Robotica (2026), **44**, pp. 518–539
doi:10.1017/S026357472610318X

CAMBRIDGE UNIVERSITY PRESS

RESEARCH ARTICLE

Trajectory tracking of multi-section tendon-driven continuum robots using virtual actuation space control

Md Modassir Firdaus[1], Shail Jadav[2] and Madhu Vadali[1]

[1]Mechanical Engineering, Indian Institute of Technology Gandhinagar, India
[2]Technische Universitat Wien, Austria
Corresponding author: Madhu Vadali; Email: madhu.vadali@iitgn.ac.in

Received: 16 May 2025; **Revised:** 3 November 2025; **Accepted:** 10 January 2026; **First published online:** 27 February 2026

Keywords: tendon-driven continuum robots; trajectory tracking; compliant robot; tendon mapping

Abstract

Tendon-driven Continuum Robots (TDCRs) excel at operating in confined and complex environments due to their inherent flexibility and high degrees of freedom. Despite these advantages, achieving precise control in multi-section TDCRs remains challenging because of their infinite degrees of freedom, nonlinear behaviour, and coupled tendon actuation. This paper introduces a generalised virtual actuation space framework that significantly simplifies kinematic modelling and control of multi-section TDCRs. By formulating a novel concept of virtual actuation space that converts each bending section of the TDCR two-input actuation system, the proposed method reduces the complexity of TDCR modelling and control, enabling unique and simplified representations of each section. Furthermore, the formulation decouples each section via a generalised tendon mapping matrix that handles coupling effects. Experimental validation on a two-section TDCR demonstrates tracking performance with a root mean square error below 3 mm, corresponding to approximately 1% of the robot's length. These results confirm the method's efficacy in improving TDCR precision for minimally invasive surgery, industrial applications, and other constrained environments.

Nomenclature

Symbol	Description
a_i, b_i, c_i	Actuating tendons for the i-th section
γ_i	Angle of tendon a_i relative to the Y_i axis of the section base disk
L_i	Backbone length of the i-th section
r_d	Radius of each disk per section
$r_{a_i}, r_{b_i}, r_{c_i}$	Radii of the circles on which the interaction points of tendons a_i, b_i, c_i move during disk rotation
$\mathbf{l}$	Tendon space vector, dimensions $3n \times 1$
$l_{a_i}, l_{b_i}, l_{c_i}$	Lengths of the physical tendons for the i-th section
$\mathbf{l_v}$	Vector of virtual tendon lengths, dimensions $n \times 1$
l_{v_i}	Length of the virtual tendon for the i-th section
$\boldsymbol{\alpha}$	Vector of bending plane angles, dimensions $n \times 1$
α_i	Bending plane angle for the i-th section (angle with respect to Y_i axis)
$\mathbf{v}$	Virtual actuation space vector, dimensions $2n \times 1$, combining $\mathbf{l_v}$ and $\boldsymbol{\alpha}$
R_{ci}	Radius of curvature for the i-th section
θ_i	Bending angle for the i-th section
x_i, y_i, z_i	Local tip coordinates of the i-th section in its body frame

© The Author(s), 2026. Published by Cambridge University Press.

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

Symbol	Description
$\boldsymbol{\omega}$	Unit normal vector to the bending plane of the i-th section
$\mathbf{R}_{i-1}^{i}$	Rotation matrix from frame $i-1$ to frame i
$\hat{\boldsymbol{\omega}}$	Skew-symmetric matrix representation of $\boldsymbol{\omega}$
$\mathbf{H}_{i-1}^{i}$	Homogeneous transformation matrix from frame $i-1$ to frame i
$\mathbf{O}_{i-1}^{i}$	Translation vector from frame $i-1$ to frame i
$\boldsymbol{\zeta}_i$	Global coordinates of the tip (free end) of the i-th section
$\boldsymbol{\zeta}$	Vector of all generalised coordinates for n sections, dimensions $3n \times 1$
$\mathbf{T}$	Generalised tendon mapping matrix, dimensions $3n \times n$
$\mathbf{T}_i$	Submatrix of $\mathbf{T}$ for the i-th section, dimensions 3×1 (for 3 tendons)
$\mathbf{O}_i$	Zero matrix block in $\mathbf{T}$, dimensions $3 \times (n-i)$
$\dot{\boldsymbol{\zeta}}, \ddot{\boldsymbol{\zeta}}$	First and second time derivatives (velocity and acceleration) of $\boldsymbol{\zeta}$
$\mathbf{J}_v$	Jacobian matrix relating virtual actuation space velocity to task space velocity, dimensions $3n \times 2n$
$\mathbf{H}_v$	Hessian matrix for second-order terms, dimensions $3n \times 2n$
Δt	Time step in discretization
$\dagger$	Moore-Penrose pseudoinverse
$\mathbf{e}$	Task space error, $\boldsymbol{\zeta}_d - \boldsymbol{\zeta}$
$\boldsymbol{\zeta}_d$	Desired generalised coordinates, dimensions $3n \times 1$
$\mathbf{K}_d$	Derivative gain matrix, dimensions $3n \times 3n$, positive definite diagonal
$\mathbf{K}_p$	Proportional gain matrix, dimensions $3n \times 3n$, positive definite diagonal
r_p	Radius of the motor pulley
$\boldsymbol{\Omega}$	Vector of motor pulley angles
m	Number of modules per section
$L_{m_i}^{j}$	Length of the j-th module in the i-th section
$\overline{\theta}_i$	Bending angle per module in the i-th section (θ_i/m)
$\mathbf{o}_i^{j-1}\mathbf{o}_i^{j}$	Vector connecting centres of consecutive disks $j-1$ and j in section i
$\mathbf{o}_i^{j}\mathbf{p}_i^{j}$	Vector from disk centre j to virtual tendon interaction point in section i
$\mathbf{p}_i^{j-1}\mathbf{p}_i^{j}$	Vector along the virtual tendon between interaction points $j-1$ and j in section i
$l_{v_i}^{j}$	Virtual tendon length change for the j-th module in the i-th section
$l_{a_i}^{j}, l_{b_i}^{j}, l_{c_i}^{j}$	Arch lengths (changes) for physical tendons in the j-th module of the i-th section
$\mathbf{R}_{j-1}^{j}$	Rotation matrix for the j-th module (using Rodrigues formula)

1. Introduction

Continuum robots (CRs) excel in navigating confined, cluttered environments, making them indispensable in minimally invasive surgeries, industrial operations, and space exploration [1–4]. Unlike rigid-link robots, CRs adapt to constraints, enabling operations in restricted spaces like turbine and aeroengine inspections [5–8]. CRs are classified by actuation systems, including tendon-driven [9], concentric tube [10], magnetically actuated [11], and shape memory alloy-based robots [12].

TDCRs offer distinct advantages over other types. Their compact, lightweight design and precise, high-DOF control make them ideal for space-constrained applications like surgeries [13, 14]. Unlike pneumatic and hydraulic systems, TDCRs rely on energy-efficient mechanical actuation [15], and their scalability supports microscale surgical robots and larger industrial tools [16]. Tendon actuation minimises backlash, ensuring smooth, accurate motion for tasks such as surgical suturing [17], while their simple, cost-effective construction using cables and pulleys enhances accessibility [18]. Additionally, TDCRs operate silently, an important advantage in hospital and research settings [19]. Advancing TDCR kinematics, dynamics, and control, which enable flexible backbone bending through tendon actuation, is critical for applications requiring precision and manoeuvrability in restricted environments. Recent

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

research has focused on advanced dynamic modelling and control strategies for tendon-driven continuum robots. Piecewise constant curvature (PCC) and Euler–Lagrange-based robust controllers such as inverse dynamic PID, fuzzy logic, and sliding mode control have been developed to ensure reliable trajectory tracking under varying loads and disturbances [20, 21]. Adaptive schemes employing function approximation have further enhanced robustness by compensating for model uncertainties and external perturbations [22]. More recently, data-driven and learning-based methods – including model-based nonlinear controllers [23], LSTM-based force estimation [24], and reinforcement learning approaches such as fuzzy RL and deep deterministic policy gradients [25, 26] – have demonstrated superior precision and stability compared to traditional PCC-based designs. Complementary sensing and feedback innovations – combining dual displacement sensors with on-board visual tracking [27] and model-free adaptive Kalman filters [28] – have further improved tip-tracking accuracy and path consistency while streamlining sensor requirements.

Despite the progress in control and sensing, modelling tendon-driven continuum robots remains challenging due to their infinite degrees of freedom and non-linear material behaviour [29]. To address these challenges, several frameworks have been proposed: the PCC and its variant variable PCC, finite element models [30], and Cosserat-rod formulations [31–33]. While finite element and Cosserat-rod approaches provide high fidelity, their computational demands limit real-time use [34]. In contrast, PCC-based kinematic models strike a practical balance, offering sufficient accuracy and the speed required for quasi-static control [13].

Beyond tip trajectory control, there is a growing need for shape and section trajectory control, especially in multi-section TDCRs deployed in unstructured environments. Each section requires at least three actuating tendons to achieve motion in 3D space, as tendons can only actuate under tension [14]. Figure 1 shows a schematic of a two-section TDCR with three actuating tendons for each section. Under the PCC assumption, one component of the tip coordinates depends on the other two [14], allowing the length of any two tendons to uniquely define the tip coordinate, with a constrained equation defining the length of the third tendon. If the number of tendons per section increased in a multi-section TDCR, it would increase the number of constrained equations, leading to modelling and control complexities in existing approaches.

This paper exploits these observations and introduces the concept of virtual actuation space to address such limitations. A generalised tendon mapping matrix, applicable to TDCRs with any number of sections and any number of actuating tendons per section, relates the virtual actuation space to the tendon lengths, enabling fully decoupled control of each section and simplifying multi-tendon, multi-section designs beyond the three-tendon PCC constraints of prior methods.

A generalised forward and inverse kinematics framework for multi-section TDCRs in virtual actuation space is formulated in this paper, maintaining scalability across varying robot configurations. It considers two inputs in virtual actuation space per section to uniquely define the configuration of the section. The generalised tendon mapping matrix captures the constrained and coupling effects of the tendons and their lengths, regardless of the total number of tendons or sections. Subsequently, control formulation in virtual actuation space for multi-section TDCRs is proposed, emphasising both section and trajectory control. The approach utilises a virtual actuation space to decouple the control of individual sections. The controller can effectively manage the complex relationship between task and tendon space, enabling control of TDCRs in confined spaces. The control methodology is validated experimentally on a multi-section TDCR, demonstrating its effectiveness in achieving accurate and reliable tip position control. The results indicate that the proposed approach can significantly improve the performance of TDCRs in confined and cluttered environments, paving the way for their broader adoption in medical and industrial applications.

The primary contributions of this study, which extend beyond established PCC-based methods by enabling scalable, decoupled shape and trajectory control for arbitrary multi-tendon/multi-section TDCRs, are as follows:

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

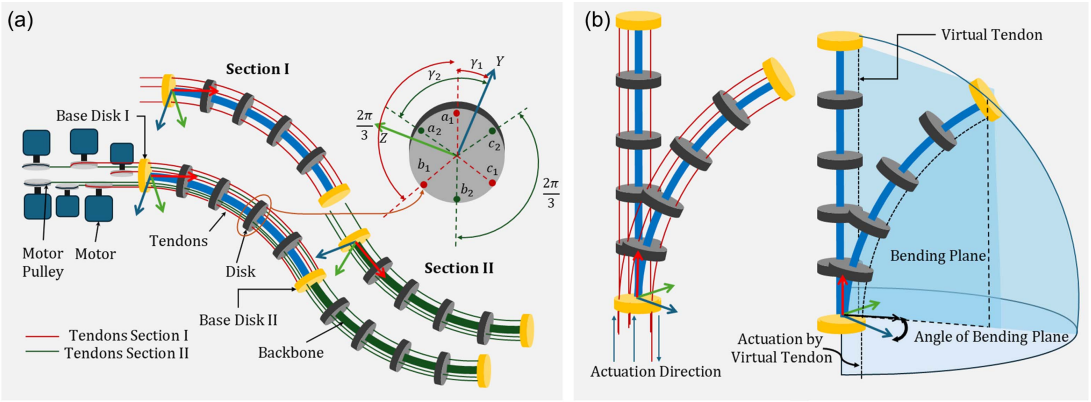


Figure 1. *Schematic of two-section tendon-driven TDCR. (a) Routing of physical (a_1, b_1, & c_1) for section I and (a_2, b_2, & c_2) for section II and distribution of tendons along the disk. (b) Conceptual representation of a virtual tendon (in a black dashed line) that lies in the bending plane (sky blue plane).*

1. A generalised virtual actuation space kinematics framework for tendon-driven continuum robot sections of any configuration, using two inputs per section to define 3D poses, overcoming the three-tendon constraints and modelling complexities of prior PCC approaches.
2. A generalised tendon mapping matrix from the virtual actuation space to the actual tendon space, enabling decoupled control of individual sections to follow any trajectory in their workspace (including shape control alongside tip trajectory adherence) and resolving inter-section couplings absent in existing methods.
3. A control formulation in the virtual actuation space for point-to-point and trajectory tracking tasks, incorporating task-space constraints to manage tendon complexities in confined environments.
4. Theoretical proofs of control convergence alongside experimental validation on a two-section TDCR, confirming accurate tip position control and enhanced reliability in cluttered settings.

The rest of the paper is organised as follows: an overview of multi-section TDCR and foundational assumptions for theoretical formulation is provided in Section 2. A brief description of virtual actuation space formulations is given in Section 3. Subsequently, generalised forward kinematics is presented in Section 4. Section 5 presents the concept of decoupling TDCRS and the generalised tendon mapping matrix. Section 6 describes the state estimation. The control architecture is provided in Section 7. The details of the experimental setup are given in Section 8. Hardware implementation, point-to-point tracking, trajectory tracking with section control and the detailed discussion are presented in Section 9. Section 10 provides final remarks and future work.

2. Multi-section TDCR

TDCRs, like the example shown in Figure 1, have a flexible backbone that allows continuous bending along their length and can be configured as a single-section or multi-section. In a multi-section design, each section has a flexible backbone, three actuating tendons, and evenly spaced disks. Each disk has a central hole for the backbone attachments and peripheral holes for tendons, with at least three tendons around each disk. The tendons, labelled a_i, b_i, and c_i for the i^{th} section, are attached to motor pulleys to adjust the length for bending. The other end of each tendon attaches to the intermediate disk between sections. Figure 1 shows a two-section TDCR. For Section I, the tendons end at the disk between Sections I and II, which serves as the free end of Section I and the base for Section II. The other tendons end at the free end of the two-section TDCR, as shown in Figure 1(a). Tendons in the same section are separated

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

by $\frac{2\pi}{3}$ radians, with γ_i representing the angle of tendon a_i relative to the Y_i axis of the section base disk. The backbone's elasticity allows it to return to its original shape when tendons are relaxed, and varying tendon lengths create different bending shapes. The theoretical model for the behaviour of tendon-driven continuum robots (TDCRs) is derived based on the following assumptions:

1. **Constant curvature approximation:** The pulling of tendons causes the TDCR to bend into a concave shape. Therefore, the kinematic model is based on the piecewise constant curvature framework [14].
2. **Backbone properties:** The backbone, which primarily determines the behaviour of the TDCR, is made of a single type of material and is considered isotropic and homogeneous.
3. **Tendon characteristics:** The tendons are assumed to be inextensible and frictionless. Given that the tendon diameter is on the order of 10^{-3} of the backbone diameter, the tendons are considered massless.
4. **Equal initial lengths:** The tendons are assumed to have equal initial lengths to prevent buckling.

While these assumptions simplify the theoretical framework, they do not fully capture the real-world behaviour of TDCRs. Therefore, a closed-loop control methodology is employed to manage and control the actual behaviour of the system accurately.

3. Conceptualising virtual actuation space

To understand the generalisation of virtual actuation space, we will first focus on one section of a multi-section TDCR. For articulation in a 3D workspace, each section needs at least three actuating tendons. Despite the complexity of 3D articulation, each section can be simplified to planar articulation within a specific *bending plane*, the plane in which a section bends. Figure 1(b) shows the representation of virtual tendons for a single section instead of three tendons to achieve the same results. For multiple sections of TDCR, a single tendon within each section's bending plane achieves this. Thus, any 3D articulation can be represented as a combination of planar movements of all sections, each controlled by a single *virtual tendon*, as shown by the dashed line in Figure 2. Changes in the lengths of physical tendons, or joint space, correlate directly with changes in virtual tendons, defining the *virtual actuation space*. The *tendon space* is given by,

$$\mathbf{l} = \begin{bmatrix} l_{a_1} & l_{b_1} & l_{c_1} & l_{a_2} & l_{b_2} & l_{c_2} & \cdots \end{bmatrix}^T_{3n\times 1} \tag{1}$$

where $\{l_{a_i}, l_{b_i}, l_{c_i}\} \in \mathbb{R}$. The *virtual actuation space* is given by,

$$\mathbf{l_v} = \begin{bmatrix} l_{v_1} & l_{v_2} & l_{v_3} & \cdots & l_{v_n} \end{bmatrix}^T_{n\times 1} \tag{2}$$

$$\boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 & \cdots & \alpha_n \end{bmatrix}^T_{n\times 1} \tag{3}$$

$$\mathbf{v} = \begin{bmatrix} \mathbf{l_v} & \boldsymbol{\alpha} \end{bmatrix}^T_{2n\times 1} \tag{4}$$

where $l_{v_i} \in \mathbb{R} \setminus \{0\}$ and $\alpha_i \in \mathbb{R}$, with l_{v_i} representing the length of the virtual tendons and α_i representing the angle of the bending plane of the sections. The angle made by the bending plane with the Y_i axis of the base frame of the i^{th} section is called the bending plane angle α_i. Let L_i represent the backbone length of the i^{th} section, and let r_d denote the radius of each of the k disks per section. Each section has a frame attached to the base disk, with the base frame $\{OXYZ\}_i$ and $\{OXYZ\}_{i+1}$ attached to the free end of the i^{th} section, which is the base frame for $i+1^{th}$, as shown in Figure 2. In this frame, the X axis is normal to the base disk surface, while the Y and Z axes follow the right-hand coordinate convention. The base disk of section I is grounded. The free end disk of the i^{th} section serves as the base disk for the $i+1^{th}$ section, except for the last section, where $i = 1, 2, \cdots, n$. Each section can bend in a different bending plane. The bending of all sections influences the position of the tip of the TDCR in the task

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

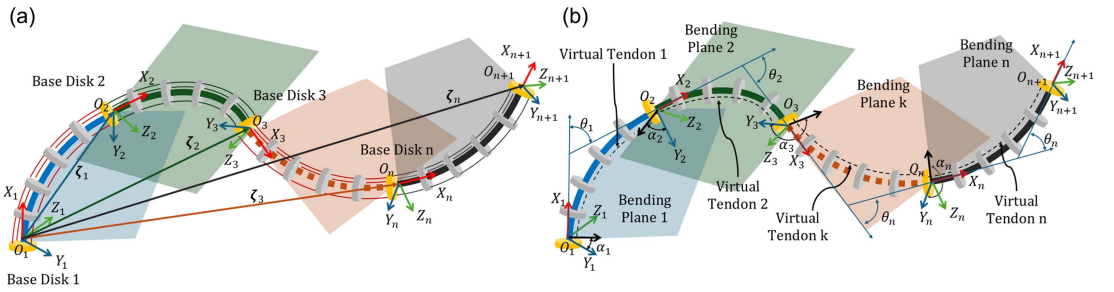


Figure 2. *Multi-section TDCR with a yellow base and free disks. (a) Each section is shown with its three actuating tendons and bending planes. (b) Virtual tendons on the concave side of each section in the respective bending plane.*

space, which is the free end of the n^{th} section. The unbent state of all sections is defined as the natural state of the TDCR.

4. Generalised forward kinematics in virtual actuation space

In the virtual actuation space of the i^{th} section, denoted by l_{v_i} and α_i, these parameters represent the change in length of the virtual tendon and the angle of the bending plane between the $i+1^{th}$ and i^{th} frames, as illustrated in Figure 2(b). Assuming constant curvature for each section, the relationship between the virtual actuation space, the TDCR bending angle, and the radius of curvature of the section is given by,

$$\begin{aligned} L_i - l_{v_i} &= \theta_i(R_{ci} - r_d) \\ L_i &= \theta_i R_{ci} \end{aligned} \tag{5}$$

where R_{ci} and θ_i are the radius of curvature and the bending angle, respectively. The relationship is further expressed as,

$$\frac{L_i}{R_{ci}} = \frac{L_i - l_{v_i}}{R_{ci} - r_d}; R_{ci} = \frac{L_i r_d}{l_{v_i}}; \theta_i = \frac{L_i}{R_{ci}} = \frac{l_{v_i}}{r_d} \tag{6}$$

The tip coordinates of the i^{th} section in the local frame are given by

$$\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \begin{bmatrix} R_{ci}\sin\theta_i \\ (R_{ci} - R_{ci}\cos\theta_i)\cos\alpha_i \\ (R_{ci} - R_{ci}\cos\theta_i)\sin\alpha_i \end{bmatrix} \tag{7}$$

where $[x_i\ y_i\ z_i]^T$ are the generalised coordinates of the i^{th} section in its body frame.

4.1. Coordinates transformation

The coordinates of each section can be transformed into the global frame using a homogeneous transformation matrix. The rotation matrix can be calculated using the Rodrigues rotation formula [35]. The angle of rotation is the bending angle of the section θ_i, and the axis of rotation is the normal vector of the bending plane. The normal vector of the i^{th} section, $\boldsymbol{\omega}$, is given by,

$$\boldsymbol{\omega} = \frac{\mathbf{o_i o_{i+1}} \times \mathbf{x_i}}{\|\mathbf{o_i o_{i+1}} \times \mathbf{x_i}\|} \tag{8}$$

The rotation matrix of the i^{th} frame relative to the $(i-1)^{th}$ frame is given by,

$$\mathbf{R}_{i-1}^{i} = e^{\hat{\boldsymbol{\omega}}_i \theta_i} = \mathbf{I} + \hat{\boldsymbol{\omega}}\sin\theta_i + \hat{\boldsymbol{\omega}}^2(1 - \cos\theta_i) \tag{9}$$

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

where $\mathbf{I}$ is the identity matrix of order three, $\mathbf{R}_{i-1}^{i}$ is an element of rotation group $\mathbb{SO}(3) \in \mathbb{R}^3$ and $\hat{\boldsymbol{\omega}}$ is an element of Lie algebra $\mathfrak{so}(3)$ which is a skew-symmetric matrix. The homogeneous transformation obtained by the translation and rotation from the $(i-1)^{th}$ frame to the i^{th} frame is defined as $\mathbf{H}_{i-1}^{i}$, given by,

$$\mathbf{H}_{i-1}^{i} = \begin{bmatrix} \mathbf{R}_{i-1}^{i} & \mathbf{O}_{i-1}^{i} \\ 0 & 1 \end{bmatrix} \tag{10}$$

where $\mathbf{O}_{i-1}^{i}$ is the translation vector of the i^{th} frame from the $(i-1)^{th}$ frame.

$$\boldsymbol{\zeta}_i = \mathbf{H}_1^2 \mathbf{H}_2^3 \ldots \mathbf{H}_{i-1}^{i} \begin{bmatrix} x_i & y_i & z_i \end{bmatrix}^T \tag{11}$$

where $\boldsymbol{\zeta}_i$ are the generalised coordinates of the tip of the i^{th} section in the global frame $\{OXYZ\}_1$, which are functions of l_{v_i} and α_i. Thus, the generalised coordinates of an n-section TDCR are given by,

$$\boldsymbol{\zeta} = \begin{bmatrix} \boldsymbol{\zeta}_1 & \boldsymbol{\zeta}_2 & \boldsymbol{\zeta}_3 & \cdots & \boldsymbol{\zeta}_n \end{bmatrix}_{3n\times 1}^{T} \tag{12}$$

where $\boldsymbol{\zeta} \in \mathbb{R}^{3n}$ represents the generalised coordinates of the TDCR section's free ends, which are functions of $\mathbf{v}$.

5. Section decoupling

Each section of a TDCR inherently possesses only two independent coordinates of its free end in the local body frame, even though three tendons are required to uniquely define its configuration through a constraint equation. As the number of tendons per section increases, the number of such constraint equations also increases, resulting in greater mathematical complexity in modelling the kinematics, Jacobian, and control laws.

To address this, the Virtual Actuation Space (VAS) framework simplifies the system by reducing it to a two-input, two-output structure per section, effectively decoupling the multiple sections of a TDCR. The tendon constraints and coupling effects arising from routing are handled through a tendon mapping matrix. Consequently, decoupling the TDCR into individual n-section units simplifies the overall control and analysis.

5.1. Utilising virtual actuation space inverse kinematics

Consider a n section TDCR with frame $\{OXYZ\}_1$ as the fixed frame shown in Figure 2(a). Inverse kinematics involves determining l_{v_i} and α_i from the given generalised coordinates of each section in global frame $\boldsymbol{\zeta}_i$, depicted in Figure 2(a), which describe the free end of each section in a global frame. The generalised coordinates are transferred into the local frame for each section. By transforming the problem into local coordinates, each section can be treated as an independent entity, thereby providing a straightforward approach to identifying the elements of the virtual actuation space.

The transformation of generalised coordinates into the local frame propagates from frame 1 to frame n. The transferred local coordinates $[x_i, y_i, z_i]^T$ using the homogeneous transformation matrix are given by,

$$[x_i \quad y_i \quad z_i]^T = \mathbf{H}_{i-1}^{0}(\boldsymbol{\zeta}_i - \boldsymbol{\zeta}_{i-1}) \tag{13}$$

To compute $\mathbf{H}_{i-1}^{0}$, the bending angle θ_i and the offset $\mathbf{O}_{i-1}^{i}$ are utilised for each section and can be obtained through straightforward trigonometric relations [36]. Since the coordinate of each section's free end is already established in its local frame using Eq. (13), the virtual joint space parameters l_{v_i} and α_i for the i^{th} section are determined.

5.2. Elements of virtual actuation space

For local frame coordinates from Eq. (13) can be used to determine l_{v_i} and $\alpha_i = atan2\left(\frac{z_i}{y_i}\right)$ for the i^{th} section of an n-section TDRC. Each TDRC section is divided into m modules, with each module

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

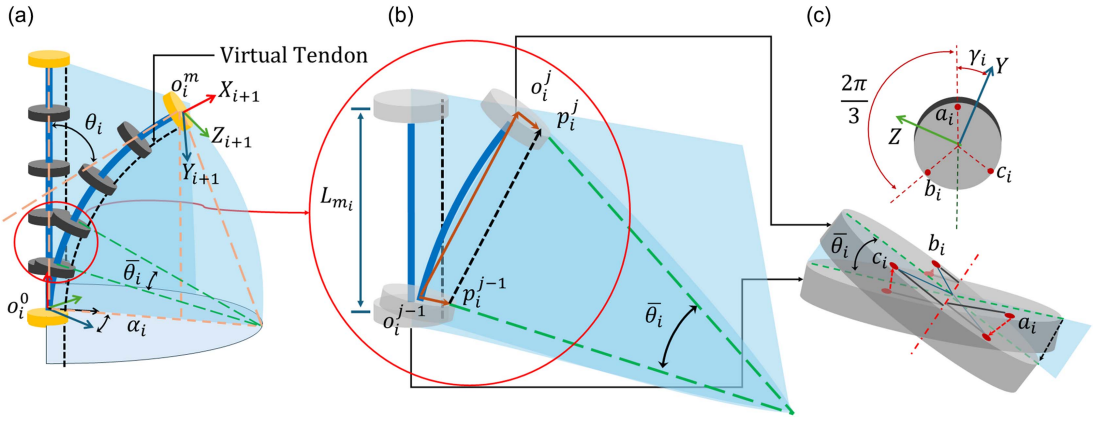


Figure 3. *(a) Virtual tendon ofi^{th}section of the muti-sections TDCR. (b) Details of a module with a module bending angle. (c) Angle made by the disks of the single module.*

spanning between two consecutive disks, as shown in Figure 3(b). The length of the j^{th} module is $L_{m_i}^j$, where j ranges from 1 to m. Each module contains part of the flexible backbone and a virtual tendon represented as the dashed line in Figure 3, and the disk thickness is assumed to be zero. Because the disks are equidistant, the bending angle of each module $\bar{\theta}_i$, is obtained by dividing θ_i by m. This angle also represents the rotation of the top disk of the module relative to the bottom disk about an axis perpendicular to the plane of bending and passing through the centre of the bottom disk, as shown in Figure 3(c). The vectors $\mathbf{o}_i^{j-1}\mathbf{o}_i^j$ (connecting centres of consecutive disks) and $\mathbf{o}_i^j\mathbf{p}_i^j$ (connecting the disk centre to the virtual tendon interaction point) are given by [36],

$$\begin{aligned}\mathbf{o}_i^{j-1}\mathbf{o}_i^j &= \frac{L_i}{\theta_i}\left[\sin\bar{\theta}_i \;\; \cos\alpha_i(1-\cos\bar{\theta}_i) \;\; \sin\alpha_i(1-\cos\bar{\theta}_i)\right]^T \\ \mathbf{o}_i^j\mathbf{p}_i^j &= [0,\; r_d\cos\alpha_i,\; r_d\sin\alpha_i]^T \end{aligned} \tag{14}$$

Using Eq. (14), the vector $\mathbf{p}_i^{j-1}\mathbf{p}_i^j$, which represents the vector of the virtual tendon in the bent state of the module, is given by,

$$\mathbf{p}_i^{j-1}\mathbf{p}_i^j = \mathbf{o}_i^{j-1}\mathbf{o}_i^j + [\mathbf{R}]_{j-1}^j\mathbf{o}_i^j\mathbf{p}_i^j - \mathbf{o}_i^{j-1}\mathbf{p}_i^{j-1} \tag{15}$$

where the $[\mathbf{R}]_{j-1}^j$ rotation matrix is found out by using Rodrigues rotation formula [35] with angle $\bar{\theta}_i$ and axis perpendicular to the plane of bending. The required length of the virtual tendon of the module is given by,

$$\begin{aligned} l_{v_i}^j &= \|\mathbf{p}_i^j\mathbf{p}_i^{j+1}\| - L_{m_i}^j \\ l_{v_i} &= \sum_{j=1}^{m} l_{v_i}^j \end{aligned} \tag{16}$$

where $l_{v_i}^j$ is the virtual tendon length of a module and l_{v_i} is of a section. Now that the l_{v_i} and α_i are known, the individual sections of the multi-section TDRC can be controlled individually. However, a virtual tendon is not a physical tendon; therefore, a mapping is required that can relate the virtual tendons to the actual TDCR tendons.

5.3. Generalised tendon mapping matrix

We will first focus on a single module and a section to understand the tendon mapping matrix. Then, the idea can be generalised for all sections of the TDCR. The bending angle of a module is equivalent to the angle made by two consecutive disks. It can be further understood as the rotation of one disk about

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

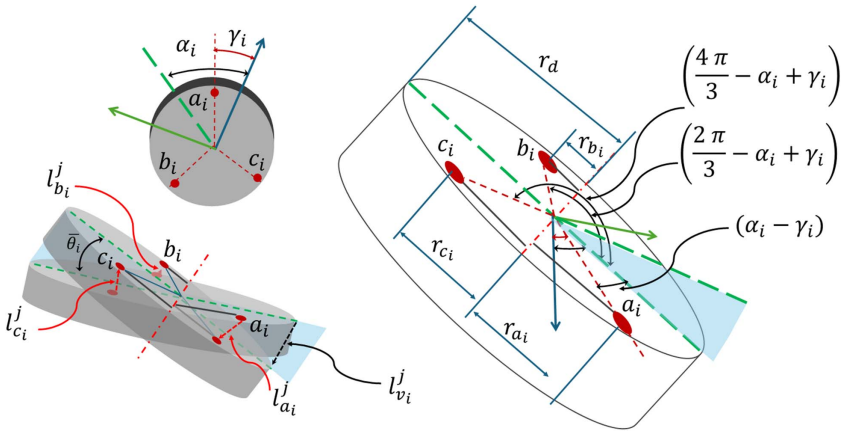


Figure 4. *Depiction of actuation space formalisationon a module basis from virtual space. The green dashed line lies in the bending plane.*

the axis perpendicular to the bending plane passing through the disk's centre, as depicted in Figure 4. When the disk rotates, the interaction point of actual tendons a_i, b_i, and c_i with the disks will move on the circle of radius r_{a_i}, r_{b_i}, and r_{c_i} around the axis of rotation. Thus, the arch $l^j_{a_i}$, $l^j_{b_i}$, and $l^j_{c_i}$ are the required changes in actuation space for a module of a section of TDCR. γ_i is the smallest angle of the line joining the disk's centre and the actual tendon's interaction point with one of the axes (Y-axis) of the body fixed frame. γ_i is a constant that describes the distribution of the tendons around the disks as shown in Figure 4. The arch $l^j_{a_i}$, $l^j_{b_i}$, and $l^j_{c_i}$ are given by,

$$\begin{aligned}
l^j_{a_i} &= r_{a_i}\bar{\theta}_i = \cos(\alpha_i - \gamma_i) r_d \bar{\theta}_i = \cos(\alpha_i - \gamma_i) l^j_{v_i} \\
l^j_{b_i} &= r_{b_i}\bar{\theta}_i = \cos\left(\frac{2\pi}{3} - \alpha_i + \gamma_i\right) r_d \bar{\theta}_i = \cos\left(\frac{2\pi}{3} - \alpha_i + \gamma_i\right) l^j_{v_i} \\
l^j_{c_i} &= r_{c_i}\bar{\theta}_i = \cos\left(\frac{4\pi}{3} - \alpha_i + \gamma_i\right) r_d \bar{\theta}_i = \cos\left(\frac{4\pi}{3} - \alpha_i + \gamma_i\right) l^j_{v_i}
\end{aligned} \tag{17}$$

From (16) and (17), a mapping matrix can be formulated for a single section with three actuating tendons as follows:

$$\begin{bmatrix} l_{a_i} & l_{b_i} & l_{c_i} \end{bmatrix}^T = \begin{bmatrix} \cos(\alpha_i - \gamma_i) \\ \cos\left(\frac{2\pi}{3} - \alpha_i + \gamma_i\right) \\ \cos\left(\frac{4\pi}{3} - \alpha_i + \gamma_i\right) \end{bmatrix} l_{v_i} \tag{18}$$

The mapping matrix in Eq. (18) has three elements for three actual tendons of the same section. If there is an increase in the number of tendons per section, say 4, the tendon mapping matrix is given by,

$$\begin{bmatrix} l_{a_i} & l_{b_i} & l_{c_i} & l_{d_i} \end{bmatrix}^T = \begin{bmatrix} \cos(\alpha_i - \gamma_i) \\ \cos\left(\frac{2\pi}{4} - \alpha_i + \gamma_i\right) \\ \cos\left(\frac{4\pi}{4} - \alpha_i + \gamma_i\right) \\ \cos\left(\frac{6\pi}{4} - \alpha_i + \gamma_i\right) \end{bmatrix} l_{v_i} \tag{19}$$

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

The tendon mapping matrix for a single section, regardless of the number of tendons, can be formulated by calculating elements in the virtual space. Consider a two-section TDCR, where each section is actuated by three tendons, as shown in Figure 1. Section I is actuated by six tendons, while Section II is actuated by three tendons. When Section I bends, all six tendons must be actuated. In contrast, bending Section II requires actuating only its three respective tendons. Consequently, the tendon mapping matrix for Section I consists of six elements, whereas the mapping matrix for Section II comprises three elements. The mapping of two sections and the actuating length of actual tendons is given by,

$$\begin{bmatrix} l_{a_1} \\ l_{b_1} \\ l_{c_1} \\ l_{a_2} \\ l_{b_2} \\ l_{c_2} \end{bmatrix} = \begin{bmatrix} \cos(\alpha_1 - \gamma_1) \\ \cos\left(\frac{2\pi}{3} - \alpha_1 + \gamma_1\right) \\ \cos\left(\frac{4\pi}{3} - \alpha_1 + \gamma_1\right) \\ \cos(\alpha_1 - \gamma_1 - \gamma_2) \\ \cos\left(\frac{2\pi}{3} - \alpha_1 + \gamma_1 + \gamma_2\right) \\ \cos\left(\frac{4\pi}{3} - \alpha_1 + \gamma_1 + \gamma_2\right) \end{bmatrix} l_{v_1} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \cos(\alpha_2 - \gamma_1 - \gamma_2) \\ \cos\left(\frac{2\pi}{3} - \alpha_2 + \gamma_1 + \gamma_2\right) \\ \cos\left(\frac{4\pi}{3} - \alpha_2 + \gamma_1 + \gamma_2\right) \end{bmatrix} l_{v_2} \tag{20}$$

Similarly, the virtual joint space is mapped into the actual joint space for multi-section TDCR using the generalised mapping matrix given by,

$$\mathbf{l} = \mathbf{T}\mathbf{l}_{\mathbf{v}} \tag{21}$$

where, $\mathbf{T}$ is a $3n \times n$ matrix and given by,

$$\mathbf{T} = \begin{bmatrix} \mathbf{T}_1 & \mathbf{T}_2 & \cdots & \mathbf{T}_i \\ \mathbf{O}_1 & \mathbf{O}_2 & \cdots & \mathbf{O}_i \end{bmatrix}^T ; i = 1, 2, \cdots n \tag{22}$$

where, $\mathbf{O}_i$ is a $3 \times n - i$ zero matrix and $\mathbf{T}_i$ is a $3 \times i$ matrix, given by,

$$\mathbf{T}_i = \begin{bmatrix} \cos\left(\alpha_1 - \sum_{j=1}^{j=i} \gamma_j\right) & \cdots\cdots & \cos\left(\alpha_i - \sum_{j=1}^{j=i} \gamma_j\right) \\ \cos\left(\frac{2\pi}{3} - \alpha_1 + \sum_{j=1}^{j=i} \gamma_j\right) & \cdots\cdots & \cos\left(\frac{2\pi}{3} - \alpha_i + \sum_{j=1}^{j=i} \gamma_j\right) \\ \cos\left(\frac{4\pi}{3} - \alpha_1 + \sum_{j=1}^{j=i} \gamma_j\right) & \cdots\cdots & \cos\left(\frac{4\pi}{3} - \alpha_i + \sum_{j=1}^{j=i} \gamma_j\right) \end{bmatrix} \tag{23}$$

The tendon mapping matrix is utilised to determine the actual tendon lengths of each section from the length of the virtual tendons. The actual tendon lengths are converted to TDCR's motor pulley angles to control the TDCR's tip position.

6. State estimation in virtual actuation space

Differentiating Eq. (12), the relation of task space velocity and acceleration with virtual actuation space velocity and acceleration is given by,

$$\begin{aligned} \dot{\boldsymbol{\xi}} &= \mathbf{J}_v \mathbf{v} \\ \ddot{\boldsymbol{\xi}} &= \mathbf{J}_v \mathbf{v} + \mathbf{H}_v \mathbf{v} \end{aligned} \tag{24}$$

where, $\mathbf{J}_v$ and $\mathbf{H}_v$ are $\in \mathbb{R}^{3n \times 2n}$ called the Jacobian matrix and the Hessian matrix, respectively. $\mathbf{J}_v$ mapped the virtual actuation space's first derivative to that of the task space and mapped the virtual actuation

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

space's second derivative to that of the task space. Using Euler's discretisation method, Eq. (24) is reformulated as,

$$\begin{aligned}\mathbf{v}(t+\Delta t) &= \mathbf{v}(t) + (\mathbf{J}_v^{\dagger}\dot{\boldsymbol{\zeta}})\Delta t \\ \dot{\mathbf{v}}(t+\Delta t) &= \dot{\mathbf{v}}(t) + (\mathbf{J}_v^{\dagger}\ddot{\boldsymbol{\zeta}} - \mathbf{H}_v\dot{\mathbf{v}}(t))\Delta t\end{aligned} \tag{25}$$

In Eq. (25), $\mathbf{v}(t)$ and $\dot{\mathbf{v}}(t)$ are the current state of virtual actuation space, and $\mathbf{v}(t+\Delta t)$ and $\dot{\mathbf{v}}(t+\Delta t)$ are the updated state of virtual actuation space, $t \in \mathbb{R}$ is the current time, and Δt is the time step. † signifies the Moore-Penrose pseudoinverse.

7. Control methodology

Virtual actuation space control methodology utilises the task space error to provide control input in the virtual actuation space. The mapping matrix maps the control input into the actual joint space. The control input is provided to the actuators to control the tendon length. The task space error is defined as the difference between the desired generalised coordinate and the current generalised coordinates of TDCR. The error is given by,

$$\mathbf{e} = \boldsymbol{\zeta}_d - \boldsymbol{\zeta} \tag{26}$$

where, $\boldsymbol{\zeta}_d \in \mathbb{R}^{3n}$ is desired generalised coordinates and $\boldsymbol{\zeta} \in \mathbb{R}^{3n}$ is generalised current coordinates. The control law for acceleration and velocity tracking is given by,

$$\begin{aligned}\ddot{\mathbf{v}} &= \mathbf{J}_v{}^{\dagger}\left(\ddot{\boldsymbol{\zeta}}_d - \mathbf{H}_v\dot{\mathbf{v}} + \mathbf{K}_d\,\mathbf{e} + \mathbf{K}_p\mathbf{e}\right) \\ \dot{\mathbf{v}} &= \mathbf{J}_v{}^{\dagger}\left(\dot{\boldsymbol{\zeta}}_d + \mathbf{K}_p\mathbf{e}\right)\end{aligned} \tag{27}$$

where, $\mathbf{K}_d \in \mathbb{R}^{3n\times 3n}$ and $\mathbf{K}_p \in \mathbb{R}^{3n\times 3n}$ are derivative and proportional positive definite diagonal gain matrices.

7.1. Convergence

Multiplying Eq. (27) by $\mathbf{J}_v$ both sides and with little rearrangement, we have,

$$\begin{aligned}\mathbf{J}_v\,\mathbf{v} + \mathbf{H}_v\mathbf{v} &= \ddot{\boldsymbol{\zeta}}_d + \mathbf{K}_d\,\mathbf{e} + \mathbf{K}_p\,\mathbf{e} \\ \mathbf{J}_v\dot{\mathbf{v}} &= \dot{\boldsymbol{\zeta}}_d + \mathbf{K}_p\mathbf{e}\end{aligned} \tag{28}$$

from Eqs. (24 and (28) we have,

$$\begin{aligned}\ddot{\boldsymbol{\zeta}} - \ddot{\boldsymbol{\zeta}}_d &= \mathbf{K}_d\dot{\mathbf{e}} + \mathbf{K}_p\mathbf{e} \\ \dot{\boldsymbol{\zeta}} - \dot{\boldsymbol{\zeta}}_d &= \mathbf{K}_p\mathbf{e}\end{aligned} \tag{29}$$

$$\begin{aligned}\ddot{\mathbf{e}} + \mathbf{K}_d\dot{\mathbf{e}} + \mathbf{K}_p\mathbf{e} &= 0 \\ \dot{\mathbf{e}} + \mathbf{K}_p\mathbf{e} &= 0\end{aligned} \tag{30}$$

If $\mathbf{K}_p$ and $\mathbf{K}_d$ are positive definite matrices, then Eq. (30), error converges to zero when time tends to infinity [37]. The control block diagram is shown in Figure 5 to explain the control architecture of the TDCR. The control algorithm is implemented in Python and LabView.

8. The experimental setup

The experimental setup used to validate the proposed methodology is shown in Figure 6. It consists of a two-section tendon-driven continuum manipulator using a feedback loop with a motion capture sensor to track the desired trajectory. The setup includes a 370 mm long, 16 mm diameter polyurethane rod as the backbone, divided into two equal sections. Six 0.7 mm steel strand tendons pass through 3D-printed PLA disks, with three tendons actuating each section. For Section I, γ_1 is 0, and for Section II, γ_2 is

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

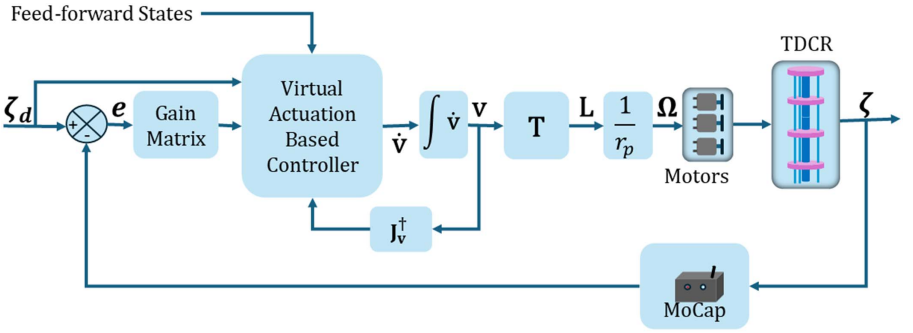


Figure 5. *Overview of the proposed virtual actuation space-based control block diagram.*

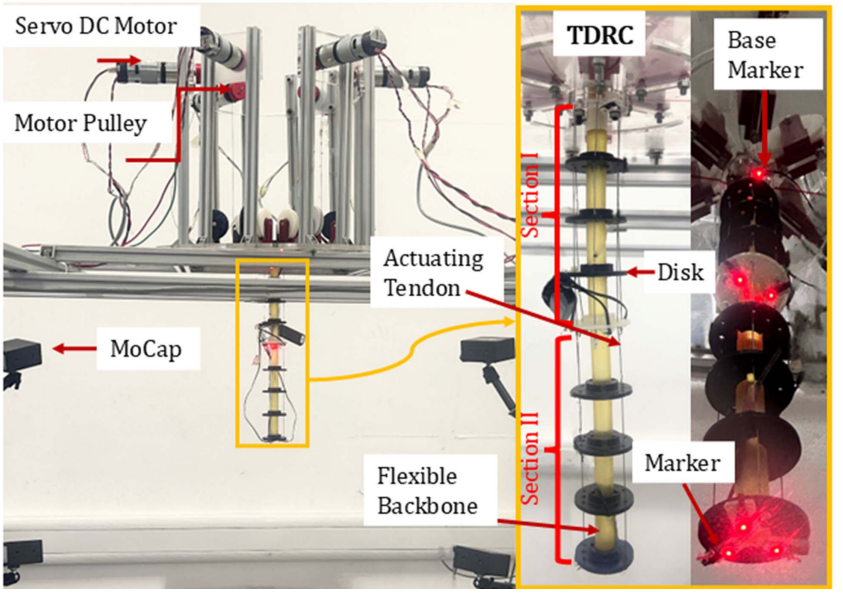


Figure 6. *Experimental setup of two sections of TDCR.*

$\pi/3$ radians. Six DC servo motors, three for each section, precisely control tendon lengths for accurate TDCR bending using an optical encoder and real-time controller (SbRio 9,629).

LED markers provide tip coordinates using a high-resolution motion capture (MoCap) camera (PhaseSpace Impulse X2), with eight cameras mounted around the setup that operate at 960 Hz. Each section tip has three markers forming an equilateral triangle, with the centroid providing generalised coordinates. PLA disks keep tendons parallel to the backbone and guide them to an 18 mm radius motor pulley. Guiding pulleys direct tendons to the motor pulley, with the setup mounted on aluminium extrusions.

The real-time controller operates in a closed-loop system, managing motor movements based on a LabView-based script and encoder feedback. The motor rotates to the desired angle at a speed specified by the host computer via USB. The current tip position of the TDCR is captured by a motion capture (MoCap) system, which sends the data to the host computer using a Python script.

The host computer executes the control algorithm, computing the necessary motor inputs based on the error between the desired and current coordinates provided by the MoCap system. These control inputs, defined in virtual space, are converted into tendon lengths. Dividing the tendon lengths by the pulley radius r_p yields the motor angles Ω, which are used by the real-time controller and motor driver to achieve the desired configuration.

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

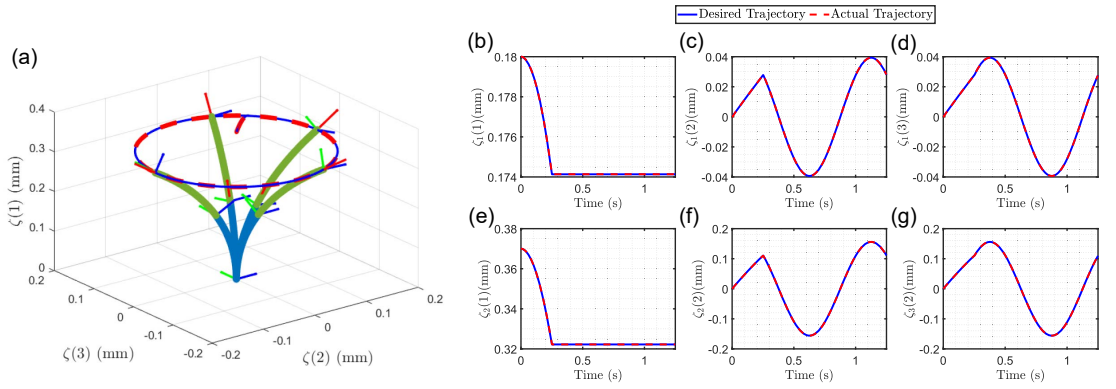


Figure 7. *(a) Tracking of circular trajectory by two-section TDCR, (b–d) tracking of generalisedcoordinates of Section I, and (e–g) tracking of generalisedcoordinates of Section II.*

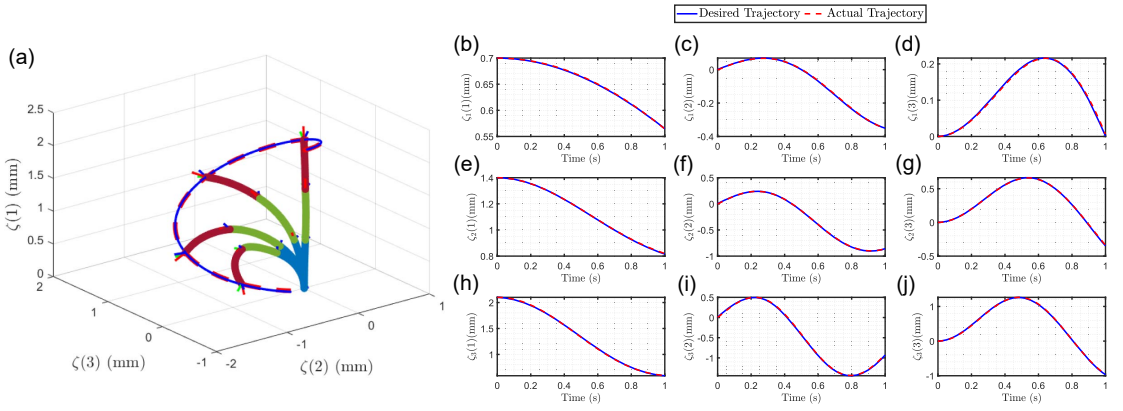


Figure 8. *(a) Tracking of spiral trajectory by three-section TDCR, (b–d) tracking of generalisedcoordinates of Section I, (e–g) tracking of generalisedcoordinates of Section II, and (h-j) tracking of generalisedcoordinates of Section III.*

9. Results and discussion

9.1. Trajectory tracking simulation

The simulation results for trajectory tracking are presented in Figures 7–9. For the two-section TDCR, Figure 7(a) demonstrates accurate tracking of the desired circular trajectory, while Figure 7(b–g) illustrates the evolution of generalised coordinates for both sections. The three-section TDCR simulation, shown in Figure 8(a), successfully tracks the desired trajectory, with each section coordinating to achieve precise tip positioning. The corresponding generalised coordinates for all three sections are presented in Figure 8(b–j). Similarly, Figure 9(a) depicts trajectory tracking of a varying curvature path using a four-section continuum robot, with the generalised coordinate evolution for each section shown in Figure 9(b–m). These simulation results validate the scalability and effectiveness of the proposed virtual actuation space formulation and control framework across different robot configurations.

9.2. Point-to-point tracking

In the workspace of the TDCR, five sets of points forming a pentagon are tracked using a virtual space-based control methodology. The goal is for the TDCR tip to reach each of these points sequentially. Figure 10 shows the tracking of these points by the TDCR tip. The desired points are depicted in

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

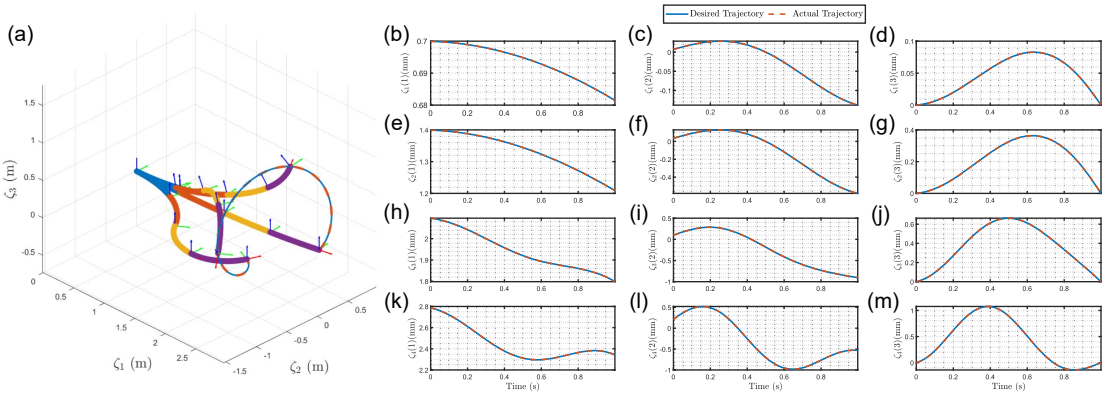


Figure 9. *(a) Tracking of changing curvature trajectory by four-section TDCR, (b–d) tracking of generalisedcoordinates of Section I, (e–g) tracking of generalisedcoordinates of Section II, (h–j) tracking of generalised coordinates of Section III, and (k–m) tracking of generalised coordinates of Section IV.*

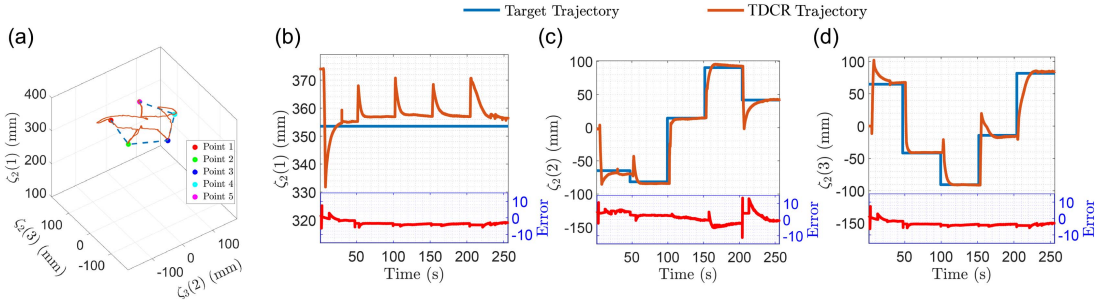


Figure 10. *(a) Pentagon set points, (b), (c), and (d) generalisedcoordinates of the tip, with position errors. The dynamics of the error (—) are labelledon the blue y-axis on the right side.*

Figure 10(a). Changes in the generalised coordinates of the TDCR tip denoted $\zeta_2(1)$, $\zeta_2(2)$, and $\zeta_2(3)$, over time, are shown in Figures 10(b), 10(c), and 10(d), respectively. The tracking error for each coordinate is displayed in red plots on the corresponding plots, with the error axis labelled on the right side of each plot. The TDCR starts from its initial state, reaches the first point in 25 s, and remains there for 50 s. It then moves to the subsequent points, taking 10 s to reach each one. The root mean squared errors (RMSE) for $\zeta_2(1)$, $\zeta_2(2)$, and $\zeta_2(3)$ are 3.4 mm, 3 mm, and 1.6 mm, respectively, and their relative errors with respect to the length of TDCR are 0.9%, 0.8%, and 0.4%. Because the pentagon's vertices lie in a plane parallel to the $\zeta_2(2)$–$\zeta_2(3)$ plane, the desired $\zeta_2(1)$ remains constant. As shown in Figure 10(b), there are five peaks because the TDCR briefly leaves the pentagon plane when tracking these points.

9.3. Trajectory tracking with section control

Several feasible trajectories are generated for both sections using generalised forward kinematics, collectively referred to as Tr $1-5$. These include a twopetalled flower shape and a spiral trajectory shown in Figure 11, a circular trajectory shown in Figure 12, a curve in a different plane shown in Figure 13, and a curve in the same plane with different orientations shown in Figure 14. The generalised coordinates at the tip of section i are shown in each figure's subplots (a–c), while those of section ii are in subplots (d–f). The 3D trajectories are depicted in subplots (g). The tracking error of each generalised coordinate is shown in the red-coloured plot at the bottom of the respective plots with the blue-coloured axes.

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

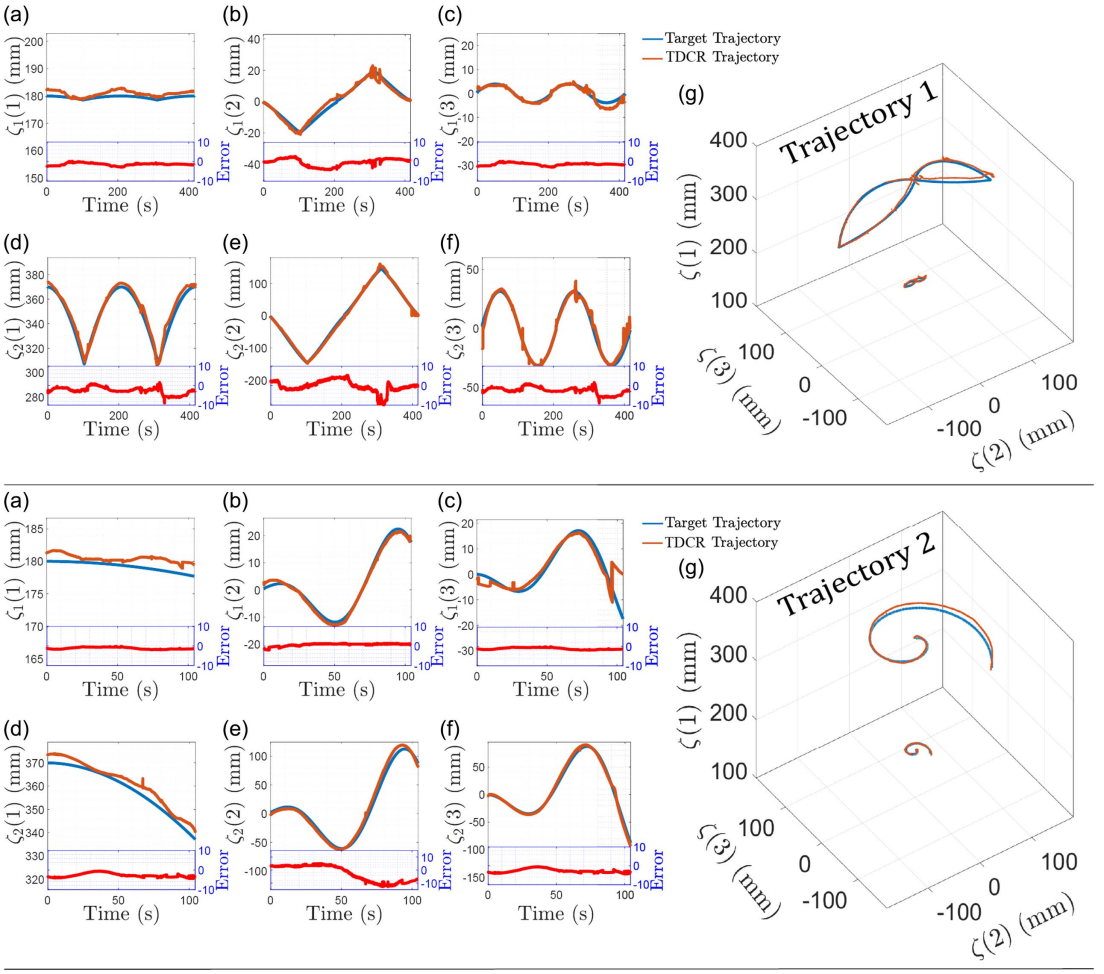


Figure 11. *(a), (b), and (c) show the generalisedcoordinates and tip tracking errors for section i, while (d), (e), and (f) depict the same for section ii. The error dynamics (—) are plotted with the right y-axis in blue colour, just beneath each plot, corresponding to their respective trajectories. (g) Illustrates both the required trajectories and the actual tracking trajectories.*

In Tr 1 and Tr 2, both sections' tips follow similar paths, effectively behaving as a single-section TDCR, as shown in Figure 11. Tr 3 highlights trajectory tracking with section-shape control: during the first half of the motion, section i bends while section ii remains unbent, and during the second half, section ii bends to track the circular path, leaving section i fixed as shown in Figure 12(a–f). Minor deviations in $\zeta_1(1)$, $\zeta_1(2)$, and $\zeta_1(3)$ occur due to the compliant nature of the TDCR, but the controller works to minimise or bind these errors. The camera view of the TDCR is shown in Figure 12(h) for the first half of the motion, and Figure 12(i) for the second half, displayed frame by frame from top to bottom. The red box indicates camera 1, and the blue box indicates camera 2. In Figure 12(h), it can be observed that only section i bends during the first half, lying in the plane passing through camera 1, while camera 2 also captures it bending in a single plane. For the second half of the motion, Figure 12(i) shows that the tip of section i remains almost fixed (intersecting the white dashed line in all frames), while the tip of section ii tracks the circular trajectory.

In Tr 4, TDCR follows trajectories in different planes, resulting in a twisted "**S**" shape bending, as shown in Figure 13(h). In Tr 5, TDCR follows trajectories in the same planes but with different

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

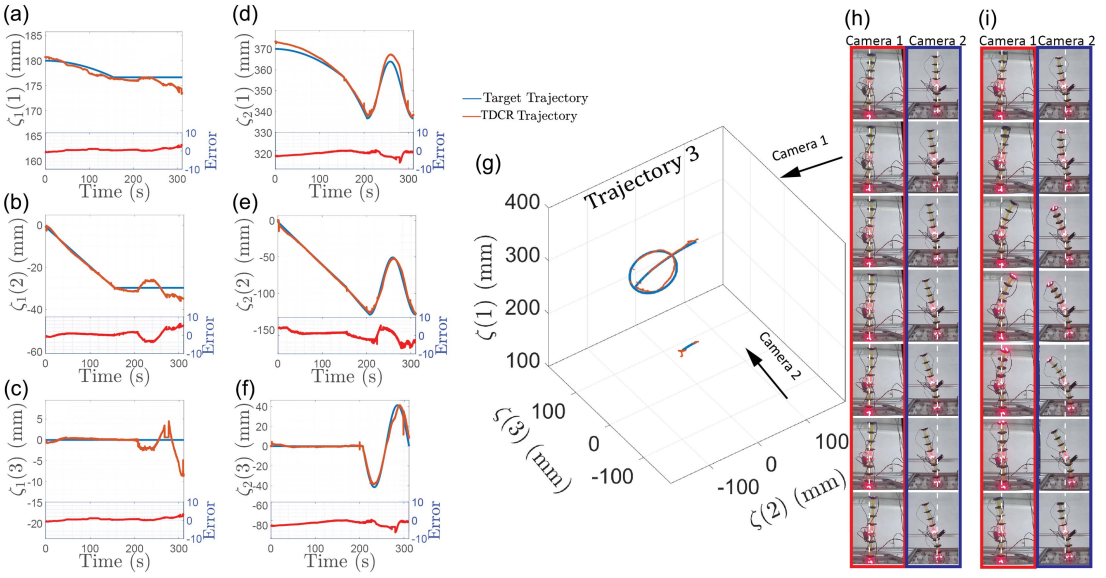


Figure 12. *The description of the plot is the same as Figure 11. Trajectory tracking with independent section control, (h) and (i) show a camera view of TDCR with different frames from top to bottom.*

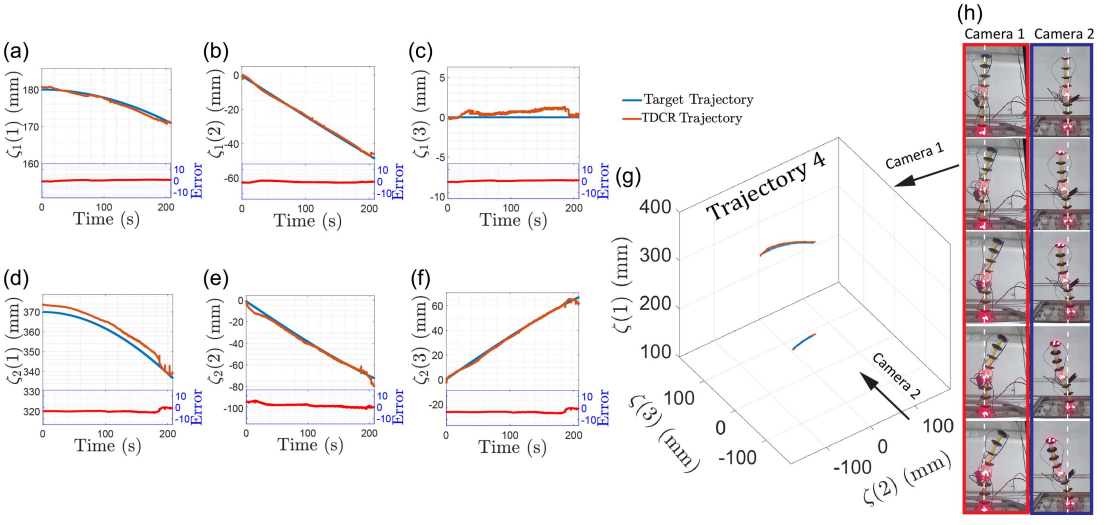


Figure 13. *The description of the plot is the same as Figure 11. (g) Trajectory in different planes, (h) shows a camera view of TDCR with frames from top to bottom.*

orientations, creating a twisted "?" shape bending, as illustrated in Figure 14(h). Figure 13(g) illustrates that the tip of section I bends in the $\zeta(1) - \zeta(2)$ plane, which is orthogonal to camera 1 and parallel to camera 2. The white dashed line in Figure 13(h) represents the intersection line of the orthogonal planes of both cameras. From Figure 13(h), it is clear that the tip of section i stays on this dashed line in camera 1, while the tip of section ii moves away. Figure 14(g) shows a trajectory in the same plane but with a different orientation. Since the plane of Tr 5 is orthogonal to camera 1 and parallel to camera 2, the TDCR remains aligned with the dashed line in camera 1 throughout the tracking, as the tip of section i moves away and the tip of section ii coincides with the dashed line of camera 2, as shown in Figure 14(h).

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

Table I. *RMSE of tracking for different trajectories in mm.*

	$\zeta_1(1)$	$\zeta_1(2)$	$\zeta_1(3)$	$\zeta_2(1)$	$\zeta_2(2)$	$\zeta_2(3)$	$\|\|e_I\|\|$	$\|\|e_{II}\|\|$
Error **Tr 1**	1.5	1.9	1	2.5	3	1.2	2.6	4
Error **Tr 2**	1	0.8	1.4	2.7	**5**	2.4	1.8	6
Error **Tr 3**	0.8	2	2	1.6	2.2	1.7	2.9	3.2
Error **Tr 4**	0.5	0.5	0.6	3.3	2	0.8	0.9	3.9
Error **Tr 5**	2.6	1.3	0.3	3.3	1.6	0.8	2.9	3.7
Ave RMSE	1.2	1.3	1.06	2.68	2.7	1.3	–	–
Ave RMSE$_L$	0.3	0.35	0.3	0.7	0.73	0.35	–	–

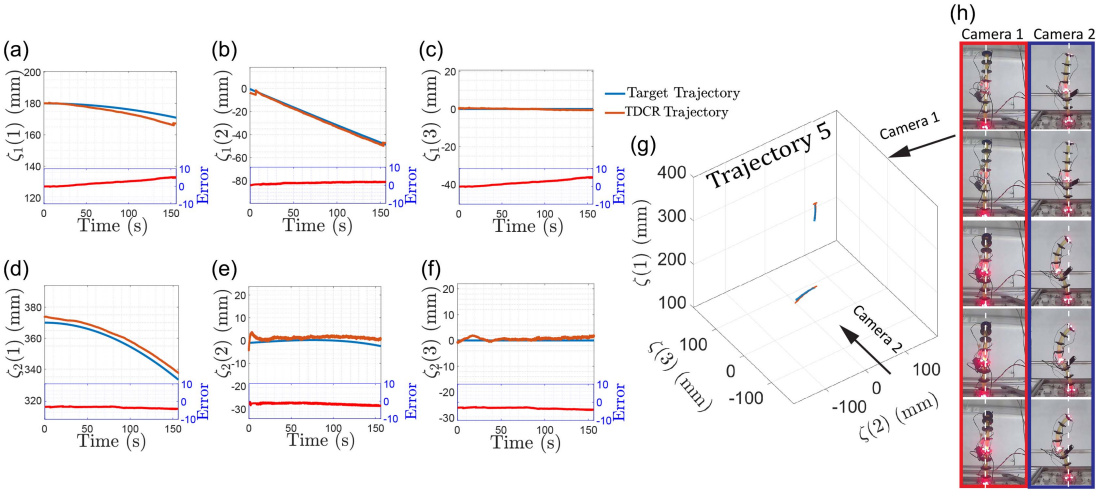


Figure 14. *The description of the plot is the same as Figure 11. (g) Trajectory in different orientations, (h) shows a camera view of TDCR with frames from top to bottom.*

The RMSE for each trajectory is presented in Table I. Each row of Table I provides the tracking error of the generalised coordinates of the TDCR, followed by the RMSE and RMSE$_L$ (RMSE normalised by the length of the TDCR). The last two columns list the average error norm for generalised tip coordinates of each section, denoted as $\|e_I\|$ for section I and $\|e_{II}\|$ for section II. $\|e_i\|$ is less than 3 mm and $\|e_{II}\|$ is 4 mm, which shows that the TDCR can follow the trajectory nicely.

TDCR has tracked all given trajectories Tr (1–5) within the acceptable bounds of error. The maximum RSME is 5 mm amongst all generalised coordinates for all the trajectories, as shown in Table I. The tracking error of generalised coordinates for both sections tracks all trajectories. Tr (1–5) Error plots for each generalised coordinate exhibit spikes when the MoCap marker leaves the field of view or is disturbed by varying environmental lighting conditions. During these instances, the system receives no feedback, leading to brief tracking lapses. However, once the markers re-enter the field of view, the system resumes tracking with bounded errors, rapidly correcting itself as feedback is restored.

9.4. Comparison with existing methods

To contextualise the performance of our virtual actuation space-based controller, we compare it against recent state-of-the-art approaches for trajectory tracking for TDCRs, focusing on positional accuracy (e.g., RMSE or tip error) for similar tasks like circular and spiral paths. Table II summarises key metrics from representative works, alongside ours. Our method achieves the lowest average RMSE

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

Table II. *Comparison of trajectory tracking accuracy (RMSE/Tip error in mm) across methods.*

Method	RMSE/Tip Error (% with respect to length of TDCR)	Trajectories	Sections
Virtual actuation space	Avg 1–3 (max 5) ($\sim$ 1%)	Circular, spiral, flower, twisted	2
Vertical-tip PCC [38]	$\leq$7 ($\sim$ 3–4 %)	Planned circular, eight shaped	2
ANN-SMC w/uncertainties [41]	15 ($\sim$ 5%)	Spiral, circular	1
Actuation-based estimation [40]	17.9 ($\sim$ 7%)	Position	1
Cosserat feedforward [39]	$\sim$ 9.5%	Circular (2D)	1
ADRC w/mapping [42]	2–4 ($\sim$ 6%)	Circular, spiral	1

(1.2–2.7 mm across coordinates, max 5 mm) and relative errors (0.3–0.73% of TDCR length), outperforming Lai et al. [38] (RMSE $\leq$7 mm for vertical-tip circular tracking in a two-segment soft robot) and Pekris et al. [39] (accuracy within 9.5% relative error for 2D circular paths using Cosserat-based feedforward control). For dynamic shape estimation during tracking, Feliu-Talegon et al. [40] report higher average tip errors (1.79 cm or 17.9 mm) relying solely on actuation readings, without explicit decoupling for multi-sections. Chen et al. [41] demonstrate robust tracking under uncertainties (e.g., fluid viscosity) via ANN-FNITSMC but require computationally intensive neural approximation; our model-based decoupling yields bounded errors post-FOV loss with simpler, scalable kinematics for arbitrary sections/tendons. Similarly, Cao et al. [42] improve ADRC-based tracking for circular/spiral paths (RMSE 2–4 mm in simulations), yet struggle with multi-section coupling; our approach extends this to independent section trajectories (e.g., Tr 3–5) with sub-4 mm section-specific errors ($\|e_I\| \leq 2.9$ mm, $\|e_{II}\| \leq 4$ mm), enabling shape control alongside tip adherence in cluttered spaces. Overall, the proposed framework advances beyond these by providing fully decoupled, hardware-agnostic control with enhanced accuracy and scalability, validated on a two-section TDCR.

9.5. Discussion

The experimental results validate the efficacy of the proposed virtual actuation space framework for kinematic modelling and control of multi-section TDCRs. By reformulating the actuation problem into a decoupled, theoretically two-input-two-output system per section, the VAS enables precise trajectory tracking while allowing independent shape control of each section. Simulations of two-, three-, and four-section TDCRs demonstrate excellent scalability, with accurate replication of the prescribed trajectories across generalised coordinates as illustrated in Figures 7, 8, and 9. Hardware experiments yielded compelling performance metrics: point-to-point tracking of pentagonal vertices achieved RMSE values of 1.6–3.4 mm (0.4–0.9% of TDCR length), while trajectory tracking (Tr 1–5) produced an average RMSE of 1–3 mm per coordinate as presented in Table I, with normalised errors below 1% of the 370 mm backbone length. In point-to-point tracking, $\zeta_2(1)$ represents the coordinate along the robot's length, while $\zeta_2(2)$ and $\zeta_2(3)$ represent the planar coordinates at a given $\zeta_2(1)$ height. Although steady-state errors in $\zeta_2(2)$ and $\zeta_2(3)$ are effectively zero, a small steady-state error of approximately 2 mm persists in $\zeta_2(1)$ due to actuator resolution limitations. These outcomes confirm millimetre-level precision relative to the robot's scale, underscoring the framework's practical effectiveness in quasi-static operations.

The enhanced performance stems primarily from section decoupling via the generalised tendon mapping matrix $\mathbf{T}$ and the VAS representation itself. Traditional multi-section TDCR control grapples with coupled tendon interactions, where actuation of one section inadvertently influences others due to shared tendon routings (e.g., the six tendons for Section I in a two-section design, as in Figure 1. This coupling amplifies modelling errors and control complexity [14, 43, 44]. In contrast, the VAS collapses the three-tendon actuation per section into a single virtual tendon length l_{v_i} within the bending plane,

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

defined by α_i, while $\mathbf{T}$ explicitly maps these virtual inputs to physical tendon lengths $\mathbf{l}$ Eqs. (18–20). This mapping, structured as a block-diagonal matrix with submatrices $\mathbf{T}_i$ and zero blocks $\mathbf{O}_i$, isolates section-specific contributions, preventing propagation of actuation errors across sections. For instance, in Tr 3 Figure 12, Section I bends independently in the first half-trajectory with Section II unactuated ($\mathbf{T}_2$ effectively zeroed), yielding bounded deviations in ζ_1 (<2 mm) despite compliance-induced perturbations. The decoupling reduces the effective dimensionality from $3n$ (tendon space) to $2n$ (VAS), simplifying Jacobian $\mathbf{J}_v$ Eq. (24), which enhances real-time solvability and minimises numerical ill-conditioning.

Furthermore, the VAS inherently leverages the PCC assumption more effectively than direct tendon-space methods. By parameterising motion in the bending plane Figure 1(b), the framework enforces uniqueness via the virtual tendon arc length $l_{v_i} = r_d\theta_i$ Eqs. (7–16), eliminating the need to resolve geometric constraints among the three physical tendons. This structural consistency translates to lower tracking errors, as evidenced by the module-wise discretisation Figure 3, which captures discrete disk rotations more accurately than continuum approximations alone. The control law Eq. (27), operating in VAS with task-space feedforward and PD gains, exploits this by inverting $\mathbf{J}_v^\dagger$ to directly compensate for nonlinearities, achieving exponential error convergence Eq. (30) without aggressive tuning that could excite unmodelled dynamics.

Regarding potential singularities, the VAS framework encounters degeneracy when $l_{v_i} = 0$, corresponding to the unbent (straight) configuration of a section. From Eq. (6), the radius of curvature is $R_{c_i} = L_i r_d / l_{v_i}$; as $l_{v_i} \to 0$, $R_{c_i} \to \infty$, making $\mathbf{J}_v$ singular due to vanishing bending angle $\theta_i = l_{v_i}/r_d \to 0$. This kinematic singularity implies a loss of instantaneous controllability in the bending plane, potentially leading to degraded accuracy or control failure near straight postures, akin to workspace boundaries in rigid manipulators. To mitigate this, the domain of l_{v_i} is restricted to $\mathbb{R} \setminus \{0\}$, and in practice, if l_{v_i} approaches zero (e.g., due to numerical drift or near-straight desired poses), it is regularised by replacing it with a small positive ϵ (e.g., 10^{-5} times the section length).

In the experimental validation, feedback from the motion capture system is acquired at 960 Hz but downsampled to 100 Hz to accommodate the quasi-static motion assumption of the kinematic model and avoid inertial effects. At this sampling rate and under quasi-static conditions, the experimental tracking exhibits no observable latency, confirming real-time control performance. The kinematic-based controller imposes minimal computational burden, enabling real-time implementation at the selected sampling rates, while the quasi-static assumption remains appropriate for the demonstrated manipulation tasks and ensures model validity throughout the experiments.

Overall, the VAS framework advances TDCR control by bridging kinematic simplicity with decoupled execution, enabling shape-aware tracking essential for cluttered environments like minimally invasive surgery or duct inspection. Its generalisability to $n > 2$ sections (simulations) and variable tendon counts Eq. (20) positions it for hybrid actuation paradigms, fostering broader adoption in precision robotics.

10. Conclusions

This study presents a generalised virtual actuation space framework for forward and inverse kinematics of TDCRs, alongside a control strategy that leverages this space with task-space feedforward velocity or acceleration terms to enable precise trajectory tracking. Theoretical analysis confirms the convergence of task-space errors.

A primary innovation lies in the decoupling of individual sections, reducing each to a theoretically two-input, two-output system, while the tendon mapping matrix accommodates arbitrary tendon counts per section and inter-tendon couplings. This facilitates independent trajectory following for each section, supporting diverse robot shapes within the workspace. Experimental validation on a two-section TDCR demonstrates effective tracking across varied paths (e.g., circular, spiral, floral, and twisted), with average root-mean-square errors (RMSE) of 1–3 mm across generalised coordinates and length-normalised RMSE_L below 1% of TDCR length. These errors can be further minimised through higher-precision actuators and encoders.

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

As detailed in Table II, our method yields superior positional accuracy compared to recent benchmarks in multi-section scenarios. This scalability extends to TDCRs with additional sections, unlocking potential in medical robotics (e.g., minimally invasive procedures), industrial automation, and confined-space inspection tasks. Future work may explore integration with real-time sensing for enhanced adaptability in dynamic environments.

Acknowledgements. The authors thank the Gujarat Council on Science and Technology for financial support. The authors specifically thank Prof Harish J. Palanthandalam-Madapusi for his discussions and support and the committee members. Special thanks to Smit Prajapthi for helping in developing the experimental setup. They also acknowledge feedback from the IITGN Robotics Lab team and thank IIT Gandhinagar for their ongoing support.

Author contributions. **Md Modassir Firdaus**: conceptualisation, formal analysis, investigation, methodology, software, validation, visualisation, writing – original draft. **Shail Jadav**: conceptualisation, formal analysis, software. **Madhu Vadali**: conceptualisation, funding acquisition, methodology, project administration, resources, supervision, writing – review and editing.

Financial support. This work is supported by the Gujarat Council on Science and Technology under the grant GUJCOST/2020-21/875.

Supplementary material. The supplementary material for this article can be found at https://doi.org/10.1017/S026357472610318X.

Competing interests. The authors declare that they have no conflicts of interest related to this work. No financial or personal relationships exist that could inappropriately influence or bias the research reported in this paper.

Ethical standards. This research does not involve human subjects, human data, or human tissue. Therefore, ethical approval was not required for this study.

Declaration of generative AI in scientific writing. During the preparation of this work, the author(s) used ChatGPT in order to improve the language and readability. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

References

[1] D. Rus and M. T. Tolley, "Design, fabrication and control of soft robots," *Nature* **521**(7553), 467–475 (2015).
[2] P. E. Dupont, N. Simaan, H. Choset and C. Rucker, "Continuum robots for medical interventions," *Proc. IEEE* **110**(7), 847–870 (2022).
[3] M. Calisti, G. Picardi and C. Laschi, "Fundamentals of soft robot locomotion," *J. R. Soc. Interface* **14**(130), 20170101 (2017).
[4] M. G. Tsegay, P. M. Pathak, A. K. Samantaray and R. Merzouki, "Bond graph modeling of a spatial multi-section soft bionic robot," *Mech. Mach. Theory* **174**(2022), 104902 (2022).
[5] D. A. Xin Dong, S. C. David Palmer, M. Raffles, A. Rabani and J. Kell, "Development of a slender continuum robotic system for on-wing inspection/repair of gas turbine engines," *Robot. Comput.-Integr. Manuf.* **44**(2017), 218–229 (2017).
[6] M. Wang, X. Dong, W. Ba, A. Mohammad, D. Axinte and A. Norton, "Design, modelling and validation of a novel extra slender continuum robot for in-situ inspection and repair in aeroengine," *Robot. Comput.-Integr. Manuf.* **67**(2021), 102054 (2021).
[7] M. Hwang and D.-S. Kwon, "Strong continuum manipulator for flexible endoscopic surgery," *IEEE/ASME Trans. Mechatron.* **24**(5), 2193–2203 (2019).
[8] P. Rao, O. Salzman and J. Burgner-Kahrs, "Towards contact-aided motion planning for tendon-driven continuum robots," *IEEE Robot. Autom. Lett.* **9**(5), 4687–4694 (2024).
[9] F. Janabi-Sharifi, A. Jalali and I. D. Walker, "Cosserat rod-based dynamic modeling of tendon-driven continuum robots: A tutorial," *IEEE Access* **9**(2021), 68703–68719 (2021).
[10] H. J. Zhang, M. T. C. Sven Lilge and J. Burgner-Kahrs, "Cooperative Control of Dual-arm Concentric Tube Continuum Robots," **In**: *2022 International Conference on Manipulation, Automation and Robotics at Small Scales (MARSS)* (IEEE, 2022) pp. 1–6.
[11] Z. Yang, H. Yang, Y. Cao, Y. Cui and L. Zhang, "Magnetically actuated continuum medical robots: A review," *Adv. Intell. Syst.* **5**(6), 2200416 (2023).

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

[12] C. Yang, S. Geng, I. Walker, D. T. Branson, J. Liu, J. S. Dai and R. Kang, "Geometric constraint-based modeling and analysis of a novel continuum robot with shape memory alloy initiated variable stiffness," *Int. J. Robot. Res.* **39**(14), 1620–1634 (2020).
[13] I. D. Walker, "Continuous backbone "continuum" robot manipulators," *International Scholarly Research Notices* **2013**, 726506 (2013).
[14] R. J. Webster III and B. A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *Int. J. Robot. Res.* **29**(13), 1661–1683 (2010).
[15] D. C. Rucker and R. J. Webster III, "Statics and dynamics of continuum robots with general tendon routing and external loading," *IEEE Trans. Robot.* **27**(6), 1033–1044 (2011).
[16] M. W. Hannan and I. D. Walker, "Kinematics and the implementation of an elephant's trunk manipulator and other continuum style robots," *J. Robot. Syst.* **20**(2), 45–63 (2003).
[17] N. Simaan, R. Taylor and P. Flint, "A Dexterous System for Laryngeal Surgery," **In**: *IEEE International Conference on Robotics and Automation, 2004*. Proceedings. ICRA'04, vol. 1 (IEEE, 2004) pp. 351–357.
[18] M. Mahvash and P. E. Dupont, "Stiffness control of surgical continuum manipulators," *IEEE Trans. Robot.* **27**(2), 334–345 (2011).
[19] B. A. Jones and I. D. Walker, "Kinematics for multisection continuum robots," *IEEE Trans. Robot.* **22**(1), 43–55 (2006).
[20] W. Ba, X. Dong, A. Mohammad, M. Wang, D. Axinte and A. Norton, "Design and validation of a novel fuzzy-logic-based static feedback controller for tendon-driven continuum robots," *IEEE/ASME Trans. Mechatron.* **26**(6), 3010–3021 (2021).
[21] E. A. Aner, M. I. Awad and O. M. Shehata, "Modeling and trajectory tracking control for a multi-section continuum manipulator," *J. Intell. Robot. Syst.* **108**(3), 49 (2023).
[22] S. Xu and B. He, "Adaptive approximation tracking control of a continuum robot with uncertainty disturbances," *IEEE Trans. Cybern.* **54**(1), 230–240 (2022).
[23] G. Zhang, F. Du, X. Zhang, K. Wu, G. Zheng, Y. Li and R. Song, "Continuum robots: A real-time model-based data-driven nonlinear controller," *IEEE Trans. Ind. Electron.* **71**(12), 16176–16186 (2024).
[24] P. Xiang, J. Zhang, D. Sun, K. Qiu, Q. Fang, X. Mi, Y. Wang, R. Xiong and H. Lu, "Learning-based high-precision force estimation and compliant control for small-scale continuum robot," *IEEE Trans. Autom. Sci. Eng.* **21**(4), 5389–5401 (2023).
[25] M. Goharimanesh, A. Mehrkish and F. Janabi-Sharifi, "A fuzzy reinforcement learning approach for continuum robot control," *J. Intell. Robot. Syst.* **100**(3-4), 809–826 (2020).
[26] T. C. Kargin and J. Kołota, "A reinforcement learning approach for continuum robot control," *J. Intell. Robot. Syst.* **109**(4), 77 (2023).
[27] Y. Fang, X. Dong, A. Mohammad and D. Axinte, "Design and control of a multiple-section continuum robot with a hybrid sensing system," *IEEE/ASME Trans. Mechatron.* **28**(3), 1522–1533 (2023).
[28] M. Li, R. Kang, D. T. Branson and J. S. Dai, "Model-free control for continuum robots based on an adaptive kalman filter," *IEEE/ASME Trans. Mechatron.* **23**(1), 286–297 (2017).
[29] M. K. Mishra, A. K. Samantaray and G. Chakraborty, "Joint-space kinematic control of a bionic continuum manipulator in real-time by using hybrid approach," *IEEE Access* **10**(2022), 47031–47050 (2022).
[30] M. M. Firdaus, O. Paranjape and M. Vadali, "FEM-based dynamic modelling framework for flexible continuum manipulators," *J. Mech. Robot.* **16**(10), 101013 (2024).
[31] A. Chawla, C. Frazelle and I. Walker, "A Comparison of Constant Curvature Forward Kinematics for Multisection Continuum Manipulators," **In**: *2018 Second IEEE International Conference on Robotic Computing (IRC)* (IEEE, 2018) pp. 217–223.
[32] T. Mahl, A. E. Mayer, A. Hildebrandt and O. Sawodny, "A Variable Curvature Modeling Approach for Kinematic Control of Continuum Manipulators," **In**: *2013 American Control Conference* (IEEE, 2013) pp. 4945–4950.
[33] J. Till, V. Aloi and C. Rucker, "Real-time dynamics of soft and continuum robots based on cosserat rod models," *Int. J. Robot. Res.* **38**(6), 723–746 (2019).
[34] L. Sciavicco and B. Siciliano, *Modelling and Control of Robot Manipulators* (Springer Science & Business Media, 2012).
[35] R. M. Murray, S. S. Sastry and L. Zexiang, *A Mathematical Introduction to Robotic Manipulation*. 1st edition, (CRC Press Inc., USA, 1994).
[36] M. M. Firdaus and M. Vadali, "Virtual Tendon-based Inverse Kinematics of Tendon-driven Flexible Continuum Manipulators," **In** *Proceedings of the 2023 6th International Conference on Advances in Robotics (AIR '23)* (Association for Computing Machinery, New York, NY, USA, 2013) Article 72, 1–5. https://doi.org/10.1145/3610419.3610491.
[37] S. Jadav and H. J. Palanthandalam-Madapusi, "Utilization of manipulator redundancy for torque reduction during force interaction," *ASME Lett. Dyn. Syst. Control* **4**(2), 021005 (2024).
[38] J. Lai, K. Huang, B. Lu, Q. Zhao and H. K. Chu, "Verticalized-tip trajectory tracking of a 3D-printable soft continuum robot: Enabling surgical blood suction automation," *IEEE/ASME Trans. Mechatron.* **27**(3), 1545–1556 (2021).
[39] S. Pekris, R. D. Williams, T. Atkins, I. Georgilas and N. Bailey, "Model-based trajectory tracking of a compliant continuum robot," *Front. Robot. AI* **11**(2024), 1358857 (2024).
[40] D. Feliu-Talegon, A. T. Mathew, A. Y. Alkayas, Y. A. Adamu and F. Renda, "Dynamic shape estimation of tendon-driven soft manipulators via actuation readings." *IEEE Robot. Autom. Lett.* **10**(1), 780–787 (2024).
[41] Q. Chen, C. Ming and Y. Qin, "Adaptive neural network-based sliding mode control for trajectory tracking control of cable-driven continuum robots with uncertainties," *Robotica* **43**(1), 316–331 (2025).
[42] Y. Cao, W. Yang, S. Wang, S. He, Z. Liu and L. Xie, "Improved trajectory tracking control for a continuum robot using error-driven adrc with input mapping method," *IEEE Trans. Ind. Electron.* **71**(10), 12696–12706 (2024).

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press

[43] P. Rao, S. L. Quentin Peyron and J. Burgner-Kahrs, "How to model tendon-driven continuum robots and benchmark modelling performance," *Front. Robot. AI* **7**, 630245 (2021).
[44] L. Raimondi, M. Russo, X. Dong and D. Axinte, "Understanding friction and superelasticity in tendon-driven continuum robots," *Mechatronics* **104**, 103241 (2024).
[45] Y. Qin, Q. Chen and C. Ming, "Adaptive recursive sliding mode based trajectory tracking control for cable-driven continuum robots," *ISA Trans.* **147**, 501–510 (2024).

Cite this article: M. M. Firdaus, S. Jadav and M. Vadali (2026). "Trajectory tracking of multi-section tendon-driven continuum robots using virtual actuation space control", Robotica **44**, 518–539. https://doi.org/10.1017/S026357472610318X

https://doi.org/10.1017/S026357472610318X Published online by Cambridge University Press