005.
- [26] Xiaomei Wang, Yingqi Li, and Ka Wai Kwok. A survey for machine learning-based control of continuum robots. *Frontiers in Robotics and AI*, 8:730330, 2021. doi: 10.3389/frobt.2021.730330.
- [27] Robert J. Webster and Bryan A. Jones. Design and kinematic modeling of constant curvature continuum robots: A review. *International Journal of Robotics Research*, 29 (13):1661–1683, 2010. doi: 10.1177/0278364910368147.
- [28] Keyu Wu, Guoniu Zhu, Liao Wu, Wenchao Gao, Shuang Song, Chwee Ming Lim, and Hongliang Ren. Safety-enhanced model-free visual servoing for continuum tubular robots through singularity avoidance in confined environments. *IEEE Access*, 7:21539–21558, 2019. doi: 10.1109/ACCESS.2019.2891952.
- [29] Lin Xiao, Bolin Liao, Shuai Li, Zhijun Zhang, Lei Ding, and Long Jin. Design and analysis of FTZNN applied to the real-time solution of a nonstationary Lyapunov equation and tracking control of a wheeled mobile manipulator. *IEEE Transactions on Industrial Informatics*, 14(1):98–105, 2018. doi: 10.1109/TII.2017.2717020.
- [30] Lin Xiao, Yingkun Cao, Jianhua Dai, Lei Jia, and Haiyan Tan. Finite-time and predefined-time convergence design for zeroing neural network: Theorem, method, and verification. *IEEE Transactions on Industrial Informatics*, 17(7):4724–4732, 2021. doi: 10.1109/TII.2020.3021438.
- [31] Michael C. Yip and David B. Camarillo. Model-less feedback control of continuum manipulators in constrained environments. *IEEE Transactions on Robotics*, 30(4):880–889, 2014. doi: 10.1109/TRO.2014.2309194.
- [32] Peng Yu, Ning Tan, Mao Zhang, and Fenglei Ni. Inverse-free tracking control of continuum robots with unknown models based on gradient neural networks. In *IEEE 11th Data Driven Control and Learning Systems Conference*, pages 648–653, 2022. doi: 10.1109/DDCLS55054.2022.9858510.
- [33] Yunong Zhang and Dongsheng Guo. *Zhang Functions and Various Models*. Springer, Berlin, Heidelberg, 2015. doi: 10.1007/978-3-662-47334-4.
- [34] Yunong Zhang, Danchi Jiang, and Jun Wang. A recurrent neural network for solving Sylvester equation with time-varying coefficients. *IEEE Transactions on Neural Networks*, 13(5):1053–1063, 2002. doi: 10.1109/TNN.2002.1031938.