An efficient follow-the-leader strategy for continuum robot navigation and coiling

Abdelkhalick Mohammad, Matteo Russo, *Fellow, IEEE*, Yihua Fang, Xin Dong, Dragos Axinte, and James Kell

Abstract—**Efficient path planning for hyper-redundant continuum and snake-like robots is a challenging task due to limited sensing capabilities, high computational loads, multiple possible solutions, and non-linear models. This paper presents a new approach to snake robot navigation and coiling, with an algorithm that enables online step-by-step position adjustment with a follow-the-leader strategy, significantly improving the performance of the robot when compared to previous methods. The proposed algorithm is demonstrated on a 16-degree-of-freedom snake-like robot for inspection and maintenance tasks in nuclear facilities.**

Index Terms—**Continuum Robot, Robotics in Hazardous Fields; Coiling; Motion and Path Planning; Collision Avoidance**

I. Introduction

Post-production operations such as maintenance, repair, and overhaul (MRO) are essential to achieve safety and reliability in key industries, such as aerospace, nuclear and telecommunication. These processes often involve challenging scenarios with hazards for human operators and significant associated costs and facility downtime (e.g., grounding aircrafts for engine inspection) [1].

Conventional industrial robotic manipulators can provide high positioning accuracy and have been extensively used in structured assembly-line applications such as automotive manufacturing [2]. However, these robots are designed for a predefined open environment and cannot operate in narrow tortuous spaces such as pipes or turbines. Conversely, continuum and snake-like robots can navigate in difficult-to-reach areas thanks to their compliant hyper-redundant architectures [3], as proved by many successful prototypes for nuclear [4-5], aerospace [6-7] and medical applications [8-9].

Generally, the main challenge for robotic in-situ MRO is given by tight constraints due to a narrow environment. Hence, MRO robots should be designed such that they can adopt to the working environments. Although the many degrees of freedom (DoF) of continuum robots increase the maneuverability of their lean snake-like bodies, they also require bulky actuation units. Furthermore, they generally have to be stored in straight configurations to avoid damage to the compliant elements in the system, requiring wide open areas next to the MRO target for deployment.

In order to solve this issue, we previously proposed a snake-like robot able to coil around its actuation unit [10], which is designed as a rotating drum with a spiral groove to accommodate the continuum robot. Therefore, the space occupied by the manipulator is significantly reduced, making its transportation and storage easier. Once the robot is located in the working situ, it can be uncoiled into a prescribed shape, and then inserted in the work environment through a linear feeding system, which is required by the proposed navigation strategy but increases the required deployment space and introduces an additional DoF in the system. The main limitation of the algorithm in [10] is given by the long computation time (up to 170 s), which prevents the robot to be operated in real time. It depends on optimizing the robot shape to fit the desired path. First, the desired path is divided into a point cloud in 3D. Then, for each step in the optimization process, Euclidean distances between each desired point on the path and nearest points on the robot are calculate numerically. These distances then are used to form an objective function which is minimized through the optimization process to find the optimal configuration of the robot. Furthermore, as the algorithm optimizes the shape of the robot from a fully coiled to a fully uncoiled configuration, which makes it inappropriate for constrained environments. If an unexpected change is needed during operation, the robot needs to be retracted.

In this paper, we present an analytical solution for the shape fitting of the continuum robot to a desired path. The advantage of the proposed method is that it does not need any time-consuming calculation like the Euclidean distances to find the optimal shape of the snake-like robot. This enables the robot to navigate and uncoil based on a step-by-step follow-the-leader strategy. Moreover, the proposed algorithm removes the need for a linear stage as it uses the uncoiling motion for insertion in the target environment.

Manuscript received: March 1, 2021; Revised May 21, 2021; Accepted June 18, 2021.

This paper was recommended for publication by Editor Youngjin Choi upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by UK EPSRC project Robotics and Artificial Intelligence in Nuclear (RAIN) under grant EP/R026084/1.

A. Mohammad, M. Russo, Y. Fang, X. Dong, and D. Axinte are with the Rolls-Royce UTC in Manufacturing and On-Wing Technology, University of Nottingham, Department of Mechanical, Materials and Manufacturing Engineering, Nottingham, NG8 1BB, UK (e-mail: {abd.mohammad1; matteo.russo; yihua.fang, xin.dong; dragos.axinte}@nottingham.ac.uk)

J. Kell is with Rolls-Royce plc, Derby, DE24 8BJ, UK (e-mail: james.kell@rolls-royce.com)

Digital Object Identifier (DOI): see top of this page.

II. Problem Formulation

A. *Piecewise Continuous Curvature Kinematics*

Conventionally, the shape of a continuum robot is modelled as a sequence of independent 2-DoF sections with piecewise constant curvature (PCC) [11-12]. The schematic diagram in Fig. 1 represents a generic i^{th} section of the robot. The configuration of this section can be defined as an arc with section length s_i, bending angle θ_i, and direction angle φ_i (with positive values when rotating around the Z_i-axis counter-clockwise). Among the many formulations for continuum robot kinematics [13], in this study we adopt the model in [14] for their limited computational complexity.

The position of the distal extremity of the section is defined as frame $\{S_i\}$, and the base coordinate frame $\{S_{i-1}\}$ of the same section is centred on its proximal extremity. Thus, the translation along the circle arc $\boldsymbol{t}_{\boldsymbol{i-1}}^{\boldsymbol{i}} \in \mathbb{R}^3$ can be expressed as

$$\boldsymbol{t}_{\boldsymbol{i-1}}^{\boldsymbol{i}} = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \frac{s_i}{\theta_i}\begin{bmatrix} \cos\varphi_i\,(1-\cos\theta_i) \\ \sin\varphi_i\,(1-\cos\theta_i) \\ \sin\theta_i \end{bmatrix}. \tag{1}$$

The rotation matrix between the proximal and distal frames of the section $\{S_{i-1}\}$ and $\{S_i\}$ results from three consecutive rotations around axes Z_i, Y_i, and Z_i of angles φ_i, θ_i, and $-\varphi_i$, respectively. Hence, the rotation matrix $\boldsymbol{R}_{\boldsymbol{i-1}}^{\boldsymbol{i}} \in SO(3)$ can be written as

$$\boldsymbol{R}_{\boldsymbol{i-1}}^{\boldsymbol{i}} = \boldsymbol{R}_{\boldsymbol{z_i}}(\varphi_i)\cdot\boldsymbol{R}_{\boldsymbol{y_i}}(\theta_i)\cdot\boldsymbol{R}_{\boldsymbol{z_i}}(-\varphi_i). \tag{2}$$

Therefore, the homogeneous transformation $\boldsymbol{T}_{\boldsymbol{i-1}}^{\boldsymbol{i}} \in SE(3)$ from $\{S_{i-1}\}$ to $\{S_i\}$ can be defined as

$$\boldsymbol{T}_{\boldsymbol{i-1}}^{\boldsymbol{i}} = \begin{bmatrix} \boldsymbol{R}_{\boldsymbol{i-1}}^{\boldsymbol{i}} & \boldsymbol{t}_{\boldsymbol{i-1}}^{\boldsymbol{i}} \\ \boldsymbol{0}_{\boldsymbol{1\times 3}} & \boldsymbol{1}_{\boldsymbol{1\times 1}} \end{bmatrix}. \tag{3}$$

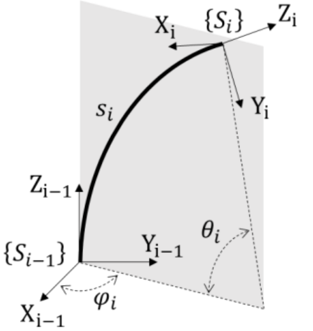


Figure 1. Design parameters and motion variables of a continuum robot section modelled with PCC kinematics.

The model for the entire robot can be obtained by adding further sections in a serial chain, and the kinematics of the entire system can be evaluated by equalling the proximal frame of each section to the distal frame of the previous one, or to the base frame in the case of the most proximal section (i = 1). This assumption implies that the distal extremity of a section corresponds to the proximal extremity of the following one, and that the two sections share tangent in that point. The recursive nature of this function allows to easily obtain an overall profile by building the robot up section by section.

B. *Tendon-based Actuation*

The control strategy that we propose in this work does not depend on the technology that is used to actuate the continuum robot. However, here we introduce tendon-driven design as the most common class of continuum robots (see examples in [6-8]), on which our algorithm will be validated. A tendon-driven continuum robot achieves the independent actuation of each section by using three or more steel or fiber tendons attached to the distal extremity of that section and driven by rotary or linear motors at the base of the system (e.g. [10]). The popularity of tendon-driven designs can be attributed to their strength, the accurate motion control given by motor encoders, and an intuitive functioning. However, they have some drawbacks: first of all, as tendons follow the shortest path to one attachment/routing point to the following one, they need to be routed through several guides to be kept at an approximately constant distance from the backbone of the robot; furthermore, as the actuation is delocalized rather than distributed, the kinematics of a tendon-driven distal section of the robot will depend on the configuration of its proximal sections; finally, all the tendons need to be kept in tension at all times to enable the correct functioning of the robot. Moreover, a distinct kinematic mapping is needed to model the transformation from section configuration to tendon displacement [15].

C. *Continuum Robot Navigation*

The main challenge in continuum robot navigation is the need to maintain the shape of the body along the path traveled while advancing the tip of the robot to a new position. In the last decades, three main methods for motion planning were identified: follow-the-leader motion, Jacobian inversion, and pose optimization. Hereafter, we discuss the advantages and limitation of these algorithms:

a) *Section-wise follow the leader:* In this method, the snake robot moves in discrete steps, with the most distal section advancing to a new pose while each generic i^{th} section assumes the pose previously held by section i + 1 [8, 16]. This approach is similar to computer graphics algorithms and can be seen in videogames such as the classic "Snake." Conceptually, this strategy is simple and effective. However, it is limited by the discrete step size, forcing the sections of the snake to be identical in length and achieving "perfect" tip following only when the advancement step is equal to a multiple of the section length. Furthermore, the transition between poses needs to be carefully planned.

b) *Continuous follow the leader*: In this approach, described in [17-18], the shape of the robot is described by using the time-varying arc-length parametrized transformation $\boldsymbol{g}(\sigma,t) \in \mathrm{SE}(3)$ with centerline position $\boldsymbol{p}(\sigma,t) \in \mathbb{R}^3$ and disk orientation $\boldsymbol{R}(\sigma,t) \in \mathrm{SO}(3)$ with respect to arc length $\sigma \in [0,\ell(t)]$ along the centerline. While the domain of σ depends on t, the two parameters are independent from each other, and their symmetry is assumed in mixed partial derivatives with respect to σ and t. The time-dependent length function $\ell(t)$ represents the exposed length of the robot, defined as the length of the robot that has been inserted in the operational workspace, rather than the full length of the snake. The evolution of the transformation $\boldsymbol{g}(\sigma,t)$ can be expressed by differential kinematic equations as

$$\boldsymbol{g}'(\sigma,t)=\frac{\partial \boldsymbol{g}(\sigma,t)}{\partial\sigma}=\boldsymbol{g}(\sigma,t)\hat{\boldsymbol{\xi}}(\sigma,t) \tag{4}$$

where $\hat{\boldsymbol{\xi}}(\sigma,t)$ is the Lie algebra $\mathfrak{se}(3)$ corresponding to the vector $\boldsymbol{\xi}\in\mathbb{R}^6$, which contains the body frame twist coordinates $\boldsymbol{\xi}(\sigma,t)=[\boldsymbol{v}(\sigma,t)^{\mathrm{T}}\quad \boldsymbol{u}(\sigma,t)^{\mathrm{T}}]^{\mathrm{T}}$. The components of the transformation are defined as

$$\begin{aligned}\boldsymbol{R}'(\sigma,t)&=\frac{\partial \boldsymbol{R}}{\partial\sigma}=\boldsymbol{R}(\sigma,t)\hat{\boldsymbol{u}}(\sigma,t)\\ \boldsymbol{p}'(\sigma,t)&=\frac{\partial \boldsymbol{p}}{\partial\sigma}=\boldsymbol{R}(\sigma,t)\boldsymbol{v}(\sigma,t)\end{aligned} \tag{5}$$

When the z-axis of $\boldsymbol{R}$ is tangent to the curve, this expression can be simplified in $\boldsymbol{p}'(\sigma,t)=\boldsymbol{z}(\sigma,t)$**.** We can also assume that at insertion point $\sigma=0$ both rotation and position are fixed as $\boldsymbol{R}(0,t)=\boldsymbol{R}_{\boldsymbol{z}}(\varphi_0(t))$ and $\boldsymbol{p}(0,t)=\boldsymbol{0}$. When using a continuous follow-the-leader approach, the position at any arc length of the curve does not change in time. This condition can be mathematically expressed as

$$\dot{\boldsymbol{p}}(\sigma,t)=\frac{\partial \boldsymbol{p}}{\partial t}=\boldsymbol{0} \tag{6}$$

where the dot denotes the partial derivative with respect to time. While this method provides a geometrically exact solution for continuum designs (e.g., concentric tube robots), it cannot represent the discrete nature of snake-like robots that alternate rigid disks to flexible elements. Furthermore, both convergence and stability issues limit the practical application of this model.

c) *Jacobian inversion:* Similarly to conventional robotic manipulators, continuum robots can be controlled by solving inverse kinematics. The Jacobian matrix $\boldsymbol{J}(\boldsymbol{q})$ of a robot represents the relationship between the velocity of the robot in the operational space $\dot{\boldsymbol{x}}$ and the corresponding velocity in the joint space $\dot{\boldsymbol{q}}$, as

$$\dot{\boldsymbol{x}}=\boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} \tag{7}$$

where $\boldsymbol{J}(\boldsymbol{q})$ is a function of the joint variables $\boldsymbol{q}$. The joint space is easy to identify in conventional robot with distributed actuation, where each revolute or prismatic joint is directly actuated by a rotary or linear motor in the body of the manipulator, resulting in a simple and fast solution to inverse kinematics. Conversely, the joint space of continuum robots with delocalized actuation can be divided in two layers: a configuration space, which describes the position of the robot with bending and direction angles, and an actuation space, which contains the linear or angular displacement of the motors corresponding to that configuration. This additional transformation and the hyper-redundant architectures of many continuum robots lead to a complex problem with longer times to solution and convergence and drift issues [3].

d) *Pose optimization:* An optimization problem is solved to find the best possible configurations of the continuum robot. For an accurate path planning, the position error of the robot, defined as the displacement between the prescribed pose and a pose that the continuum robot can achieve, is minimized given problem constraints (e.g., initial position, obstacle geometry and location, joint limits) [10]. While pose optimization can lead to the best numerical results, its application is limited by the accuracy of the robot modelling, the prescribed constraints and search space, and a long computational time to solution.

D. Problem Definition

With reference to Fig. 2, which illustrates an example of two consecutive configurations of a continuum robot navigating in a restricted environment, we assume the robot to be in a known pose (configuration A), for which all the kinematic parameters (i.e. direction and bending angles; location of the base) are determined. From this configuration, the robot has to advance to a new configuration (B), moving with a prescribed discrete step in the operational space. This work addresses the following research question: *assuming a motion step with negligible dynamic response, what are the new configuration and actuation parameters that advance the robot to the prescribed position while maintaining the shape of its body if possible, or minimizing its variation if not?*

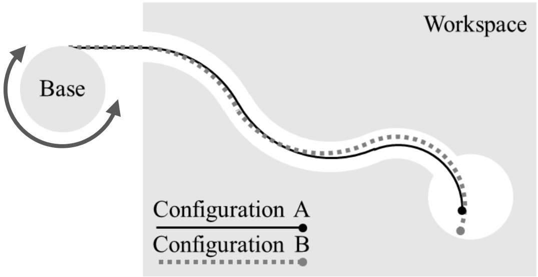


Figure 2. A schematic representation of two consecutive configurations of a continuum robot advancing along a narrow tortuous path.

III. NAVIGATION ALGORITHM

In order to provide the reader with a better understanding of the proposed algorithm, here we first explain a simplified formulation to find the optimal configuration of a single non-extensible section in the local X_i Z_i plane ($\varphi_i=0$), as shown in Fig. 3.

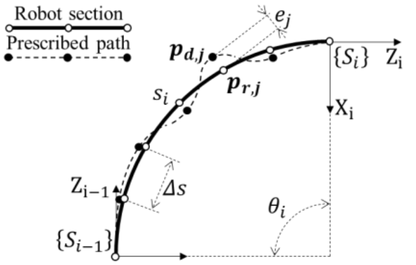


Figure 3. A graphical representation of the proposed navigation method for a single-section robot in a planar configuration.

The prescribed path to be followed by the robot is obtained by a curve that describes the ideal configuration for the section and defined by discretizing that curve with a fixed step Δs into an ordered point cloud $P_d=\{\boldsymbol{p}_{\boldsymbol{d},\boldsymbol{j}}\}$. The body of the robot is divided into segments of fixed length, equal to the step of the path discretization, described by the ordered point cloud $P_r=\{\boldsymbol{p}_{\boldsymbol{r},\boldsymbol{j}}\}$. Given a number of points in the continuum robot equal to n, the discretization step Δs can be computed as s_i/n. The error between a desired position of the ideal curve and the corresponding point of the robot body can be defined as

$$e_j = \left|\boldsymbol{p_{r,j}} - \boldsymbol{p_{d,j}}\right| = \left\|\begin{bmatrix} x_{r,j} - x_{d,j} \\ y_{r,j} - y_{d,j} \\ z_{r,j} - z_{d,j} \end{bmatrix}\right\|, \tag{8}$$

where $j = \{1, 2, \ldots, n\}$. By substituting (1) into (8), the squared error E_j on a point of a single planar section ($\varphi_i = 0$) becomes

$$E_j = {e_j}^2 = \frac{{s_i}^2}{{\theta_i}^2}\left(2 - 2\cos\frac{j\theta_i}{n}\right) + {x_{d,j}}^2 + {z_{d,j}}^2 - 2\frac{s_i}{\theta_i}\left(x_{d,j}\left(1 - \cos\frac{j\theta_i}{n}\right) + z_{d,j}\sin\frac{j\theta_i}{n}\right). \tag{9}$$

This procedure can be extended to other robot configurations and not limited to constant curvature. The section-wise cumulative squared error G_i along the path can be evaluated as

$$G_i = \sum_{j=1}^{n} E_j \tag{10}$$

In order to find the optimal configuration of the robot, we need to find the value of θ_i that minimizes this function. This condition is expressed by

$$\frac{\partial G_i}{\partial \theta_i} = \sum_{j=1}^{n}\left(-2\frac{s_i}{\theta_i}\left(x_{d,j}\sin\frac{j\theta_i}{n} + z_{d,j}\cos\frac{j\theta_i}{n}\right) + 2\frac{s_i}{{\theta_i}^2}\left(x_{d,j}\left(1 - \cos\frac{j\theta_i}{n}\right) + \left(z_{d,j} + s_i\right)\sin\frac{j\theta_i}{n}\right) - 2\frac{{s_i}^2}{{\theta_i}^3}\left(2 - 2\cos\frac{j\theta_i}{n}\right)\right) = 0 \tag{11}$$

As this function is not linear, we cannot solve it for theta in closed form. However, as shown in Fig. 4, it is possible to consider a linearize around an operating point $\theta_{0,i}$, as

$$\frac{\partial G_i(\theta_i)}{\partial \theta_i} = \frac{\partial G_i(\theta_{0,i})}{\partial \theta_i} + \left.\frac{\partial^2 G_i(\theta_i)}{\partial {\theta_i}^2}\right|_{\theta_i = \theta_{0,i}}\left(\theta_i - \theta_{0,i}\right) = 0 \tag{12}$$

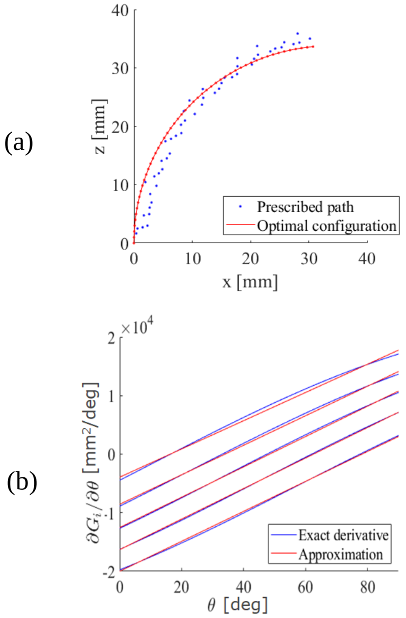


Figure 4. Optimal configuration of a single-section robot over five arbitrary paths: a. Simulation results for a sample configuration; b. Comparison of exact derivative of G_i and its linear approximation around $\theta_{0,i}$ = 45° on five randomly generated configurations.

The derivative of the cumulative squared error function has been analyzed for a continuum robot section on 1000 randomly generated configurations. The derivative (11) is almost linear in the whole operational range (0° – 90°), with a maximum linearity error of 2% when compared to its approximation (12) around an operating point $\theta_{0,i}$ = 45°, as illustrated by the five example configurations of Fig. 4. Hence, the approximation can be safely used within that range. This approach can be extended to consider the direction angle as well by considering the partial differentiation of $\mathrm{G_i}$ with respect θ_i and φ_i.

B. Navigation of a Multi-section Robot

The method presented in the previous section can be expanded to obtain a navigation strategy for a continuum robot with a generic number of sections m by including a variable direction angle in (9) and minimizing a global cumulative squared error G, obtained by summing the section-wise cumulative squared errors G_i in (10) as

$$G = \sum_{i=1}^{m} G_i \tag{13}$$

Thus, the multi-section algorithm can be summarized in the following points:

a) The current configuration of the robot (A) is defined as a point cloud $P_{rA} = \{\boldsymbol{p_{r,j}}\}$, as previously described.

b) The desired path is defined as a point cloud P_d, which can be generated either by discretizing a curve that represents an ideal configuration, similarly to the strategy in [9], or by adding new points at the tip of the current configuration of the robot P_r, in case the continuum robot is advancing step by step from a partially or fully coiled configuration (or, vice versa, by removing points at the tip in case the robot needs to withdraw).

c) The tip point ($j = n$) of the most distal section of the robot ($i = m$) is associated to the most distal point in the ideal path, thus defining the origin of frame $\{S_m\}$.

d) The path can be parametrized, and a coordinate σ can be defined as the distance from the tip point along the given path. The closest neighbor of $P_d = \{\boldsymbol{p_{d,j}}\}$ to the point defined by the coordinate $\sigma = s_m$ can be assumed to be the target point for the origin of frame $\{S_{m-1}\}$, thus defining the proximal extremity of section m.

e) By knowing the origin of frames $\{S_m\}$ and $\{S_{m-1}\}$ and by using the subset of $P_d = \{\boldsymbol{p_{d,j}}\}$ characterized by $0 \leq \sigma \leq s_m$, it is possible to use the procedure in III-A to compute the optimal values for φ_m and θ_m.

f) Once defined the configuration of the section, update the closest neighbor of $P_d = \{\boldsymbol{p_{d,j}}\}$ to the origin of $\{S_{m-1}\}$, as it might differ from the one found in stage 4.

g) Repeat the above procedure (stages 3 to 6) to compute the configuration of the section with its distal extremity attached to the proximal extremity of the section evaluated in the previous repetition (i.e. $i = i - 1$), proceeding backward from tip to base until the configuration of the whole robot is defined.

C. Advancing, Coiling and Uncoiling

During the insertion of the robot in the workspace, the advancement of the robot is controlled by modifying the exposed length of the robot ℓ (i.e., the length of the robot that has been inserted in the operational workspace, rather than its full length). When evaluating a new configuration with the procedure described in III-B, the algorithm can be used to evaluate only the sections that are part of the exposed length of the robot, whereas the remaining sections stay in a predetermined coiled configuration as shown in Fig. 6. The advancement of the robot, defined as step Δs, is obtained with a corresponding rotation of the actuation unit drum by an angle α, computed as

$$\alpha = \frac{\Delta s}{r} \tag{14}$$

where r is the radius of the drum.

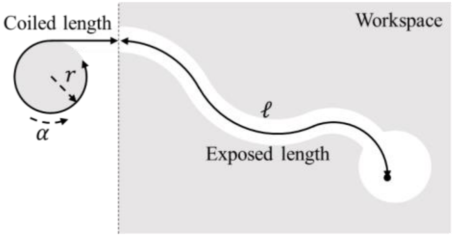


Figure 5. Exposed and coiled length of the proposed continuum robot.

One of the advantages of the proposed method is that the new configuration of the robot does not depend on the previous ones, which enables online adjustments of the shape of the robot rather than having to follow a predetermined path only as in [10]. Furthermore, the computation time for a new position is relatively small (less than 0.1 s for a two-section example in MATLAB on a Windows 10 computer with an Intel i7-6700HQ 2.60 GHz quad-core CPU), leading to a fast and accurate (maximum displacement from desired path: less than 3% of robot length) control method for the proposed class of robotic systems.

C. Simulation Results

The uncoiling and navigation are validated through a computer simulation by following a path defined by the user as distinct steps in the global coordinates. These inputs are added to the current robot tip position to generate a new path to be followed by the body of the robot while uncoiling. In Figure 6, an example path of the tip of each section is shown whilst following the leader and uncoiling simultaneously.

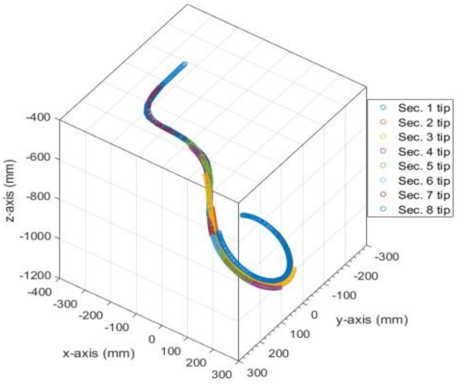


Figure 6. Path of the tips of sections {1,2, …, 8} of the continuum robot during an uncoiling simulation with the proposed follow-the-leader strategy.

In total, 183 steps of approximately 5 mm along the x-, y-, and z-axes are traveled, with an average tip error (relative to the robot's length) of 0.05% and a maximum deviation of 0.71%. The reason is that the robot consists of sections which not always can take the same curvature as the desired profile.

IV. Experimental Validation

A. Experimental Setup

The proposed navigation algorithm has been demonstrated on a snake-like robotic manipulator with 8 sections (16 independent DoFs) and an additional motor for the rotation of the actuation unit, shown in Fig. 7. The robot has been developed for maintenance tasks in nuclear industries, and it is an evolution of the robotic system presented in [6] and [10].

The inspection or dismantling of components of nuclear plants often involves inaccessible and/or hazardous environments and dangerous items, which need to be assessed before the inspection or dismantling process starts. An example of these challenges is the mapping of legacy glove boxes in nuclear plants. Before dismantling the gloveboxes, the items inside them need to be identified. However, the opacity of their glass windows hinders a visual inspection, and human operators expose themselves to hazards in retrieving the contents. Thus, a snake-like robotic manipulator can be used to scan and retrieve items from legacy gloveboxes.

The design of the robot used for the tests is an evolution of the twin-pivot architecture in [6], with each section actuated with three cables, each attached to a single rotational motor, whereas the previous design in [6] actuated each section with two pairs of antagonistic cables connected to a single motor. By having a single cable per motor, a torque sensor on the motor can be used to identify the tension on the cable. This capability is used before each operation to pre-tension all the actuation cables with prescribed forces and to identify faults by monitoring tension drops during operation. The tip diameter of the robot is 20 mm, whereas the length of each section is reported in Table I.

TABLE I
Snake-like Manipulator Geometry

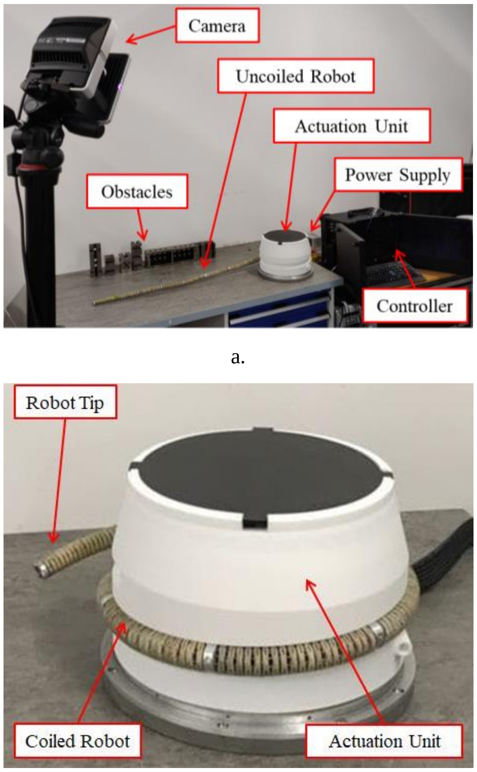
Section	Length	Section	Length
1	150 mm	5	118 mm
2	150 mm	6	118 mm
3	150 mm	7	102 mm
4	125 mm	8	102 mm

B. Results: Navigation Test

The experimental setup for a first navigation test is shown in Fig. 7a, where the snake is shown in a straight, fully uncoiled configuration and the obstacles have not been deployed yet in the workspace.

The robot has been calibrated by tensioning all the cables to a predetermined tension in a straight configuration, and then fully coiled around its actuation pack before each acquisition, as shown in Fig. 7b. As the joint values are computed by an optimization problem while coiling, the coiled configuration might slightly vary between experiments.

After coiling the robot, obstacles have been deployed in the workspace to resemble an operational scenario in the nuclear industry, with the robot navigating through teleoperated tip position control in a cluttered workspace, where five different obstacles of different shape and size have been deployed as per Fig. 8.



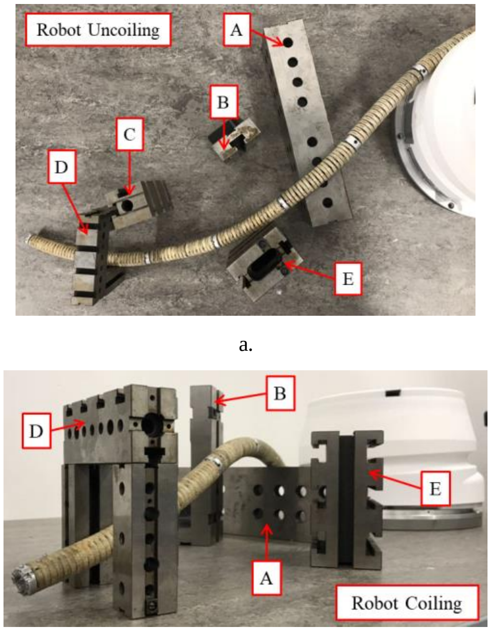
a.

b.

Figure 7. Experimental setup: a. Main elements of the setup, with the snake-like robot in a fully uncoiled configuration, the robot's controller and power supply, one of the Vantage 16 cameras of the VICON motion capture system, and the obstacles; b. Initial configuration of the robot, with the snake-like body fully coiled around the actuation unit.

In the example, the robot has to navigate around the obstacles A, B, C, and E, in order to move through the arch-shaped obstacle D. To demonstrate a combined uncoiling and navigation task, the robot has been uncoiled through the workspace and then coiled back to the starting position. Once the robot reaches the target work area, the follow-the-leader control can be deactivated to independently control the distal sections for inspection or maintenance tasks. After each operation, the robot is likely to be in a different pose than at its start. Thus, an updated follow-the-lead path is required to enable further coiling and uncoiling along a new shape. Unlike the algorithm in [10], the proposed algorithm integrates the capability to toggle follow-the-leader motion on or off and automatically compute further steps from the current position rather than moving along a fully defined path. This capability has been demonstrated by performing the coiling operation along a different path from the uncoiling, as shown in the example path of the end-effector reported in Fig. 9.

During the experiment, four hemispherical reflective markers have been placed on the end-effector of the robot to measure its pose with a VICON motion capture system with four Vantage 16 cameras deployed around the workspace and calibrated through their proprietary software with an accuracy of 0.090 mm.



a.

b.

Figure 8. An example of snake-like robot navigation with integrated follow-the-leader, coiling, and uncoiling: a. Top view of a snake-like robot navigating through obstacles while uncoiling; b. Side view of a snake-like robot at the beginning of a coiling operation, after an online adjustment for obstacle avoidance.

The acquisition results are reported in Fig. 9. During the uncoiling phase (Fig. 9a), the robot moves close to the obstacle E, risking a collision. In the coiling operation, an online adjustment enables a safer return to the starting position, with the snake body more equally distanced from the obstacles B, C and E (Fig. 9b). This difference in path can be observed in the comparison of the end-effector path during the uncoiling and coiling operation in Fig. 8c.

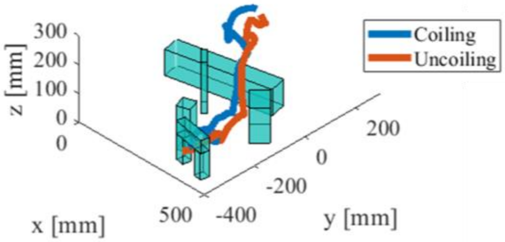
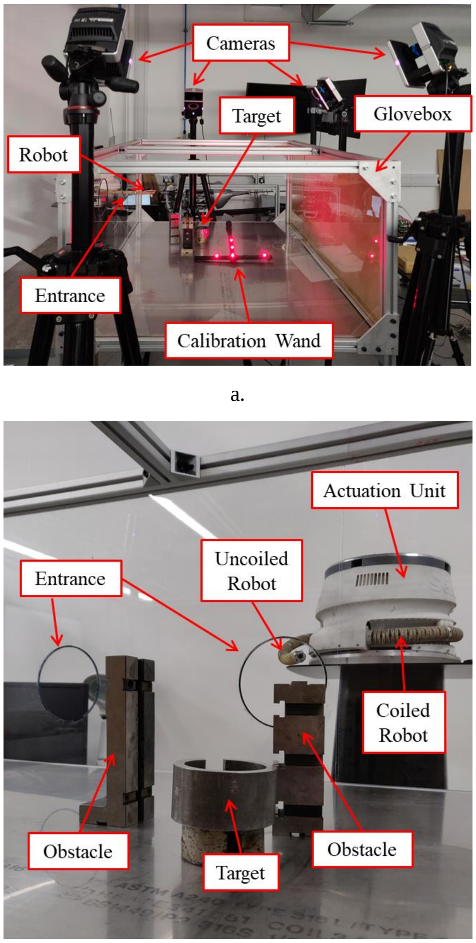


Figure 9. Path of the end-effector of the snake-like robot acquired by the motion capture system during the navigation test.

C. *Results: Glovebox Inspection Test*

A second experimental test demonstrates an application of the robot in another practical scenario for the nuclear industry, glovebox inspection. In this task, the robot has to access the glovebox mockup through a narrow entrance (diameter 150 mm), navigate around obstacles to find a cylinder-shaped inspection target (diameter 72 mm) and inspect its inner volume. The experimental layout is reported in the picture in Fig. 10, with an obstacle on the path from the glovebox entrance to the inspection target. In the test, the robot enters the glovebox from the partially covered access port, navigates around the obstacle and enters the target cylindrical volume, as shown in Fig. 11, where the final configuration of the snake-like robot can be observed with close-ups of the tip inspecting the inner volume of the target.

The robot motion as captured by the VICON system is reported in Fig. 12, with accuracy 0.050 mm from the calibration data in Table III. Once in position, the online tip adjustment provided by the novel strategy proposed in this work enables a remotely operated inspection of the unknown environment which cannot be achieved with previous predetermined path planning methods.



a.

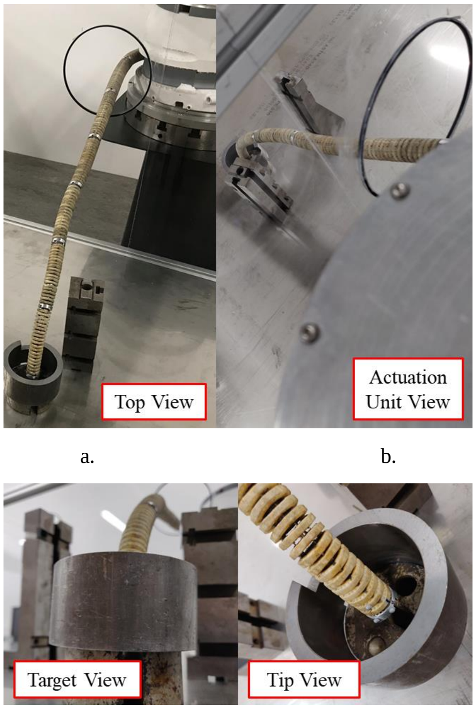
b.

Figure 10. Experimental setup of a snake-like robot for the inspection of a glovebox: a. Side view of the glovebox mockup, with the motion capture system and its calibration setup; b. Inner view of the glovebox mockup layout, with access port, obstacles, and cylindrical inspection target.

D. Results: Uncoiling Test

A third experimental test reports an uncoiling operation, which has been captured through three VICON markers, respectively on the fourth, sixth, and eight section of the continuum robot. In Fig. 13, fifty poses of the robot have been recorded to show how its shape evolves during operation. The time interval between successive poses is set to 0.5 s.

The uncoiling operation has been repeated 20 times over different paths to evaluate the stability and reliability of the proposed algorithm when applied to a real robot, and the motion acquired from five consecutive tests is reported in Figure 14 as example.



a. b.

c. d.

Figure 11. An example of snake-like robot performing a glovebox inspection task with integrated follow-the-leader, coiling, and uncoiling after reaching the inspection target while avoiding any collision with nearby obstacles: a. Top view from a camera inside the glovebox mockup; b. View from a camera on the actuation unit; c. View from the inspection target; d. detailed view of the snake-like robot tip inspecting the inner volume of the cylindrical object.

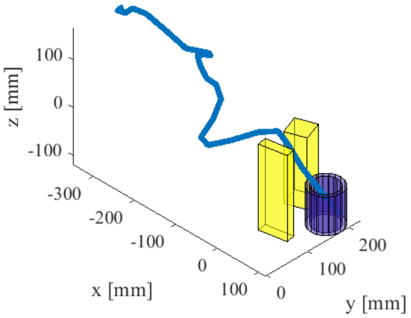


Figure 12. Path of the end-effector of the snake-like robot acquired by the motion capture system during the glovebox inspection test.

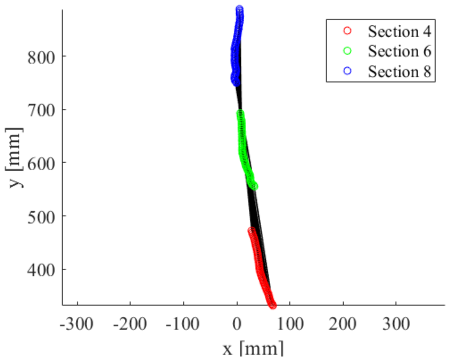


Figure 13. Path of the tips of sections 4, 6, and 8 (corresponding to the end-effector) of the snake-like robot acquired by the motion capture system during an uncoiling test, with the black regions estimating the volume occupied by the body of the robot.

The maximum follow-the-leader deviation error acquired in the reported test is 1.9% of the robot's length, while the repeatability over the different trials is within 1.0% of the robot's length.

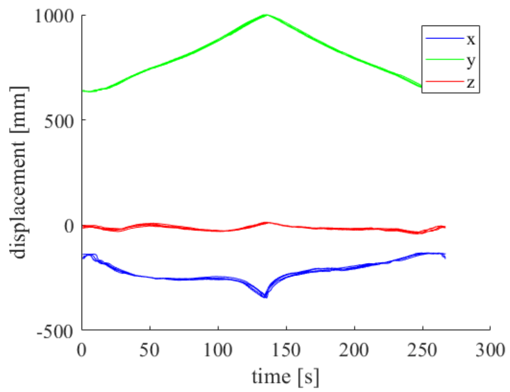


Figure 14. Acquired displacement along x, y, and z of the end-effector during consecutive uncoiling tests. The different acquisitions are overlapped to show the motion repeatability of the continuum robot.

Overall, the experiments validate the proposed method as an efficient navigation strategy. Whereas the previous follow-the-leader method was able to achieve an efficient motion only in a known environment on predetermined tasks with a significant pre-calculation required (170s in [10]), the algorithm introduced in this study is fully integrated with the coiling and uncoiling motion of the robot and it enables online adjustments and teleoperation in unknown environments.

V. Conclusion

In this work, we introduced a novel control algorithm to integrate follow-the-leader strategies for snake-like and continuum robots with coiling and uncoiling capabilities. After analyzing the main navigation strategies for snake-like and continuum robots, we identified an error minimization method for the evaluation of the optimal configuration of a single section and propose a novel linearized formulation for an efficient computation. Then, we expanded the proposed formulation to include follow-the-leader and coiling conditions as constraints and demonstrated the algorithm on a 16-DoF snake-like robot.

The novel algorithm has been successfully demonstrated on a snake-like robotic system for applications in the nuclear industry in three operational scenarios. In the first one, the robot navigates in a cluttered environment. In the second the robot simulates a nuclear decommissioning task by inspecting the inner volume of a narrow cylindrical item, inaccessible for conventional industrial manipulators, in the mockup of a legacy glovebox of a nuclear plant. In the third one, the follow-the-leader deviation error and the repeatability error are evaluated as respectively 2% and 1% of the robot's length for an uncoiling motion.

This study improves existing coiling and uncoiling methods by enabling online adjustments to the position of the end-effector of the robot, as well as by integrating an online follow-the-leader strategy with robot uncoiling and coiling. Thus, the proposed algorithm provides an efficient solution for the online control of both teleoperated and autonomous continuum and snake-like robots.

Possible future developments include refinements to the robot modelling, for example by considering non-uniform bending due to cable friction or by including robot stiffness to compensate undesired bending due to the robot's weight, and shape sensing to improve accuracy and efficiency with closed-loop control.

References

[1] UK Department for Business Innovation & Skills, "UK Aerospace Maintenance, Repair, Overhaul & Logistics Industry Analysis", accessed on 19/10/2020, available online at: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/502588/bis-16-132-uk-mrol-analysis.pdf,

[2] V. Nguyen, J. Johnson and S. Melkote, "Active vibration suppression in robotic milling using optimal control," International Journal of Machine Tools and Manufacture, 152, pp. 2020.

[3] I.D. Walker, H. Choset, and G.S. Chirikjian, "Snake-like and continuum robots," in *Springer handbook of robotics*, B. Siciliano and O. Khatib, Eds. Cham: Springer, 2016, pp. 481-498.

[4] R. Buckingham, A. Graham, "Nuclear snake arm robots." *Industrial Robot: An International Journal* 39, no. 1 (2012): 6-11.

[5] F. Trebuňa, I. Virgala, M. Pástor, T. Lipták, and Ľ. Miková, "An inspection of pipe by snake robot." *International Journal of Advanced Robotic Systems* 13, no. 5 (2016): 1729881416663668.

[6] X. Dong, D. Axinte, D. Palmer, S. Cobos, M. Raffles, A. Rabani, and 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.

[7] M. Russo, L. Raimondi, X. Dong, D. Axinte, J. Kell, "Task-oriented optimal dimensional synthesis of robotic manipulators with limited mobility." *Robotics and Computer-Integrated Manufacturing,* 69 (2021):102096.

[8] E. Amanov, T.D. Nguyen, and J. Burgner-Kahrs, "Tendon-driven continuum robots with extensible sections—A model-based evaluation of path-following motions." *The International Journal of Robotics Research* (2019): 0278364919886047.

[9] J. Burgner-Kahrs, D.C. Rucker, and H. Choset, "Continuum robots for medical applications: A survey." *IEEE Transactions on Robotics* 31, no. 6 (2015): 1261-1280.

[10] D. Palmer and D. Axinte, "Active uncoiling and feeding of a continuum arm robot." *Robotics and Computer-Integrated Manufacturing* 56 (2019): 107-116.

[11] B.A. Jones and I.D. Walker, "Kinematics for multisection continuum robots." *IEEE Transactions on Robotics* 22, no. 1 (2006): 43-55.

[12] R.J. Webster III and B.A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review." *The International Journal of Robotics Research* 29, no. 13 (2010): 1661-1683.

[13] P. Rao, Q. Peyron, S. Lilge, and J. Burgner-Kahrs. "How to model tendon-driven continuum robots and benchmark modelling performance." *Frontiers in Robotics and AI* 7 (2021): 223.

[14] S. Neppalli, M.A. Csencsits, B.A. Jones, and I.D. Walker, "Closed-form inverse kinematics for continuum manipulators." *Advanced Robotics* 23, no. 15 (2009): 2077-2091.

[15] X. Dong, M. Raffles, S. Cobos Guzman, D. Axinte, and J. Kell, "Design and analysis of a family of snake arm robots connected by compliant joints." *Mechanism and Machine Theory* 77 (2014): 73-91.

[16] H. Choset and W. Henning, "A follow-the-leader approach to serpentine robot motion planning." *Journal of Aerospace Engineering* 12, no. 2 (1999): 65-73.

[17] H.B. Gilbert, J. Neimat, and R.J. Webster, "Concentric tube robots as steerable needles: Achieving follow-the-leader deployment." *IEEE Transactions on Robotics* 31, no. 2 (2015): 246-258.

[18] A. Garriga-Casanovas and F. Rodriguez y Baena, "Complete follow-the-leader kinematics using concentric tube robots." *The International Journal of Robotics Research* 37, no. 1 (2018): 197-222.