



RESEARCH ARTICLE

A data density-based measure of dexterity for continuum robots and its comparative study

Shailesh Bamoriya , Roshan Kumar Hota  and Cheruvu Siva Kumar

Department of Mechanical Engineering, Indian Institute of Technology, Kharagpur, India

Corresponding author: Shailesh Bamoriya, Email: bamoriya.415@iitkgp.ac.in

Received: 6 January 2024; **Revised:** 8 August 2024; **Accepted:** 26 September 2024

Keywords: continuum robot; dexterity; data density; workspace; surgical robot; and kinematics

Abstract

Continuum robot-based surgical systems are becoming an effective tool for minimally invasive surgery. A flexible, dexterous, and compact robot structure is suitable for carrying out complex surgical operations. In this paper, we propose performance metrics for dexterity based on data density. Data density at a point in the workspace is higher if the number of reachable points is higher, with a unique configuration lying in a small square box around a point. The computation of these metrics is performed with forward kinematics using the Monte Carlo method and, hence, is computationally efficient. The data density at a particular point is a measure of dexterity at that point. In contrast, the dexterity distribution property index is a measure of how well dexterity is distributed across the workspace according to desired criteria. We compare the dexterity distribution property index across the workspace with the dexterity index based on the dexterous solid angle and manipulability-based approach. A comparative study reveals that the proposed method is simple and straightforward because it uses only the position of the reachable point as the input parameter. The method can quantify and compare the performance of different geometric designs of hyper-redundant and multisegment continuum robots based on dexterity.

1. Introduction

Surgical robots provide numerous advantages, including precise and predictable movements by means of complex three-dimensional paths, the capability to perform highly accurate motions during surgery, and are popular as they reduce postoperative in-hospital case time [1]. A key category of surgical robots is the continuum robots [2], popular for their flexibility and dexterity (with multiple segments) [3]. Conventional multi-port robotic systems for minimally invasive surgery have rigid surgical tools resulting in conical workspace, insufficient dexterity, and a high payload. They cannot operate in deep surgical sites with space limitations. In such cases, some of the interesting options are emerging such as snake-like flexible continuum robots and micro robots. The dexterity of the continuum robot helps to perform complex tasks in constrained workspace. Therefore, dexterity is essential for designing flexible surgical manipulators/ instruments (forceps and endoscopes) to access deep, confined surgical spaces [4] and repositioning its base in a surgical environment to position the robot end-effector in the dexterous region. Continuum robots have applications in Natural Orifice Transluminal Endoscopic Surgery and Single Port Access surgeries [5] to perform complex procedures by dexterous manipulation of the robot, avoid extra incision requirements, and damage to peripheral tissues during surgeries.

There are different types of actuation mechanisms [6, 7] that are used to drive these kinds of flexible mechanisms. Some of the continuum robots are often remotely actuated with external actuators to minimize robot size due to environmental constraints [8]. An actuation mechanism for a tendon-driven continuum robot uses tendons, which are pulled by actuators such as servo motors, linear actuators,

and shape memory alloys [9–12]. Li et al. [13] did a comparative study of surgical tendon-driven continuum robot structure, tendon-driven serpentine manipulator, and concentric tube-type continuum robot based on their workspace and dexterity based on dexterous solid angle (DSA) [14, 15] approach and found that the number of segments and their length ratios influence the workspace and dexterity. The study also confirmed that the tendon-driven continuum robot could have better distal-end dexterity among these three.

Dexterity is a fundamental measure of the number of potential orientations to achieve at a certain point in the workspace or the count of inverse kinematics solutions that can be found [16]. Various notions of dexterity and the corresponding measures have been adopted in the literature. The measures broadly belong to two categories – measures that quantify the ability of the robot to produce motions in different directions and measures that quantify the number of orientations that a robot can achieve at a particular position. Examples for methods in the first category include condition number of the Jacobian [17–19] and the volume of the manipulability ellipsoid [20–22]. These works define dexterity in terms of the ability of the robot to produce motions in different directions. Methods in the second category compute the number of orientations that the robot can achieve at a particular position. For example, researchers in ref. [23] compute the number of orientations in which the object can be grasped at a particular position. In 1994, Abdelmalek et al. [24] introduced the concept of the DSA for a hyper-redundant robotic arm. Later, Wu et al. [14] defined a DSA-based measure of dexterity and dexterity indices to investigate the distribution property of dexterity across the workspace. Burgner et al. [25] characterize the workspace of a concentric tube robot, computing robot redundancy across the reachable volume by counting configurations per discretized workspace volume. However, some of these methods involve inverse kinematics and the use of jacobian [26], which is computationally intensive for continuum robots [27, 28]. Solving inverse kinematics for multisegment robots is a challenge due to its complexity, hyper-redundant DOF, and nonlinearities [27]. One needs to adopt a sophisticated regression method, as in deep learning [29, 30] to get some solution. Here, we are avoiding the need for inverse kinematics using the data density approach stated. The design of a multisegment continuum robot for a surgical system will need a description of the number of segments, their lengths, and the total length. Considering a given cavity size, the length of robot segments has a major influence on the kinematics of the multisegment robot.

In this paper, we chose the tendon-driven three-segment continuum robot (6 DOF), where each segment has 2 DOF; therefore, it has been considered for the study. Such a 6 DOF version provides a comprehensive representation of both position and orientation in three-dimensional space and is therefore considered for dexterity analysis by utilizing a newly proposed dexterity measure based on data density, which is computationally simpler due to the lesser number of computation steps required than other, as it only requires end-effector position data computed from the sample of uniformly distributed random robot configuration in the workspace using forward kinematics. Similar approaches are used to compute the forward kinematics for multisegment continuum robots [6, 13, 31, 32]. For the simulation, bending of the robot segment in constant curvature [33–35] is assumed for simplification of the robot kinematics; there is also the scope using another kinematic model. In the first simulation, dexterity distribution is analyzed based on the proposed data density-based approach, DSA, and manipulability-based approach for different robot designs. In the second simulation, the dexterity distribution property index, dexterity index, and manipulability index distribution are calculated based on data density, DSA, and manipulability-based approach, respectively, by varying the segment length of the robot, and the results are verified.

In the proposed approach, the measures of dexterity and dexterity distribution property index of a multisegment continuum robot are based on data density. Various methods are studied in the literature for dexterity and related measures (dexterity distribution property) to analyze the robot design. A comparison has been made based on required input parameters to compute the dexterity and its distribution for various types of continuum robot design. Our key contributions to this paper are

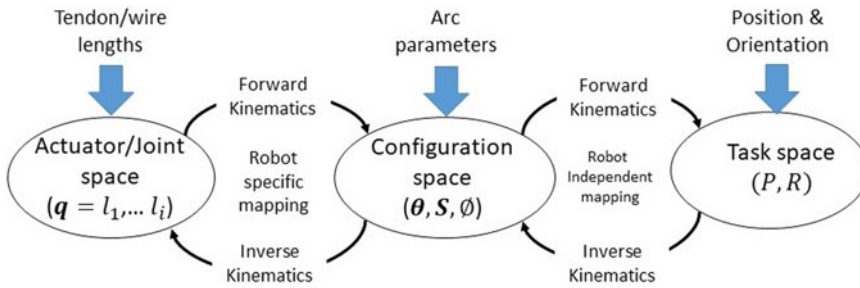


Figure 1. Kinematic classification of continuum robot manipulator.

1. A new, simple, and effective measure of dexterity based on data density.
2. A new measure of dexterity distribution property that characterizes the distribution according to desired criteria (higher dexterity away from the robot base is preferred).
3. The proposed method is compared and validated with state-of-the-art methods like the DSA-based dexterity measure and manipulability index.

The first key contribution is about a new measure of dexterity at a point, whereas the second contribution is about the new measure of dexterity distribution across the entire workspace. The dexterity distribution computed by the proposed data density-based dexterity measure is validated by comparison with the existing dexterity measures. The variation in dexterity distribution property, measured by varying the segment length, follows a trend similar to the existing approaches in the literature.

The proposed method utilizes forward kinematics of the tendon-driven continuum robot, which is explained in Section 2. In Section 3, the dexterity distribution property based on data density is discussed, and a qualitative comparative study is conducted. In Section 4, the simulation results are discussed.

2. Kinematics and simulation of continuum robot

In this section, we present the kinematics of a three segment continuum robot. The key assumptions for the kinematic modelling are

1. Each of the three segments of the continuum robot segment bends in constant curvature due to the resultant moment acting on the end tip of the segment. However, when all three segments are taken together, the overall robot has multiple curved sections (3D curvature) in all directions.
2. The elastic structure along the center line of the segment is symmetric and homogeneous around its longitudinal axis, called the backbone of the robot. Hence can bend in all directions uniformly.
3. Each point in the workspace is obtained by forward kinematics from a random sample of configuration using Monte Carlo (nonrepetitive robot configuration).

Continuum robot’s kinematics is divided into two phases of kinematic mappings, the first of which is robot-specific mapping and the second of which is robot independent mapping [6, 36–38], as illustrated in Figure 1. Robot-independent forward kinematics [31] compute the end-effector’s pose using the continuum segment’s configuration space parameters (3D curvature) can be described as $\psi_i = [\theta_i, \phi_i, S_i]^T$. Where ϕ is the angle of the bending plane from the x_0 -axis, θ is the bending angle of curvature, and S_i is the length of the curvature of the continuum segment, which is considered as the length of the segment.

The comparable rigid body model’s motion can be used to depict the bending of the continuum segment by local frame transformations shown in Figure 2. The Denavit–Hartenberg (DH) parameters

Table I. Equivalent modified DH parameters of the constant curvature continuum robot [6].

Link	θ	d	a	α
1	ϕ	0	0	$-\pi/2$
2	$\theta/2$	0	0	$\pi/2$
3	0	$d = s \times \frac{\sin \theta/2}{\theta/2}$	0	$-\pi/2$
4	$\theta/2$	0	0	$\pi/2$
5	$-\phi$	0	0	0

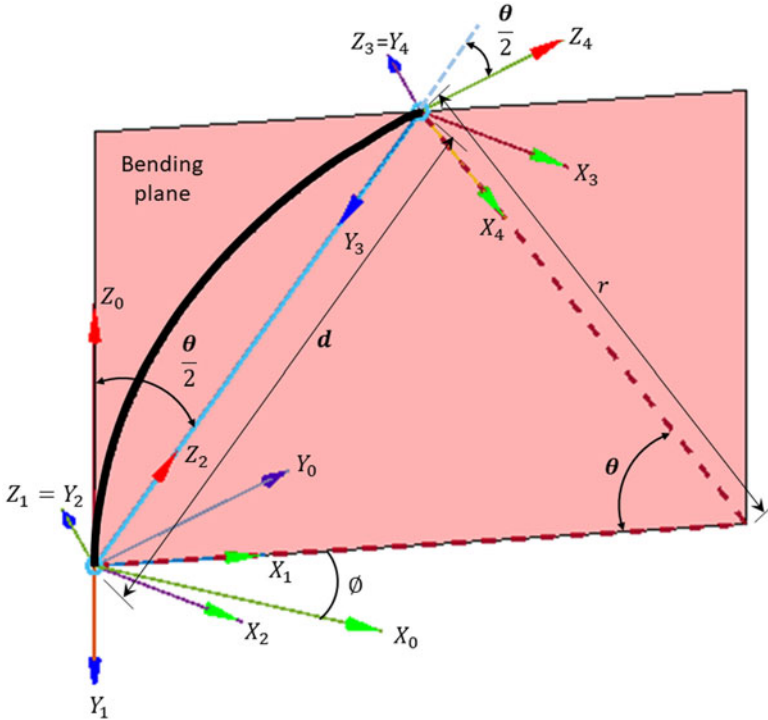


Figure 2. The continuum robot's frame representation of the updated DH parameters.

of the equivalent rigid body model of the continuum segment are the function of the configuration space parameters of the bent arc segment, which are used to obtain the robot-independent kinematic relations.

The DH parameters for a single segment are given in Table I, where d is the distance between the base and the end tip of the continuum segment. The homogeneous transformation matrix for single-segment continuum robot (forward kinematics) in Eq. (1). The end tip position of the continuum segment is

$$T_i^{i-1} = \begin{bmatrix} \cos^2 \phi_i (\cos \theta_i - 1) + 1 & \sin \phi_i \cos \phi_i (\cos \theta_i - 1) & \cos \phi_i \sin \theta_i & \frac{S_i \cos \phi_i (1 - \cos \theta_i)}{\theta_i} \\ \sin \phi_i \cos \phi_i (\cos \theta_i - 1) & -\cos^2 \phi_i (\cos \theta_i - 1) + \cos \theta_i & \sin \phi_i \sin \theta_i & \frac{S_i \sin \phi_i (1 - \cos \theta_i)}{\theta_i} \\ -\cos \phi_i \sin \theta_i & -\sin \phi_i \sin \theta_i & \cos \theta_i & \frac{S_i \sin \theta_i}{\theta_i} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$x = \frac{S_i \cos \phi_i (1 - \cos \theta_i)}{\theta_i} \tag{2}$$

$$y = \frac{S_i \sin \phi_i (1 - \cos \theta_i)}{\theta_i} \tag{3}$$

$$z = \begin{cases} \frac{S_i \sin \theta_i}{\theta_i}, & \forall \theta_i \neq 0 \\ S_i, & \text{For } \theta_i = 0 \end{cases} \tag{4}$$

The configuration space parameters for the i^{th} segment are $\psi_i = [\theta_i, \phi_i, S_i]^T$. This paper follows the constant curvature bending assumption and keeps the segment length S_i constant. In the special case when $\theta_i = 0$, the radius curvature of the segment becomes infinite; hence, the end tip position of the segment becomes $x = 0, y = 0$, and $z = S_i$. The homogeneous transformation matrix of the end tip coordinate frame w.r.t. the base coordinate frame of the i^{th} segment of continuum robot is T_i^{i-1} in (1). Now, the transformation matrix of the end tip of the distal segment w.r.t. the base of a three-segment robot can be written as

$$T_3^0 = T_1^0.T_2^1.T_3^2 = \begin{bmatrix} R_3^0 & P_3^0 \\ 0 & 1 \end{bmatrix} \tag{5}$$

where in equation (5), T_3^0 is the total transformation matrix of a three-segment continuum robot w.r.t. the base of the first segment. The end-effector pose of the robot is represented by P_3^0 and R_3^0 , respectively.

We have taken an example case with a 50 mm segment length for which the geometric model is shown in Figure 3. In Figure 3, the tendons in red, blue, and green color are used to actuate the third, second, and first segments, respectively, where the tendons of the distal segment pass through proximal segments through the guiding hole on disks. Therefore, guiding holes are provided symmetrically with equal angular offset.

2.1. Generation of workspace point by Monte Carlo approach

The reachable points of the workspace are generated using the forward kinematics model of the three-segment continuum robot mechanism from samples of the configuration parameter obtained using the Monte Carlo method. This method produces pseudo-random numbers between the range of configuration parameters [39]. This ensures the generated samples can simulate the random points in the workspace, where the random values of configuration variables are uniformly distributed. Using its DH matrix parameters as the function of robot configuration parameters (ψ_i), a finite number of configuration parameter’s samples are generated by the Monte Carlo method [40] based on the given range in Table II as follows;

$$\psi_i^k = \psi_{imin} + rand(\psi_{imax} - \psi_{imin}); i = 1, 2, 3 \quad k = 1, 2, 3, 4, \dots, m \tag{6}$$

where $\psi_i^k = [\theta_i^k, \phi_i^k]^T$ represents the k^{th} sample values for each configuration variable θ_i and ϕ_i of the i^{th} segment obtained by Monte Carlo sampling, with the sample size of $m = 10^8$. The value of configuration variables ranges $[\theta_{imin}, \theta_{imax}] \in [-\pi, \pi]$, and $[\phi_{imin}, \phi_{imax}] \in [0, 2\pi]$.

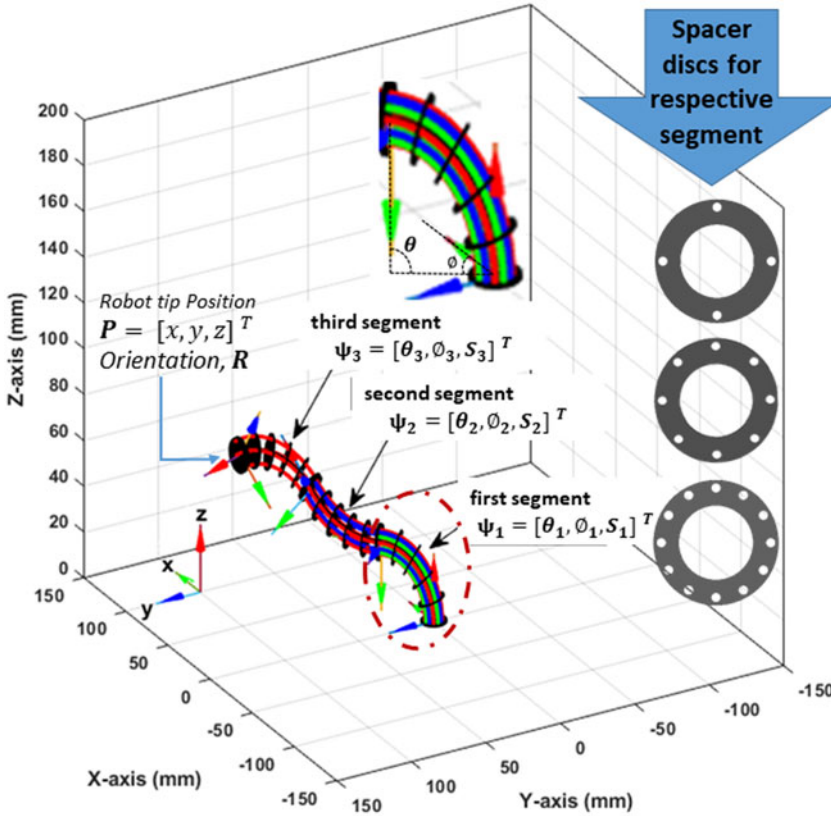
The continuum robot workspace is symmetric about the z-axis; hence, reachable points (m_p) in the planar workspace cross-section in the XZ-plane at $y = 0$ are chosen for simulating the dexterity distribution in a planar space. The planar dexterity distribution simulation and corresponding configuration parameters are used to compute the manipulability across the workspace cross-section. The m_p is a number of points in workspace cross-section $m_p \in m$.

2.2. Workspace

The robot workspace measures the robot’s reachability and region’s volume that the robot end-effector can reach. The forward kinematics, in equations (1–5), is used to compute the robot workspace by

Table II. Simulation parameters used for 3D and planar workspace plots of three-segment continuum robots.

No. of segment	Segment length (mm)	θ Range (rad)	ϕ Range (rad)	Samples
3	50, 50, and 50	$[-\pi, \pi]$	$[0, 2\pi]$ for 3D	10^8

**Figure 3.** Geometric model of three-segment continuum robot, four tendons drive each segment [33].

using input configuration parameters generated based on Monte Carlo method [39] for the configuration parameter θ_i , ϕ_i ranges $[-\pi$ to $\pi]$ and $[0$ to $2\pi]$, respectively, for each i^{th} segment of continuum robot according to predefined simulation parameters, as shown in Table II.

Due to the flexible backbone structure, continuum robots can bend uniformly in all directions. Hence, the robot workspace will be symmetric about the z-axis. It is also assumed that each continuum segment can bend by a 180° bending angle. The three-segment continuum robot's workspace in 3D space and a planar workspace were plotted with simulation parameters (see Table II) and illustrated in Figures 4, and 5, respectively. The reachable points of the end-effector are dense in some regions of the workspace, which signifies the reachability with multiple configurations. The workspace cross-section (planar) is utilized to compute the dexterity distribution property index across the workspace. The area of the planar workspace is computed by calculating the area of the polygon shape created by the robot's reachable positions at the boundary region of the cross-section of the workspace.

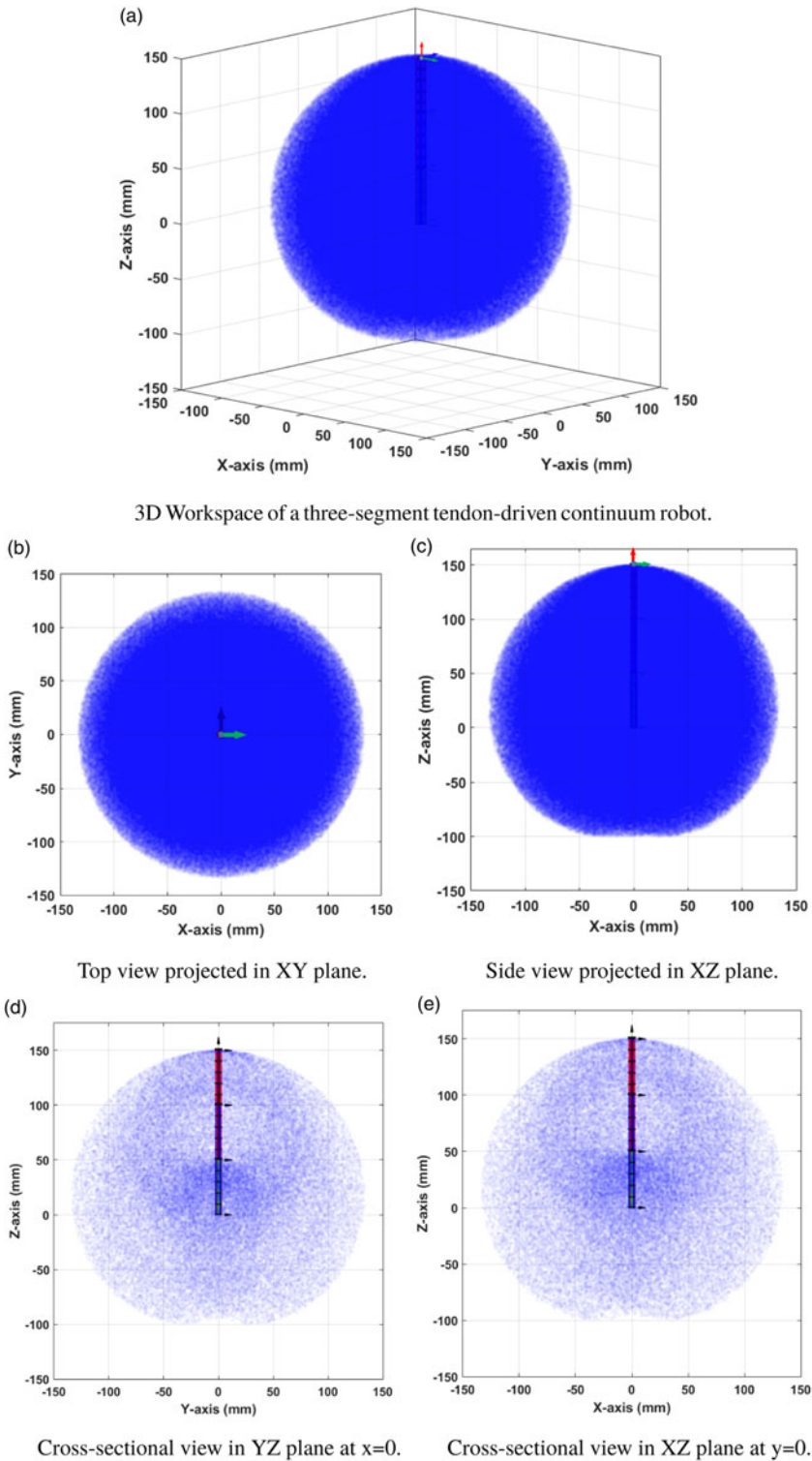


Figure 4. 3D workspace of a three-segment tendon-driven continuum robot.

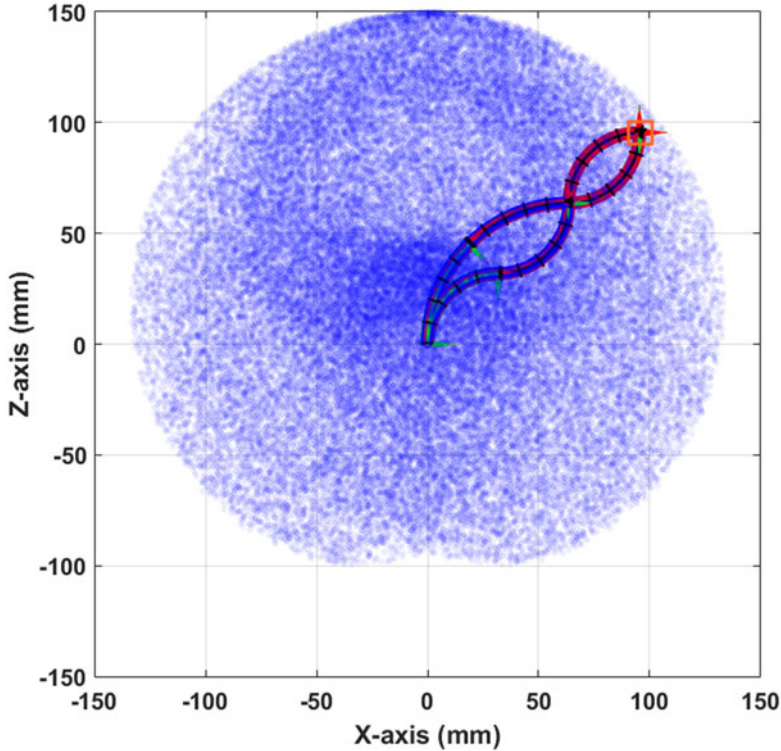


Figure 5. A three-segment tendon-driven continuum robot's workspace cross-section in X-Z plane.

3. A new method of evaluating dexterity and a dexterity distribution property index using data density approach

In surgical workspaces, a robot end-effector should access distant target regions with sufficient dexterity to perform complex tasks like cutting, suturing, and endoscopy [41]. Obtaining such a dexterous region in the distal end of the robotic device will require special attention to some desirable features. We define a term called dexterity distribution property index of these workspaces based on only data density of the position points in the workspace as the number of such points in such unit space will be directly proportional to the no of orientations possible to the end effector in the same unit space (or the dexterity of the end effector). Such dexterity property measure should also capture the distance of the high dexterity region from the base of the continuum robot. The farther the dexterity region is the better the suitability for the surgical application. As this also depends on the number of segments considered as well as their relative lengths, which are different from that of the articulated robots, this index would be suitable for evaluating design variants that are being explored or optimized.

Table III presents a comparison of the various measures of dexterity based on the approaches used, parameters computed, and insight provided (dexterity and dexterity distribution). This comparison considers the kinematics approaches, dexterity, dexterity distribution property, and input parameters. Badescu and Mavroidis [15], Abdel et al. [24], and Li et al. [13] adopted the DSA approach for the dexterity analysis; in addition to this Wu et al. [14] defined the dexterity indices for analyzing the distribution property of dexterity along the axis directions. The DSA-based approach is the forward kinematics-based approach that utilizes the position and orientation of the robot from sample configuration and computes DSA. The DSA ($D(k)$) at a position is defined as the ratio of the area of the service region ($A_R(k)$) of the unit sphere constructed at the position of k^{th} point to the total area of the sphere (A_{sp}) [14]. The total

Table III. Required parameters based comparison of the proposed method with existing approaches to measure dexterity.

Authors	Approach	Standard parameters			Extended parameters	
		Position	Orientation	Jacobian	Dexterity	Dexterity distribution property
M. Badescu and C. Mavroidis [15], Abdel et al. [24], Li et al. [13]	DSA	✓	✓	×	✓	×
Wu et al. [14], Wang and Lau [17], Cardou et al. [19]	DSA Condition number of the Jacobian matrix	✓	✓	×	✓	✓
Abdel et al. [43], Leibrandt et al. [44]	Manipulability	✓	✓	✓	✓	×
Du et al. [45]	Inverse kinematics and DSA	✓	✓	×	✓	×
Hota and Kumar [46]	Inverse kinematics	✓	×	✓	✓	×
Present approach	Data density	✓	×	×	✓	✓

area of the identical area patches of the service sphere is intersected by the end tip tangent of the robot called service region ($A_R(k)$). A dexterous solid angle can be written as

$$D(k) = \frac{A_R(k)}{A_{sp}} \tag{7}$$

$$D_g = \sum_{k=1}^m D(k).d_p(k) \tag{8}$$

where D_g is the dexterity index defined to quantify the dexterity distribution property across the workspace based on the DSA approach, and $d_{p(k)}$ is the distance of the point from the base of the robot.

To compare the proposed method with manipulability-based approaches for dexterity [42–44], we have also computed the manipulability index (w_1) on each point in the workspace using the same configuration samples generated to compute data density $\rho(k)$ and DSA $D(k)$ a dexterity measure. The comparison shows that the results are quite similar. However, the manipulability index takes a longer time to compute as it involves Jacobian computation. In addition, the term manipulability index distribution property w_g is defined similarly to the dexterity distribution property based on the data density approach. Manipulability of a manipulator proposed by Yoshikawa [20] manipulability is

$$w = \sqrt{\det(JJ^T)} = S_1, S_2, \dots S_n \tag{9}$$

where w is a quantitative measure of the manipulability of a robot arm that can be written as the multiply of the singular values $S_1, S_2, \dots S_n$, which also represents the volume of the manipulability ellipsoid. The defined manipulability index $w_1(k)$ corresponding to the k^{th} configuration of the robot and the manipulability index distribution property w_g of the workspace in the paper are

$$w_1(k) = \frac{S_{min}}{S_{max}} \tag{10}$$

$$w_g = \sum_{k=1}^m w_1(k).d_p(k) \tag{11}$$

where S_{min} , S_{max} are the minimum and maximum singular values corresponding to the k_{th} configuration of robot or reachable point in workspace. We use an index k to represent a reachable k^{th} point with robot tip position coordinates $[x, y, z]^T \in P_3^0$ having a configuration of the robot parameters $(\theta_i, \phi_i$ where $i = 1, 2, 3)$, the maximum and minimum singular value is the square root of the maximum and minimum eigenvalues of (JJ^T) [42].

The DSA approach requires the end-effector poses to evaluate the dexterity. On the other hand, the Jacobian matrix is required to compute the dexterity based on the condition number of Jacobian and the robot manipulability at a point. Du et al. [45] combined the DSA and inverse kinematics approach to minimize the computation time in dexterity evaluation. Hota and Kumar [46] used the inverse kinematics approach to calculate the dexterity in terms of counting the possible no of orientations at a point as the inverse kinematic solution, but inverse kinematics of multisegment continuum robots is quite difficult.

In the present data density-based approach, only the position of the reachable points in the workspace is used to evaluate the dexterity measure, and the dexterity distribution property index is a measure of distribution property of the dexterity away from the robot base across the workspace [47, 48] of a multisegment continuum robot without the need to solve inverse kinematics [49]. Though memory intensive, this eases the determination in practical situations due to its simpler computation requirement. The proposed data density-based approach is described in the following sections.

3.1. Comparison of various approaches

Multisegment continuum robots are redundant manipulators that involve complex inverse kinematics. The proposed approach utilizes forward kinematics to compute the reachable points in the workspace using the sample of robot configuration parameters for a finite range of robot configurations based on the Monte Carlo method discussed in section 2.1. Each point in the workspace is determined from a random set of configuration variables, resulting a random end-effector orientations. When the positional data points are denser in the workspace, as shown in Figure 5, the robot can reach these positions in more configurations. This leads to positional data density (ρ) for repetitive end-effector positions with different robot configurations. On this basis, robot dexterity is analogous to the robot workspace's positional data density at a point (ρ). The measure dexterity distribution property index (σ) is calculated to quantify the distribution property of the proposed dexterity measure across the workspace.

The dexterity distribution property index (σ) helps to visualize the high-density region (more dexterous region away from the robot origin). This parameter is useful for flexible surgical robot design and repositioning of the robot base during surgery to get high dexterity of the tooltip to perform complex surgical procedures in a confined space [50].

3.2. Methodology to determine data density and simulation

The data density at a point is considered the density of the reachable points inside a unit square area around that point in the workspace cross-section in XZ-plane at $y = 0$ and volume in the case of a 3D workspace. The proposed algorithm only considers a cross-section because the workspace is asymmetric about the Z-axis. Data density at a point corresponds to the k^{th} configuration and is calculated by assigning a square block around the point, counting the number of reachable points (N_k) within the block, and dividing it by the area of the block in equation (12) as shown in Figure 6. For example, P_1 and P_2 are two random reachable points in the workspace indicated in Figure 6 to compute the density square around point P_1 and P_2 are in sky blue and yellow color, respectively. Each robot configuration obtains a unique reachable point in the box. Therefore, we propose the definition of a data density-based measure of dexterity ρ (where $\rho_{(area)}, \rho_{(vol)}$ for workspace cross-section and workspace volume, respectively) and dexterity distribution property index (σ) based on data density as follows:

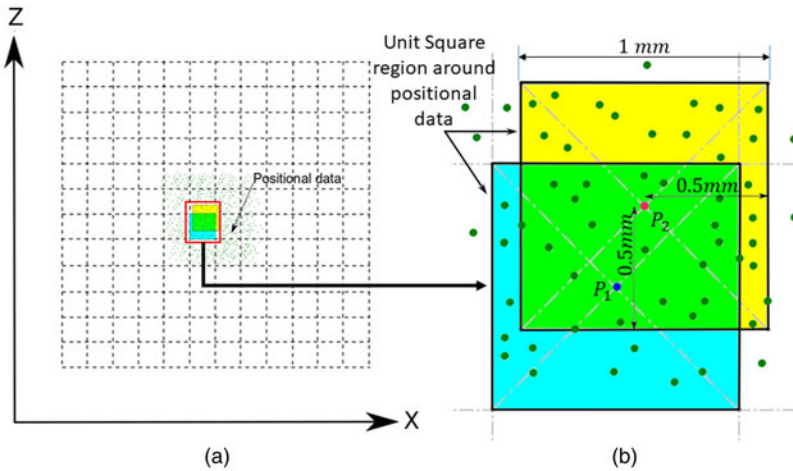


Figure 6. Representation of the data density ($\rho_{(area)}$) based approach for dexterity distribution property in the workspace cross-section in XZ-plane at $y = 0$.

$$\rho_{(area)}(k) = \frac{N_k}{A_{square}} \quad (12)$$

$$\sigma = \sum_{k=1}^{m_p} \rho_{(area)}(k) \cdot d_p(k) \quad (13)$$

Similarly, the volumetric data density-based measure of dexterity $\rho_{(vol)}$ and the dexterity distribution property index are as follows:

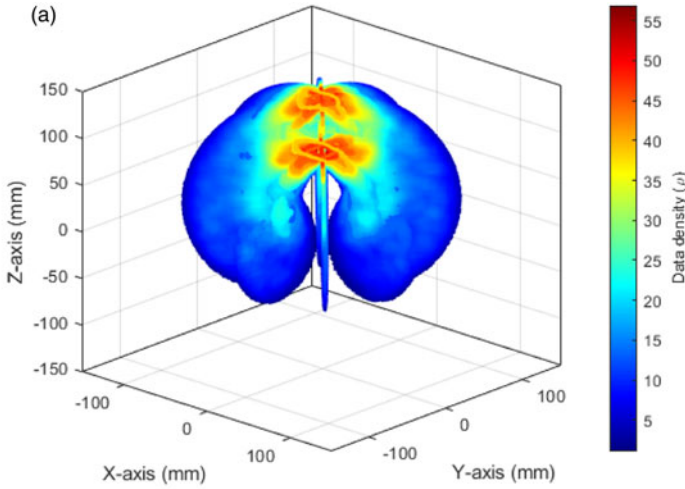
$$\rho_{vol}(k) = \frac{N_k}{V_{cube}} \quad (14)$$

$$\sigma = \sum_{k=1}^m \rho_{vol}(k) \cdot d_p(k) \quad (15)$$

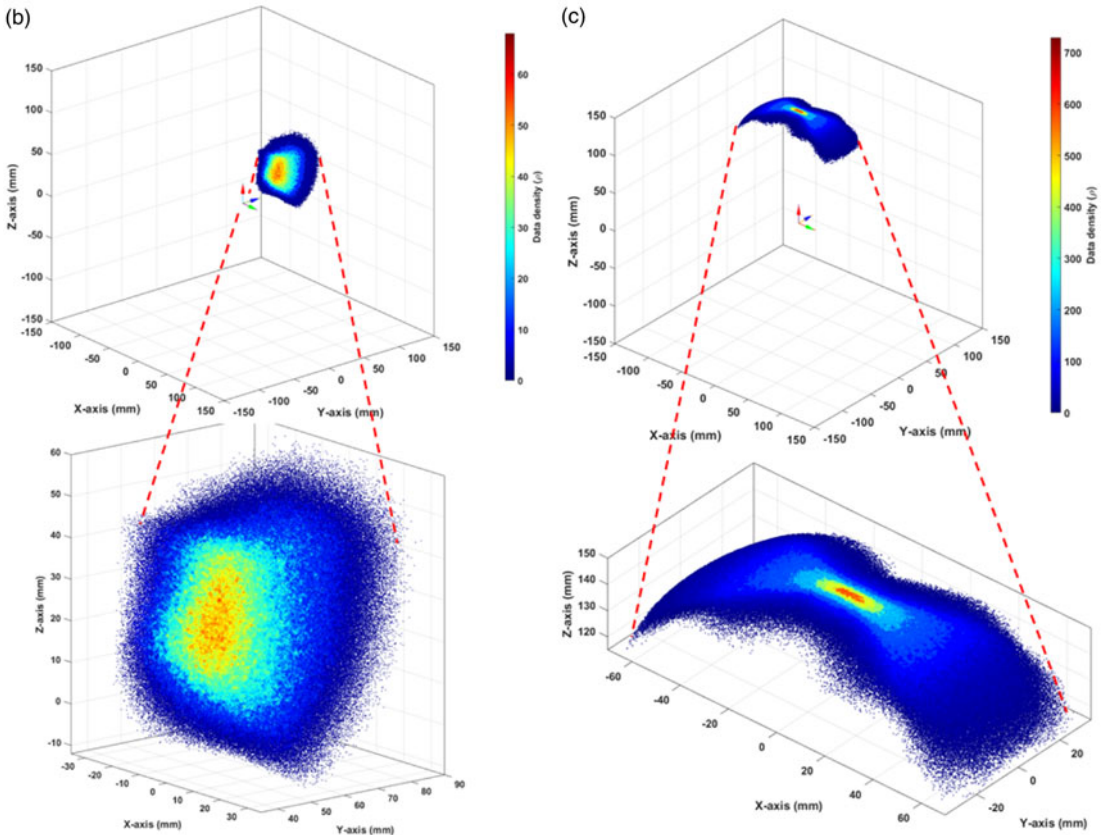
where $\rho_{(area)}(k)$ and $\rho_{(vol)}(k)$ are the data density of the unit square block having area A_{square} and volume V_{cube} , respectively, at the k^{th} point of m_p number of reachable points in the workspace cross-section. $d_p(k)$ is the distance of the k^{th} point from the robot base. Dexterity distribution property index σ is defined as the summation of the product of data density (the measure of dexterity) at a point and the distance of the respective point from the origin of the robot. As stated before this definition gives weightage to the concentration of data density away from the base, near the workspace boundary.

As the proposed method is based on forward kinematics, we use simulation criteria for the 3D workspace plot given in Table II to generate the reachable point from configuration parameters using the Monte Carlo approach. The data density algorithm is implemented at the reachable point of the workspace cross-section, in a vertical plane through the Z-axis, because the workspace is symmetrical about the Z-axis. In the case of a serial robot, the joints have actuation limits due to physical or assembly constraints, which results in a nonsymmetric workspace. However, in the case of the tendon-driven continuum robot, the actuation line is far from the robot assembly; therefore, it is free from actuation constraints, which results in a symmetric workspace. A nonsymmetric workspace is possible when we assign the actuation constraints in the configuration space variable.

The dexterity distribution in the 3D workspace is shown in Figure 7 by taking a typical example of a three-segment continuum robot design having the segment lengths $S_1 = 90$ mm, $S_2 = 30$ mm, and $S_3 = 30$ mm, with a total of 150 mm length, where the dexterity may vary with the robot design. Where Figure 7(a) shows the dexterity distribution in the workspace cross-section in vertical planes through the Z-axis. The reachable points for a nonsymmetric workspace are computed using forward kinematics



Dexterity distribution across the 3D workspace based on data density-based approach $\rho_{(area)}$ for workspace cross-section in vertical planes through z-axis at 45 degrees from each.



Configuration constraint $\psi_i \in [\pi/3, 2\pi/3]$.

Configuration constraint $\psi_i \in [-\pi/6, \pi/6]$.

Figure 7. Dexterity distribution across the workspace using proposed data density-based approach (a) using $\rho_{(area)}$ for workspace cross-section in a vertical plane through the z-axis (b) Using $\rho_{(vol)}$ for workspace volume for given configuration (c) Using $\rho_{(vol)}$ for workspace volume for given configuration.

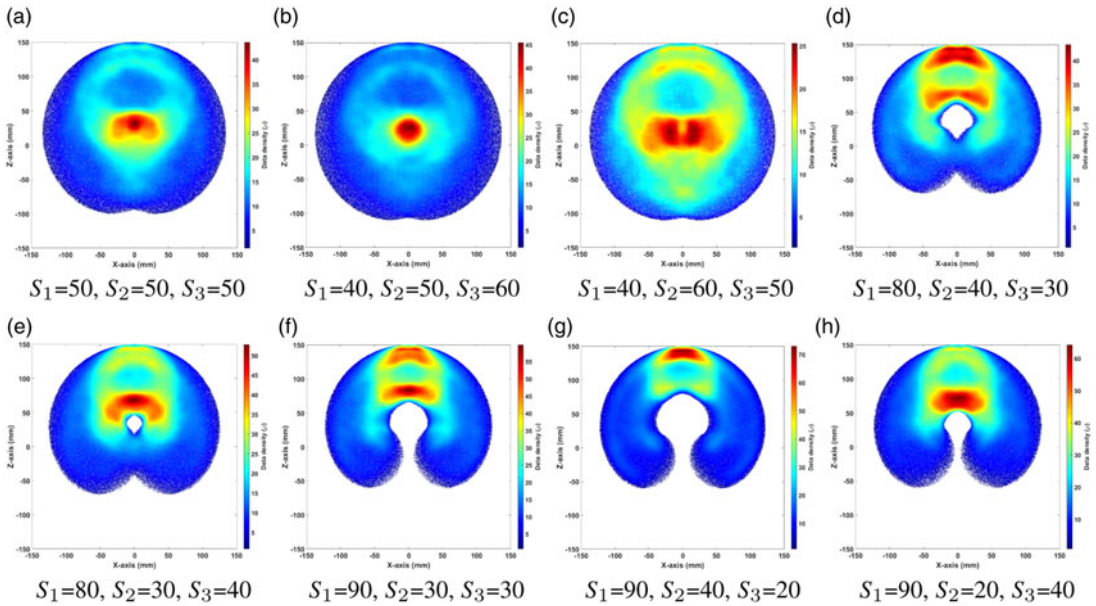


Figure 8. Dexterity distribution across the workspace cross-section in XZ plane at $y = 0$ using data density ($\rho_{(area)}$) based approach.

from the uniformly distributed random sample of the configuration parameters for the given range of configuration. There are one million configuration samples generated for the configuration range of $[\pi/3, 2\pi/3]$ and $[-\pi/6, \pi/6]$ for 3D non-symmetric workspace named workspace1 and workspace2, respectively, the corresponding dexterity distribution is shown in Figures 4(b) and (c), respectively.

The dexterity distribution across the workspace cross-section in XZ plane at $y = 0$ for different segment lengths is shown in Figure 8. This simulation is conducted for different sets of robot designs with varying segment lengths by keeping the total robot length ($S_1 + S_2 + S_3$) constant, say, 150 mm of three-segment continuum robots, for which proposed data density-based measure of dexterity $\rho_{(area)}$ is computed. Dexterity measures across the workspace based on Eq. (12), red: high dexterity, blue: low dexterity region. The color bar shows the workspace’s minimum to maximum data density range which is the proposed measure of dexterity for a workspace of reachable point. The same variants of the continuum robot designs are used to compute the dexterity distribution across the cross-sectional workspace in the XZ plane at $y = 0$ using DSA ($D(k)$) and manipulability index($w_1(k)$) based approaches as shown in Figure 9 and Figure 10 respectively.

In the second simulation, the study focuses on analyzing the effect of segment length of three segment robots on dexterity distribution. Therefore, we quantified the terms dexterity distribution property index σ , dexterity index, and Manipulability index distribution property (w_g) for workspace cross-section in XZ plane at $y = 0$ for which the simulation criteria are given in Table IV. The nonzero length of the first and second is varied with a step size of 5mm by keeping the distal segment length as constant $S_3 = 30mm$. For each set of segment lengths, the dexterity distribution property index, dexterity index, and manipulability index distribution property are computed and shown in Figure 11.

4. Results

Simulation results show the dexterity distribution across the workspace using various measures of dexterity for comparison, which are described in Table V. The dexterity distribution for the 3D workspace is shown in Figure 7 by computing the dexterity on reachable points in the workspace cross-section at

Table IV. Simulation criteria to analyze the effect of segment lengths on dexterity distribution property index (σ) of three segment robot with 180° bending capability.

Segment length range	ΔS (mm)	Range θ (rad)	Range ϕ (rad)	Samples in each
$S_1 \in [5, 115]$	5	$[-\pi, \pi]$	$[0, 2\pi]$	10^8
$S_2 \in [115, 5]$	5	$[-\pi, \pi]$	$[0, 2\pi]$	10^8
$S_3 = 150 - (S_2 + S_1)$	–	$[-\pi, \pi]$	$[0, 2\pi]$	10^8

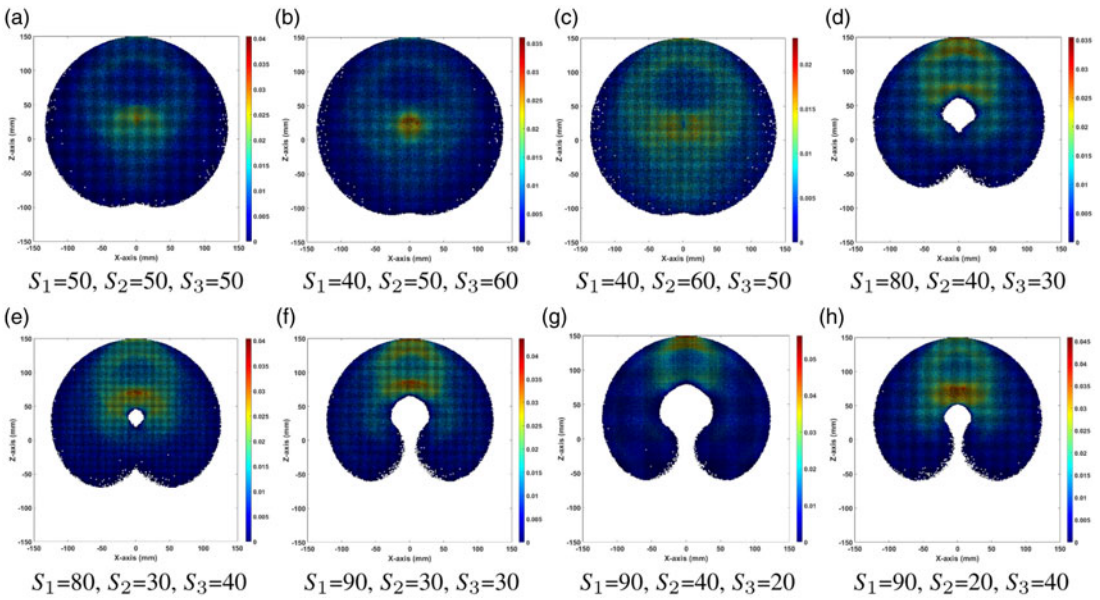


Figure 9. Dexterity (D) distribution across the workspace cross-section in XZ plane at $y = 0$ using DSA ($D(k)$) based approach.

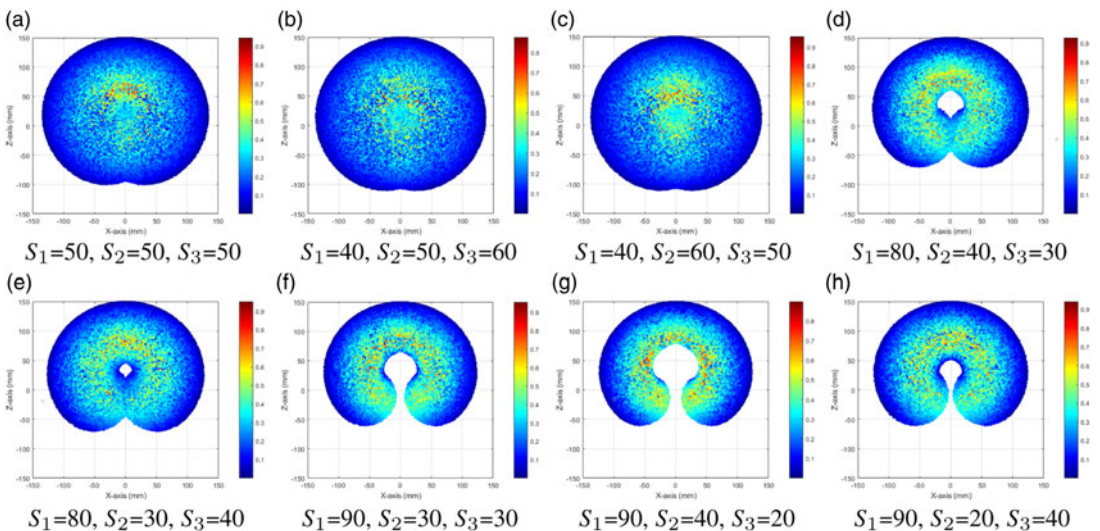


Figure 10. Manipulability index (w_1) distribution across the workspace cross-section in XZ plane at $y = 0$ based on manipulability approach.

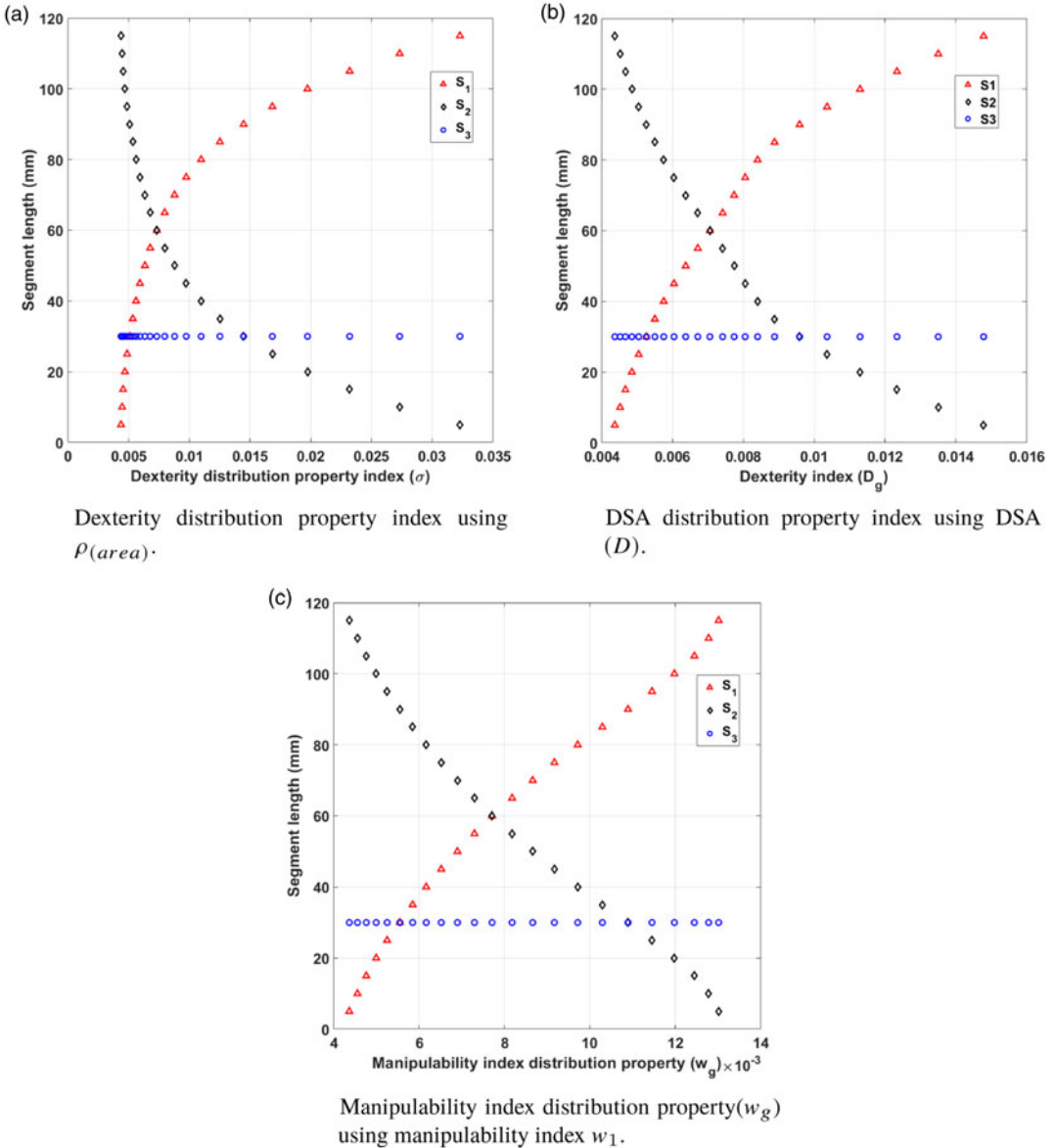


Figure 11. Effect of segment length variation on dexterity distribution property measure across the workspace cross-section in XZ plane at $y = 0$ based on (a) data density-based approach, (b) DSA-based approach, and (c) manipulability based approach, respectively.

vertical plane through the z-axis in Figure 7(a), each at 45 degrees and non-symmetric 3D workspace in Figure 7(b) and (c). A nonsymmetric workspace of a continuum robot is possible when we assign the actuation constraints to the configuration space variable. We need the same number of configuration samples for the same difference in the configuration range. The simulation results are obtained based on the proposed measure of dexterity and a dexterity distribution property in the above section for a set of segment lengths of the three-segment continuum robots in Figure 8. The results are compared with the dexterity distribution across the workspace cross-section based on DSA [15] [24] as shown in Figure 9

Table V. Comparison of simulation results.

Workspace type	Method	Measure of dexterity	Figure no.	Remarks	Number of computation steps	Computation time (sec.)
3D using workspace crosssection	Proposed	$\rho_{(area)}(k)$	Figure 7(a)	Applicable for symmetric workspace	-	3924
3D	Proposed	$\rho_{(vol)}(k)$	Figure 7(b),(c)	Visualize the relative dexterity	-	2440
Workspace crosssection in XZ pane $y = 0$	Proposed	$\rho_{(area)}(k)$	Figure 8	Dexterity distribution similar to DSA and manipulability index	4	1024.5
Workspace crosssection in XZ pane $y = 0$	Dexterous solid angle	DSA ($D(k)$)	Figure 9	Proposed and DSA approach have better visualization to dexterity distribution	7	Not available
Workspace crosssection in XZ pane $y = 0$	Manipulability index	$w_1(k)$	Figure 10	Effect of segment lengths on dexterity distribution observed.	7	Not available

and manipulability-based approach in Figure 10. The obtained dexterity distribution property across the workspace by the proposed measure is very similar to the dexterity distribution property based on the DSA-based measure of dexterity. Both the simulations, of which the conditions are listed, are carried out in MATLAB 2019b on a WINDOWS 10 pro-64-bit platform with Intel Core i5-6500 3.20 GHz CPU and 12.0 GB RAM. The number of computation steps required by each method in this paper is included in Table V. The DSA-based approach involves seven steps: configuration parameters generation (Monte Carlo), forward kinematics position, orientation, discretizing the workspace into patches, service sphere, DSA, and dexterity index. The manipulability-based approach also involves seven steps: configuration parameters generation (Monte Carlo), forward kinematics position, Jacobian, manipulability, singular values, manipulability index, and manipulability index distribution property. In contrast, the proposed data-density-based approach involves only four steps: configuration parameters generation (Monte Carlo), forward kinematics position, data density-based measure of dexterity, and dexterity distribution property index. Using the proposed approach, the computation time to determine dexterity distribution across the workspace cross-section in the XZ plane at $y = 0$ from 10^8 uniformly distributed random configuration parameter samples for the robot's full range of motion (robot workspace) is 1024.517 s. Meanwhile, the computation time is 2440.1176 s to compute the dexterity distribution for a nonsymmetric 3D workspace using 10^6 uniformly distributed random configuration parameter samples.

In the above simulation results from three different methods, it is observed that the proposed method is identical to the DSA-based approach. Further, the simulation is extended to compare the effect of the segment length variation (with step ΔS) on the data density-based, DSA-based, and manipulability-based measure of dexterity. The simulation is performed using the parameters given in Table IV for three segment continuum robots to verify the proposed measure of dexterity distribution property as dexterity distribution property index. Here, we are analyzing the effect of the length of the first and second segments on the data density-based dexterity distribution property index by following the constraint $S_3 = 150 - (S_1 + S_2)$, where the distal segment length S_3 is kept constant and compared with the dexterity index (D_g) and manipulability index distribution property (w_g) based on the DSA and manipulability based approach, respectively, using the same simulation criteria.

A scaling factor is used to normalize the dexterity distribution property index and manipulability index distribution property of all the design variants in one scale, as shown in Figure 11. Based on the comparison, the dexterity distribution property index corresponds well with the dexterity index for segment length variation. Simulation results are presented in Figure 11 to show how dexterity distribution property index, dexterity index, and manipulability index distribution property vary according to segment length in Figures 11(a), (b) and (c), respectively. Figure 11(c) shows the effect of segment length variation on additional terms as manipulability index distribution property (w_g) measure across the workspace using manipulability-based approach for comparison. The results show the manipulability approach based manipulability index distribution property follows a similar trend for segment length variation as seen in the case of DSA and data density approach, where the dexterity index (D_g) and dexterity distribution property index (σ) will be larger for a large length of the proximal segment. As seen in Figure 8, it reaffirms that higher dexterity is obtained for some designs at the distant peripheral region of the workspace and thereby increases the effectiveness of the robotic device usage. Therefore, the small change in dexterity distribution property index when $S_1 < S_2$ and increases as $S_1 > S_2$ because, when $S_2 > S_1$, dexterity is maximum nearer to the base of the robot and vice versa, as shown in Figure 11.

5. Limitations and future recommendations

This proposed method required a large number of position data inside the workspace for better visualization of the dexterity distribution. The method is more suitable for computing the dexterity analysis for hyper-redundant robots, for which inverse kinematics is also challenging. The generated configuration sample for the analysis must be uniform random samples. For robots with less than 6DOF, there is a

high computation cost. The method is computationally costly in order to compute the data density-based dexterity for 3D workspace.

There is also the scope to improve the method's computation efficiency for many reachable points in the 3D workspace. This method will help researchers explore the dexterity analysis of new different hyper-redundant robot mechanisms without solving its complex inverse kinematics. The design optimization of the mechanisms is based on the workspace and dexterity distribution property. The robot design having a high dexterity distribution property index will make it easier to do complex tasks. The dexterity distribution will help to reposition the base of the redundant manipulators.

6. Conclusion

In this paper, we present two new measures of dexterity based on a data density approach – one for dexterity at a point and the second to quantify the dexterity distribution across the workspace cross-section in the XZ plane at $y=0$ due to its symmetry about the Z-axis. The positional data density is considered as the dexterity at a point, whereas the dexterity distribution property index is computed as the sum of the positional data density, weighted by the distance of the points. The reachable positions in the workspace are calculated by forward kinematics using the sample configuration parameters obtained from the Monte Carlo simulation for the given range of configuration variables as input parameters. The dexterity distribution property index provides a measure of dexterity distribution property across the workspace cross-section. Our results show that the dexterity distribution property index also varies with the length of the continuum segments like other existing measures of similar methods to verify.

Qualitative and quantitative comparisons of the method with similar works were carried out. Qualitatively, we compare this with previous methods based on the input parameters, quantities computed, and outcomes of the dexterity measures. Quantitatively, we compare the results with that of the DSA approach and manipulability index-based approach. We observe that the dexterity distribution results from the three approaches are similar, but our method is simpler and easier as it uses only reachable positions in the workspace cross-section to compute the proposed dexterity measure. We also present the dexterity computation in 3D for symmetric and asymmetric cases.

In a second simulation, the results of the dexterity and dexterity distribution property index are compared with the DSA-based and manipulability-based approach by computing the dexterity index and manipulability index distribution properties, respectively. Here, it is also observed that all have similar responses to the segment length variation when distal segment length is fixed; however, the proposed method is simpler and easier to compute. Therefore, it can be directly used for dexterity analysis and design synthesis of hyper-redundant robots for dexterous manipulation. In the future, the computation efficiency of the method can be improved for 3D workspace.

Author contributions. SB and CSK conceived and designed the study. SB did the simulations, and RKH helped to do the analysis. SB wrote the article, and RKH and CSK contributed to the revision process.

Financial support. This research received no specific grant from any funding agency, commercial, or not-for-profit sectors.

Competing interest. The authors declare no conflicts of interest exist.

Ethical approval. Not applicable.

References

- [1] B. Davies, "A review of robotics in surgery," *Proc. Inst. Mech. Eng. Part H: J. Eng. Med.* **214**(1), 129–140 (2000).
- [2] M. Russo, S. M. H. Sadati, X. Dong, A. Mohammad, I. D. Walker, C. Bergeles, K. Xu and D. A. Axinte, "Continuum robots: An overview," *Adv. Intell. Syst.* **5**(5), 2200367 (2023). doi: [10.1002/aisy.202200367](https://doi.org/10.1002/aisy.202200367).
- [3] M. Ding, X. Zheng, L. Liu, J. Guo and Y. Guo, "Collision-free path planning for cable-driven continuum robot based on improved artificial potential field," *Robotica* **42**(5), 1350–1367 (2024).

- [4] J. Burgner-Kahrs, D. C. Rucker and H. Choset, "Continuum robots for medical applications: A survey," *IEEE Trans. Robot.* **31**(6), 1261–1280 (2015).
- [5] K. Xu, J. Zhao and X. Zheng, "Configuration comparison among kinematically optimized continuum manipulators for robotic surgeries through a single access port," *Robotica* **33**(10), 2025–2044 (2015).
- [6] 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).
- [7] H. M. Le, T. N. Do and S. J. Phee, "A survey on actuators-driven surgical robots," *Sens. Actuators A: Phys.* **247**, 323–354 (2016).
- [8] J. Barrientos-Diez, X. Dong, D. Axinte and J. Kell, "Real-time kinematics of continuum robots: Modelling and validation," *Robot. Com-Integr. Manuf.* **67**, 102019 (2021).
- [9] Z. Xu, "Research on kinematics and attitude control model of a surgical interventional catheter," *Int. J. Adv. Robot. Syst.* **16**(5), 1729881419874639 (2019). doi: [10.1177/1729881419874639](https://doi.org/10.1177/1729881419874639).
- [10] Y. Haga, T. Mineta, W. Makishi, T. Matsunaga and M. Esashi, "Active Bending Catheter and Electric Endoscope Using Shape Memory Alloy," **In: Shape Memory Alloys** (IntechOpen, 2010)
- [11] M. R. A. Kadir, D. E. O. Dewi, M. N. Jamaludin, M. Nafea and M. S. M. Ali, "A multi-segmented shape memory alloy-based actuator system for endoscopic applications," *Sens. Actuators A: Phys.* **296**, 92–100 (2019).
- [12] C. Cheng, J. Cheng and W. Huang, "Design and development of a novel sma actuated multi-dof soft robot," *IEEE Access* **7**, 75073–75080 (2019).
- [13] Z. Li, L. Wu, H. Ren and H. Yu, "Kinematic comparison of surgical tendon-driven manipulators and concentric tube manipulators," *Mech. Mach. Theory* **107**, 148–165 (2017).
- [14] L. Wu, R. Crawford and J. Roberts, "Dexterity analysis of three 6-dof continuum robots combining concentric tube mechanisms and cable-driven mechanisms," *IEEE Robot. Autom. Lett.* **2**(2), 514–521 (2016).
- [15] M. Badescu and C. Mavroidis, "New performance indices and workspace analysis of reconfigurable hyper-redundant robotic arms," *Int. J. Robot. Res.* **23**(6), 643–659 (2004).
- [16] Z. Li, H. Ren, P. W. Y. Chiu, R. Du and H. Yu, "A novel constrained wire-driven flexible mechanism and its kinematic analysis," *Mech. Mach. Theory* **95b**, 59–75 (2016b).
- [17] J. Wang and H. Y. Lau, "Dexterity analysis based on jacobian and performance optimization for multi-segment continuum robots," *J. Mech. Robot.* **13**(6), 061012 (2021).
- [18] J. K. Salisbury and J. J. Craig, "Articulated hands: Force control and kinematic issues," *Int. J. Robot. res.* **1**(1), 4–17 (1982).
- [19] P. Cardou, S. Bouchard and C. Gosselin, "Kinematic-sensitivity indices for dimensionally nonhomogeneous jacobian matrices," *IEEE Trans. Robot.* **26**(1), 166–173 (2010).
- [20] T. Yoshikawa, "Manipulability of robotic mechanisms," *Int. J. Robot. Res.* **4**(2), 3–9 (1985).
- [21] M. Bamdad and M. M. Bahri, "Kinematics and manipulability analysis of a highly articulated soft robotic manipulator," *Robotica* **37**(5), 868–882 (2019).
- [22] J. Lenarcic, T. Bajd and Stanišić M. M., "*Robot Mechanisms*. vol. **60** (Springer Science and Business Media, 2012) .
- [23] A. Mehrkish and F. Janabi-Sharifi, "Grasp synthesis of continuum robots," *Mech. Mach. Theory* **168**, 104575 (2022).
- [24] K. A. Abdel-Malek and B. Paul. "The Dexterous Solid Angle of Robotic Manipulators with a Spherical Wrist," **In: International Design Engineering Technical Conferences and Computers and Information in Engineering Conference**, American Society of Mechanical Engineers, **12860**, (1994) pp. 341–350.
- [25] J. Burgner-Kahrs, H. B. Gilbert, J. Granna, P. J. Swaney and R. J. Webster. "Workspace characterization for concentric tube continuum robots," **In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems**, IEEE, (2014) pp. 1269–1275.
- [26] S. Cobos-Guzman, D. Palmer and D. Axinte, "Kinematic model to control the end-effector of a continuum robot for multi-axis processing," *Robotica* **35**(1), 224–240 (2017).
- [27] H. Wu, J. Yu, J. Pan, G. Li and X. Pei, "Crrik: A fast heuristic algorithm for the inverse kinematics of continuum robot," *J. Intell. Robot. Syst.* **105**(3), 55 (2022).
- [28] O. M. Gergert and D. Y. Kolpashchikov, "Inverse kinematics for multisection continuum robots with variable section length," **In: Conference on Creativity in Intelligent Technologies and Data Science**, Springer (2023) pp. 131–142.
- [29] O. M. Omisore, S. Han, L. Ren, A. Elazab, L. Hui, T. Abdelhamid, N. A. Azeez and L. Wang, "Deeply-learned damped least-squares (dl-dls) method for inverse kinematics of snake-like robots," *Neural Net.* **107a**, 34–47 (2018a).
- [30] O. M. Omisore, S. P. Han, L. X. Ren, Z. C. Zhao, L. Wang and W. Lei, et al. "Adaptation of translated frame-based approach for forward kinematics in a radiosurgical snake-like robot," **In: 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)**, IEEE (2018b) pp. 3669–3672.
- [31] S. Bamoriya and C. S. Kumar, "Kinematics of three segment continuum robot for surgical application," **In: Machines, Mechanism and Robotics: Proceedings of iNaCoMM 2019**, Springer (2022) pp. 1011–1021.
- [32] A. B. Boutchouang, A. Melingui, J. M. Ahanda, O. Lakhali, F. B. Motto and R. Merzouki, "Forward kinematic modeling of conical-shaped continuum manipulators," *Robotica* **39**(10), 1760–1778 (2021).
- [33] B. Ouyang, Y. Liu, H.-Y. Tam and D. Sun, "Design of an interactive control system for a multisection continuum robot," *IEEE/ASME Trans. Mech.* **23**(5), 2379–2389 (2018).
- [34] P. Rao, Q. Peyron, S. Lilge and J. Burgner-Kahrs, "How to model tendon-driven continuum robots and benchmark modelling performance," *Front. Robot. AI* **7**, 630245 (2021).
- [35] C. F. Costa and J. C. Reis, "End-point position estimation of a soft continuum manipulator using embedded linear magnetic encoders," *Sensors* **23**(3), 1647 (2023).
- [36] B. A. Jones and I. D. Walker, "Kinematics for multisection continuum robots," *IEEE Trans. Robot.* **22**(1), 43–55 (2006).

- [37] Y. Tian, M. Luan, X. Gao, W. Wang and L. Li, “Kinematic analysis of continuum robot consisted of driven flexible rods,” *Math. Probl. Eng.* **2016**(1), 6984194 (2016).
- [38] Z. Li and R. Du, “Design and analysis of a bio-inspired wire-driven multi-section flexible robot,” *Int. J. Adv. Robot. Syst.* **10**(4), 209 (2013).
- [39] J. Li, F. Zhao, X. Li and J. Li, “Analysis of robotic workspace based on monte carlo method and the posture matrix,” **In: 2016 IEEE International Conference on Control and Robotics Engineering (ICCRE)**, IEEE, (2016a) pp. 1–5.
- [40] S. Mahadevan. Monte carlo simulation. Mechanical Engineering-New York and Basel-Marcel Dekker-, 123-146 (1997).
- [41] A. Alian, E. Zari, Z. Wang, E. Franco, J. P. Avery, M. Runciman, B. Lo, F. R. y Baena and G. Mylonas, “Current engineering developments for robotic systems in flexible endoscopy,” *Tech. Innovat. Gastroi. Endosc.* **25**(1), 67–81 (2023).
- [42] I. Mansouri and M. Ouali, “A new homogeneous manipulability measure of robot manipulators, based on power concept,” *Mechatronics* **19**(6), 927–944 (2009).
- [43] K. Abdel-Malek, W. Yu and J. Yang, “Placement of robot manipulators to maximize dexterity,” *Int. J. Robot. Autom.* **19**(1), 6–14 (2004).
- [44] K. Leibrandt, P. Wisanuvej, G. Gras, J. Shang, C. A. Seneci, P. Giataganas, V. Vitiello, A. Darzi and G.-Z. Yang, “Effective manipulation in confined spaces of highly articulated robotic instruments for single access surgery,” *IEEE Robot. Autom. Lett.* **2**(3), 1704–1711 (2017).
- [45] F. Du, G. Zhang, Y. Xu, Y. Lei, R. Song and Y. Li, “Continuum robots: Developing dexterity evaluation algorithms using efficient inverse kinematics,” *Measurement* **216**, 11–2925 (2023).
- [46] R. K. Hota and C. Kumar, “Effect of hand design and object size on the workspace of three-fingered hands,” *Mech. Mach. Theory* **133**, 311–328 (2019).
- [47] P. Berthet-Rayne, K. Leibrandt, K. Kim, C. A. Seneci, J. Shang and G.-Z. Yang, “Rolling-joint design optimization for tendon driven snake-like surgical robots,” **In: 2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)**, IEEE (2018) pp. 4964–4971.
- [48] H. Dong, Z. Du and G. S. Chirikjian, “Workspace density and inverse kinematics for planar serial revolute manipulators,” *Mech. Mach. Theory* **70**, 508–522 (2013).
- [49] F. Qi, F. Ju, D. M. Bai and B. Chen, “Kinematics optimization and static analysis of a modular continuum robot used for minimally invasive surgery,” *Proc. Inst. Mech. Eng. Part H: J. Eng. Med.* **232**(2), 135–148 (2018).
- [50] P. E. S. Dupont, N. Choset, H and C. Rucker, “Continuum robots for medical interventions,” *Proc. IEEE* **110**(7), 847–870 (2022).