Real-Time Kinematics of Continuum Robots: Modelling and Validation

Jorge Barrientos-Díez[a], Xin Dong[a], Dragos Axinte[a,*], James Kell[b]

[a] *University of Nottingham, Nottingham, NG8 1BB, United Kingdom*
[b] *Rolls-Royce Plc, Derby, DE24 8BJ, United Kingdom*

Abstract

Kinematics of flexible backbone continuum robots is highly non linear and its complexity quickly escalates with the number of sections of the robot, which is usually more than three. This paper introduces a kinematic modelling of actuation and configuration spaces that greatly simplifies the computational requirements compared to the commonly used Piecewise Constant Curvature Kinematics which results in a faster algorithm at a rate proportional to the number of sections. This new algorithm is firstly developed for Twin Pivot Compliant Joint continuum robots but then extended to a general single neutral axis backbone configuration, both achieving a very low error of approximation (0.7% for the prototype developed), which results in several advantages such as the avoidance of highly non-linear functions and singularities, great reduction of computational complexity and an user-friendly graphical representation to help operation and status monitoring of this kind of robots. Moreover, a slender, small diameter and hyper-redundant (175 mm length, 6 mm diameter, 10 Degrees of Freedom) continuum robot prototype is developed and tested in a real-case industrial application for inspection and repair of aero engines in order to validate the proposed model.

Keywords: Kinematics, Continuum robots, Tendon driven mechanism

*Corresponding author. Tel.: +44 1159 514117.
Email addresses: jorge.barrientosdiez@nottingham.ac.uk (Jorge Barrientos-Díez), xin.dong@nottingham.ac.uk (Xin Dong), dragos.axinte@nottingham.ac.uk (Dragos Axinte), james.kell@rolls-royce.com (James Kell)

Nomenclature

β	Bending angle of a single TPCJ segment.
δ, δ_m	Cable offset angle (of the m-th section).
$\mathbf{A}$	Cable - Joint disposition matrix.
$\mathbf{R_{x,y,z}}$	Rotation matrix around the axis (x, y, z).
$\mathbf{T_{x,y,z}}$	Translation matrix along the axis (x, y, z).
ϕ	Direction angle of a constant curvature section.
θ	Bending angle of a constant curvature section.
h	Guiding disc height.
$K(f, p)$	Executions of algorithm f with respect to p.
L	Joint length of a single segment.
l, l_i	Cable length (of the i-th cable).
l', l'_i	Approximation of the i-th cable length.
M	Number of sections in a robot.
N	Number of segments in a section
r_c	Cable disposition radius.
r_{disc}	Guiding disc external radius.

1. Introduction

Continuum robots have been subject of research in the last decades due to their advantages against classic rigid link manipulators in certain tasks where the working environments are highly constrained.

The main characteristics that these applications share is a target workspace

located far from the robot access point, and a high number of obstacles between them. Continuum robots are ideal for these tasks since they have the ability to acquire convolute shapes that allow them to avoid these obstacles.

The environmental requirements for this kind of robots can then be identified as follows: (i) Small cross section in order to be able to pass through small gaps along their path, (ii) high length over cross-section aspect ratio (or slenderness ratio $length/diameter$) due to the remoteness of the area of interest and lastly, (iii) hyper-redundancy of the robot to have enough Degrees of Freedom (DoFs) to perform the task and to navigate through the environment to reach the area of interest.

These requirements define the characteristics that make continuum robots unique. First, they are usually composed of a set of compliant joints which gives them great flexibility in order to navigate through environments with high obstacle density. Also, due to the size restrictions of the environment, continuum robots tend to be remotely actuated. Thus, the actuators are not placed inside the robot body, but externally to reduce the size of the robot. The actuation of the robot is then transmitted to the robot in different ways, such as with pneumatic actuators [1] or tendon driven as presented in [2]. As commented in [3], after reviewing several continuum robot prototypes, tendon driven or variations of it is the most common actuation strategy for this type of robots.

One of the main applications that has pushed progress in this kind of robots is the surgical interventions [4]. Minimally invasive surgery has motivated robotics to be miniaturised and several solutions for it have been proposed [5]. Those that can be defined as "continuum robots" include concentric tube constructions as presented in [6] that have been successfully used in several applications such as heart surgery [7]. Tendon-driven continuum robots have also been developed for medical applications with the inconvenience of having a larger outer diameter or having to sacrifice the number of DoFs in order to keep this dimension small enough. It is the case of the prototype presented in [8], a robot of 1.7 mm in diameter and 60 mm long which suppose a high slenderness ratio of 35.3 but is very limited in mobility, with only 2 DoFs. The miniaturisation of this kind of robots is a challenge and even in these reports the experiments were carried out with a 10:1 scale prototype [8] and some preliminary tests were made with a 2:1 prototype [8],[9].

Regarding industry, the applications tend to be less restrictive than in the medical field in terms of outer diameter but need of a larger number of

DoFs or higher payload. For example, snake-like robots are used for inspection and repair of nuclear facilities as shown in [10], one of them having 20 DoFs but an external diameter of 40 mm, which prevents its use in very confined environments. However, there are examples of industrial applications of small diameter and high length to diameter ratio such as the Tendril robot for inspection of NASA equipment [11] being 10 mm in diameter, 1 m in length (slenderness ratio 100) but again at the expense of its mobility having 4 DoFs, or those created to carry out inspection and repair operations of gas turbine engines for aviation such as the ones presented in [12, 13, 14]. There is a general lack of highly slender, low diameter and high mobility continuum robots that can tackle complex tasks in narrow environments with remote areas of operation where the aforementioned designs will not be able to perform due to large diameter, low reach or lack of DoFs.

Moreover, continuum robots also present challenges in terms of kinematics. The shape modelling of the compliant joints is not as straightforward as rigid links, since the external actuation adds an extra step for computing the kinematics and the hyper-redundancy makes real time computing difficult since the number of possible solutions and operations needed increases with the number of extra DoFs.

There are several methods to model the kinematics of continuum robots such as those that apply Cosserat Rod theory [15], static analysis [16], minimisation of the energy stored in the elastic element [17], and constant curvature [18]. Cosserat rod approaches tend to be more precise when external loads or gravitational forces are not negligible but they are more complex than a constant-curvature approach. Even if results shown in [19] prove that these algorithms can run quite fast ($\sim$ 8 ms for a single section robot), the processor used for benchmark this algorithm is a high-performance CPU which makes it unrealistic for cost-effective real-time hardware. Similarly, the static analysis presented in [16] yields good results to characterize the robot pose under external forces, but its solving has to be numeric and computational time is high for real time control ($\sim$ 170 ms) again using a high-performance CPU. The energy based formulation of [17] is applied to concentric tube robots and achieves small error, but the model has either be solved numerically or taking only two concentric tubes (4 DoFs) to be solved analytically. The aim of this model is to achieve a good characterisation of the robot shape but is not applied to control it.

Lastly, Piecewise Constant Curvature Kinematics (PCC) divides the robot in kinetically independent parts and approximates the bending of each of

this parts to the circular arc with constant length. This results in a simpler model, and it is one of the most used ones. Many of the different approaches and contributions for this method are summarized in [3] where the distinction between actuation, configuration and task spaces and their kinematics are discussed. This isolates the kinematics of the robot shape in 3D space from how this shape is changed by the robot actuators, allowing to model separately the type of actuation (pneumatic, tendon driven, etc..).

Regarding the implementation of this kinematic modelling for real-time operation, it is worth to mention the work presented in [20], where a PCC approach is proposed taking into account the physical limitations of actuator lengths. For this, it is assisted by a 3D representation only used to identify the workspace reach derived from the actuator limitations. Moreover, it presents a "tangle/untangle" algorithm to tackle the problem of the routing of the driving tendons in a multisection continuum robot. However, this algorithm contains highly non-linear functions with asymptotic values and the number of operations needed to compute the effective driving tendon lengths rapidly increases with the number of sections of the robot, making this method not ideal to be utilised in real-time control.

In this paper, a new approach to the kinematics from actuation to configuration space for tendon driven continuum robots is presented based on a linearisation of PCC. This approach allows both for direct and inverse transformations, resulting in a much less resource-intensive algorithm for the robot controller when compared with previous approaches making it more appropriate for real-time control of multi-DoF continuum robots. The developments of this paper are done step by step for Twin Pivot Compliant Joints design presented in [21], but the model is extended to the more broadly used single backbone design with the configuration parameters of direction and bending angles. Moreover, a novel 2D representation of the actuation to configuration space kinematics is presented that can be used not only off-line to represent the physical limitations of the robot but also as a powerful visual aid to operate and monitor continuum robots as well as troubleshoot possible problems during their operation. Finally a high mobility, small diameter and slender continuum robot (10 DoFs, 6 mm diameter, 175 mm length) is designed using the Twin Pivot Compliant Joint structure, on which the new algorithm presented in this paper is implemented. This prototype is then tested in a relevant industrial task with a crammed environment: Inspection of the latter compressor stage of an aero-engine.

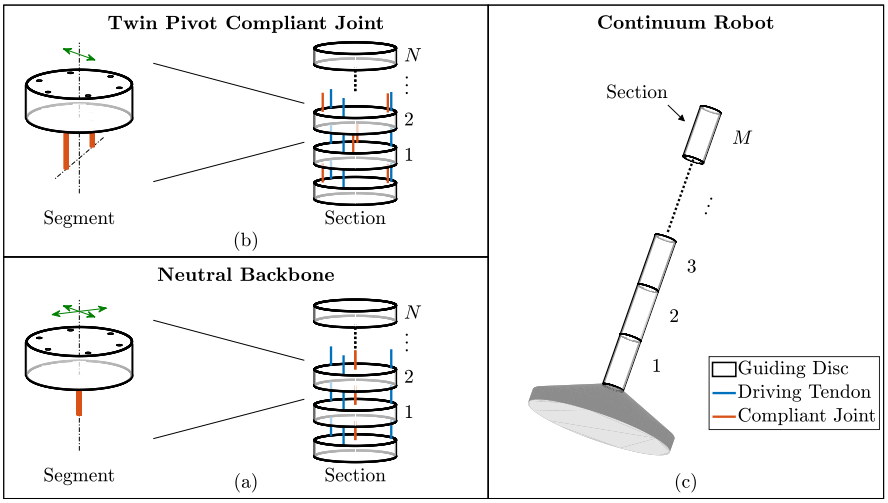

Figure 1: Schematic of construction for continuum robots. Continuum robots are made of M sections stacked consecutively. These sections can be either composed of N neutral backbone segments stacked or of N TPCJ segments alternatively stacked $\pi/2$.

2. Principles of tendon driven continuum robots and Piecewise Constant Curvature Kinematics.

In order to properly explain the contributions of this paper, a brief description of the design basis of tendon driven continuum robots, both Twin Pivot Compliant Joint (TPCJ) and Neutral Backbone has to be done, as well as an outline of the PCC kinematic modelling of these robots.

2.1. Construction of Tendon Driven continuum robots.

The basic unit of a tendon driven continuum robot is called segment. A segment consists of a rigid body, known as guiding disc, and an elastic body, known as backbone. The main purpose of the guiding disc is to route (or guide) the driving tendons through the body of the robot, and to serve as a link to constrain the flexible backbone. The function of the flexible backbone is to provide compliance that gives the robot the ability to adopt different poses. In the case of neutral backbone continuum robot, which is the most used design, this flexible backbone is a single element placed in the neutral axis of the guiding disc and it can deform in two principal directions, giving the segment two DoFs (Fig. 1a). However, the disadvantage of this design is that due to its geometry, it is prone to suffer torsional deformations along

the main axis and it is not possible to deliver services for the end effector through the centreline of the robot. To solve both problems, TPCJ robots alter the design of the backbone, having two elastic elements diametrically opposed around the main axis (Fig. 1b), providing high torsional rigidity to the segment, and freeing space in the neutral axis of the robot that can be used to deliver wires, fibre optics, and others. However, this construction limits the DoFs of the segment to a single one, being the bending plane of the segment perpendicular to the position of both flexible elements.

Stacking N segments consecutively and adding driving tendons results in a section. The driving tendons (or cables) are the elements that transmit motion to the structure. Each section has a set of usually 3 or 4 driving tendons routed through the guiding discs and evenly distributed around its main axis. By properly pulling or releasing the driving tendons, the flexible backbone elements are deformed, in order to control the pose of the robot. All of the segments of each section are kinetically linked to each other and actuated by the same driving tendons, evenly distributing the deflection of the elastic elements (ideally). This implies that sections are the smallest kinetically independent units of a continuum robot. In the case of single neutral backbone construction, stacking segments results directly in two DoFs, since each segment already has these two DoFs (Fig. 1a). On the other hand, TPCJ segments have only one DoF, so they are stacked consecutively alternating their bending planes $\pi/2$ in order to have 2 DoFs per section, and thus achieving a workspace with a similar 3D surface than its neutral backbone counterpart (Fig. 1b).

Lastly, a Tendon Driven continuum robot is built by successively stacking M sections (Fig. 1c) which affects the total number of DoFs of the robot. Since the driving tendons of the distal (tip) section have to be routed through the rest of previous sections to the base of the robot, where the actuators are placed, the sets of 3 or 4 cables of each section cannot be placed where the ones of the previous sections are. This is typically solved by applying an angular offset δ (Fig. 2b) around the main axis of the robot to all the driving cables of a same section, with each cable set having a different one.

2.2. Piecewise Constant Curvature Kinematics.

Piecewise Constant Curvature Kinematics [18] supposes that the flexible backbone deforms in an uniform way, adopting a circular arc shape of constant length L when the length of the driving tendons l_i is modified. This driving tendon is routed at a distance from the main axis r_c and an offset

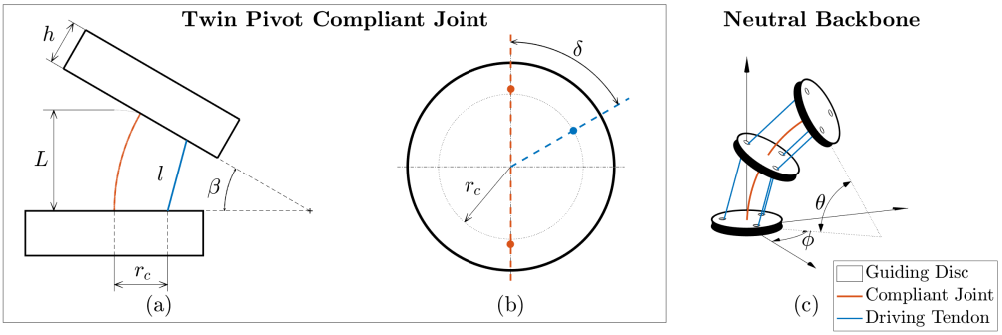

Figure 2: Kinematic parameters of Neutral Backbone and Twin Pivot Compliant Joint continuum robots.

angle from a reference plane δ (Fig. 2a,b), which usually is the common offset of the cables of the section δ_m plus the angle of each cable within the rest of the cables of a section, spaced $2\pi/3$ for a 3-cable configuration and $\pi/2$ for a 4-cable configuration.

In the case of neutral backbone robots, its deformation is described by two angles: the circular arc angle defined as bending angle θ, and the angle of the plane where the robot is bent with respect a reference plane, defined as direction angle ϕ (Fig. 2c). Since all the segments of a same section are supposed to deflect uniformly, the bending angle θ is usually referred to a whole section, being θ/N the bending angle of each flexible element that composes the section. This DoF is often described by the curvature κ, which is θ/L. The cable length of a single driving tendon along a section composed of N segments is then defined by (1) as adapted from [18].

$$l_i = 2 \cdot sin\left(\frac{\theta}{2N}\right) \cdot \left(\frac{L \cdot N}{\theta} - r_c \cdot sin\left(\phi + \delta_i\right)\right) \tag{1}$$

On the other hand, TPCJ segments have only one DoF, described by the bending angle β (Fig. 2a), whereas the rest of the parameters still the same, but when building a section the bending plane of each segment is alternated by $\pi/2$ resulting in two bending angles (β_1, β_2). The length l_i of a tendon that runs through a TPCJ robot section is then defined by (2).

$$l_i = N \cdot \sum_{j=1}^{2} \left(\frac{L}{\beta_j} - r_c\right) \cdot sin\left(\frac{\beta_j}{2}\right) \tag{2}$$

Finally, to obtain the pose in the Cartesian space of the robot from the configuration angles, spatial transformations can be made segment by segment to achieve the homogeneous transformation matrix for a whole section being the one for neutral backbone shown in (3) and the one for a TPCJ section as described in (4).

$$\mathbf{R_z}(\phi) \cdot \left[\mathbf{T_y}\left(\frac{L \cdot N}{\theta}\right) \cdot \mathbf{R_x}\left(\frac{\theta}{N}\right) \cdot \mathbf{T_y}\left(\frac{-L}{\theta}\right) \cdot \mathbf{T_z}(h) \right]^N \tag{3}$$

$$\begin{aligned} &\left[\mathbf{T_y}\left(\frac{L}{\beta_2}\right) \cdot \mathbf{R_x}(\beta_2) \cdot \mathbf{T_y}\left(\frac{-L}{\beta_2}\right) \cdot \mathbf{T_z}(h) \cdot \mathbf{R_z}\left(\frac{-\pi}{2}\right) \cdot \right. \\ &\left. \mathbf{T_y}\left(\frac{L}{\beta_1}\right) \cdot \mathbf{R_x}(\beta_1) \cdot \mathbf{T_y}\left(\frac{-L}{\beta_1}\right) \cdot \mathbf{T_z}(h) \cdot \mathbf{R_z}\left(\frac{\pi}{2}\right) \right]^{\frac{N}{2}} \end{aligned} \tag{4}$$

To compute the transformation of the whole robot the matrix of all its sections have to be successively multiplied. However, the developments of this paper are focused on the kinematics between actuation and configuration spaces, focusing mainly in (1) and (2).

The focus of this paper has been made in TPCJ continuum robots

3. New Kinematic Simplification of Piecewise Constant Curvature Kinematics.

Piecewise Constant Curvature kinematics has some disadvantages when applied to compute in real-time continuum robots with a large number of DoFs. Taking the PCC approach as a starting point, the first order of the Maclaurin series is applied to the Configuration to Actuation space Kinematics that evolves to a novel Linearised Segment Kinematics (LSK) modelling, which greatly simplifies the computation resources needed and avoiding singularities.

3.1. Existing computational challenges of Piecewise Constant Curvature Kinematics.

Equation (1) shows that each time a cable length in a particular section has to be calculated given the desired bending and direction angles, two sine functions and a reciprocal have to be computed. Furthermore, the domain of the bending angle θ includes zero which does not yield in a mathematical

asymptote ($\lim_{\theta \to 0} l_i(\theta) = L \cdot N$) but, when computing a numerical solution, a division by zero is made. Having to handle this case when implementing the kinematics in a controller results in less accurate results the closer the bending angle gets to this value, effect caused by round off errors.

Also, when computing the kinematics of a M-section robot, the cable lengths of a particular section have to be computed not only for that section, but also for all the previous ones where the cables pass through, as shown in section 3 of [20]. To do so, (1) has to be executed a number of iterations K defined by (5) in the case of a 3-cable configuration.

$$K(l_i, M) = 3 \cdot \sum_{m=1}^{M} m = \frac{3}{2} \cdot \left(M^2 + M\right) \tag{5}$$

Resulting in an algorithm of growth rate $\mathcal{O}(M^2)$, which becomes computationally expensive for real-time control when the number of sections increases.

Taking into account all of the above, it can be seen the benefits of having a simplified kinematics that overcomes the problems exposed.

3.2. Linearised Segment Kinematics (LSK) for Twin-Pivot Compliant Joint continuum robots.

To greatly simplify the kinematic modelling between actuation and configuration spaces, a linearisation can be applied to the kinematics that describes the most basic unit of a continuum robot (a segment). From this new approach, the equations that define the kinematics of a section and of a full robot can be easily derived, resulting in much more straightforward expressions.

Taking a single segment of TPCJ construction the relationship between the length of a generic driving cable l with the bending angle β supposing that the cable offset angle is $\delta = 0$ is as shown in (6).

$$l \geq 2 \cdot \left(\frac{L}{\beta} - r_c\right) \cdot sin\left(\frac{\beta}{2}\right) \tag{6}$$

Note that the inequality is due to the particular consideration that the robot is cable driven. Considering that cables do not stretch under tension, any bending angle value β that requires a greater cable length is unattainable, but any value that needs a smaller cable length is reachable since under compression they become slack.

The maximum bending angle of a full section tend to be large enough such that linearising trigonometric functions would cause a large error. However, this bending angle is obtained piecewise segment by segment by constraining the driving tendon through the guiding discs. Consequently, the angle that each segment has to bend in order to attain a large bending angle is considerably smaller (e.g. a section that bends 90° and has $N = 8$ segments needs for each segment to bend a maximum of $90/N = 11.25°$), thus a linearisation of the relationship between cable length and individual segment angle can be applied yielding a reasonably small error, which will be evaluated in Section 5.

In particular, calculating the first order of the Taylor expansion of (6) around the value $\beta = 0$ (Maclaurin series) results in (7). Both the original and the simplified functions are represented in Fig. 3 for their comparison.

$$l \approxeq l' \geqslant L - r_c \cdot \beta \tag{7}$$

With this approximation, the kinematics is greatly simplified, and existing problems of the constant curvature kinematics such as singular values, highly non-linear functions or non direct reversibility, get solved.

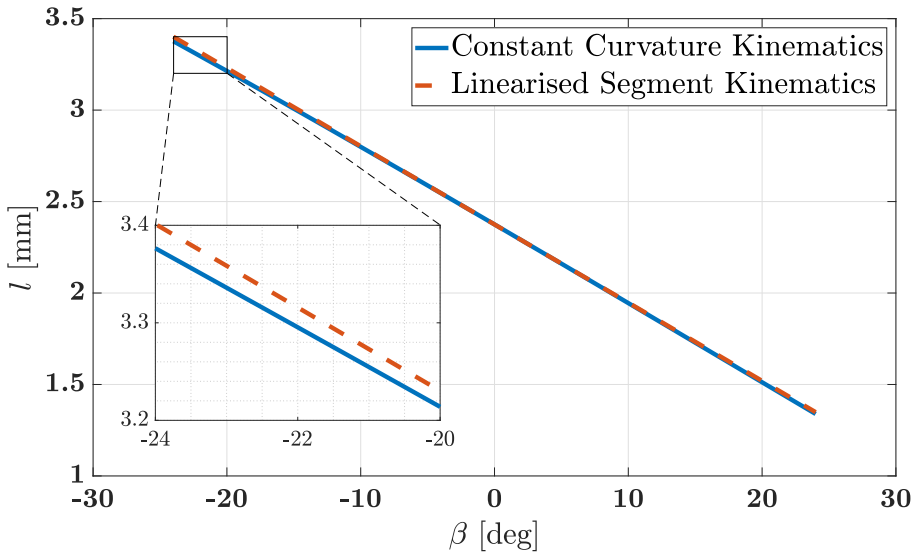

Figure 3: Constant Curvature Kinematics (blue) and Linearised Segment Kinematics (red) are overlaid to show their similarity. The max error is 0.7%. The robot parameters used are those shown in table 2.

3.3. Singular section kinematics as an outcome of the LSK approach.

To calculate the kinematics of a section, the offset angle of the cable δ from the joint plane of β (as shown in Fig. 2) has to be taken into account. Moreover, a tendon-driven robot section is composed of N segments which in the case of TPCJ continuum robots, is an even number and each segment alternates its bending angle direction $\pi/2\ rad$ from the previous segment around the main robot axis as seen in Fig. 1b. This results in two bending angles per section (β_1, β_2) with offset angles $\delta_1 = \pi/2 - \delta$ and $\delta_2 = \delta$. The adaptation of LSK shown in (7) for a single cable l_i that runs through a TPCJ N-segment section is then defined by (8).

$$\begin{aligned} l_i' \geq & \frac{N}{2} \cdot \left(L - r_c \cdot sin(\frac{\pi}{2} - \delta_i) \cdot \beta_1 \right) \\ & + \frac{N}{2} \cdot \left(L - r_c \cdot sin(\delta_i) \cdot \beta_2 \right) \\ = & - \frac{N \cdot r_c}{2} \cdot (cos(\delta_i) \cdot \beta_1 + sin(\delta_i) \cdot \beta_2) + N \cdot L \end{aligned} \tag{8}$$

To extend (8) to all the cables of a section, the angular arrangement between them has to be taken into account. This can be easily done by decomposing the offset angle of each cable δ_i in a common offset of the cables of a section δ and the offset between each cable evenly distributed around the guiding disc being $0\ rad$, $2\pi/3\ rad$ and $4\pi/3\ rad$ the corresponding angles in the case of a 3-cable configuration. Moreover, considering that for the kinematics is only important the amount of cable that has to be released or retracted from the neutral position ($\beta = 0,\ l = N \cdot L$), the cable length can be taken as increments Δl by eliminating the constant height of the segments $N \cdot L$, resulting in an homogeneous linear relationship between the bending angles (β_1, β_2) and the increment of cable lengths $(\Delta l_1, \Delta l_2, \Delta l_3)$. The configuration space to actuation space kinematics of a section using LSK can then be represented as a linear matrix equation as shown in (9).

$$\begin{bmatrix} \Delta l_1 \\ \Delta l_2 \\ \Delta l_3 \end{bmatrix} \gtrsim \underbrace{\frac{-N \cdot r_c}{2}}_{\text{Scaling factor}} \cdot \underbrace{\begin{bmatrix} cos\,(\delta) & sin\,(\delta) \\ cos\left(\delta + \frac{2\pi}{3}\right) & sin\left(\delta + \frac{2\pi}{3}\right) \\ cos\left(\delta + \frac{4\pi}{3}\right) & sin\left(\delta + \frac{4\pi}{3}\right) \end{bmatrix}}_{\text{Cable disposition matrix } \mathbf{A}(\delta)} \cdot \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} \tag{9}$$

Note that the variables are only the bending angles β and the increment of cable lengths Δl, since all the rest of the terms are constants design parameters specific to each particular robot. The scaling factor indicated in (9) depends on the number of segments N and radial cable position r_c, which is common for all the cables of a section. On the other hand, the matrix $\mathbf{A}(\delta)$ represents the relative position around the main axis of the robot between the bending planes of (β_1, β_2) and the placement of the driving tendons.

Finally, the direct kinematics can be solved by isolating the configuration angles vector from (9) which is as simple as doing the reciprocal of the scaling factor and transposing the cable disposition matrix resulting in (10).

$$\begin{bmatrix}\beta_1\\ \beta_2\end{bmatrix} \lesssim -\frac{2}{N \cdot r_c} \cdot \mathbf{A}(\delta)^T \cdot \begin{bmatrix}\Delta l_1\\ \Delta l_2\\ \Delta l_3\end{bmatrix} \tag{10}$$

Equation (10) is valid as long as the combination of all three incremental cable lengths are coherent.

3.4. LSK method applied to Multisection Robots.

As stated in the introduction of this paper, continuum robots are usually hyper-redundant due to the nature of tasks that they are meant to perform. Because of this, the number of sections tends to be $M \geq 3$ and a solution for the kinematics of the full robot needs to be obtained. This has to include a solution for the existing coupling between the cable lengths $(\Delta l_1, \Delta l_2, \Delta l_3)$ of a particular section to the bending angles of the previous ones. However, solving this problem with the novel LSK method applied to a section as presented in (9) is straight forward. Taking into account that each triad of cables of a section m has a common angular offset δ_m through all the previous sections, computing the cable lengths that control this section taking into account the coupling with the previous ones is shown in (11) (being $\beta_{j,k}$ the bending angle j of section k).

$$\begin{bmatrix}\Delta l_1\\ \Delta l_2\\ \Delta l_3\end{bmatrix} \gtrsim -\frac{N \cdot r_c}{2} \cdot \mathbf{A}\left(\delta_m\right) \cdot \begin{bmatrix}\beta_{1,1} + ... + \beta_{1,m}\\ \beta_{2,1} + ... + \beta_{2,m}\end{bmatrix} \tag{11}$$

By analysing (11), the number of executions K that a cable length has to be computed with the linearised method (8) for a M-section robot is as shown in (12).

$$K(l'_i, M) = 3 \cdot M \tag{12}$$

Comparing these results with the ones obtained for conventional PCC kinematics (shown in (5)), an algorithm of lower order of growth rate is obtained ($\mathcal{O}(M)$ against $\mathcal{O}(M^2)$) when using the method proposed in this paper, as will be demonstrated in the experiments of section 6. Moreover the type of operations required to be done are simpler, being reduced to addition and multiplication operations instead of trigonometric functions and reciprocals. This results in a much higher efficiency for real-time operation specially for robots that are composed of a high number of sections.

3.5. Generalising the LSK method to different continuum robot configurations.

The simplification presented up to now has been applied to the case of TPCJ continuum robots, as the ones presented in [21]. However, it can be extended to any other kinematics, such as the more generally used of bending and orientation angles (θ, ϕ). To do so, it has to be applied to each single segment in order to have a small angle span that does not lead into big approximation errors. From (1), the cable length is calculated for an N number of segments so the bending angle of the whole section (θ) is evenly distributed amongst its segments, each one bending θ/N. Then, the application of the Maclaurin series to (1) with respect to θ when the offset and direction angles are $\delta = \phi = 0$ yields in (13).

$$l \approxeq l' \geqslant L - r_c \cdot \frac{\theta}{N} \tag{13}$$

To model the behaviour of a whole section, robot and its direct kinematics, the direction angle ϕ has to be taken into account. To do so, the bending and direction angle approach of a single segment $(\phi, \theta/N)$ can be considered as the polar coordinates of the (β_1, β_2) approach, so a change of coordinate system has to be made as shown in (14).

$$\begin{bmatrix} \theta_x \\ \theta_y \end{bmatrix} = \frac{\theta}{N} \cdot \begin{bmatrix} cos\,(\phi) \\ sin\,(\phi) \end{bmatrix} \tag{14}$$

Once done this, an analogue procedure to the one shown in Section 3.3 can be done to conclude that the section kinematics of this configuration applying the LSK method is as shown in (15).

$$\begin{bmatrix} \Delta l_1 \\ \Delta l_2 \\ \Delta l_3 \end{bmatrix} \gtrsim -r_c \cdot \mathbf{A}\left(\delta\right) \cdot \theta \cdot \begin{bmatrix} cos\left(\phi\right) \\ sin\left(\phi\right) \end{bmatrix} \tag{15}$$

Note that there is a 1/2 missing in (15) in comparison with (9) since with the (β_1, β_2) design, each segment has a single DoF which direction is alternated consecutively along a section, hence having half of the segments bending for a particular angle β_1 and the other half to β_2, whereas a segment of the (θ, ϕ) construction has both DoFs.

Lastly, the corresponding kinematics of the section m of a multisection robot with a (θ, ϕ) configuration with radially distributed driving tendons adapted from 3.4 is presented in (16).

$$\begin{bmatrix} \Delta l_1 \\ \Delta l_2 \\ \Delta l_3 \end{bmatrix} \gtrsim -r_c \cdot \mathbf{A}\left(\delta_m\right) \cdot \sum_{i=1}^{m} \left(\theta_i \cdot \begin{bmatrix} cos\left(\phi_i\right) \\ sin\left(\phi_i\right) \end{bmatrix} \right) \tag{16}$$

This results again in an algorithm of $\mathcal{O}(M)$ growth rate.

In this section, a linearised model is proposed for solving the kinematics between actuation and configuration space of continuum robots. This method allows an easy computation of segment-by-segment, section-by-section and whole robot kinematics (both direct and inverse) resulting in a more straight forward set of equations. In particular, the algorithm efficiency of the inverse kinematics is greatly improved with the LSK method, avoiding singularities during calculations and reducing the calculations needed at a rate directly proportional to the number of sections that the robot has.

4. LSK-based graphical representation of a Section for intuitive online monitoring.

By using the LSK modelling developed in the previous section, a novel method of graphical representation of the actuation to configuration space kinematics is developed that helps the user to visualize the state of the robot during operation.

4.1. Construction of an user-friendly graphical representation.

Having the kinematics between the actuation and configuration spaces been simplified to a two dimensional overdetermined homogeneous system

as shown in (10), a planar representation of this relationship for a single section can be built. This eases the interpretation of the kinematics, which is particularly useful during the robot operation to infer the robot configuration angles or detect if it is being properly actuated. As a visual aid to follow the explanation of how this representation is made, the reader can consult Fig. 4.

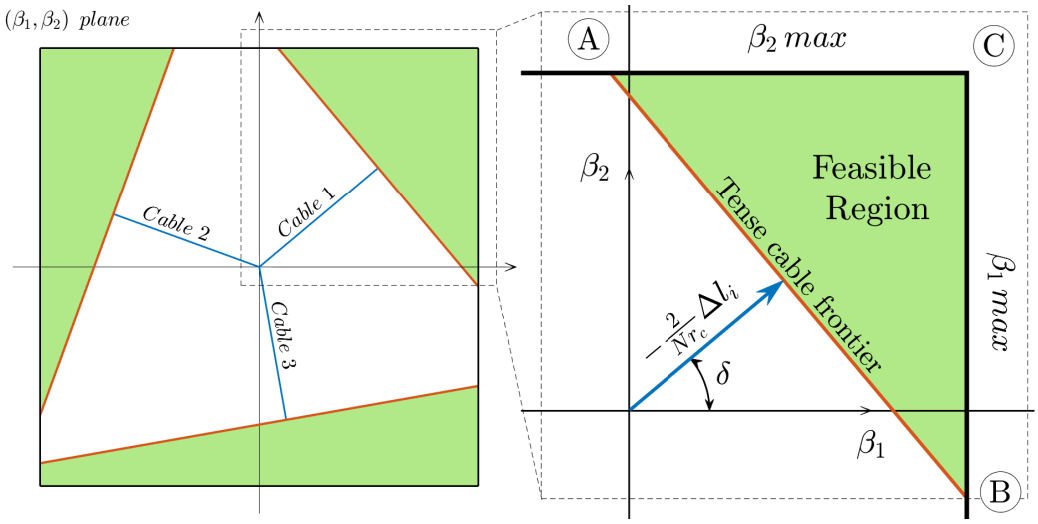

Figure 4: Representation of construction of the 2D graph for a particular length increment of cable 1. The cable length is scaled and represented along the offset angle δ in the (β_1, β_2) plane, generating a frontier that together with the limits of the maximum values of bending angles, delimit a region of all possible combinations of bending angles for that particular cable length.

The 2D graph is built in the (β_1, β_2) plane, choosing indistinctively this variables as the ones to represent in the abscissa and ordinate axes. The workspace envelope of the robot is then delimited by a square with sides parallel to the main axis that pass trough $\pm\beta_{max}$ (Fig. 4 left).

The value of β_{max} can be defined by the physical limits of the robot such as the angle when two consecutive segment discs collide, the elastic strain limit of the flexible element of the backbone, or any others that may apply to each particular case.

To represent a particular cable length increment Δl_i in the 2D graph, it has to be applied the relationship between it and the bending angles (β_1, β_2) shown in (10). The vector to represent the cable length increment can be obtained multiplying the value of Δl_i by the scaling factor from (10) that

defines it modulus, whereas its direction is defined by the offset angle of the cable δ_i from the β_1 axis (blue vector in Fig. 4).

By drawing a perpendicular line to this vector passing through its tip (segment AB, red), the square workspace envelope is divided into two parts. The first one (workspace square minus the surface ABC, green), is the region where the inequality condition of (10) is not met, and physically means that the robot cannot reach the combinations of β_1 and β_2 defined by this area since the driving tendon is unable to stretch (ideally). On the flip side, the region defined by the surface ABC (green) shows where the inequality is met and contains all the feasible combinations of bending angles that can be achieved with that particular cable length since the cable could be slack. Finally, all the combinations of β_1 and β_2 that are included in the frontier line between both regions (segment AB, red) are those that can be reached with the cable fully tense.

To obtain the representation of the whole section, the previous procedure has to be applied to all the cables of a section. The directions of these vectors are determined by the relative position of the cables from the bending planes δ plus the corresponding angular offset of each cable. This is done in the case of a 3-cable configuration by adding $(0\ rad,\ 2\pi/3\ rad,\ 4\pi/3\ rad)$ respectively to each cable and $(0\ rad,\ \pi/2\ rad,\ \pi\ rad, 3\pi/2\ rad)$ in the case of each cable of the 4-cable configuration. The unit vector of each one of them corresponds to each row of matrix $\mathbf{A}(\delta)$ in (9).

This results in a square workspace envelope with divided by 3 tense cable frontiers (red) in three different feasible regions (Fig. 4). The aim is then to achieve a particular configuration of bending angles with all cables fully tensioned , being represented by a single point in the (β_1, β_2) plane where all 3 frontier lines of the cable lengths meet.

However, the manipulation of the driving tendons is physically independent, so there are 3 possible situations that have to be contemplated when operating the robot. These 3 situations can be easily identified in the graphical representation depending on how the reachable areas of each cable interfere amongst each other, being the following:

1. **Exact solution - Fig. 5(a):** All cable lengths satisfy (9), meaning that there is only one combination possible of bending angle values β_1, β_2. This is represented in the graph by all 3 region frontiers concurring in the same plane point (the desired bending angles).
2. **Indeterminate solution - Fig. 5(b):** The cables are not completely

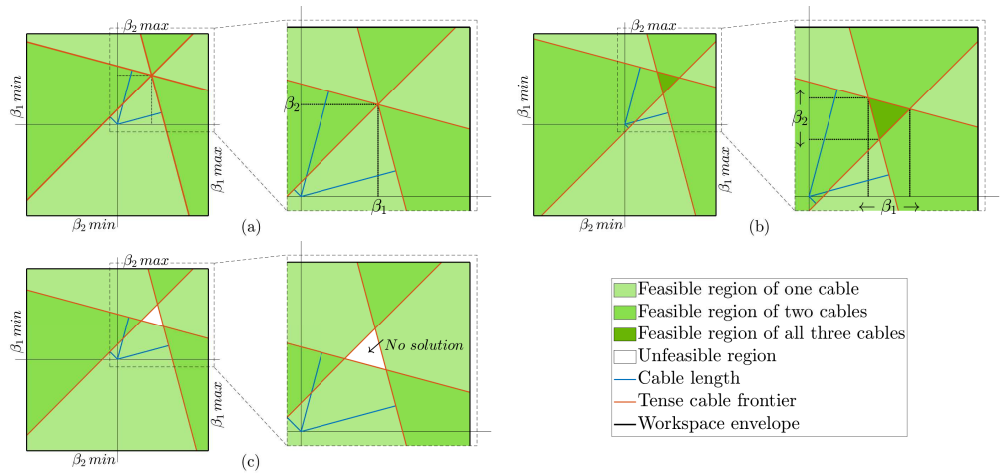

Figure 5: $2D$ representation of the kinematics between actuation and configuration spaces. In (a), all three inequations of 9 match in a single point meaning that the robot is fully constrained at a particular configuration. In (b) the solution of the set of inequations is indeterminate, shown as a darker triangle between all 3 perpendiculars to the cable vectors, meaning that the system is underconstrained, being possible for the robot to be at any bending angle combination inside said triangle. In (c), the combination of the three cable lengths are not compatible, being impossible to solve the set of inequations. This is shown as an empty triangle and causes overtensioning of the backbone and risks to buckle it.

tensioned, giving room to the robot to adopt different bending angles. In the graph these results in an area where the feasible regions of all three cable lengths meet. The possible combinations of β_1 and β_2 are those contained in this area.

3. **Inconsistent solution - Fig. 5(c):** The cables are being tensioned over the physical constraints of the robot, risking its buckling. The combination of these cable lengths in the graphical representation do not match in the same point nor region, leaving an empty area between the frontiers.

This LSK-based graphical representation results in a powerful and easy-to-implement tool for the operation of continuum robots. Including these diagrams for each section in an HMI for controlling the robot can help the operator to easily track its status. Moreover, this representation can be taken into account when doing fast point-to-point trajectories, since during the movement the robot can be induced to be in an indeterminate state (Fig. 5(b)) to leave room for the dynamic of the actuators to response without

overstraining the robot. Lastly, when the desired point is reached, the cable lengths can be adjusted to the exact solution (Fig. 5(a)) to precisely position the robot, or even overshoot slightly to an over-constrained situation (Fig. 5(c)) in order to stiffen that section and compensate for the possible elongation of the driving cables.

4.2. Application of LSK-based representation to different kinematics.

Equivalent to the developments shown in Section 3.5, the graphical representation above shown for a TPCJ continuum robot can be also applied to the more usual (θ, ϕ) construction (neutral backbone design). To do so, instead of a Cartesian set of 2 orthogonal axis, a polar representation achieves a more intuitive interface. The phase of this polar representation then corresponds to the direction angle ϕ and the radius to the bending angle θ (Fig. 6).

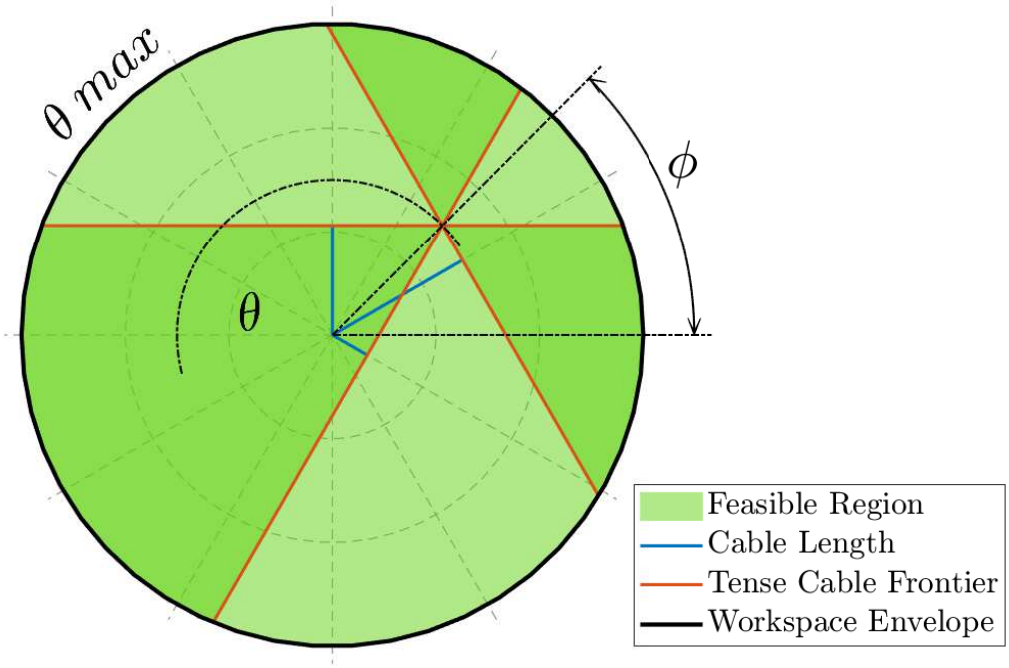

Figure 6: Example of the 2D representation for a neutral backbone kinematics defined by its direction and bending angles (θ, ϕ). The point where the fully tense frontier of all three cables match defines a point which polar coordinates match (θ, ϕ). The reachable workspace is defined by a circle centred in the origin and of radius θ_{max}.

Regarding the cable lengths, these are represented in the same way that in the twin-pivot compliant joint case, but applying its corresponding scaling factor to the neutral backbone case, $-1/r_c$.

Lastly, the workspace envelope limits have to be evaluated. Again, the previous approach can be used to do this by considering the physical limits of the robot. As a result, a robot where the physical limits are set by the collision of two consecutive segments will be represented as a circle of radius $\theta_{max}/N = L/r_{disc}$ (as the example shown in Fig.6) whereas one which limits are defined by the maximum and minimum actuator length values will be represented as an hexagon with edges perpendicular to each cable length direction placed at a distance from the centre equal to $\Delta l_{max} \cdot r_c$. This case is equivalent to the section made by the intersection of a plane perpendicular to the line $l_1 = l_2 = l_3$ to the reachable configuration space cube presented in [20]. The graphical representation proposed in this paper then can be also used for continuum robots of variable backbone length but its application is left to the reader.

5. Error quantification of the LSK method.

The kinematic approach presented in 3.2 is based on an approximation, so a quantification of its error has to be done in order to evaluate its validity.

A percent error has been chosen as a general performance index to compare robots with different geometries. This percent error can be defined by $e_l\% = 100 \cdot |(l - l')/l|$, with $l(\beta)$ being the cable length and $l'(\beta)$ its approximation.

By applying this percent error equation to (6) and (7) and simplifying the expression, the constants L and r_c get cancelled, resulting in an error that depends solely on the bending angle β as can be expressed as in (17).

$$e\%\,(\beta) = 100 \cdot \left| \frac{\beta}{2 \cdot sin\left(\frac{\beta}{2}\right)} - 1 \right| \tag{17}$$

To extract relevant magnitudes for the error, (17) can be considered only between reasonable values of $\beta \in (-45°, 45°)$. This interval of the function (shown in Fig. 7) can be fitted by a purely polynomial expression very accurately to ease the calculation of the maximum error and the mean error.

$$e\% \approx 1.289 \cdot 10^{-3} \cdot \beta^2\,[deg] \tag{18}$$

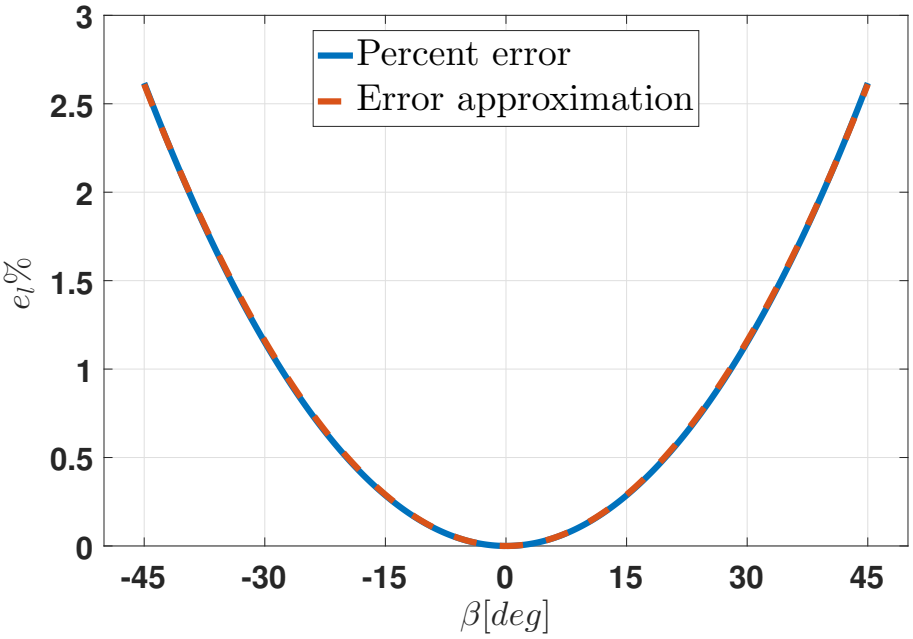

Figure 7: Percentual relative error of the LSK approximation shown in (7) along the span of bending angle $(-45°, 45°)$ overlaid with the pure quadratic fitting shown in 18.

$$e\%_{max} = e\%\left(\beta_{max}\right) \tag{19}$$

$$\overline{e\%} = \frac{1}{2 \cdot \beta_{max}} \cdot \int_{-\beta_{max}}^{\beta_{max}} e\%\left(\beta\right) \cdot d\beta = \frac{e\%_{max}}{3} \tag{20}$$

This error is for a single segment, and when having a multisection continuum robot, the stacking of the errors in multiple sections has to be taken into account. The maximum error is achieved in the driving tendons of the distal section, since they travel along all the sections, and when the offset angle of the cable is $\delta = \pi/4\ rad$ since the influence of both bending angles to its length is maximised. The expression of this cable length along the whole robot both of constant curvature kinematics and of the simplified one are shown in (21) and (22) respectively, (supposing that $\beta_1 = \beta_2 = \beta$).

$$M \cdot N \cdot \left(\frac{L}{\beta} - \sqrt{2} \cdot r_c\right) \cdot sin\left(\frac{\beta}{2}\right) \tag{21}$$

$$M \cdot N \cdot \left(L - \sqrt{2} \cdot r_c \cdot \beta\right) \tag{22}$$

Applying the percent error to (21) and (22) yields the same result as the shown in (17).

Therefore, for any existing flexible backbone continuum robot with a design analogue to disc-guided driving tendons, the error can be easily evaluated in function of the maximum bending angle of each segment and infer if the kinematic simplification presented in this paper yields a permissible error.

6. Implementation benchmarking of the LSK method.

One of the great advantages of the method presented in this paper is the simplicity of the algorithm to compute the kinematics of a robot, having lower order of computational growth rate than previous approaches, plus the mathematical functions that requires (addition and multiplication) against the complexity of the original kinematic modelling, which includes a reciprocal function of a value that passes through 0 and trigonometric functions. This is a great advantage when implementing in a real-time hardware making it both faster to compute and more robust when an simpler data type for the variables is used. In this section, the kinematics presented is particularised to a particular prototype and benchmarked in a target hardware.

6.1. Time performance in Real-Time hardware.

Equations (5) and (12) show that the proposed kinematics in this paper greatly simplifies the computation of the cable lengths. This means that potentially, the algorithm presented in (11) is going to run M times faster than the previous models for full robot kinematics being M the number of sections, and to prove this, execution time of the algorithms has been measured. The pseudo-code of the algorithm used to implement the original kinematic method is as follows:

for $m = 1$ **to** M **do**
  $\overline{\Delta l_m} = \bar{0}$
  for $i = 1$ **to** m **do**
    $\overline{\Delta l_m} = \overline{\Delta l_m} + 2N \cdot sin\left(\frac{\theta_i}{2N}\right) \cdot \left(\frac{L}{\theta_i} - r_c \cdot sin\left(\phi_i + \bar{\delta}_i\right)\right)$
  end for
end for

Whereas the one to implement the method proposed in (16) is:

Table 1: Variable values used for benchmarking

	PCC Kinematics (θ, ϕ)	LSK method (θ, ϕ)	LSK method (β_1, β_2)
Angle span	$\theta \in (0°, 96°)$, $\phi \in (0°, 360°)$		$\beta_1, \beta_2 \in (-24°, 24°)$
Angle step	$\theta_{step} = 4.8°$, $\phi_{step} = 15°$		$\beta_{step1,2} = 2.4°$
Number of sections	$(1, ..., 10)$		
Sweep repetitions	5		

$\overline{\theta}_{sum} = \bar{0}$
for $m = 1$ **to** M **do**
$\bar{\theta}_{sum} = \bar{\theta}_{sum} + \frac{\theta_m}{N} \cdot \left[cos\left(\phi_m + \delta_m\right) \quad sin\left(\phi_m + \delta_m\right)\right]$
end for

These algorithms, as well as the linear one applied for the (β_1, β_2) construction, have been implemented in an *ATMega2560* processor and tested for different number of robot sections M. For each value of M, different values of configuration angles ((θ, ϕ) and (β_1, β_2)) along their span have been swept several times, and the execution time for each set of angles has been measured. The characteristics of the experiments are contained in table 1, giving the results shown in Fig. 8. The PCC kinematics has a quadratic growth as expected and execution time peaks also grow exponentially, making this algorithm not convenient to implement in a real-time controller since determinism and constancy in time execution (Low jitter) is crucial in order to ensure that the processor is not overloaded and time-critical tasks can be launched periodically without affecting the behaviour of the system. On the other hand, the LSK method presented in this paper results in a linear growth both in the mean time and in the maximum execution time, outperforming the conventional PCC kinematics when the number sections M is. Is worth to mention that a (β_1, β_2) construction results in an even faster algorithm since there are no trigonometric functions involved as shown in (11) whereas the simplified version of the (θ, ϕ) construction have, as can be seen in (16).

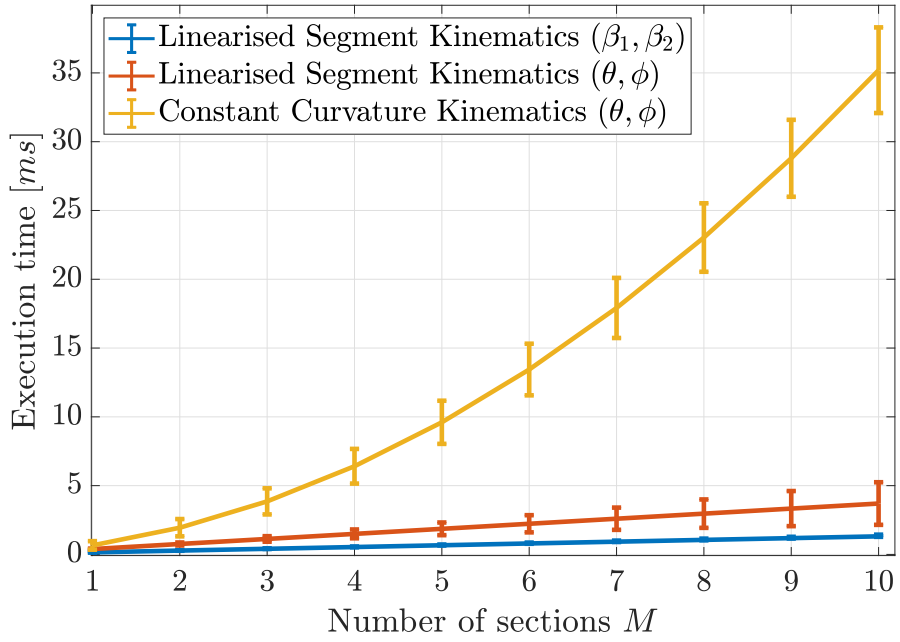

Figure 8: Mean execution time with maximum and minimum values of different kinematic algorithms when increasing the number of sections M.

6.2. Precision robustness.

When implementing in a real-time controller, robustness of an algorithm within the computational limitations of the hardware target has to be taken into account. For this matter, using highly non-linear operations should be avoided since loss of precision or small fluctuations such the ones created for round-off errors may result in large errors at the output. The conventional PCC kinematic modelling shown in (6) suggest robustness problems since there is a variable (θ) which range passes through zero in the denominator of a fraction. These results in high values near the asymptote that when operated with the rest of the terms in the equation of much lower exponent, can induce to inaccurate values. In contrary, the operations present in the basic expression of the LSK method (7) only include linear operations of values of similar ranges, which presumably means that is a much more robust algorithm.

To benchmark this, both algorithms have been tested using different precision data-types. Equations (6) (conventional PCC) and (7) (LSK method)

have been particularised to the parameters shown in table (2) and run through the whole bending angle span. Then, the mean percentage error along this span has been computed taking as a reference the results given by (1) with a double precision floating point data type. The results, shown in Fig.(9) show that the kinematics presented in this paper is much more robust to data-type downgrading, maintaining a mean error under 1% whereas constant curvature kinematics yields a larger mean error when fixed-point arithmetic is used.

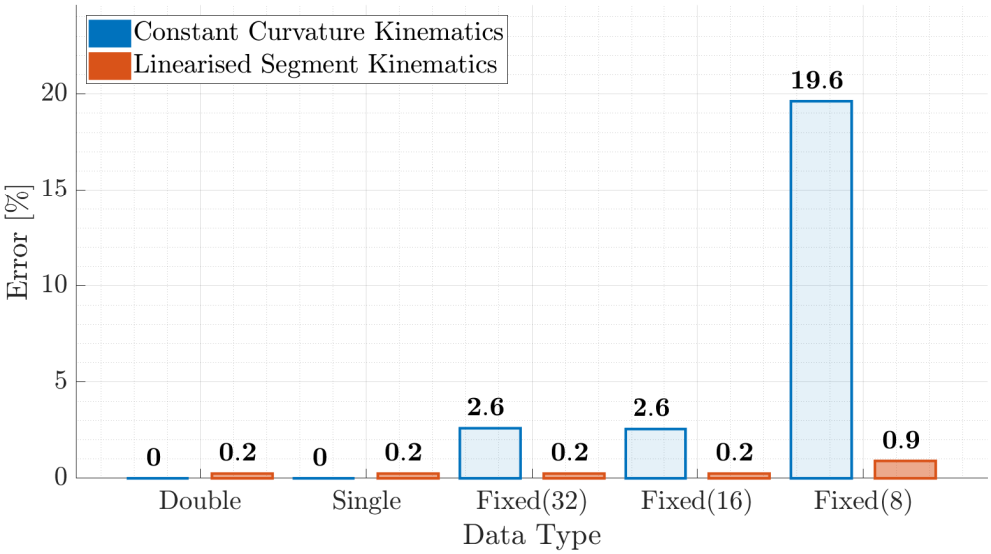

Figure 9: Mean percent error of the original 6 (Blue) and the proposed kinematics in 7 (Red) along the whole angle span $(\beta_{min}, \beta_{max})$ when the used data type is downgraded. The parameters used are those shown in table 2.

The performance results of the LSK algorithm presented in this paper show its superiority when implemented in real-time controllers against conventional PCC kinematics, allowing to reduce the resources used by the kinematics algorithm, This allows either use a controller with lower specifications or use the released resources to execute other tasks such as control algorithms, sensing of cable tension (which, being an analogue measurement, is usually a time spending task due to the nature of analogue to digital converters), and so on.

7. Physical prototype and experiments

As stated in the introduction, the tasks where continuum robots are preferred against traditional rigid link robots share some characteristics in common. More precisely, the environments where these tasks are needed are usually very obstacle dense and remoteness to access the task area from the possible accessing points of the robot. However, there is a general lack in the literature of low diameter robots (less than 10 mm) that can reach locations far from the access point relative to their size and are able to perform the task having successfully avoided the environment obstacles. Because of this, a 6 mm diameter , 175 mm long, 10 $DoFs$ continuum robot has been developed, to undertake those tasks that have small spaces and highly convoluted paths from the access point to the task area with. Moreover, this robot has been tested in a mock-up of an aero engine to perform inspection tasks in the latter stages of the high pressure compressor, where accessibility is very limited. To make the system as self contained as possible, the navigation of the robot has been done implementing the LSK algorithm described in this paper.

7.1. Prototype description

The system developed is composed of all the subsystems to fully operate the robot. The prototype can be divided in 4 subsystems: Robot, Cable Routing System, Actuation System and Electronic System as it can be seen in Fig.10.

7.1.1. Robot

The prototype is a TPCJ tendon driven continuum robot with the characteristics presented in table 2. The guiding discs are made of Titanium and have the functionality of guiding the driving tendons and holding the backbone elements. These backbone elements consist on insertions of 0.44 mm diameter wire of superelastic Nitinol. Lastly, the driving tendons consist on 0.39 mm stainless steel wire.

7.1.2. Cable routing system

The driving tendons have to be routed from the geometry that have trough the robot body to the disposition of the driving mechanism that manipulates them. This routing has been found to be critical for such a small robot. From (9) the sensitivity between the actuation and configuration spaces can be computed as presented in (23).

Table 2: Parameters of developed robot prototype

Outer diameter	6 mm
Total length	175 mm
Number of sections M	5
Segments per section N	8
Cables per section	3
Bending angle per segment β_{max}	24°
Disc height h	2 mm
Joint length L	2.375 mm
Cable disposition radius r_c	2.45 mm
Cable offset angle of the m-th section δ_m	$m \cdot 24°$

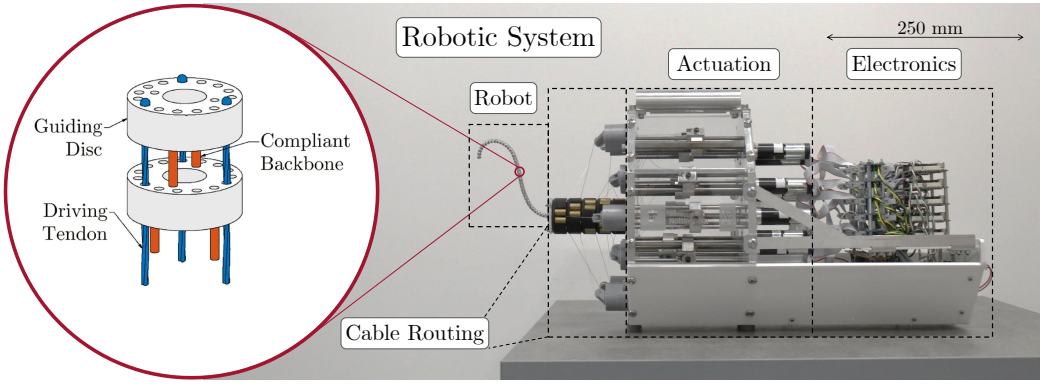

Figure 10: Developed prototype including all the subsystems.

$$\frac{\delta\beta_{section}}{\delta\Delta l} = \frac{1}{r_c} = 23.38\frac{[deg]}{[mm]} \tag{23}$$

This means that for every mm of increment in one cable, the potential maximum angle of a section is 23.38°. Potential since it depends on how the cables are positioned in respect of the bending plane (represented by the parameter δ) and in the status of the other two cables of that same section. This implies that any slip-stick friction or slack in the tendons can have a great impact on the robot position. Taking this into account, a pulley system

has been designed to guarantee as much as possible the correct transmission without friction and uneven tensioning (Fig. 10).

7.1.3. Actuation system

In order to operate the driving tendons, an actuation system is required. This consists on a total of 15 sliding carriages that hold the tendons via friction and move back and forth along screw drives actuated by Maxon brushed DC motors of the A-max series (Fig. 10). The maximum span needed for the carriage can also be deducted from (11) as shown in (24).

$$\Delta l_{max} = \frac{N \cdot r_c}{2} \cdot M \cdot (\beta_{max} - \beta_{min}) = 41.05\ mm \tag{24}$$

7.1.4. Electronic system

For the hardware implementation, each motor is driven by a servo drive that accepts as inputs direction and duty cycle signals. A line receiver decodes the output signal of the incremental encoder that is included in each motor. A National Instruments sbRIO-9606 embedded controller has been used for the control. The FPGA of the sbRIO takes the count of the encoder signals, closes the control loop of the motor and generates the PWM and direction signals. Since the LSK algorithm presented in this paper is computationally so simple, it has been implemented as well in the FPGA. The Real Time processor of the unit preprocess the command data and the feedback data of the robot and communicates with the computer, which only serves as a control panel (Fig. 10).

7.2. Experiment description

As previously stated, the prototype described in this section has been developed to perform inspection and repair tasks in the last stage of the high pressure compressor of an aeroengine. To access this stage, the robot has to access from a small diameter inspection port, and navigate through two stator stages and one rotor stage as it can be seen in Fig. 11. However, due to the high intricacy of the environment mainly due to the fact that rotor and stator blades face opposite directions, the experiment presented in this paper includes only stator stages as a first milestone and experiments including rotor blades will be conducted in the future of this research.

In order to achieve a path to arrive at the area of interest, the robot has been simulated in the engine environment to manually choose the poses that the robot can take to reach its objective. This is done in a inverse approach,

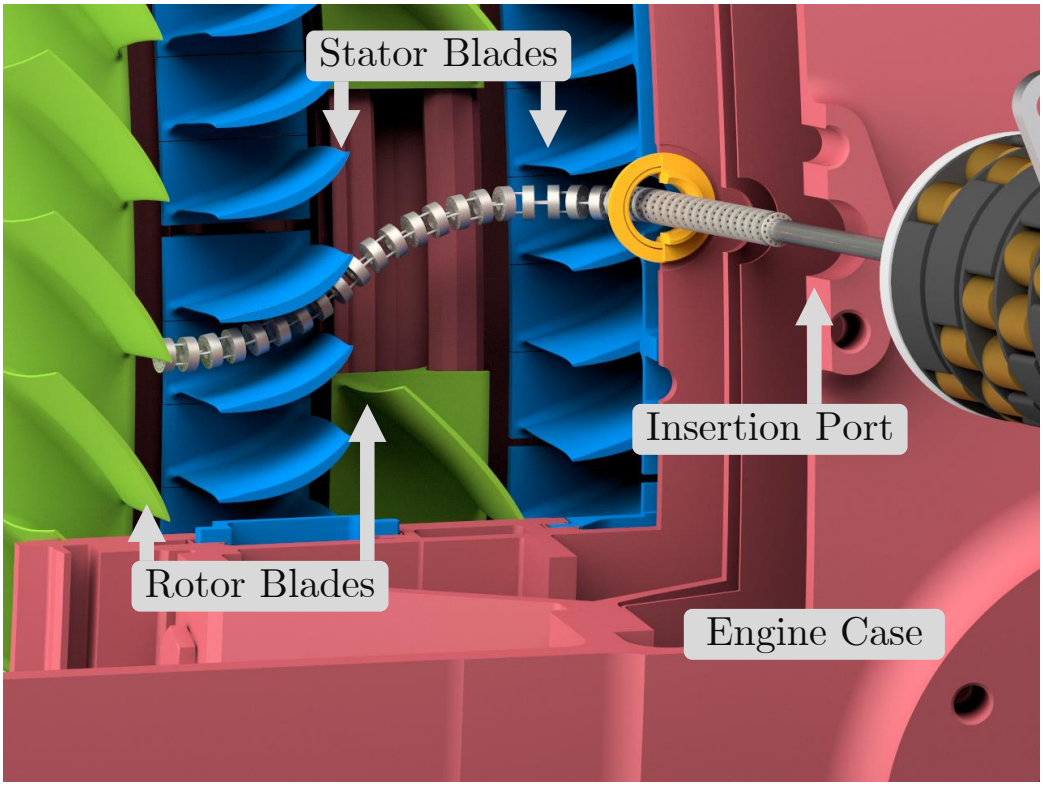

Figure 11: Schematic of the experiment environment. The insertion port of the robot goes through the case of the engine up to the airstream between two blades of the first stator. Then, the robot has to navigate through a rotor stage (where some blades have been removed for this experiments) and a second stator stage to arrive to the area of interest, where the blades of the rotor of the high pressure compressor are located.

first finding a final pose where the robot will reach its target area without colliding with the engine and from that point, trying to find a possible way out for the robot. This path is done by simulating the extraction of the robot through the inspection port 1 mm each step, and by trial and error, finding a new pose that avoids collision. Once all the poses between the robot reaching the area of interest and full extraction of the robot are found, this are interpolated and reversed, obtaining an insertion trajectory..

This set of valid points from the final pose to the initial one (extraction of the robot) is then concatenated to its reverse and all points of each configuration angle are connected by a spline to make the whole navigation smoother, achieving the full trajectory of insertion and extraction of the robot into the

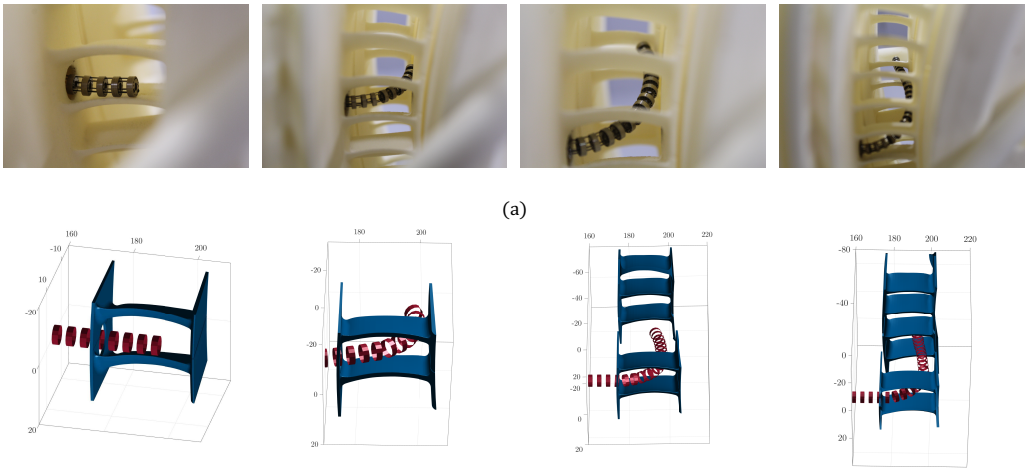

Figure 12: Developed robot prototype at different phases of navigation through the latter compressor stages of a 3D printed engine mock-up (a) and simulation of the same navigation (b).

engine. To implement these experiments with the physical prototype, a section of the engine has been 3D printed and placed on a linear stage aligning the insertion port of the engine with the main axis of the robot. The actuation of the linear stage is then connected to the control hardware of the robot in order to perform synchronously the insertion of the robot with its change of pose.

7.3. Results of navigation in relevant scenario.

The solution for the trajectory obtained in the simulation environment is then exported to the robot hardware, where is read and sent as reference to the kinematic algorithm presented in this paper to control the actuators of the robot and the linear stage. This supposed a great advantage since prior to the development of this algorithm, the kinematics was not possible to be computed in the control hardware of the robot and had to be run in a computer, slowing the process due to the latency of the communication between the computer and the target hardware. However, the algorithm proposed in this paper greatly simplifies the computation requirements of the kinematics that it can be implemented into the control hardware.

Finally, some measurement of the position of the robot had to be taken in order to validate the experiment. Unfortunately the tracking of the robot pose during insertion has not been possible with the available high-precision methods (e.g. VICON tracking system) due to the highly cluttered environment that prevents to any vision system to have sight of the robot and the absence of any other sensing system to measure the pose of the robot. However, the engine mock-up used is open from the side of the area of interest, so measurements of the position of the robot tip at his final pose have been taken to test its accuracy.

Once the robot navigates following the previously calculated poses to the area of interest, its position has been captured by a stereo-camera system, and its distance to recognisable points in the engine has been measured to triangulate its position and compared it to the one obtained in simulation, achieving an accuracy of 3 mm in positioning. However, this does not pose a problem since once arrived to the area of interest, an operator can take over the control to make minor adjustments of the position. After this manipulation, no positioning error has been measured, considering that the precision of the stereo vision system is 1.5 mm. The precision and accuracy of the stereo-camera system has been characterised by comparing it with the measures taken with a VICON motion tracking system in free space so that all cameras had direct sight of the robot during its movement. An example of the robot navigating through the engine stages can be seen in Fig. 12.

8. Conclusions

This paper presents a linearisation to the constant-curvature kinematics between actuation and configuration spaces for tendon driven continuum robots. This simplification leads to a modelling of a section and robot by a homogeneous system of linear equations which not only simplifies the mathematics involved but greatly reduces the computation of cable lengths along a multisection robot. Other of the advantages of this system is the possibility of represent the kinematic relationship between both spaces in a 2D plane, giving a tool for real time operation where it can be easily diagnosed if the robot is being properly actuated. Even though the model presented is a simplification, the error of it has been calculated yielding an expression dependant on the bending angle and it is acceptable for typical values of maximum bending angles of continuum robot segments.

The implementation of the proposed kinematics has led to a faster and more robust algorithm than previous ones increasing its advantage the more number of sections that the robot have, being this a key feature since continuum robots tend to be hyper-redundant to succeed to reach their target while avoiding the obstacles of the task environment.

Lastly, a hyper-redundant (10 DoFs) and slender low-diameter robot (6 mm diameter 175 mm length) has been developed to perform inspection and repair tasks in the last high pressure compressor stage of an aero engine, being this first experiments of its kind and setting a milestone in robotic maintenance of gas turbines. The algorithm presented in this paper has been implemented in the hardware of this prototype allowing it to be self-contained without the need of high-spec external hardware, which eases the portability of the robot.

Future lines of research of this work include tackling the kinematics between configuration and task spaces, optimization of the navigation through cluttered environments and trials in the complete engine scenario.

Acknowledgment

This research was supported by UK EPSRC Platform Grant: Through-life performance: From science to instrumentation (grant no. EP/P027121/1), UK EPSRC RAIN (grant no. EP/R026084/1) and Rolls-Royce Plc.

[1] K. Suzumori, S. Iikura, H. Tanaka, Development of flexible microactuator and its applications to robotic mechanisms, in: Proceedings. 1991 IEEE International Conference on Robotics and Automation, 1991, pp. 1622–1627 vol.2. doi:10.1109/ROBOT.1991.131850.

[2] C. Li, C. D. Rahn, Design of continuous backbone, cable-driven robots, Journal of Mechanical Design (Jun 2002). doi:10.1115/1.1447546.

[3] I. Robert J. Webster, B. A. Jones, Design and kinematic modeling of constant curvature continuum robots: A review, The International Journal of Robotics Research 29 (13) (2010) 1661–1683. doi:10.1177/0278364910368147.

[4] J. Burgner-Kahrs, D. C. Rucker, H. Choset, Continuum robots for medical applications: A survey, IEEE Transactions on Robotics 31 (6) (2015) 1261–1280. doi:10.1109/TRO.2015.2489500.

[5] M. N. Huda, H. Yu, S. Cang, Robots for minimally invasive diagnosis and intervention, Robotics and Computer-Integrated Manufacturing 41 (2016) 127 – 144. doi:10.1016/j.rcim.2016.03.003.

[6] P. E. Dupont, J. Lock, B. Itkowitz, E. Butler, Design and control of concentric-tube robots, IEEE Transactions on Robotics 26 (2) (2010) 209–225. doi:10.1109/TRO.2009.2035740.

[7] A. H. Gosline, N. V. Vasilyev, E. J. Butler, C. Folk, A. Cohen, R. Chen, N. Lang, P. J. del Nido, P. E. Dupont, Percutaneous intracardiac beating-heart surgery using metal mems tissue approximation tools, The International Journal of Robotics Research 31 (9) (2012) 1081–1093. doi:10.1177/0278364912443718.

[8] T. Kato, I. Okumura, S. Song, A. J. Golby, N. Hata, Tendon-driven continuum robot for endoscopic surgery: Preclinical development and validation of a tension propagation model, IEEE/ASME Transactions on Mechatronics 20 (5) (2015) 2252–2263. doi:10.1109/TMECH.2014.2372635.

[9] T. Kato, I. Okumura, H. Kose, K. Takagi, N. Hata, Extended kinematic mapping of tendon-driven continuum robot for neuroendoscopy, in: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2014, pp. 1997–2002.

[10] R. Buckingham, A. Graham, Nuclear snake-arm robots, Industrial Robot: An International Journal 39 (1) (2012) 6–11. doi:10.1108/01439911211192448.

[11] J. S. Mehling, M. A. Diftler, M. Chu, M. Valvo, A minimally invasive tendril robot for in-space inspection, in: The First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, 2006. BioRob 2006., 2006, pp. 690–695. doi:10.1109/BIOROB.2006.1639170.

[12] X. Dong, D. Axinte, D. Palmer, S. Cobos, M. Raffles, A. Rabani, J. Kell, Development of a slender continuum robotic system for on-wing inspection/repair of gas turbine engines, Robotics and Computer-Integrated Manufacturing 44 (2017) 218 – 229. doi:10.1016/j.rcim.2016.09.004.

[13] D. Alatorre, B. Nasser, A. Rabani, A. Nagy-Sochacki, X. Dong, D. Axinte, J. Kell, Robotic boreblending: The future of in-situ gas turbine repair, in: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018, pp. 1401–1406.

[14] M. Wang, D. Palmer, X. Dong, D. Alatorre, D. Axinte, A. Norton, Design and development of a slender dual-structure continuum robot for in-situ aeroengine repair, in: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018, pp. 5648–5653. doi:10.1109/IROS.2018.8594142.

[15] D. Trivedi, A. Lotfi, C. D. Rahn, Geometrically exact models for soft robotic manipulators, IEEE Transactions on Robotics 24 (4) (2008) 773–780. doi:10.1109/TRO.2008.924923.

[16] H. Yuan, Z. Li, Workspace analysis of cable-driven continuum manipulators based on static model, Robotics and Computer-Integrated Manufacturing 49 (2018) 240 – 252. doi:10.1016/j.rcim.2017.07.002.

[17] D. C. Rucker, I. Robert J. Webster, G. S. Chirikjian, N. J. Cowan, Equilibrium conformations of concentric-tube continuum robots, The International Journal of Robotics Research 29 (10) (2010) 1263–1280. doi:10.1177/0278364910367543.

[18] M. W. Hannan, I. D. Walker, Novel Kinematics for Continuum Robots BT - Advances in Robot Kinematics, Springer Netherlands, Dordrecht, 2000, pp. 227–238. doi:10.1007/978-94-011-4120-8_24.

[19] B. A. Jones, R. L. Gray, K. Turlapati, Three dimensional statics for continuum robotics, in: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009, pp. 2659–2664. doi:10.1109/IROS.2009.5354199.

[20] B. A. Jones, I. D. Walker, Practical kinematics for real-time implementation of continuum robots, IEEE Transactions on Robotics 22 (6) (2006) 1087–1099. doi:10.1109/TRO.2006.886268.

[21] S. C.-G. D. A. Xin Dong, Mark Raffles, J. Kell, A novel continuum robot using twin-pivot compliant joints: Design, modeling, and validation, Journal of Mechanisms and Robotics 8 (2015). doi:10.1115/1.4031340.

- Simplification of actuation to configuration space kinematics of continuum robots.
- Graphic representation for easy interpretation of robot state during operation.
- Application to small and high DoF robot to run tests in industrial application.