



Modeling and Trajectory Tracking Control for a Multi-Section Continuum Manipulator

Elsayed Atif Aner^{1,2} · Mohammed Ibrahim Awad² · Omar M. Shehata²

Received: 23 December 2022 / Accepted: 28 May 2023
© The Author(s) 2023

Abstract

This paper presents a dynamic modeling and trajectory tracking control design for a hyper-redundant continuum manipulator (HRM), which is able to continuously bend along its length. The HRM has the advantages of high flexibility, large workspace, and low inertia, but also poses challenges of complex kinematics, and dynamic coupling. To address these challenges, the kinematics model was developed based on the piecewise constant curvature (PCC) assumption that captures the relationship between the cable lengths, joint angles, and end-effector pose. Inverse kinematics is presented geometrically by solving trigonometric equations. The dynamic model for the continuum robot (CR) is derived using the Euler–Lagrange representation, which incorporates the effects of gravity and elasticity. Accordingly, three control strategies are developed and applied to a two-section continuum robot, which are the inverse dynamic proportional integral derivative controller (PID), fuzzy logic controller (FLC), and sliding mode controller (SMC). We validate our proposed methods through simulations on various 3D trajectories, utilized MATLAB symbolic math toolbox in conjunction with Simulink simulation; demonstrate and evaluate different close-loop dynamic control responses for different scenarios. This simulation used in the creation of a unique animated simulation for a whole 3D continuum robot while tracking its desired trajectories. Finally, the simulation results demonstrate that our proposed methods can effectively control the HRM with high tracking performance, exhibiting enhanced responses in terms of settling time, rising time, and steady-state error. While the developed SMC outperforms Poth's PID and FLC in terms of robustness and settling time (51% and 10.3%), respectively.

Keywords Continuum robotic arm · Dynamics · Hyper-redundant manipulator · Position control · Sliding mode controller · Trajectory control

1 Introduction

Continuum robots are a type of under actuated hyper-redundant manipulators (HRMs), which are typically inspired by natural biological features, they emulate nature in their movement, such as snakes, elephant trunks, and squid tentacles [1],

hyper-redundant is a term which denote robots that have large or infinite degrees of freedom [2].

Continuum Robots (CRs) are fundamentally built based on a flexible backbone on which is mounted a series of rigid discs; their structures continuously bend through elastic deformation along their length, in contrast to conventional robots, which typically consist of a number of rigid links and rotating joints. They can be built with multiple sections; thus, they are theoretically capable of having an endless degree of freedom. Where it shows a significant compliance and great operational capabilities for manipulating objects and interacting with the environment, there for it is a most suitable for minimal invasive surgery [3], medical application [4], and operating in a complex, unstructured environment.

Various techniques can be used to actuate CR, including motor-driven tendons [5], pneumatically operated CRs [6], and an intelligent material, shape memory alloy (SMA) [7],

✉ Elsayed Atif Aner
alsayedatif88@gmail.com; elsayed-atef@eru.edu.eg

Mohammed Ibrahim Awad
mohammed.awad@eng.asu.edu.eg

Omar M. Shehata
omar.shehata@eng.asu.edu.eg

¹ Department of Mechatronics Engineering, Egyptian Russian University (ERU), Badr, Cairo 11829, Egypt

² Department of Mechatronics Engineering, Ain Shams University (ASU), Cairo, Cairo 11517, Egypt

which can alter its shape by passing current through, where the first two actuation methods use significantly large driving hardware. At the same time, the SMA is best-suitable for miniature CRs.

Continuum robots are complex systems that are challenging to model due to their nonlinearity and infinite degrees of freedom. However, On the basis of various simplifications and assumptions, several research investigations have been attempted to model the kinematics of soft continuum robots.

To the best of our knowledge, the HRM was introduced in 1995 by Chirikjian, G.S. [8]. From then on various methodologies and theories, such as Serret–Frenet frames, Euler–Bernoulli beam equation, and Denavit–Hartenberg parameters (DH), have been utilized to develop a kinematic model [9–11]. While the commonly used methods for the dynamic model: finite element method [12] where it is used to solve the boundary value problem of a partial differential equation by discretizing the domain of interest into a finite number of subdomains called finite elements and applying a variational principle to minimize the error function and obtain a stable solution. This method is applicable to a variety of complex structures, however obtaining numerical solutions when internal or external motion constraints are present poses a challenge due to the high computational complexity. To obtain a more extensive description of soft robotic arms, the Cosserat theory was proposed. The Cosserat technique has been used to control and manipulate flexible robotic arm [13]. Despite the high accuracy of the model created by this technique, the partial differential equations in this model cannot be used in effective control design.

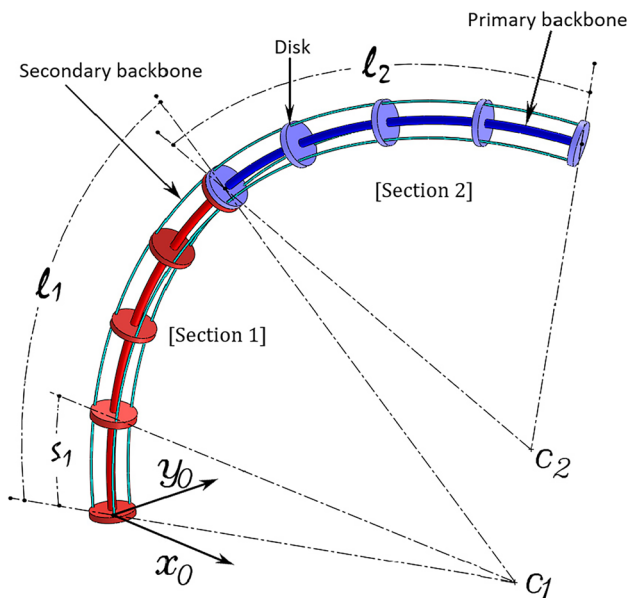


Fig. 1 Three-dimensional design of two sections continuum robot with defined components

Kazemipour et al. [14] presents a continuum robot dynamic model based on the Euler–Lagrange approach which does not include oversimplified assumptions. In addition they introduce an adaptive sliding mode controller that can adjust its parameters online based on the tracking error. The results show that the controller can achieve accurate task-space trajectory tracking under different payloads and external forces. Thuruthel et al. [15] proposes a method for modeling and controlling soft robots using first-order dynamical systems based on Cosserat-rod mechanics. In addition they present two closed-loop control strategies based on the reduced-order state feedback. The results indicate that controllers developed on this assumption can compensate the errors in modeling accuracy with the increased control frequency. Several studies have explored different techniques for controlling and stabilizing robotic systems using fuzzy logic, neural networks, passivity and flatness [16–20].

We adopted Piece wise constant curvature (PCC) approximation as the first step in modeling the kinematics of HRMs, where the bending section of the CR is defined as an arc of a circle and the bending plane is capable of rotating along a fixed axis [21]. While [22, 23] covers the inverse kinematics of the PCC model for CRs.

By solving the forward and inverse system kinematics, we developed a system dynamics model using the Euler–Lagrange representation that incorporates the effects of gravity, and elasticity. Where symbolic Math Toolbox is used to derive and apply inverse kinematics to robot arms by using symbolic variables for solving system partial derivatives.

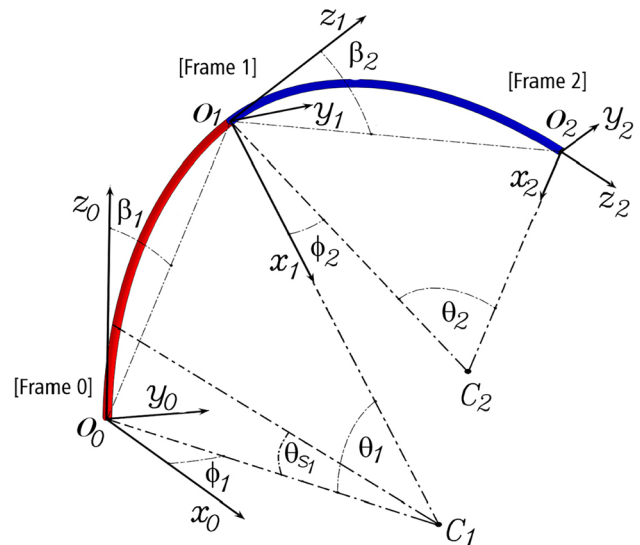
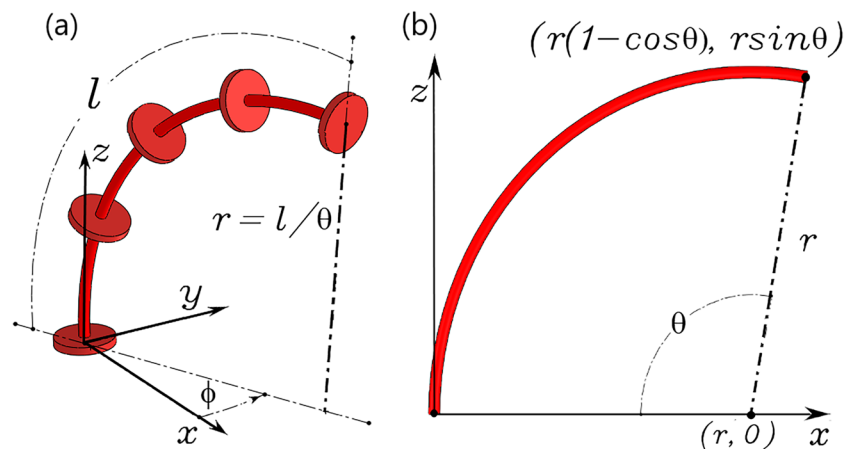


Fig. 2 Two-section continuum robot diagram with defined parameters

Fig. 3 Single section diagram, **a** The arc is rotated away from the x - z plane by the angle φ , **b** When φ equals zero, the arc is located on the x - z plane



Eventually, we developed and applied three various control strategies to control the position and trajectory tracking of the robot. (a) The inverse dynamic PID controller, which utilized the system inverse dynamics to cope with system nonlinearity. (b) The FLC combined with the system's inverse dynamics, which addressed the nonlinearity and coupling effect of the two-link robot arm. (c) The nonlinear SMC with a designed boundary layer δ based on the system stability analysis, which eliminate the chattering phenomena.

By the end, MATLAB Simulink used to simulate and assess different dynamic control loop responses for a given configuration space values, and desired trajectories. This simulation used in the creation of a unique animated simulation for whole 3D continuum robot while tracking its desired trajectories (Online Resource 1).

The contribution of this work can be summarized as follows:

- Developing a system dynamic model based on Euler Lagrange representation that incorporates the effects of gravity, and elasticity, using MATLAB symbolic Math Toolbox for solving system dynamics.
- For accurate position and trajectory tracking at high speeds, three various control algorithms, inverse dynamic PID, intelligent FLC, and nonlinear SMC, are precisely developed for our particular CR model.
- We validate our developed methods through MATLAB Simulink simulations, where a comparative study is performed for illustrating and clarifying the various dynamic responses for each pre-designed controller in mapped 2D and 3D space trajectories.

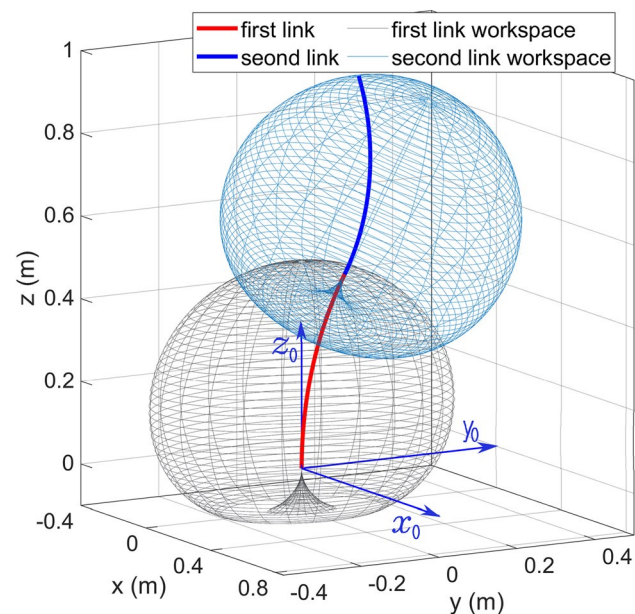


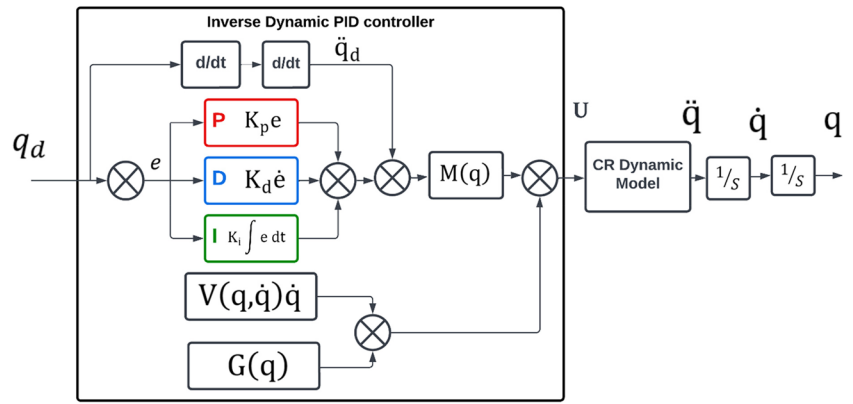
Fig. 4 Two section continuum robot workspaces

The remainder of this paper is organized as follows. In Section 2, the robot mechanical structure and modelling assumptions are introduced. Section 3 presents the kinematic model for two section CR based on constant curvature (CC) assumption, followed by the dynamic model developed using classic Lagrangian representation, which are shown in Section 4. Three control strategies are developed and investigated in Section 5. Dynamic model validation presented in Section 6. Section 7 presents the simulation study for step and trajectory responses, conducted using MATLAB Simulink. Finally, a conclusion and further research plans are stated in Section 8.

2 Continuum Robot Design

Figure 1 shows the continuum robot structure. Each section is comprised of three main parts: a series of disks, a flexible backbone (primary backbone) (PB), and three flexible wires (secondary backbones) (SBs). Each section is made up of a 50 cm long flexible backbone formed

Fig. 5 The inverse dynamic PID controller block diagram for the continuum robot dynamic system



of composite materials and five disks that are uniformly distributed and rigidly mounted. Each disk features three holes for three driving wires, distributed in a circular pattern 120° apart from one another. In order to control the behavior of the CR, the wires for each section are utilized to apply a moment to the flexible backbone tip. It is crucial to emphasize that the flexible backbone deflection can control the robot's spatial mobility when adequate tension forces are applied to one or two cables at once, which achieves a 2-DOF bending motion of the continuum robot. By using a series of 2-DOF links, a multi-DOF continuum robot arm can be achieved.

3 The Two-section Continuum Robot Kinematics and Modeling Assumption

Constant curvature has frequently been viewed as a desirable modeling characteristic in hyper-redundant continuum robots due to its simplifications in kinematic modeling.

According to the PCC assumption on modelling CRs presented in [21]. The bending of each section of the robot is considered an arc. The arc parameters are as follows:

- θ_i , the angle corresponding to the bending of each section.
- φ_i , the angle of the arc plane.

Fig. 6 FLC membership function, **a** first input vector \dot{q} , **b** second input vector \ddot{q} , **c** output vector $\ddot{\theta}_{i(f)}$, and $\ddot{\varphi}_{i(f)}$

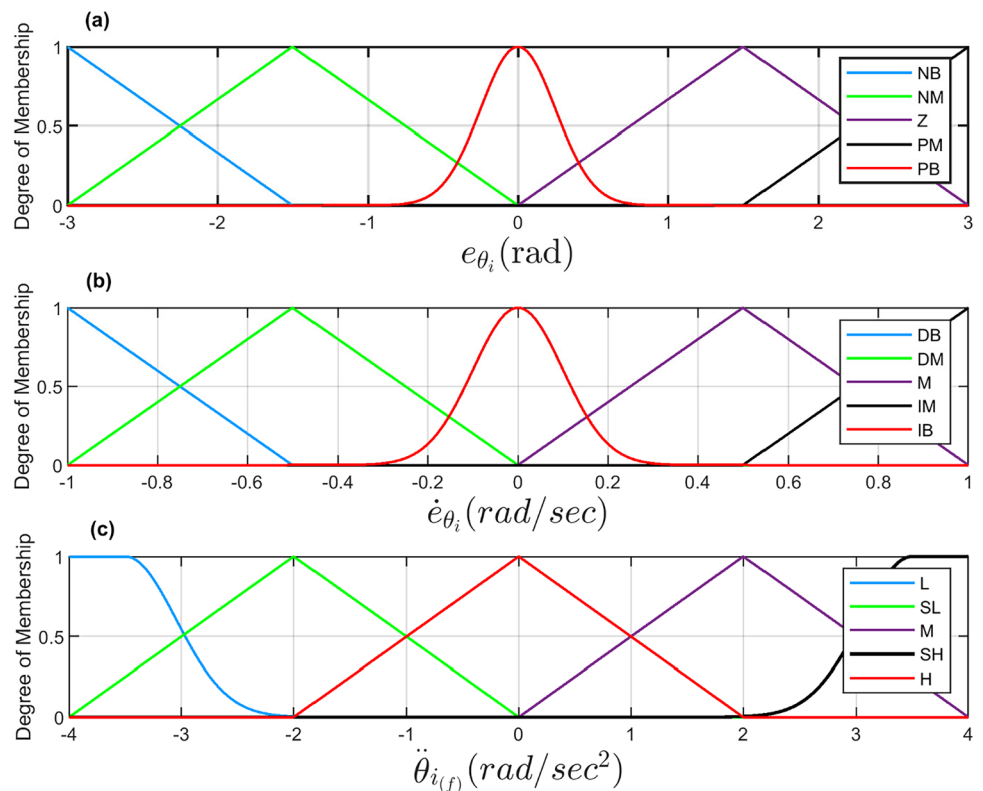
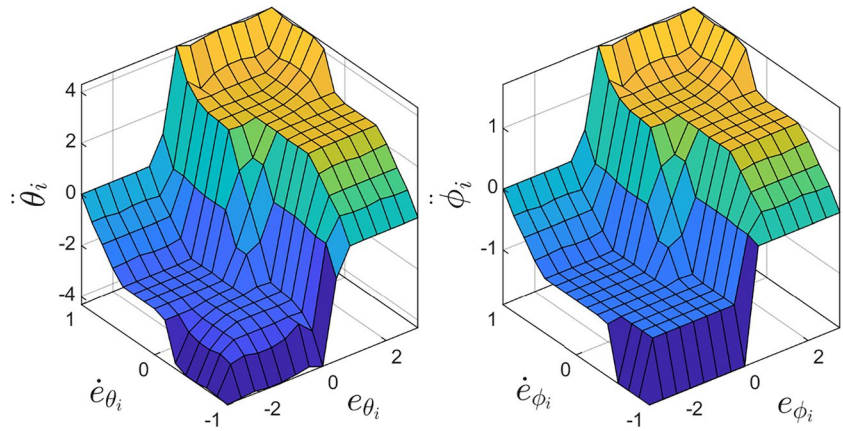


Fig. 7 Control Surface for $\dot{\theta}_{i(t)}$, and $\dot{\phi}_{i(t)}$



- l_i , length of each section's arc.
- ri , radius of curvature.
- β_i , the bending angle of the primary backbone tangent.
- si , the length of a subsection's arc.

Where $i = 1, 2$. in the first section when $i = 1$ or in the second section when $i = 2$. Where, as illustrated in Fig. 1 each subsection has a corresponding bending angle θ_{s_i} .

$$\theta_{s_i} = s_i \theta_i / l_i \tag{1}$$

In Which, $\mathbf{q} \in \mathbb{R}^4$ can be used to express the robot configuration space. Where $\mathbf{q} = [\theta_1 \varphi_1 \theta_2 \varphi_2]^T$. As shown in Fig. 2, there are three frames with three axes each x_i, y_i , and z_i axis for a two sections CR, with all parameters defined.

3.1 Forward Position Kinematics

A continuum robot's geometry provides a method for figuring out how the points along it are posed [21, 24, 25]. As shown in Fig 3, when $\phi = 0$ the coordinates of a point on the circular arc of radius r in the $x - z$ plane centered at $[r \ 0 \ 0]^T$ are:

$$p = [r(1 - \cos \theta) \ 0 \ r \sin \theta]^T \tag{2}$$

Also, note that this motion includes a rotation $\mathbf{R}_y(\theta) \in \mathbb{R}^{3 \times 3}$ about the positive y -axis. And rotating the entire arc by

$\mathbf{R}_z(\phi) \in \mathbb{R}^{3 \times 3}$ about the positive z -axis by ϕ moves the robot out of the (x, z) plane, producing a transformation from arc base to tip.

$$T = \underbrace{\begin{bmatrix} \mathbf{R}_z(\phi) & 0 \\ 0 & 1 \end{bmatrix}}_{\text{Rotation}} \underbrace{\begin{bmatrix} \mathbf{R}_y(\theta) & p \\ 0 & 1 \end{bmatrix}}_{\text{Inplane transformation}} \tag{3}$$

$\mathbf{T}_i^{i-1} \in \mathbb{R}^{4 \times 4}$ express the Transformation matrix of each i th section with respect to [Frame $i - 1$] [21, 26], where general transformation matrix form $\mathbf{T}_i^{i-1} = \begin{bmatrix} \mathbf{R}_i^{i-1} & P \\ 0 & 0 & 0 & 1 \end{bmatrix}$, by substituting (2) and (3):

$$\mathbf{T}_i^{i-1} = \begin{bmatrix} \cos \varphi_i \cos \theta_i & -\sin \varphi_i \cos \theta_i & \sin \varphi_i \cos \theta_i & r_i \cos \varphi_i (1 - \cos \theta_i) \\ \sin \varphi_i \cos \theta_i & \cos \varphi_i \cos \theta_i & \sin \varphi_i \cos \theta_i & r_i \sin \varphi_i (1 - \cos \theta_i) \\ -\sin \theta_i & 0 & \cos \theta_i & r_i \sin \theta_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

The following matrix, express the end position of each subsection with respect to [Frame $i - 1$], where θ_{s_i} substitute θ_i in the position vector of (4):

$$\mathbf{O}_{s_i}^{i-1} = \begin{bmatrix} \mathcal{O}_{s_i(x)}^{i-1} & \mathcal{O}_{s_i(y)}^{i-1} & \mathcal{O}_{s_i(z)}^{i-1} \end{bmatrix}^T = \begin{bmatrix} r_i \cos \varphi_i (1 - \cos \theta_{s_i}) \\ r_i \sin \varphi_i (1 - \cos \theta_{s_i}) \\ r_i \sin \theta_{s_i} \end{bmatrix} \tag{5}$$

where, $r_i = l_i / \theta_i$.

Table 1 Fuzzy logic controller rule base table

		e_{θ_i}, e_{ϕ_i}				
		NB	NM	Z	PM	PB
$\dot{\theta}_i,$ $\dot{\phi}_i$	DB	L	L	L	M	M
	DM	L	SL	SL	SH	SH
	M	SL	SL	M	SH	SH
	IM	SL	SL	SH	SH	H
	IB	M	M	H	H	H

e_{θ_i} is the error in θ angle, \dot{e}_{θ_i} derivative of the error e_{θ_i} . Where $i = 1$ for first section and 2 for second section

Table 2 Amouri et al. PID parameter constant

	k_p	k_i	k_d
value	2.8	0.004	0.38

By substituting $i = 1$ and $s_1 = l_1$ in (5), the end position of the first section with respect to [Frame 0] (O_1^0) can be obtained.

The end position of a subsection in the second section with respect to [Frame 0] ($O_{s_2}^0$) can be expressed by substituting $i = 1$ in (4) to get T_1^0 , as follows:

$$\begin{bmatrix} O_{s_2}^0 \\ 1 \end{bmatrix} = \begin{bmatrix} O^0 & O^0 & O^0 & 1 \end{bmatrix}^T = T_1^0 \begin{bmatrix} O_{s_2}^1 \\ 1 \end{bmatrix} \tag{6}$$

where, $O_{s_2}^1$ represents the second section's end position relative to [Frame 1] and can be calculated using (5) for $i = 2$.

By substituting $s_2 = l_2$ in (6), the end position of the second section with respect to [Frame 0] (O_2^0) can be obtained.

3.2 Inverse Position Kinematics

The configuration angles $[\theta_1 \phi_1 \beta_1 \theta_2 \phi_2 \beta_2]$ for two section CR, can geometrically calculated by knowing the end position of each section with respect to [frame 0] shown in Fig. 2. as follows:

$$\varphi_i = \tan^{-1} \left(O_{i(y)}^{i-1} / O_{i(x)}^{i-1} \right) \tag{7}$$

$$\beta_i = \tan^{-1} \left(\sqrt{[O_{i(x)}^{i-1}]^2 + [O_{i(y)}^{i-1}]^2} / O_{i(z)}^{i-1} \right) \tag{8}$$

$$\theta_i = \begin{cases} 2\beta_i, & O_{i(z)}^{i-1} > 0 \\ \pi, & O_{i(z)}^{i-1} = 0 \\ 2\pi + 2\beta_i, & O_{i(z)}^{i-1} < 0 \end{cases} \tag{9}$$

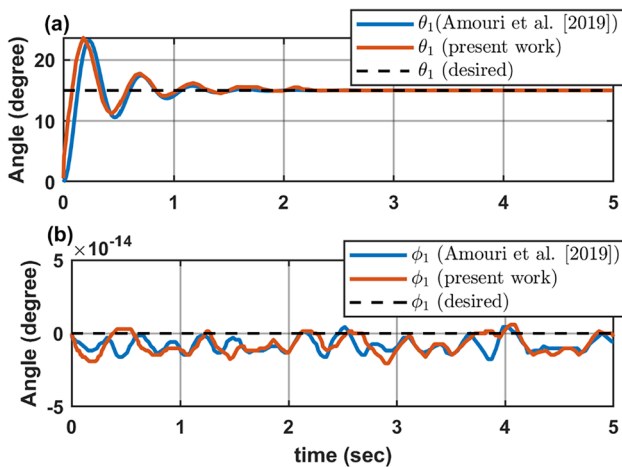


Fig. 8 dynamic model response for Amouri et al. and present work for the same PID, where **a** θ_1 response, and **b** ϕ_1 response

Figure 4 shows the CR workspace for each link based on the introduced kinematics model.

3.3 Forward Velocity Kinematics

the velocity of each section is obtained by providing the angular velocities of the configuration space $\dot{\mathbf{q}} = [\dot{\theta}_1 \ \dot{\phi}_1 \ \dot{\theta}_2 \ \dot{\phi}_2]$.

Let the task space of the robot ($O \in R^6$) to be:

$$O = [O_1^0 \ O_2^0]^T \tag{10}$$

Therefore, by differentiating (10) by the time, the velocity of end point for second section can be obtained:

$$\dot{O} = \frac{\partial O}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial t} = \mathbf{J} \dot{\mathbf{q}} \tag{11}$$

where, the Jacobian matrix is $\mathbf{J} = \frac{\partial O}{\partial \mathbf{q}} \in R^{6 \times 4}$.

3.4 Inverse Velocity Kinematics

The configuration space velocities may be acquired in the manner shown below by supplying the task space velocities:

$$\dot{\mathbf{q}} = \mathbf{J}^{-1} \dot{O} \tag{12}$$

Since the system is underactuated, the inverse Jacobian matrix, $\mathbf{J}^{-1} \in R^{4 \times 6}$, is constructed using the pseudo-inverse approach.

3.5 Forward Acceleration Kinematics

By differentiating (11), the task space acceleration (\ddot{O}) can be obtained, as the configuration space angular acceleration is provided:

$$\ddot{O} = \frac{\partial O}{\partial \mathbf{q}} \frac{\partial \dot{\mathbf{q}}}{\partial t} + \frac{\partial \dot{O}}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial t} = \mathbf{J} \ddot{\mathbf{q}} + \dot{\mathbf{J}} \dot{\mathbf{q}} \tag{13}$$

where, $\ddot{\mathbf{q}} = [\ddot{\theta}_1 \ \ddot{\phi}_1 \ \ddot{\theta}_2 \ \ddot{\phi}_2]^T$, and $\dot{\mathbf{J}} = \frac{\partial \dot{O}}{\partial \mathbf{q}}$ is the differentiation of the Jacobian matrix.

4 Dynamics of The Two-Section Continuum Robot

In this section, the Euler-Lagrange equation is used to represent whole-body dynamics [10, 26, 27]. whereby kinetic and potential energy needs to be examined. As depicted in Fig. 1, the robot is divided into sections, each of which is made up of a primary backbone (PB), three secondary backbones (SBs) or actuators, and several discs. Where the MATLAB symbolic toolbox [28, 29] is utilized to derive and solve dynamic system

Table 3 Continuum manipulator defined parameter

Parameter	Designation	Value
L_1	First link length	0.5 m
L_2	Second link length	0.5 m
α	Distance between the (PB) and each (SBs)	0.02 m
E_P	Modules of elasticity of the (PB)	65 Pa $\times 10^9$
ρ_s	Density for Secondary Backbone (SB)	5000 kg/m ³
ρ_P	Density for Primary Backbone (PB)	5000 kg/m ³
A_P	Cross section area of the (PB)	28.26 $\times 10^{-4}$ m ²
I_P	Second moment of cross-sectional area of the PB	5.15 $\times 10^{-14}$ m ⁴
A_s	Cross section area of the (SB)	7.85 $\times 10^{-7}$ m ²
g	Gravitational acceleration	9.81 m/s ²
I_D	Mass moment of inertia of each disk	3.48 $\times 10^{-7}$ kg.m ²
h	Distance between each disk	0.1 m
I_s	Second moment of cross-sectional area of the (SB)	4.91 $\times 10^{-14}$ m ⁴
m_D	Mass of each disk	0.01 kg

differential equations of robot arms by using symbolic variables and expressions.

4.1 Kinetic Energy Analysis

The three basic components of the kinetic energy analysis are the kinetic energies of the primary backbone (PB), the secondary backbones (SBs), and the kinetic energies of each disc.

The kinetic energy for a differential section of the backbone, with mass of dm and inertia moment of dI , The kinetic energy of the PBs of the two sections expressed by KE_{pb} can be obtained as follows:

$$KE_{pb} = \frac{1}{2} \sum_{i=1}^2 \int_0^{l_i} \left[\left(\frac{dO_{s_i(x)}^0}{dt} \right)^2 + \left(\frac{dO_{s_i(y)}^0}{dt} \right)^2 + \left(\frac{dO_{s_i(z)}^0}{dt} \right)^2 \right] \rho_P A_P ds_i + \frac{1}{2} \sum_{i=1}^2 \int_0^{l_i} \left[\left(\frac{s_i \dot{\theta}_i}{l_i} \right)^2 + (\dot{\varphi}_i)^2 \right] \rho_P I_P ds_i \tag{14}$$

where, ρ_P is the density, A_P is the cross-section area, and I_P is the second moment of cross-sectional area of the PB.

Calculating the derivative of (5), the velocity of point P can be expressed as follows:

$$\frac{dO_{s_i(x)}^{i-1}}{dt} = \cos \varphi_i \left(1 - \cos \theta_{s_i} \right) \frac{dr_i}{dt} - r_i \sin \varphi_i \left(1 - \cos \theta_{s_i} \right) \frac{d\varphi_i}{dt} + r_i \sin \varphi_i \cos \theta_{s_i} \frac{d\theta_{s_i}}{dt}, \tag{15}$$

$$\frac{dO_{s_i(y)}^{i-1}}{dt} = \sin \varphi_i \left(1 - \cos \theta_{s_i} \right) \frac{dr_i}{dt} r_i \cos \varphi_i \left(1 - \cos \theta_{s_i} \right) \frac{d\varphi_i}{dt} + r_i \sin \varphi_i \cos \theta_{s_i} \frac{d\theta_{s_i}}{dt}, \tag{16}$$

$$\frac{dO_{s_i(z)}^{i-1}}{dt} = \sin \theta_{s_i} \frac{dr_i}{dt} + r_i \cos \theta_{s_i} \frac{d\theta_{s_i}}{dt}, \tag{17}$$

The lengths of the SBs are obtained as follows:

$$l_{i,1} = l_i - \alpha \theta_i \cos \left(\sum_{k=1}^i \varphi_k \right), \\ l_{i,2} = l_i - \alpha \theta_i \cos \left(120^\circ - \sum_{k=1}^i \varphi_k \right), \text{ and} \\ l_{i,3} = l_i - \alpha \theta_i \cos \left(120^\circ + \sum_{k=1}^i \varphi_k \right) \tag{18}$$

where, α is the distance between the PB and each SB. and each section has its own independent actuators from all other sections. $i = 1, 2, l_{1,1}, l_{1,2}$, and $l_{1,3}$ are lengths of the first section's actuators (link 1 secondary backbone). And $l_{2,1}, l_{2,2}$, and $l_{2,3}$ are lengths of the second section's actuators (link 2 secondary backbone).

The SBs kinetic energy is divided into two components. The first part (KE_{sb_1}) can be calculated similarly to (14) by substituting each $\frac{1}{2}$ with $\frac{3}{2}$ (because of the 3 SBs), and each subscript "P" with "S" in which, I_S, A_S , and ρ_S are the second moment of cross-sectional area, the cross-section area, and the density of each SB, respectively.

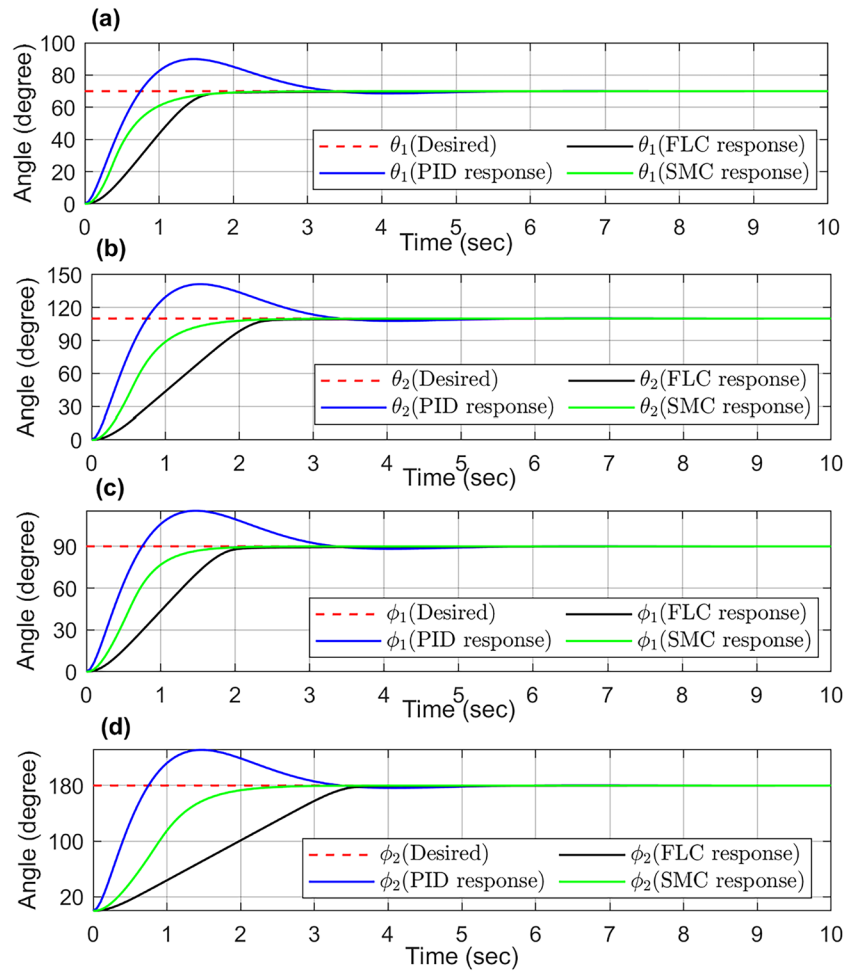
By the differentiation of the actuators' lengths the second part can be constructed as follows:

$$KE_{sb_2} = \frac{1}{2} \sum_{i=1}^2 m_S \left[\left(\frac{dl_{i,1}}{dt} \right)^2 + \left(\frac{dl_{i,2}}{dt} \right)^2 + \left(\frac{dl_{i,3}}{dt} \right)^2 \right] \tag{19}$$

where, m_S is the mass of each SB.

As mentioned in [5], the kinetic energy of the discs located on the first and second sections can be calculated by developing the velocity of each disk, in which all discs are connected to the PB, where the velocity of each disk

Fig. 9 Designed controllers response **a** θ_1 response **b** θ_2 response **c** ϕ_1 response **d** ϕ_2 response



equals the velocity of the PB at the arc location s_i where the disk is connected to the PB, by substituting $s_i = k_i h$ in $\mathbf{O}_{s_i}^0$ to produce $\mathbf{O}_{d_i}^0$, where $\mathbf{O}_{d_i}^0$ represents the positions of each disc in the first section with respect to frame 0 when $i = 1$, and $i = 2$ for each disc position in the second section

with respect to frame 0, where h is the distance between each disk and $k_i = 1, 2, 3, \dots, n_i$. The number of discs in the first and second sections of the CR is indicated by the letters n_1 and n_2 , respectively. $n_1 = n_2 = 5$ In this study. Where,

$$KE_D = \sum_{i=1}^2 \left[\frac{1}{2} \sum_{k_i=1}^{n_i} m_D \left[\left(\frac{dO_{d_i(x)}^0}{dt} \right)^2 + \left(\frac{dO_{d_i(y)}^0}{dt} \right)^2 + \left(\frac{dO_{d_i(z)}^0}{dt} \right)^2 \right] + \frac{1}{2} \sum_{k_i=1}^{n_i} I_D \left[\left(\frac{k_i h \dot{\theta}_i}{l_i} \right)^2 + (\dot{\phi}_i)^2 \right] \right] \quad (20)$$

where, KE_D denotes the kinetic energy of the disks placed on the first and the second section. The mass and the mass moment of inertia of each disk are denoted by m_D and I_D , respectively.

The total kinetic energy KE is:

$$KE = KE_{Pb} + KE_{Sb_1} + KE_{Sb_2} + KE_D \quad (21)$$

4.2 Potential Energy Analysis

The gravitational potential energy and the elastic potential form the CR's total potential energy.

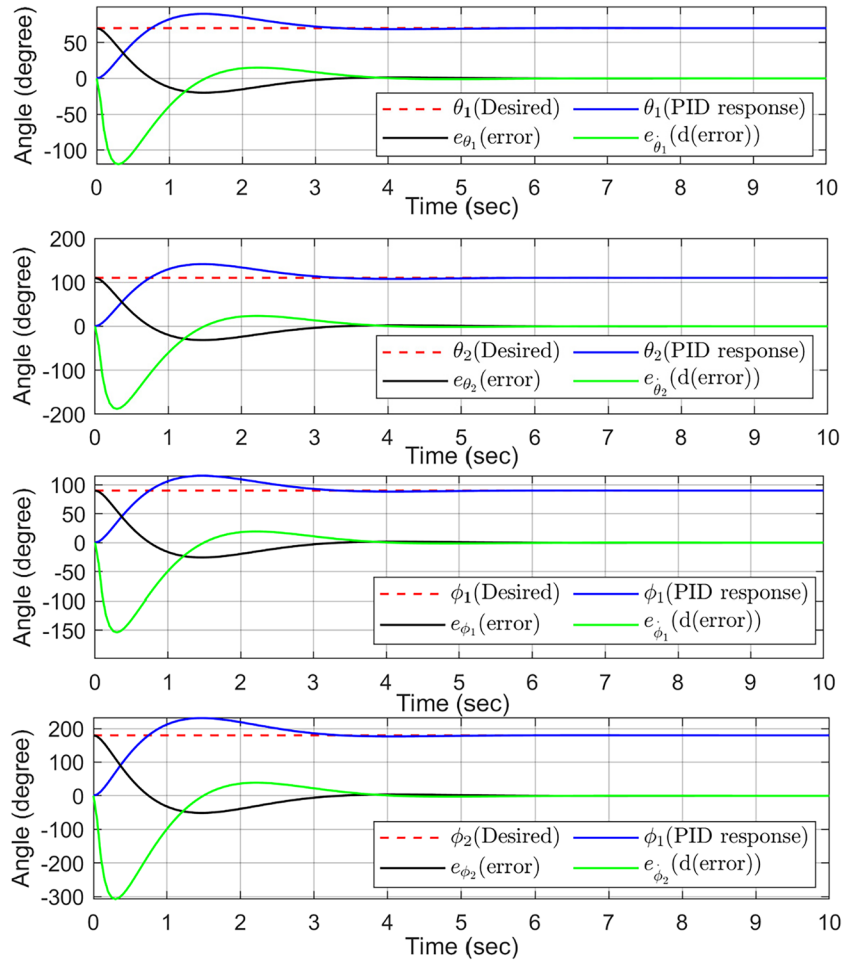
The two components of the gravitational energy analysis are the PB and the disc gravitational energies.

In order to compute the gravitational potential energy of the PB (GPE_{Pb}), the gravitational acceleration (g) is assumed to be directed in the direction of the positive x_0 -axis. Where,

$$GPE_{Pb} = \sum_{i=1}^2 \int_0^{l_i} [O_{s_i(x)}^0] \rho_P A_P g ds_i \quad (22)$$

The first and second sections discs' combined gravitational potential energy, denoted by the symbol GPE_D , are as follows:

Fig. 10 PID controller dynamic response to step input



$$GPE_D = \sum_{i=1}^2 \left[\sum_{k=1}^{n_i} m_D g O_{d_i(x)}^0 \right] \tag{23}$$

The elastic potential energy is given by:

$$EPE = \sum_{i=1}^2 \left[\int_0^{l_i} \frac{E_P I_P}{2} \left(\frac{d\theta_{s_i}}{ds_i} \right)^2 ds_i + 3 \int_0^{l_i} \frac{E_S I_S}{2} \left(\frac{d\theta_{s_i}}{ds_i} \right)^2 ds_i \right] \tag{24}$$

$$= \frac{E_P I_P}{2l_1} \theta_1^2 + \frac{3}{2} \frac{E_S I_S}{l_1} \theta_1^2 + \frac{E_P I_P}{2l_2} \theta_2^2 + \frac{3}{2} \frac{E_S I_S}{l_2} \theta_2^2$$

where, E_P and E_S are the modules of elasticity of the PB and the SBs , respectively.

(25) represent the total potential energy PE , where it is the summation of (22) and (23), and (24). Where,

$$PE = GPE_{pb} + GPE_D + EPE. \tag{25}$$

4.3 Equation of Motion

The Lagrange equation can be expressed, using (21) and (25), as follows:

$$\frac{d}{dt} \frac{\partial KE}{\partial \dot{q}_{(j)}} - \frac{\partial KE}{\partial q_{(j)}} + \frac{\partial PE}{\partial q_{(j)}} = \tau_{(j)} \tag{26}$$

where, $j = 1, 2, 3, 4$. In which, $q_{(1)} = \theta_1, q_2 = \varphi_1, q_3 = \theta_2, q_4 = \varphi_2, \dot{q}_1 = \dot{\theta}_1, \dot{q}_2 = \dot{\varphi}_1, \dot{q}_3 = \dot{\theta}_2, \text{ and } \dot{q}_4 = \dot{\varphi}_2$. (27) represent the final form for the equation of motion:

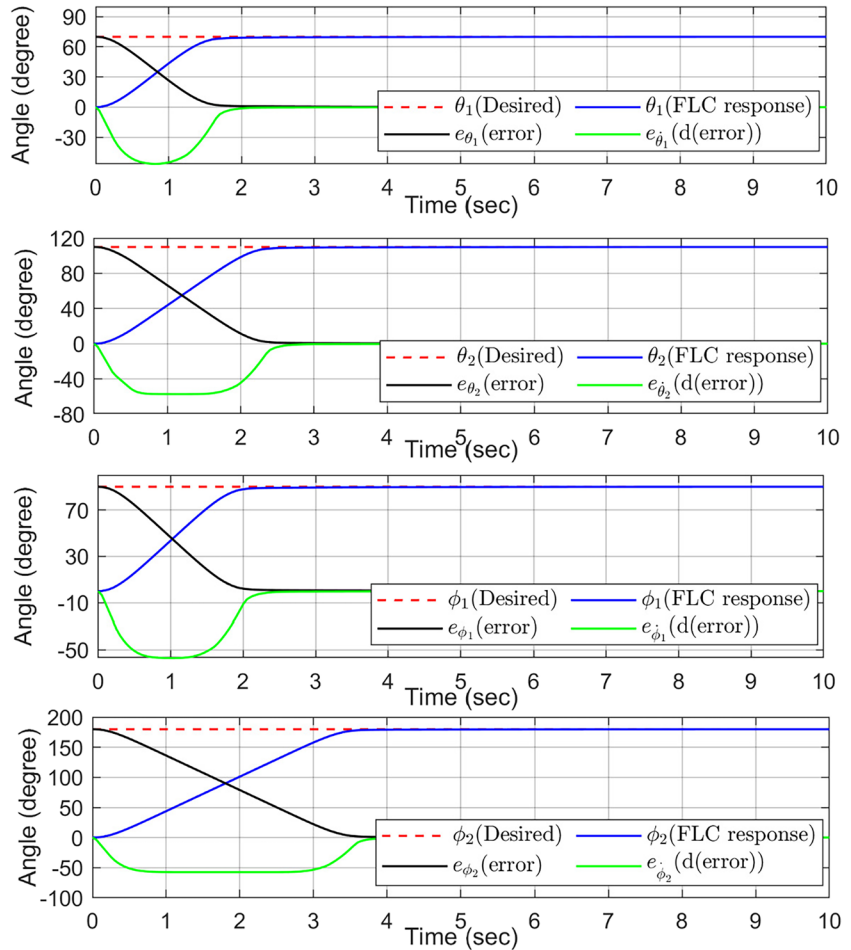
$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{V}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \boldsymbol{\tau} \tag{27}$$

where, $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{4 \times 4}$ is the inertia matrix, $\mathbf{V}(\mathbf{q}) \in \mathbb{R}^{4 \times 4}$ is the matrix of centrifugal Coriolis torque, $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^4$ is the matrix about gravitational torques and $\boldsymbol{\tau} \in \mathbb{R}^4$ is the torques of the CR. where each matrix is comprised of a partial derivatives that were calculated using the MATLAB symbolic toolbox.

5 Control System Design

Using the dynamic model given in (27) we observe that the continuum manipulator is a complex nonlinear system. In this section, we attempt to apply three nonlinear control algorithms to the manipulator system, including the inverse dynamic control PID, fuzzy logic controller and sliding mode controller.

Fig. 11 Fuzzy logic controller dynamic responses to step input



Defining $\mathbf{e}(t) \in R^4$ to be the configuration space error. Where,

$$\mathbf{e}(t) = \mathbf{q}_{(d)}(t) - \mathbf{q}(t) = \left[e_{(\theta_1)} e_{(\varphi_1)} e_{(\theta_2)} e_{(\varphi_2)} \right]^T \quad (28)$$

where, $\mathbf{q}_{(d)} = \left[\theta_{1(d)} \varphi_{1(d)} \theta_{2(d)} \varphi_{2(d)} \right]^T$ is the desired configuration space angles that the robot try to reach.

5.1 PID Controller

with the mass matrix $\mathbf{m}(\mathbf{q}) \in R^{4 \times 4}$ and the vector of nonlinear terms $(\mathbf{V}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q})) \in R^4$.

The system coordinates $q(t)$ should follow a reference time profile described by the vector $\mathbf{q}_d(t)$, with the error $\mathbf{e}(t) = \mathbf{q}_d(t) - \mathbf{q}(t)$ decaying to zero. The desired error equation should be of second order in this case. It can be chosen in the following linear form:

$$\ddot{\mathbf{e}} + K_d \dot{\mathbf{e}} + K_p \mathbf{e} + K_i \int \mathbf{e} dt = 0 \quad (29)$$

From (29), the second order time derivative of the coordinate vector q can be computed as:

$$\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_d + K_d(\dot{\mathbf{q}}_d - \dot{\mathbf{q}}) + K_p(\mathbf{q}_d - \mathbf{q}) + K_i \int (\mathbf{q}_d - \mathbf{q}) dt \quad (30)$$

Substituting expression (30) into the system dynamic model (27) produces the developed control law:

$$\mathbf{M}(\mathbf{q}) \left(\ddot{\mathbf{q}}_d + K_p \mathbf{e} + K_d \dot{\mathbf{e}} + K_i \int \mathbf{e} dt \right) + \mathbf{V}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \tau_{PID} \quad (31)$$

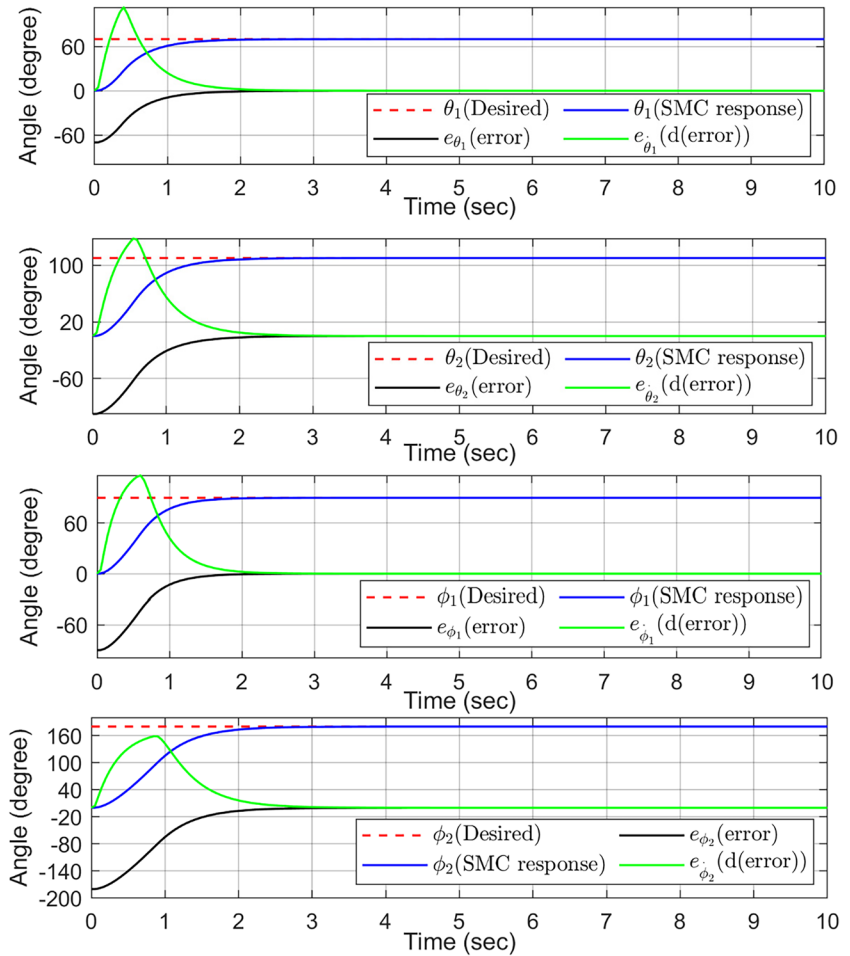
where, k_p is the proportional gain, k_d is the derivative gain, and K_i is the integral gain, $\mathbf{e}(t)$ defined in (28) and $\dot{\mathbf{e}}(t)$ is the differentiation of (28). while k_p, k_d and K_i values are chosen to minimize $\mathbf{e}(t)$.

Figure 5 shows the inverse dynamic PID controller block diagram, where the system dynamic model is solved by MATLAB symbolic math toolbox, and the values of vector q are used as runtime feedback to the controller.

5.2 Fuzzy Logic Controller

Fuzzy logic control (FLC) is a well-known control approach based on artificial intelligence. It exploits the functionary's previous knowledge of the system to be controlled. The primary

Fig. 12 Sliding mode controller dynamic responses to step input



responsibility of the functionary is to establish decision-based rules by studying the system's behavior and the linguistic variables within the framework of the controlled system.

We proposed an FLC controller in conjunction with system inverse dynamics, compensating for system nonlinearity and coupling effects:

$$\tau_{(fuzzy)} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}_{(f)} + \mathbf{V}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) \tag{32}$$

where, $\ddot{\mathbf{q}}_{(f)} = [\ddot{\theta}_{1(f)} \ddot{\phi}_{1(f)} \ddot{\theta}_{2(f)} \ddot{\phi}_{2(f)}]^T$ are the outputs of the FLC.

Taking \mathbf{e} and $\dot{\mathbf{e}}$ as the inputs to the FLC. The linguistic variables of the inputs could be expressed as follows:

$$e_{\theta_i} = e_{\phi_i} = \begin{cases} \text{Negative big (NB)} \\ \text{Negative medium (NM)} \\ \text{Zero (Z)} \\ \text{Positive medium (PM)} \\ \text{Positive big (PB)} \end{cases}, \text{ and}$$

Fig. 13 Sliding surface of SMC

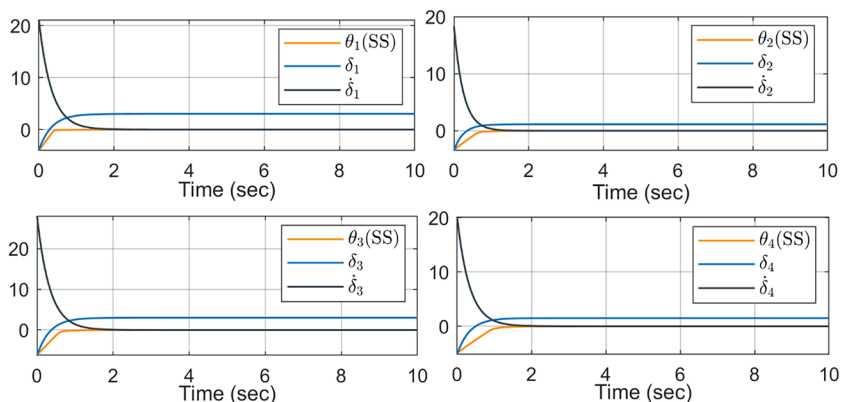


Table 4 Response parameter for designed controllers

Controller	PID	FLC	SMC
overshoot	28.4%	0%	0%
Rise-time	0.45 s	1.02 s	0.59 s
Settling time	2.85 s	1.56 s	1.40 s

$$\dot{e}_{\theta_i} = \dot{e}_{\varphi_i} = \begin{cases} \text{Decreasing big (DB)} \\ \text{Decreasing medium (DM)} \\ \text{Medium (M)} \\ \text{Increasing Medium (IM)} \\ \text{Increasing big (IB)} \end{cases}$$

The linguistic variables of the outputs from the FLC are:

$$\ddot{\theta}_{i(q)} = \ddot{\varphi}_{i(q)} = \begin{cases} \text{Low (L)} \\ \text{Slightly low (SL)} \\ \text{Medium (M)} \\ \text{Slightly high (SH)} \\ \text{High (H)} \end{cases}$$

Considering the error from (28), when $[e_{(\theta_i)}$ less than 0] and $[\dot{e}_{\theta_i}$ less than 0] it means that $[\theta_i$ greater than $\theta_{i(d)}$] and $[\dot{\theta}_i$ greater than $\dot{\theta}_{i(d)}$] which means that the value of θ_i is high and increasing. Where when $[e_{(\theta_i)}$ greater than 0] and $[\dot{e}_{\theta_i}$ greater than 0] it means that $[\theta_i$ less than $\theta_{i(d)}$] and $[\dot{\theta}_i$ less than $\dot{\theta}_{i(d)}$] which means that the value of θ_i is low and decreasing.

Figure 6 show the member ship function for designed FLC, where Fig. 7 show the control surface for $\ddot{\theta}_{i(q)}$ and $\ddot{\varphi}_{i(q)}$.

Table 1 present the rule base table for the FLC, where first input are $(e_{\theta_i}, e_{\varphi_i})$ and the second input are $(\dot{e}_{\theta_i}, \dot{e}_{\varphi_i})$.

5.3 Sliding Mode Controller

SMC is basically a nonlinear adaptive control algorithm that gives a robust performance by using discontinuous control signals to slide the dynamic system along a hypersurface.

The surface of the sliding mode $s(t) \in \mathbb{R}^4$ is:

$$s(t) = \lambda e(t) + \dot{e}(t) \tag{33}$$

where λ must satisfy Hurwitz condition, $\lambda > 0$. The tracking error and its derivative value is:

$$e(t) = \mathbf{q}(t) - \mathbf{q}_d(t), \dot{e}(t) = \dot{\mathbf{q}}(t) - \dot{\mathbf{q}}_d(t) \tag{34}$$

where $q_d(t)$ is the desired position signal.

From (32), we can see that if $s(t) = 0$, then $\lambda e(t) + \dot{e}(t) = 0$, and we can get $e(t) = e(0)\exp(-\lambda t)$. That is, when $t \rightarrow \infty$,

position tracking error and speed tracking error will tend to zero exponentially with λ value.

$$\begin{aligned} \dot{s}(t) &= \lambda \dot{e}(t) + \ddot{e}(t) \\ &= \lambda \dot{e}(t) + \ddot{\mathbf{q}}(t) - \ddot{\mathbf{q}}_d(t) \\ &= \lambda \dot{e}(t) + \frac{1}{M_{(q)}}(u_{(t)} - V_{(q,\dot{q})}\dot{\mathbf{q}} - \mathbf{G}_{(q)}) - \ddot{\mathbf{q}}_d(t) \end{aligned} \tag{35}$$

and,

$$s\dot{s} = s \left(\lambda \dot{e}(t) + \frac{1}{M_{(q)}}(u_{(t)} - V_{(q,\dot{q})}\dot{\mathbf{q}} - \mathbf{G}_{(q)}) - \ddot{\mathbf{q}}_d(t) \right) \tag{36}$$

To guarantee $s\dot{s} < 0$, and avoiding Chattering phenomenon due to high frequency switching of SMC, the proposed control signal designed as:

$$\tau_{(smc)} = (M_{(q)}\ddot{\mathbf{q}}_d(t) + V_{(q,\dot{q})}\dot{\mathbf{q}} + \mathbf{G}_{(q)}) - M_{(q)}\lambda \dot{e}(t) - M_{(q)}\mu \text{sat}\left(\frac{s}{\delta}\right) \tag{37}$$

while $\left[M_{(q)}\mu \text{sat}\left(\frac{s}{\delta}\right) \right]$, is a term responsible for the boundary layer. here, $\mu \in \mathbb{R}^{4 \times 4}$ is the gain of the controller which is a diagonal matrix. Where,

$$\text{sat}(\mathbf{s}) = \begin{cases} \text{sgn}(\mathbf{s}), \text{ if } |\mathbf{s}| > \delta \\ \frac{\mathbf{s}}{\delta}, \text{ if else} \end{cases}, \text{sgn}(\mathbf{s}) = \begin{cases} 1, \text{ if } \mathbf{s} > 0 \\ 0, \text{ if } \mathbf{s} = 0 \\ -1, \text{ if } \mathbf{s} < 0 \end{cases}$$

while the value of δ provided by solving (38) online with model.

$$\delta = \mu - \lambda\delta \tag{38}$$

where, δ is the boundary layer in which once the state trajectory reaches this layer, then it remains there.

6 Validation

Our proposed multi-section continuum robot dynamic model was validated using previously published data from Amouri et al. [30]. Amouri et al. present a 2-DOF cable-driven continuum robot dynamic model using system kinetic and potential energy. A comparison study was conducted using the PID controller developed by Amouri et al. on our proposed dynamic model.

Table 2 displays the PID controller constants (k_p, k_i , and k_d) chosen by Amouri et al. Figure 8 illustrates a comparison between the dynamic responses for the bending angle (θ_i) developed by our model and those of Amouri et al. Figure 8a, extracted using a data export program. The comparison reveals only slight differences between the two outputs, confirming the validity of our proposed model. Additionally, we compared (φ_i) fluctuation response between our proposed work and that of Amouri

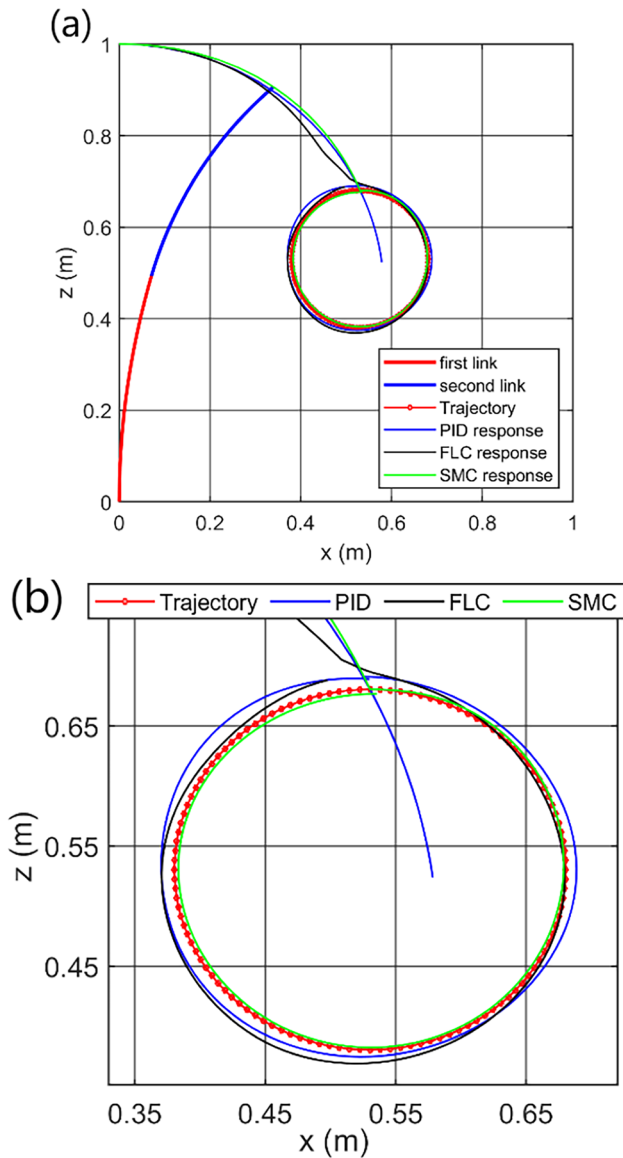
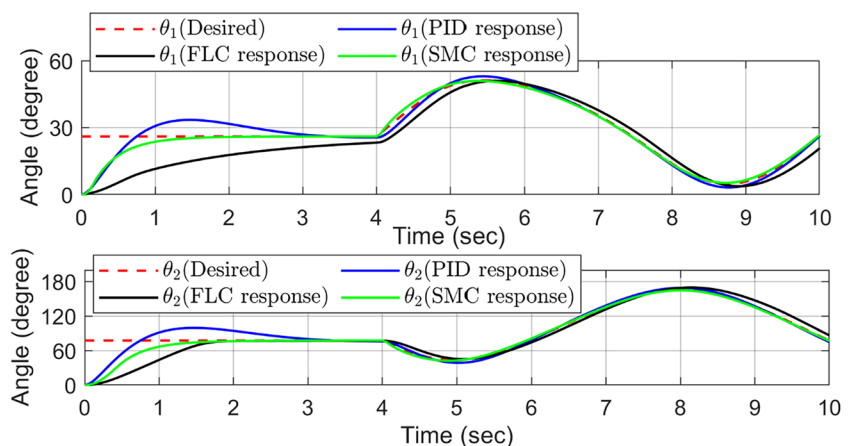


Fig. 14 The tracking performance of different controllers for in plane circular path, **a** Full trajectory response with CR PBs, **b** Magnified response for the circular trajectory

Fig. 15 Dynamic response for 2D circular trajectory



et al. Figure 8b. The response comparison demonstrates similar fluctuations between the two systems, further validating our proposed model.

7 Simulation and Results

In order to simulate the continuum manipulator behavior with two bending sections, the dynamics model has been implemented using MATLAB 20201b. The material and geometric properties of the continuum manipulator under consideration are given in Table 3.

7.1 Step Response

The following control strategies try to move the two-section CR into the desired configuration space $q(d)$, which is equivalent to the intended task space $0_{(d)} = [0^0_{1(d)} 0^0_{2(d)}]^T$, in order to evaluate each controller's reliability and effectiveness.

The initial configuration space values are chosen to be $q = [0.001^0 0^0 0.001^0 0^0]^T$ to avoid its singularity. While the desired configuration space values are $q_{(d)} = [80^0 90^0 110^0 180^0]^T$ which are acting as step input. For the PID controller, the values of k_p, k_d , and k_i are 16, 8.5, and 17, respectively, which is perfectly designed to deliver a satisfying response, as it produce 28.4% overshoot, 0.45 s rise time and 2.85 s settling time, which represents a significant improvement above earlier studies.

For the FLC, Fig. 6c describes the membership function of the FLC output $\hat{\theta}_{i(j)}$ and $\hat{\phi}_{i(j)}$, while Fig. 6a and b describe the membership function of the FLC input.

For SMC, the values for designed $\lambda = 3I_{4 \times 4}$, and $\mu = 9I_{4 \times 4}$. The saturation function limit δ , are provided by solving (38) online with simulation. Where $I_{4 \times 4}$ is the identity matrix that belongs to $R^{4 \times 4}$.

Figures 9, 10, 11 and 12 show complete dynamic responses for predesigned controllers, including error $e_{(t)}$ and

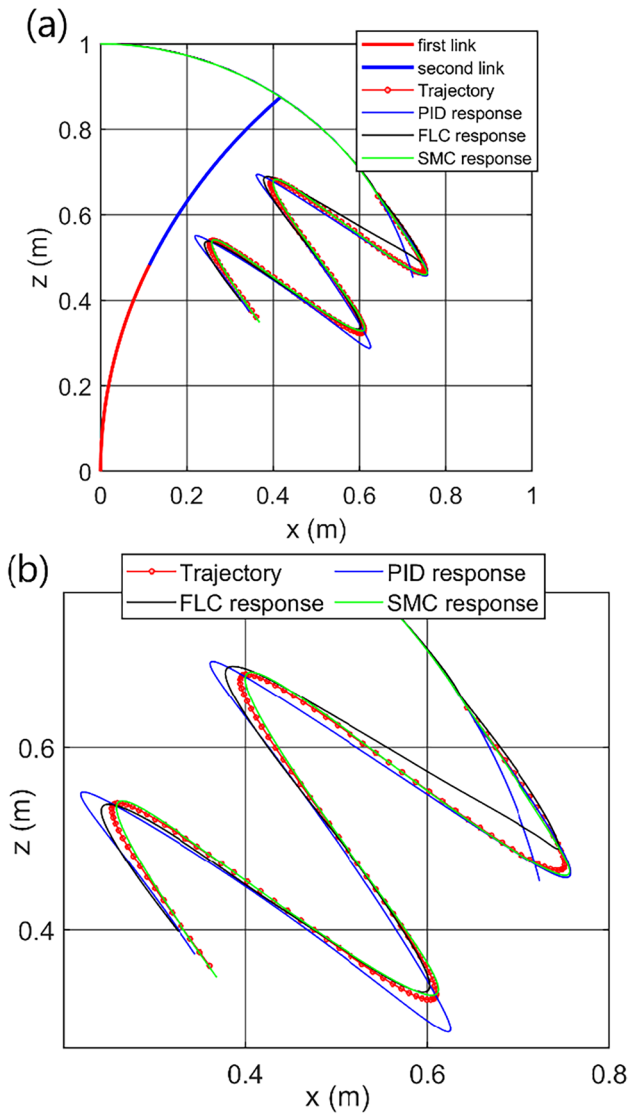
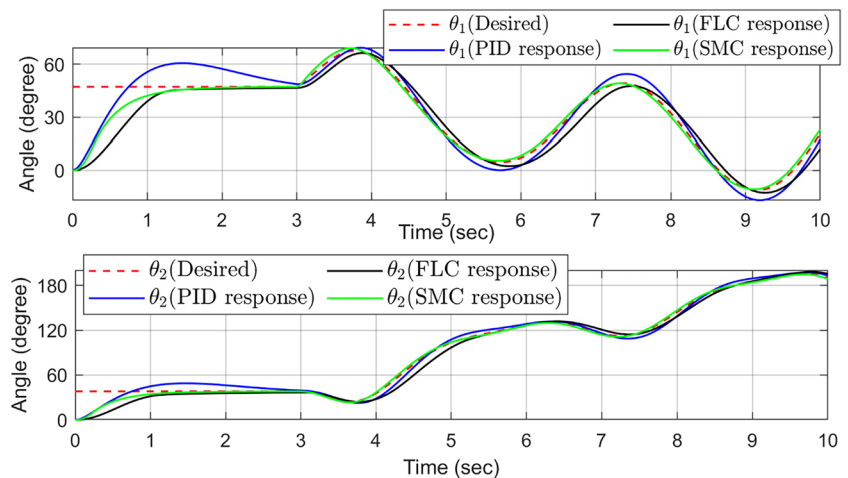


Fig. 16 The tracking performance of different controllers for in plane sinusoidal path, **a** Full trajectory response with CR PBs, **b** Magnified response for the sinusoidal trajectory

Fig. 17 Dynamic response for the 2D sinusoidal trajectory



change of error $\dot{e}(t)$ and SMC sliding surface Fig. 13. Table 4 shows the values of overshoot, rise-time, and settling-time.

Where Fig. 9 reveals that the PID controller with designed gain values, despite the exceptional settling-time t_s achieved, the PID produces a relatively high overshoot followed by a small and short oscillation, which is caused by the lowest rising time t_r of the controller, in its endeavors to achieve zero steady-state error. Due to the overshoot and oscillation, we must emphasize that using the PID in real-time applications is not the ideal choice.

As opposed to that, the FLC shows no overshoot and oscillation. And produces an acceptable rise time t_r , while achieving zero steady-state error.

The robust nonlinear sliding mode controller shows an astounding response. where it produces zero overshoot and no oscillation, while achieving the lowest rising time and zero steady-state errors.

7.2 Trajectory Tracking Response

This subsection covers the inverse dynamic responses of the designed controller for a given trajectory. To this point, three simulation examples are performed in two and three-dimensional spaces, tracking a circular and sinusoidal trajectory in a 2D plane and a conical helix in a 3D plane.

The first example is a circular trajectory in the $x-z$ plan, where CR tracks a perimeter of a circular path with a diameter of 50 mm in 6 s in the curvature planes $\varphi_1 = \varphi_2 = 0$.

Figure 14a shows the different controller responses for tracking the circular trajectory. Figure 14b represents a magnified view, where PID shows an overshoot from the desired set point (start point for the circular path), where FLC shows more accurate tracking for trajectory and SMC achieves the efficient tracking with minimal tracking error.

Figure 15 shows the response of the configuration space parameter $\mathbf{q} = [\theta_1 \varphi_1 \theta_2 \varphi_2]^T$ for each controller.

For the second path, a sinusoidal trajectory consisting of two full waves with an inclined angle of 45° from the x-axis in the x-z plane, where:

$$\begin{aligned} \begin{bmatrix} x \\ y \end{bmatrix} &= \frac{\begin{bmatrix} \cos(45^\circ) & -\sin(45^\circ) \\ \sin(45^\circ) & \cos(45^\circ) \end{bmatrix}}{\text{rotation matrix}} \frac{\begin{bmatrix} X \\ Y(x) \end{bmatrix}}{\text{point on graph}} \\ &= \frac{\begin{bmatrix} \cos(45^\circ)t - \sin(45^\circ)\sin(t) \\ \sin(45^\circ)t + \cos(45^\circ)\sin(t) \end{bmatrix}}{\text{point on rotated graph}} \end{aligned} \tag{39}$$

$$\begin{aligned} x &= \cos(45^\circ)X - \sin(45^\circ)Y(x) \\ y &= \sin(45^\circ)X + \cos(45^\circ)Y(x) \end{aligned} \tag{40}$$

Figure 16a shows the different controller responses for tracking the sinusoidal trajectory, where Fig. 16b represents a magnified view, where SMC shows great tracking for the trajectory with an error of less than 0.4 mm.

Figure 17 Shows the configuration space response for sinusoidal trajectory.

For the three-dimensional space, a conical helix trajectory was simulated for 12 seconds path.

Figures 18 and 19 shows the dynamic response for different controller for conical helix trajectory, where it reveals that the PID controller response suffers from a relatively high overshoot of 28.4% which leads to a high peak error; however, it rapidly handled the error, due to its lower settling time. The FLC provides a more precise response, with no overshoot and a reduced settling time of 1.56 s, resulting in an accurate tracking. The SMC shows the most responsive and effective controller, where it produces a zero percent overshoot, a rising time of less than 0.59 s, a settling time of less than 1.4 s the lowest among them all, and zero steady-state error, which leads to the most accurate tracking of trajectories.

Using the data developed during MATLAB Simulink simulation, which constituted from solving the kinematics and dynamics for each and every part and component of CR, a highly accurate dynamic animated simulation program developed using MATLAB environment, where it used to present a three-dimensional space simulation for each previously presented trajectory, plotting every configuration space variable along with CR dynamics and its workspace as shown in Fig. 20.

(Online Resource 1) represent created animated simulation for each and every trajectory, where developed MATLAB application Hyper-redundant Continuum robot simulator (HRCR simulator) was created to facilitate the simulation and dynamics plotting graphs.

8 Conclusion

For the two-link hyper-redundant continuum manipulator consisting of a primary backbone, a secondary backbone, and a series of disks, a kinematic model is derived from the piece-wise constant curvature assumption, and its dynamic model is derived using the Euler Lagrange representation. Three control strategies (inverse dynamic PID, intelligent FLC, and nonlinear SMC) were designed and applied for the dynamic model of the system in order to control the position and orientation of the CR for desired configuration values or along a given trajectory.

The inverse dynamics control rule, in particular, changes the closed-loop system into a new second-order linear system, while the fuzzy logic controller converts human experience and a thorough understanding of system dynamics into an intelligent, Robust controller, whereas the sliding mode control employs a discontinuous controller to compel the system to slide on the hypersurface with stability. All three close-loop dynamic control systems were implemented using MATLAB Simulink environment.

The first control strategy PID, which depends on its gains, is able to track the reference configuration space with high acceleration leading to a low rising time, which causes a relatively high overshoot followed by a small oscillation around desired configuration space values. Despite the fact that inverse dynamics control transforms the system into a decoupled linear second-order system where several linear control strategies can be investigated further to improve performance, the inverse dynamics calculation adds computational complexity and delay, which may affect the robot response in real-time applications. The second control strategy, FLC, on the other hand, eliminates both overshoot and oscillation, resulting in a relatively longer rising time. The third control strategy, SMC, achieves the best performance among them all, where it tracks the reference configuration space trajectory by an acceleration depending on the error

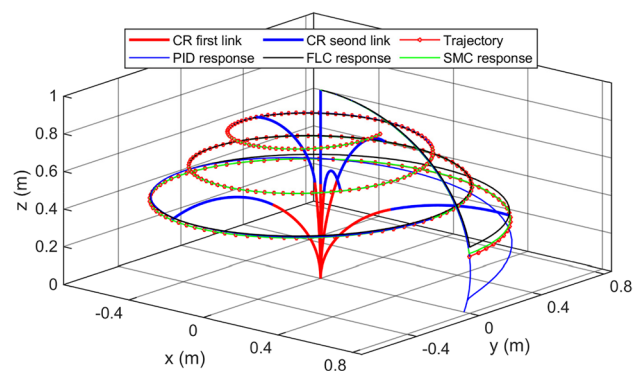
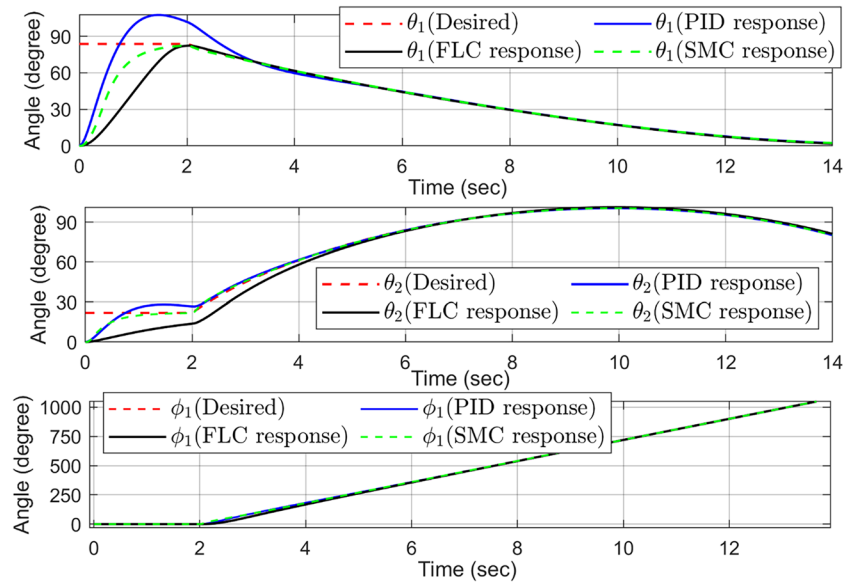


Fig. 18 Tracking performance of different controllers for 3D conical helix

Fig. 19 Dynamic response for conical helix trajectory, where $\varphi_2=0$



signal, which leads to a low rise time, no oscillation, and zero % overshoot, resulting in the most accurate trajectory tracking performance. By switching the system between two states, it gives significant robustness to the system. This provides the system with a significant advantage in dealing with uncertainty and disturbance, which can be demonstrated in further research.

Future work focuses on designing and implementing a disturbance observer for handling model uncertainty and external disturbances. Experiments will be carried out using the specified dynamic control techniques and the disturbance observer.

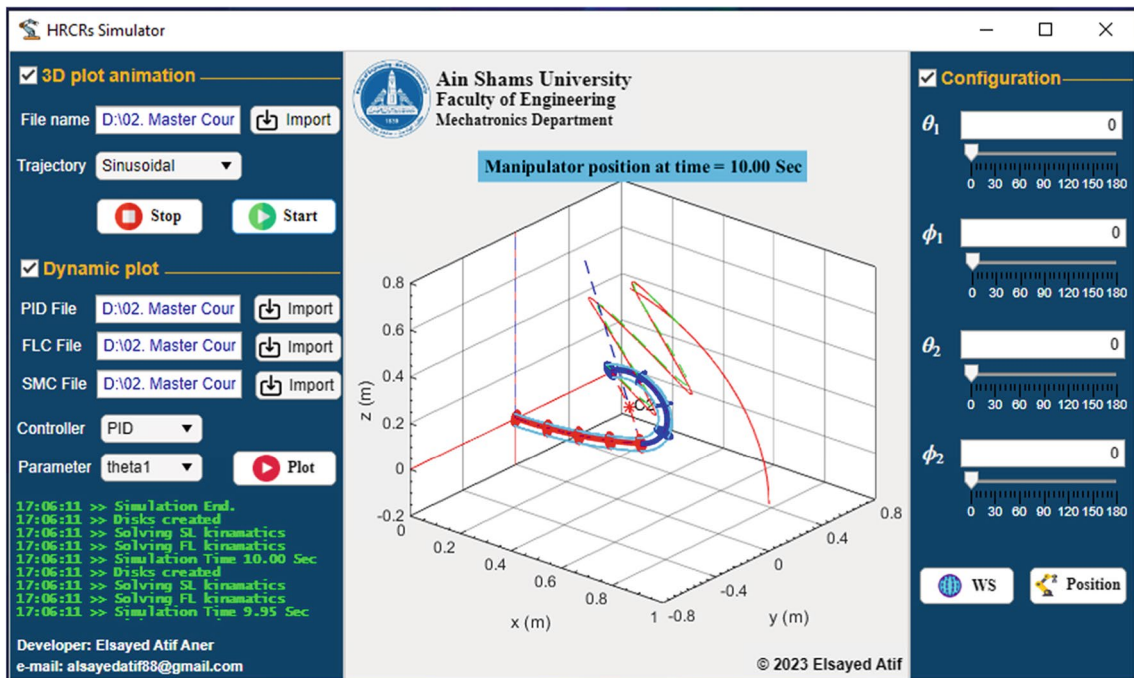


Fig. 20 HRCRs application for animating and simulation of whole CR for different trajectory

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1007/s10846-023-01896-1>.

Acknowledgements Not applicable.

Author Contributions [Elsayed Atif Aner] conceptualized the proposed system and is the author of the proposed controller design and control architecture, develop and perform the simulation, program the designed application. Data analyses were performed by [Professor Mohamed Ibrahim Awad] and [Dr. Omar Mahmoud Shehata]. The first draft of the manuscript was written by [Elsayed Atif Aner], where [Professor Mohamed Ibrahim Awad] and [Dr. Omar Mahmoud Shehata] analyzes and revises previous versions of the manuscript. All authors read and approved the final manuscript.

Funding Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB). The Authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Data Availability The presented software and simulation analysis conducted using MATLAB Simulink, where it is available from the corresponding author, [Elsayed Atif Aner], upon reasonable request.

Declarations

Ethics Approval Not applicable.

As we have not involved human or animal subjects in our research.

Consent to Participate Not applicable.

As we have not involved human subjects in our research.

Consent to Publish Not applicable.

As we have not involved human subjects in our research.

Competing Interests The authors have no relevant financial or non-financial interests to disclose.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Trivedi, D., Rahn, C.D., Kier W.M., Walker, I.D.: ("Printflatables | Proceedings of the 2017 CHI Conference on Human ...") "Soft robotics: Biological inspiration, state of the art, and future research." ("Soft robotics: Biological inspiration, state of the art, and future ...") *Appl. Bionics. Biomech.* **5**(3), 99–117 (2008)
- Chirikjian, G.S., Burdick, J.W.: A modal approach to hyper-redundant manipulator kinematics. *IEEE Trans Robot Autom.* **10**(3), 343–354 (1994)
- Shi, C., Luo, X., Qi, P., Li, T., Song, S., Najdovski, Z., Fukuda, T., Hongliang, R.: Shape sensing techniques for continuum robots in minimally invasive surgery: A survey. ("Shape Sensing Techniques for Continuum Robots in Minimally Invasive ..."). *IEEE Trans. Biomed. Eng.* **64**(8), 1665–1678 (2016)
- Burgner-Kahrs, J.D., Caleb, R., Howie, C.: Continuum robots for medical applications: A survey. *IEEE Trans. Robot.* **31**(6), 1261–1280 (2015)
- Amouri A., Mahfoudi, C., Zaatri, A.: Dynamic modeling of a spatial cable-driven continuum robot using euler-lagrange method. ("Dynamic Modeling of a Spatial Cable Driven Continuum Robot Using ... - CORE"). *Int. J. Eng. Technol. Inno.* **10**(1), 60 (2019)
- Kang R., Guo, Y., Chen, L., Branson, D.T., Dai, J.S.: Design of a pneumatic muscle based continuum robot with embedded tendons." ("Design of a Pneumatic Muscle Based Continuum Robot ... - Semantic Scholar"). *IEEE/ASME Trans. Mechat.* **22**(2), 751–761 (2016)
- Kim Y., Jaydev P.D.: Design and kinematic analysis of a neurosurgical spring-based continuum robot using SMA spring actuators. ("Publications - gatech.edu") In 2015 IEEE/RSJ Int. Conf. Intell. Robots. Syst. (IROS). 1428–1433 (2015).
- Chirikjian, G.S.: Hyper-redundant manipulator dynamics: A continuum approximation. *Adv. Robot.* **9**(3), 217–243 (1994)
- Rucker, D.C., Webster, R.J.: Parsimonious evaluation of concentric-tube continuum robot equilibrium conformation. *IEEE Trans. Biomed. Eng.* **56**(9), 2308–2311 (2009)
- He, B., Wang, Z., Li, Q., Xie, H., Shen, R.: An analytic method for the kinematics and dynamics of a multiple-backbone continuum robot." ("An Analytic Method for the Kinematics and Dynamics of a Multiple ..."). *Int. J. Ad. Robot. Syst.* **10**(1), 84 (2013)
- Hannan, M. W., and Walker I. D.: Novel kinematics for continuum robots." In *Advances in Robot Kinematics*, pp. 227–238. Springer, Dordrecht. (2009)
- Ding, L., Niu, L., Yang, Su., Yang, H., Liu, G., Gao, H., Deng, Z.: Dynamic finite element modeling and simulation of soft robots. *Chin. J. Mech. Eng.* **35**(1), 24 (2022)
- Yang, Y., Han, J., Liu, Z., Zhao, Z., Hong, K.-S.: Modeling and Adaptive Neural Network Control for a Soft Robotic Arm with Prescribed Motion Constraints. *IEEE/CAA J. Autom. Sinica* **10**(2), 501–511 (2023)
- Kazemipour A., Fischer O., Toshimitsu Y., Wong K.W., Katschmann R.K.: Adaptive dynamic sliding mode control of soft continuum manipulators. In 2022 Int. Conf. Robot Autom. (ICRA), pp. 3259–3265. IEEE (2022)
- GeorgeThuruthel, T., Renda, F., Iida, I.: First-order dynamic modeling and control of soft robots. *Front. Robot. AI* **7**, 95 (2020)
- Soriano, L.A., Zamora, E., Vazquez-Nicolas, J.M., Hernández, G., Madrigal, J.A.B., Balderas, D.: ("José Antonio Barraza Madrigal | PubFacts") "PD control compensation based on a cascade neural network applied to a robot manipulator." ("Frontiers | PD Control Compensation Based on a Cascade Neural Network ..."). *Front. Neurobot.* **14**, 577749 (2020)
- Lughofer E., Skrjanc I.: Evolving error feedback fuzzy model for improved robustness under measurement noise. ("Evolving Error Feedback Fuzzy Model for Improved Robustness under ...") *IEEE Trans. Fuzzy. Syst.* (2022)
- Meda-Campaña, J.A., Rodríguez-Manzanarez, R.A., Ontiveros-Paredes, S.D., de Jesus Rubio, J., Tapia-Herrera, R., Hernández-Cortés, T., Obregón-Pulido, G., Aguilar-Ibáñez, C.: An algebraic fuzzy pole placement approach to stabilize nonlinear mechanical systems." ("An Algebraic Fuzzy Pole Placement Approach to Stabilize Nonlinear ..."). *IEEE Trans. Fuzzy Syst.* **30**(8), 3322–3332 (2021)
- De Jesús Rubio, J., Orozco, E., Cordova, D.A., Islas, M.A., Pacheco, J., Gutierrez, J.G., Zacarias, A., Soriano, L.A., Meda-Campaña, J.A., Mujica-Vargas, D.: Modified linear technique for the controllability and observability of robotic arms. ("PDF)

- Modified Linear Technique for the Controllability and ..."). IEEE Access **10**, 3366–3377 (2022)
20. Balcazar, R., Rubio de Jesús, J., Orozco E., Cordova, D.A., Ochoa, G., Garcia, E., Pacheco, J., Gutierrez, G.J., Mujica-Vargas, D., Aguilar-Ibañez, C.: The regulation of an electric oven and an inverted pendulum." ("The Regulation of an Electric Oven and an Inverted Pendulum"). *Symmetry* **14**(4), 759 (2022)
 21. Webster III, R.J., Jones, B.A.: Design and kinematic modeling of constant curvature continuum robots: A review." ("[PDF] Design and Kinematic Modeling of Constant Curvature Continuum ..."). *The Int. J. Robot. Res.* **29**(13), 1661–1683 (2010)
 22. Neppalli, S., Csencsits, M.A., Jones, B.A., Walker, I.: A geometrical approach to inverse kinematics for continuum manipulators. In 2008 IEEE/RSJ Int. Conf. Intell. Robots. Syst. pp. 3565–3570. IEEE, (2008)
 23. Amouri, A., Mahfoudi, C., Zaatari, A., Lakhali, O., Merzouki, R.: A metaheuristic approach to solve inverse kinematics of continuum manipulators. ("A metaheuristic approach to solve inverse kinematics of continuum ..."). *Proc. Inst. Mech. Eng. Part I: J. Syst. Control Eng.* **231**(5), 380–394 (2017)
 24. Hannan, M.W., Walker, I.D.: Kinematics and the implementation of an elephant's trunk manipulator and other continuum style robots. ("Kinematics and the Implementation of an Elephant's Trunk Manipulator ..."). *J. Robot. Syst.* **20**(2), 45–63 (2003)
 25. Neppalli, S., Jones, B.A.: Design, construction, and analysis of a continuum robot." In 2007 IEEE/RSJ Int. Conf. Intell. Robots. Syst. pp. 1503–1507. IEEE, 2007
 26. Moussa, O.A., Mira, M.A., Fahmy, A.H., Morgan, E.I., Shehata, O.M.: Behavioral Assessment of Various Control Laws Formulations for Position Tracking of Multi-sectioning Modeled Continuum Robots. 2019 7th Int Conf. Contrl. Mech. Autom. (ICCA). pp. 191–196 2019. <https://doi.org/10.1109/ICCA46720.2019.8988780>.
 27. Amouri, A., Zaatari, A., Mahfoudi, C.: Dynamic modeling of a class of continuum manipulators in fixed orientation. ("Dynamic Modeling of a Class of Continuum Manipulators in Fixed Orientation"). *J. Intell. Robot. Syst.* **91**(3), 413–424 (2018)
 28. Tutsoy, O., Barkana, D.E., Colak, S.: Learning to balance an NAO robot using reinforcement learning with symbolic inverse kinematic. *Trans. Ins. Meas. Control.* **39**(11), 1735–1748 (2017)
 29. Tutsoy, O.: CPG based RL algorithm learns to control of a humanoid robot leg. *Int. J. Robot. Autom.* **30**(2), 1–7 (2015)
 30. Amouri, A., Mahfoudi, C., Zaatari, A.: Dynamic modeling of a spatial cable-driven continuum robot using euler-lagrange method. *Int. J. Eng. Technol. Innov.* **10**(1), 60 (2019)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Elsayed Atif Aner received his B.Sc. degree in Mechatronics and Robotics Engineering with High Honors from the Faculty of Engineering at the Egyptian Russian University (ERU) in 2017. His bachelor's focused on the control of a 6 DOF robotic arm. He is currently pursuing an M.Sc. degree in the Mechatronics Department at Ain Shams University in Egypt, where his research is focused on modeling and controlling a soft robotic arm. Since 2017, he has held the position of Teaching Assistant at the Egyptian Russian University, where he teaches various topics in Mechatronics Engineering and Robotics.

Mohammed Ibrahim Awad received his B.Sc. and M.Sc. degrees in mechanical engineering from the Ain Shams University, Egypt, and his Ph.D. degree in mechatronics from University of Leeds, U.K., in 2012. Then, he worked as a research fellow between 2013-2016 at Institute of Design, Robotics and Optimization (IDRO), University of Leeds, UK. He is currently the head of mechatronics engineering department at Ain Shams University. He has been PI, Co-PI, and team member in many research projects related to mechatronics, robotics and biomechatronic. His research activities in mechatronic systems design and robotic development and control in industrial and healthcare applications.

Omar M. Shehata received his first M.Sc. degree in Mechatronics Engineering in the field of multi-cooperative systems, Ain Shams University (2014). His second M.Sc. degree was completed in the field of micro-robotics used in medical applications from The German University in Cairo (2015). His Ph.D. was finished in the field of Intelligent transportation Systems specializing in Autonomous Vehicles specialization from Ain Shams University (2018). Additionally, he holds MBA degree specialized in the strategic planning and marketing majors from The German University in Cairo (2019). Dr. Shehata is currently an Assistant Professor at the Mechatronics engineering department. His current research directions include robotics, multi-cooperative systems, and evolutionary optimization techniques.