Robotica (2024), **42**, pp. 3324–3347
doi:10.1017/S0263574724001383

CAMBRIDGE UNIVERSITY PRESS

RESEARCH ARTICLE

A study on path-planning algorithm for a multi-section continuum robot in confined multi-obstacle environments

Guohua Gao, Dongjian Li, Kai Liu, Yuxin Ge and Chunxu Song

College of Mechanical and Energy Engineering, Beijing University of Technology, Beijing, China
Corresponding author: Kai Liu; Email: kailiu@bjut.edu.cn

Received: 12 December 2023; **Revised:** 23 June 2024; **Accepted:** 3 July 2024; **First published online:** 16 September 2024

Keywords: continuum robot; path planning; path-tracking motion; RRT* algorithm; navigation

Abstract

In confined multi-obstacle environments, generating feasible paths for continuum robots is challenging due to the need to avoid obstacles while considering the kinematic limitations of the robot. This paper deals with the path-planning algorithm for continuum robots in confined multi-obstacle environments to prevent their over-deformation. By modifying the tree expansion process of the Rapidly-exploring Random Tree Star (RRT*) algorithm, a path-planning algorithm called the continuum-RRT* algorithm herein is proposed to achieve fewer iterations and faster convergence as well as generating desired paths that adhere to the kinematic limitations of the continuum robots. Then path planning and path tracking are implemented on a tendon-driven four-section continuum robot to validate the effectiveness of the path-planning algorithm. The path-planning results show that the path generated by the algorithm indeed has fewer transitions, and the path generated by the algorithm is closer to the optimal path that satisfies the kinematic limitations of the continuum robot. Furthermore, path-tracking experiments validate the successful navigation of the continuum robot along the algorithm-generated path, exhibiting an error range of 2.51%–3.91%. This attests to the effectiveness of the proposed algorithm in meeting the navigation requirements of continuum robots.

1. Introduction

Mobile robots have a wide range of applications, including detection, search and rescue, and collaborative tasks, in which most mobile robots face limitations in navigating narrow spaces with numerous obstacles [1–3]. This challenge becomes evident in scenarios such as navigating intricate equipment, negotiating cluttered environments after earthquakes, and tackling multi-obstacle situations. Confined multi-obstacle environments are typically characterized by their small and intricate nature, demanding the use of smaller and more agile robots for effective task execution. Continuum robots, inspired by the grasping motions of elephants or the crawling motions of snakes, are a promising class of bionic robots [4–7]. Made from flexible materials, continuum robots boast deformable structures and muscle-like actuators, enhancing their elongated and agile form. Continuum robots, characterized by superior flexibility and degrees of freedom compared to conventional mobile robots, excel in maneuvering through confined spaces cluttered with obstacles, as illustrated in the scenarios mentioned above.

Applications of the continuum robots in medical surgery [8–12], environmental detection, and maintenance under complex and narrow environments [13–18] have received significant attention. In the medical field, continuum robots have been utilized in endoscopic surgery, as elucidated in refs. [19] and [20]. A novel continuum robot for automated surgical blood suction is demonstrated in ref. [21], demonstrating effective trajectory tracking and fluid suction capabilities. A miniature continuum robot with three flexible joints was introduced in ref. [22], designed specifically for visualizing the human bronchial tubes. Environmental inspection and maintenance are also scenarios in which continuum robots have been frequently applied in recent years. For instance, a slender continuum robot tailored

© The Author(s), 2024. Published by Cambridge University Press.

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

for aircraft engine maintenance was presented in ref. [23], while a growable continuum robot capable of firefighting in complex environments was introduced in ref. [24]. Furthermore, the utilization of continuum robots in deep-sea exploration was outlined in ref. [25].

The flexibility and dexterity of continuum robots enable them to adapt effectively to confined spaces. Yet their unique structure and kinematic mechanisms pose challenges in planning feasible paths for these robots in such environments. Path-planning methods for traditional robots were well established [26], including the A* algorithm [27], the rapidly exploring random trees (RRT) algorithm [28], and the artificial potential field method [29]. However, these traditional path-planning algorithms still encounter certain challenges. The A* algorithm suffers from low efficiency, the RRT algorithm exhibits issues of blindness, instability, poor smoothness, and limited obstacle avoidance ability, while the artificial potential field method is prone to easily falling into local minima. Furthermore, innovative techniques for navigating continuum robots in real-time inspections within complex and constrained environments were proposed [30]. However, path planning has not been taken into consideration. In recent years, significant efforts have also been devoted to the study of path planning for continuum robots. Local path-planning techniques for continuum robots have been discussed in various literature. Godage et al. [31] incorporated proximity sensors into the continuum robot to detect environmental information and prevent collisions with obstacles during its motion. Ataka et al. [32] introduced a real-time path-planning algorithm for continuum robots utilizing a potential field approach, with efficient modifications based on the constant curvature assumption. Ouyang et al. [33] introduced a human-robot interactive control system for continuum robots, which proposes a shape correspondence approach based on user-defined desired curves. Additionally, an image-based controller is designed to prevent collisions between the robot and its environment. Wu et al. [34] presented a heuristic inverse kinematics algorithm for continuum robots, broadening its applicability through backward iteration. The effectiveness of the algorithm in trajectory tracking and obstacle avoidance for a multi-section continuum robot was validated through simulations. Although the aforementioned local path-planning methods offer significant advantages in real-time navigation for continuum robots, they still struggle to prevent these robots from getting trapped in local optima while navigating through complex environments, resulting in motion deadlocks. Likewise, global path planning for continuum robots has garnered significant attention. Torres et al. [35] introduced a motion planning method tailored for concentric tube continuum robots. This method employs the rapidly exploring random graph (RRG) approach to pre-compute a collision-free roadmap. However, its applicability is limited to concentric tube continuum robots and does not extend to general continuum robots. Greigarn et al. [36] presented a task-space motion planning technique for MRI-actuated continuum robots, employing the Jacobian approach to generate trajectories guiding the robot's end-effector along a designated path. However, this method may produce inappropriate paths for the continuum robots within the task space. Seleem et al. [37] introduced a Demonstration Guided Pose Planning technique, designed for learning and planning point-to-point spatial motion for multi-section continuum robots. These methods emphasize the transfer of movement skills to the robot via teleoperation but do not include a process for generating feasible paths in the task space of continuum robots. Meng et al. [38] proposed a path-planning method for continuum robots based on their workspace (W-Space), illustrating its application in robot obstacle avoidance and trajectory tracking. However, this method does not explore further the motion characteristics of the continuum robot as well.

For a continuum robot, pre-planning a globally optimal path before it enters a workspace with multiple obstacles can result in more efficient robot movement and reduced navigation time, such as the inspection and maintenance of airplane wings, supercomputers, etc. However, from the above research, it is evident that existing global path-planning algorithms for continuum robots overlook the consideration of the kinematic constraints specific to these robots. Neglecting the kinematic constraints of the continuum robot in path planning can lead to paths that do not align well with the configuration of robots in various scenarios, thereby significantly compromising the accuracy of the path-tracking motion. This paper focuses on global path planning for continuum robots, enhancing the RRT* algorithm [39] to generate paths that adhere to the kinematic constraints of these robots. Although the RRT* algorithm is probabilistically complete, its efficiency is hindered by its inherent randomness. To address this, the

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

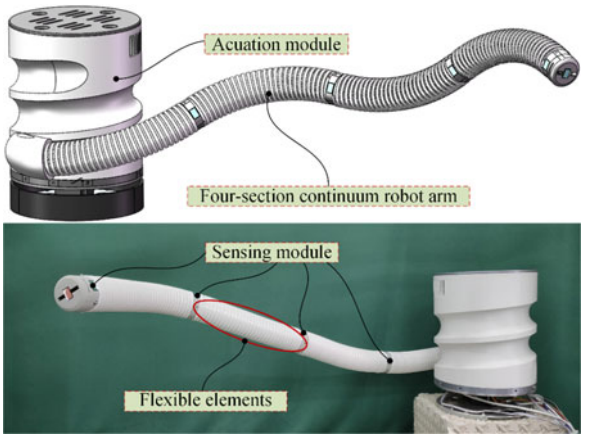


Figure 1. *Virtual prototype and physical prototype of the continuum robot.*

greedy algorithm [40] is integrated with RRT* to curtail its iteration count. To ensure optimality and adherence to continuum robot kinematic constraints, modifications are applied to the rewire rules and node numbers in RRT*. This enhanced algorithm is termed the continuum-RRT* algorithm in this study. The resulting path generated by the smoothed continuum-RRT* algorithm serves as the target path for the designed continuum robot, with path feasibility confirmed through continuum robot path-tracking experiments.

The remainder of this paper is organized as follows: The structure of the continuum robot used in this paper is introduced in Section 2. In Section 3, the motion planning algorithm of the continuum robot is elaborated. Path-tracking experiments of the continuum robot are carried out in Section 4. The conclusion is given in Section 5.

2. Prototype Design and Kinematic Model

2.1. Prototype design

The structure of the continuum robot is proposed as shown in Figure 1, which includes four flexible elements, actuation modules, and sensing modules. As displayed in Figure 2(a), the four-section continuum robot arm is comprised of four flexible elements. Each element features a flexible polyurethane rod material (TPU, Thermoplastic Polyurethane) serving as its backbone. The polyurethane rod material exhibits stable mechanical properties, uniform deformation, and excellent resilience, rendering it highly suitable as the backbone material for the continuum robot arm. On the backbone of each element, circular segments are uniformly distributed along the axial direction of the robot arm. Each circular segment is equipped with three evenly spaced circular holes circumferentially, which serve as passages for drive tendons. Corrugated hoses are wrapped around the continuum robot arm to ensure the torsional stability of the robot arm.

Each individual segment of the continuum robot arm is driven independently by three drive tendons. One end of each drive tendon is anchored to the top segment, while the other end sequentially passes through the circular holes in the middle segment. Metal rings are installed within these holes to minimize friction between the segments and the drive tendons. The connection structure between adjacent continuum robot arms is depicted in Figure 2(a) as well. Flexible tubing is situated beneath each single-section continuum robot arm to guide the drive tendons into the cavity of the following section, preventing spatial interference among the drive tendons of each individual section.

Furthermore, the sensing modules are distributed at the end of each flexible element of the continuum robot, as shown in Figure 2(b). The structure of the actuation module, as depicted in Figure 2(c), comprises a hollow shell known as the spiral drum, featuring a spiral groove on its exterior. The

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

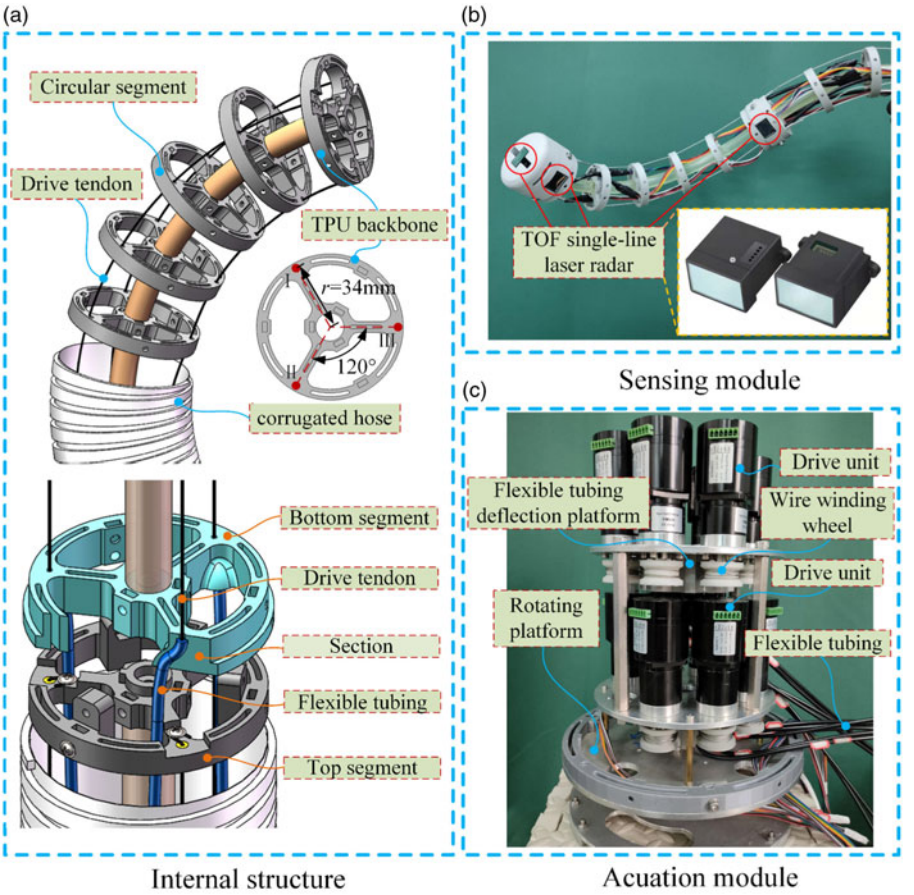


Figure 2. *Flexible elements, actuation modules, and sensing modules of the continuum robot.*

spiral drum is affixed to a rotating table, and several servo motors are positioned inside the spiral drum. In the non-working state of the continuum robot, the servo motors retract the drive tendons, causing the continuum robot arm to coil within the spiral groove of the spiral drum. Conversely, to transition the continuum robot arm into the working state, the drive tendons are released, and the rotating table is rotated, allowing the continuum robot arm to extend and enter the workspace in a linear feed [41].

2.2. Kinematic model

The concept of constant curvature kinematics [42] is employed to depict the motion of the continuum robot. Given that the primary mode of movement for the continuum robot discussed in this paper involves ground-level walking, gravitational forces have minimal impact on the posture of the robot. Regarding the flexible components, their curved configuration resembles a consistent curvature arc. Altering the effective length of the three drive tendons within each flexible element enables the bending of these elements in various planes. As shown in Figure 3, $\{x_a, y_a, z_a\}$ and $\{x_e, y_e, z_e\}$ are the base coordinate system and end coordinate system of the flexible element, respectively, φ is the rotation angle to describe the bending direction and θ is bending angle to describe the bending degree, L is the axial length, where θ and φ are attitude parameters, and L is assigned as $L = 300$ mm herein. θ is determined by the structure

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

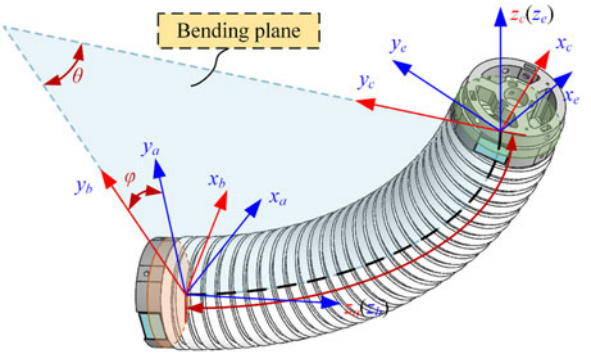


Figure 3. *Kinematic of the flexible element.*

of the flexible elements. The maximum bending angle θ_{max} of the flexible elements is assigned as $2\pi/3$ considering the deformation limits of materials. The homogeneous transformation matrix T from the base coordinate system to the end coordinate system can be expressed as:

$$T = \begin{bmatrix} \cos\theta\cos^2\varphi + \sin^2\varphi & \cos\varphi\sin\varphi(1-\cos\theta) & \cos\varphi\sin\theta & \frac{L}{\theta}\cos\varphi(1-\cos\theta) \\ \cos\varphi\sin\varphi(1-\cos\theta) & \cos^2\varphi + \cos\theta\sin^2\varphi & -\sin\varphi\sin\theta & \frac{L}{\theta}\sin\varphi(1-\cos\theta) \\ -\cos\varphi\sin\theta & \sin\varphi\sin\theta & \cos\theta & \frac{L}{\theta}\sin\theta \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

Thus, the endpoint of the flexible element can be expressed by:

$$\begin{bmatrix} x_e \\ y_e \\ z_e \end{bmatrix} = \begin{cases} \frac{L}{\theta}\cos\varphi(1-\cos\theta) \\ \frac{L}{\theta}\sin\varphi(1-\cos\theta) \\ \frac{L}{\theta}\sin\theta \end{cases} \tag{2}$$

Utilizing the constant curvature assumption, the mathematical correlation between the length of each drive tendon and the attitude parameters can be determined by leveraging the geometric distribution of each drive tendon, thereby achieving forward kinematics.

The kinematics of the driving parameters is illustrated in Figure 4. Here, ρ_i denotes the bending radius of the i-th drive tendon, r represents the diameter of the circle of drive tendon distribution, and d_i signifies the distance between the i-th drive tendon and the bending plane as well as the difference between ρ_i and ρ. Based on trigonometric derivation, the relationship between d_i and the attitude parameter φ can be obtained as follows:

$$\{|d_i|\} = \begin{cases} |d_1| = r\sin\varphi \\ |d_2| = r\sin\left(\varphi + \frac{\pi}{3}\right) \\ |d_3| = r\sin\left(\varphi + \frac{2\pi}{3}\right) \end{cases} \tag{3}$$

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

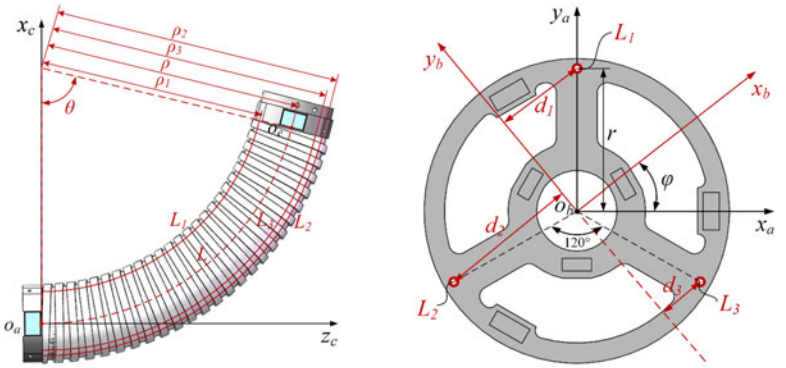


Figure 4. *Kinematics of driving parameters.*

Based on the central angle theorem, θd_i can characterize the driving amount of the drive tendons. However, since the drive tendons are distributed on different sides of the bent plane, the robot exhibits different states of elongation or contraction in the bent state. Therefore, it is necessary to assign a positive or negative property to d_i. Adjusting the expression of d_i based on Eq. (3) yields the driving amounts of each drive tendon.

$$\{\Delta L_i\} = \{\theta d_i\} = \begin{cases} \Delta L_1 = r\theta \cos\left(\varphi + \dfrac{\pi}{2}\right) \\ \Delta L_2 = r\theta \cos\left(\varphi - \dfrac{\pi}{6}\right) \\ \Delta L_3 = r\theta \cos\left(\varphi - \dfrac{5\pi}{6}\right) \end{cases} \tag{4}$$

According to Eq. (4), one can derive the relationship between the effective length of each drive tendon and the attitude parameters.

$$L_i = L + r\theta \cos\left(\varphi + \frac{\pi}{2} - (i-1)\frac{2\pi}{3}\right), i = 1, 2, 3 \tag{5}$$

According to the forward kinematic of the series robot, the expression of the homogeneous transformation matrix of the four-section continuum robot arm from its base coordinate system to the end coordinate system can be obtained as:

$${}^0T_4 = \prod_{i=1}^{4} {}^{i-1}T_i = {}^0T_1\,{}^1T_2\,{}^2T_3\,{}^3T_4 \tag{6}$$

where ${}^{i-1}T_i$ represents the homogeneous transformation matrix from the base coordinate system $\{x_{i-1}, y_{i-1}, z_{i-1}\}$ of the i-th flexible element to the base coordinate system $\{x_i, y_i, z_i\}$ of the $i+1$ flexible element, as shown in Figure 5. In light of Eq. (1), ${}^{i-1}T_i$ could be simplified as:

$${}^{i-1}\boldsymbol{T}_i = \begin{bmatrix} {}^{i-1}_{i}v_x & {}^{i-1}_{i}u_x & {}^{i-1}_{i}t_x & {}^{i-1}_{i}s_x \\ {}^{i-1}_{i}v_y & {}^{i-1}_{i}u_y & {}^{i-1}_{i}t_y & {}^{i-1}_{i}s_y \\ {}^{i-1}_{i}v_z & {}^{i-1}_{i}u_z & {}^{i-1}_{i}t_z & {}^{i-1}_{i}s_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

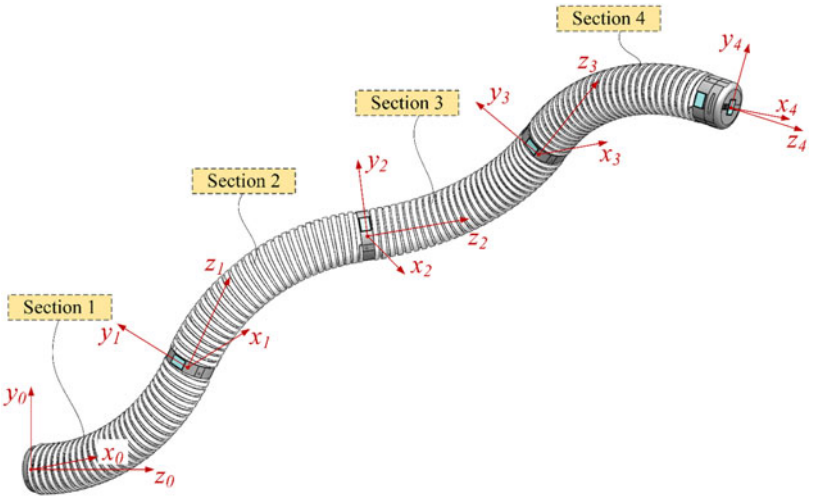


Figure 5. *Kinematic of the four-section continuum robot arm.*

If the pose information of a certain section end coordinate system relative to the base coordinate system in a multi-section robot is known, that is, the homogeneous transformation matrix ${}^{i-1}T_i$ is given, then the expression for solving the attitude parameters of this section based on ${}^{i-1}T_i$ is as follows:

$$\begin{cases} \theta_i = \arccos\left({}_i^{i-1}t_z\right) \\ \varphi_i = \begin{cases} \arctan\left(\dfrac{{}_t^{t-1}t_y}{{}_t^{t-1}t_x}\right), & {}_i^{i-1}t_x > 0 \\ \pi + \arctan\left(\dfrac{{}_t^{t-1}t_y}{{}_t^{t-1}t_x}\right), & {}_i^{i-1}t_x < 0, {}_i^{i-1}t_y \geq 0 \\ -\pi + \arctan\left(\dfrac{{}_t^{t-1}t_y}{{}_t^{t-1}t_x}\right), & {}_i^{i-1}t_x < 0, {}_i^{i-1}t_y < 0 \\ \dfrac{\pi}{2}, & {}_i^{i-1}t_x = 0, {}_i^{i-1}t_y > 0 \\ -\dfrac{\pi}{2}, & {}_i^{i-1}t_x = 0, {}_i^{i-1}t_y < 0 \\ 0, & {}_i^{i-1}t_x = 0, {}_i^{i-1}t_y = 0 \end{cases} \end{cases} \tag{7}$$

From the structural design introduction of the multi-segment continuum robot in Figure 1, it is evident that each flexible section of the robot is driven by three tendons. Therefore, the actuation between the robotic sections is decoupled. Thus, based on the forward kinematics functions of the joint space and the driving space of the flexible section, the relationship between the effective length of the drive tendons of section j and the attitude parameters of that section is as follows:

$$L_{ji} = L_j + r\theta_j \cos\left(\varphi_j + \frac{\pi}{2} - (i-1)\frac{2\pi}{3}\right), j = 1, 2, 3, 4, i = 1, 2, 3 \tag{8}$$

According to the inverse kinematics functions of the joint space and the driving space of the single-section robot, the expression describing the attitude parameters of section j of the robot in terms of the

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

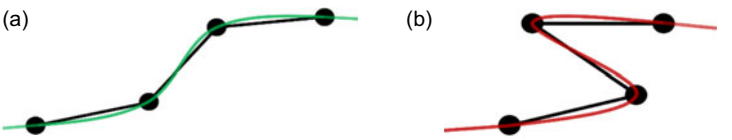


Figure 6. *Schematic diagram of feasible and infeasible paths for continuum robot.*

effective length of each drive tendon of that section is as follows:

$$\begin{cases} \theta_j = \dfrac{2A_j}{3r} \\ \varphi_j = \begin{cases} 2\arctan\left(\dfrac{\sqrt{3}\left(L_{j3}-L_{j2}\right)+2A_j}{L_{j2}-2L_{j1}+L_{j3}}\right), & L_{j2}-2L_{j1}+L_{j3}\neq 0 \\ 0, & L_{j2}-2L_{j1}+L_{j3}=0, L_{j2}>L_{j3} \\ \pi, & L_{j2}-2L_{j1}+L_{j3}=0, L_{j2}<L_{j3} \\ 0, & L_{j2}-2L_{j1}+L_{j3}=0, L_{j2}=L_{j3} \end{cases} \quad j=1,2,3,4 \\ A_j = \sqrt{L_{j1}^2-L_{j1}L_{j2}-L_{j1}L_{j3}+L_{j2}^2-L_{j2}L_{j3}+L_{j3}^2} \end{cases} \tag{9}$$

3. Path Planning for the Continuum Robot

3.1. Robot path planning

Path planning is essential for continuum robots to perform navigation in muti-obstacle environments. Their flexible structure provides remarkable adaptability. Ensuring obstacle-free motion is crucial for maintaining motion accuracy. According to the constant curvature kinematic, the shape of the four-section continuum robot arm is essentially composed of four constant curvature arcs. The bending angle θ of the flexible elements is confined by the material and structural constraints of the continuum robot. The maximum bending angle $\theta_{\max}$ of the flexible elements is assigned $2\pi/3$ herein considering the deformation limits of materials. In such a condition, Figure 6 illustrates the passable and impassable paths of the continuum robot. In Figure 6(b), the kinematic solution of the continuum robot cannot be obtained due to the excessively large bending angle of the polyline path. Therefore, path planning for continuum robots needs to satisfy the following constraints:

(i) The path curve is continuous and differentiable.

(ii) The curvature at each point along the path curve should not exceed the maximum bending curvature of the continuum robot.

For paths that satisfy the mentioned constraints, a theoretical pose exists at each point along the path that can minimize the cumulative deviation between the continuum robot and the path curve. As a result, during the path-tracking motion of the continuum robot along such a path, a substantial level of alignment with the path can be achieved.

The "follow-the-leader" motion strategy [43] is adopted for the continuum robot in this study. By employing this strategy, the process of path planning for the continuum robot is analogous to devising a trajectory for a mobile robot characterized by complex motion constraints. Drawing from mobile robot path-planning techniques, the RRT* algorithm, known for its optimality, is employed. Nevertheless, modifications are required to customize the path according to the distinctive attributes of the continuum robot.

Based on the aforementioned analysis, the following improvements are implemented to the RRT* algorithm to match the kinematic characteristics of the continuum robot.

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

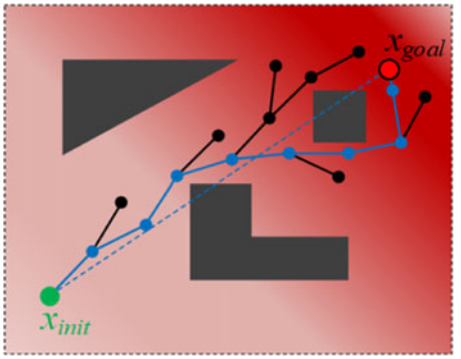


Figure 7. *RRT* based on heuristic-biased sampling.*

3.1.1. Heuristic bias sampling

In the RRT* algorithm, x_{init}, x_{goal}, and x_{rand} are used to denote the starting node, the target node, and the randomly generated nodes in an input map, respectively. To produce feasible paths for the continuum robot, it is essential for the low-dimensional polyline paths generated by the RRT* algorithm to be relatively smooth. To achieve this, it becomes necessary to intervene in the random sampling process of the RRT* algorithm, specifying the tree expansion direction to ensure smoother path generation. By modifying the sample function in the RRT* algorithm, it is possible to generate nodes x_{rand} closer to the target node x_{goal}. Consequently, the expansion of the tree in the RRT* algorithm will tend to converge toward the target node x_{goal}, resulting in the connection of the nodes in the tree also tending to be distributed near the straight line from x_{init} to x_{goal}. As a result, nodes in the vicinity of the target node x_{goal} will have a higher probability of being sampled, leading to smoother branches in the tree.

A two-dimensional random variable (X, Y) is employed to denote the positions of the randomly sampled nodes in the two-dimensional configuration space. The random variables X and Y correspond to the horizontal and vertical coordinates of the sampling nodes, respectively. The probability of the random sampling nodes being selected is determined by the probability density function $f(x, y)$ of the two-dimensional random variable (X, Y). $f(x, y)$ can be defined as:

$$f(x,y) = \frac{k}{2\pi} e^{-\frac{(x-u_{goal})^2+(y-v_{goal})^2}{2\varepsilon^2}} \tag{10}$$

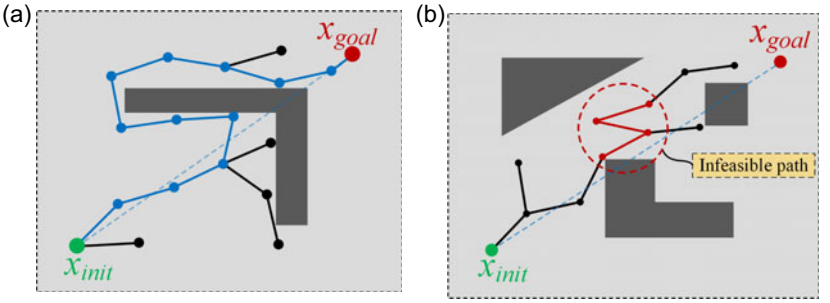
where u_{goal} and v_{goal} are the horizontal and vertical coordinates of the target node x_{goal} in the input map of the RRT* algorithm, respectively; ε is the bias coefficient, which determines the bias degree of the random sampling nodes x_{rand} toward the target node x_{goal}. k is the adaptation coefficient, which can be adjusted for different input maps according to:

$$\int_{x_{\min}}^{x_{\max}} \int_{y_{\min}}^{y_{\max}} f(x,y)\,\mathrm{d}x\mathrm{d}y = \int_{x_{\min}}^{x_{\max}} \int_{y_{\min}}^{y_{\max}} \frac{k}{2\pi} e^{-\frac{(x-u_{goal})^2+(y-v_{goal})^2}{2\varepsilon^2}} \mathrm{d}x\mathrm{d}y = 1 \tag{11}$$

The above modification to the sample function is called heuristic bias sampling. After the above modification, the path generated by the RRT* algorithm is shown in Figure 7. As evident from Figure 7, the sampling nodes demonstrate a tendency to cluster along the connection between the starting node x_{init} and the target node x_{goal}. Moreover, regions closer to the target node x_{goal} exhibit a higher probability of the sampling nodes being selected.

The value of the bias coefficient ε is the key to set the sampling rule of the algorithm. A smaller ε value directs the tree expansion more definitively from the initial node to the target node, increasing the likelihood of obtaining feasible paths while reducing the number of nodes needed to expand the tree in search of the target node. This enhances the efficiency of the algorithm. Nevertheless, as ε decreases, the nodes within the tree tend to expand with increased greediness. In relation to the greedy algorithm, it becomes apparent that excessive intervention in the sampling process can lead to suboptimal paths

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press



Weakening of path optimality *Infeasible node exploration*

Figure 8. *Limitations of the biased sampling improvement for RRT*.*

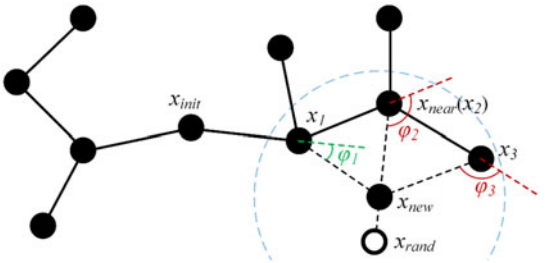


Figure 9. *Parent node selection for increased angle constraints.*

in certain scenarios. This is illustrated in Figure 8(a), where the algorithm requires more iterations to achieve the optimal path.

Besides, the modification of the sample function cannot guarantee that each expansion step will align with the kinematic characteristics of the continuum robot. For instance, as shown in Figure 8(b), certain adjacent polyline segments exhibit overly large angles, significantly diverging from the expected kinematic characteristics of the continuum robot. Therefore, beyond refining the sampling function, enhancements must also be made in parent node selection and rewire rules to ensure that the algorithm generates paths aligning with the kinematic characteristics of the continuum robot.

3.1.2. Parent nodes selection and rewire

In the low-dimensional polyline paths generated by the RRT* algorithm, the angle between adjacent polyline segments is essentially the steering angle for the mobile robots and is the bending angle θ for the single-section continuum robot. As shown in Figure 9, when the newly generated node x_{new} of the RRT* algorithm selects its parent node, it needs to traverse all nodes in the circle centered on itself to find a node with the lowest cost as its parent node. For ease of description, the parent node of x_{new} is recorded as x_{j}, and the parent node of x_{j} is recorded as x_{jparent}. To obtain a path that satisfies the kinematic characteristics of the continuum robot, φ_j in Figure 9 must satisfy $\varphi_j < \theta_{\max}$.

Once the set of x_{j} nodes satisfying the above constraint is identified, the next step is to compute the path costs from x_{init} to x_{new} through each x_{j} node. Among these costs, the one associated with the lowest value is recorded as x_{best}. Subsequently, x_{best} is chosen as the parent node of x_{new}, and x_{new} is connected to x_{best}, as depicted in Figure 10(a).

The above process can be realized by changing the ChooseParent function in the RRT* algorithm. Similarly, the wiring process in RRT* will change the connections between the nodes, so it is also necessary to add angle constraints to the rewiring rules. As shown in Figure 10(a), traverse other nodes x_j except x_{best} within the range centered on x_{new}, such as x_2, x_3. If the angle between the connection of x_{best}

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

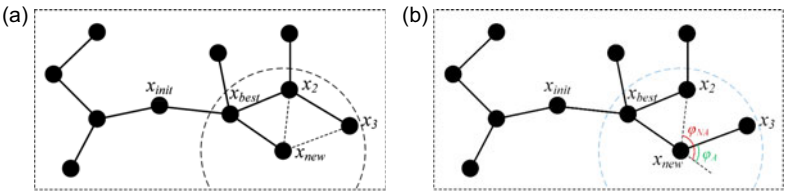


Figure 10. *Tree structure exploration based on angle constraints.*

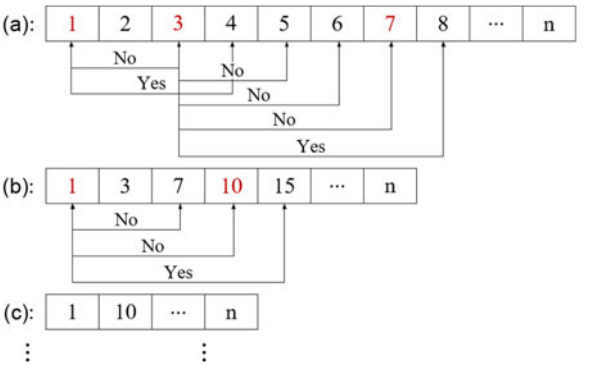


Figure 11. *Node optimization of feasible paths.*

and x_{new} and the connection of x_{new} and x_j satisfies the relation in φ_j<θ_{max} and the cost of the path from x_{init} to x_j through x_{new} is lower compared to the cost of the path through x_j in the original tree structure, change the parent node of x_j to x_{new} and delete the connection with its original parent node. As shown in Figure 10(b), the original tree structure $x_{best}x_2x_3$ is rewired into $x_{best}x_{new}x_3$ with a lower path cost.

3.1.3. Node optimization

The path obtained after the above process may still contain a substantial number of nodes with multiple turns. When the continuum robot follows such a path for path-tracking, the unevenness of the path can make it challenging to precisely conform the robot to the path, resulting in increased motion errors. To address this issue, it is necessary to optimize the output path by removing redundant nodes and minimizing steering in the path. This optimization process aims to create a path that better aligns with the kinematic characteristics of the continuum robot.

As shown in Figure 11, an array A is employed to store the path nodes output by the above modified RRT* algorithm, where element 1 represents the starting node x_{init}, element n represents the target node x_{goal}, and elements 2∼(n-1) represent the feasible nodes between the starting node x_{init} and the target node x_{goal}, and the specific steps for optimizing nodes are as follows:

(1) Defining a new array B and adding the starting node x_{init} to B;

(2) If the connection between node k and node $k+1$ does not interfere with the obstacle in the input map, determine whether the connection between node k and node $k+2$ interferes with the obstacle. If not, skip node $k+2$ to determine whether the connection between node k and node $k+3$ interferes with the obstacle; if so, add the parent node of node $k+3$ to array B, and then search backward with $k+2$ as the starting node, and so on, where node k starts from the starting node x_{init};

(3) Repeat the process in (2) until the target node x_{goal} is searched, then stop searching and add x_{goal} to array B;

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

Table I. *The algorithm for continuum robot.*

Algorithm: RRT* for Continuum Robot (Continuum-RRT*)

Input: M, x_{init}, x_{goal}
Result: A path ξ from x_{init} to x_{goal}
T.init(), T.addNode(x_{init});
for i = 1 to n **do**
  SetBiasProbability(M);
  x_{rand} = *Sample*(M);
  x_{near} = *Near*(x_{rand}, T);
  x_{new} = *Steer*(x_{near}, x_{rand}, *StepSize*);
  x_j = *NearC*(T, x_{new}, R_{new});
  C_i.init();
  x_k = *AngleCheck*(x_{new}, x_j, $x_{jparent}$);
  C_i.addNode(x_k);
  x_{best} = *ChooseParent*(C_i);
  E_i = *Edge*(x_{new}, x_{best});
  if *CollisionFree*(M, E_i) **then**
    T.addNode(x_{new}), T.addEdge(E_i);
    x_A = *AngleCheck*(x_{best}, x_{new}, x_j);
    T.rewire(x_A);
  if x_{new} = x_{goal} **then**
    ξ = *CallBack*(T);
    ξ = *NodeOptimization*(ξ);
    Success();

Note: M denotes input map, x_{init} denotes starting node, x_{goal} denotes target node, x_{rand} denotes randomly generated node, x_{near} denotes neighboring node of x_{rand}, x_{new} denotes new generated node, R_{new} denotes radius of circular area centered on x_{new}, T denotes RRT* primitive tree structure, x_j denotes parent node of x_{new}, $x_{jparent}$ denotes parent node of x_j, x_k denotes x_j that satisfies the angle constraint, x_{best} denotes x_j with lowest path cost, x_A denotes x_{new} that satisfied the angle constraint in rewire process, and C_i denotes improved RRT* tree structure.

(4) Define a new array C, insert the starting node x_{init} into array C, renumber the elements in array B and repeat the process in (2) until x_{goal} is found, and add it to array C;

(5) Repeat the process from (2) ∼ (4) until a new array can be defined where the elements of the array no longer change.

The RRT* algorithm modified by steps (a), (b), and (c) is renamed as continuum-RRT* algorithm, and its pseudo-code is shown in Table 1.

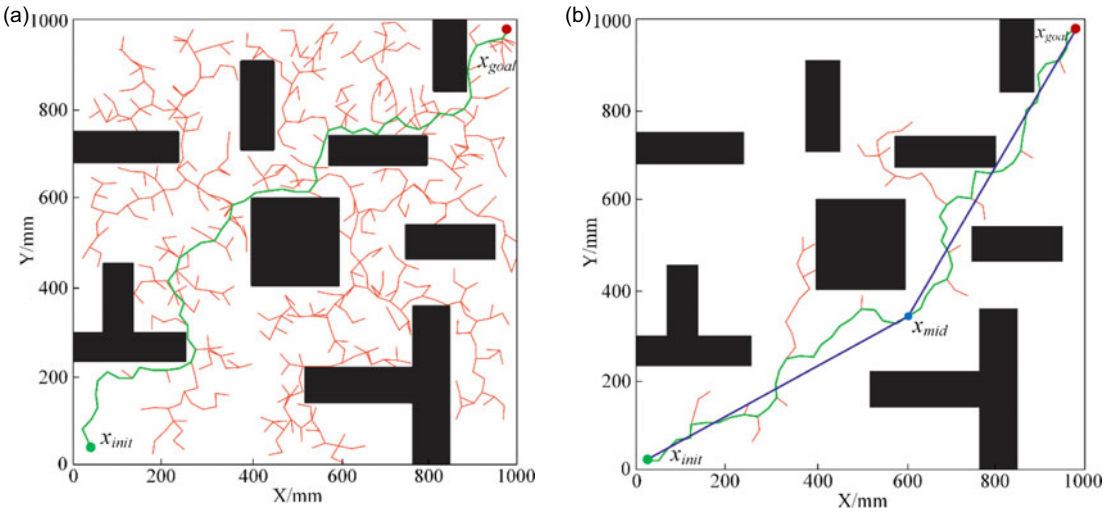
3.2. Simulation of the continuum-RRT algorithm*

To validate the feasibility of the continuum-RRT* algorithm, simulations are conducted using a 1000 mm*1000 mm raster map with a map resolution of 1 mm, as shown in Figure 12. The start node x_{init} and the target node x_{goal} are set to coordinates (20, 20) and (990, 990), respectively. Obstacles in the map are represented by dark rasters.

Figure 12(a) shows the results generated by the basic RRT* algorithm, and Figure 12(b) shows the simulation results of the continuum-RRT* algorithm. The latter produces paths with significant directionality and has reduced the number of node extensions in the RRT* algorithm.

The continuum-RRT* algorithm and the basic RRT/RRT* algorithms are simulated 50 times in randomized multi-obstacle environments, utilizing the same tree structure expansion step. In each

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press



Basic RRT simulation results* *Continuum-RRT* simulation results*

Figure 12. *Simulation analysis of heuristic bias sampling.*

simulation, the generated path length, the number of iterations, and the running time were recorded, as shown in Figure 13. Table 2 summarizes the average path length and average number of iterations obtained from both algorithm simulations. It is noteworthy that the path length is calculated as the sum of the Euclidean distances between adjacent nodes along the path generated by the algorithm.

In the aforementioned 50 simulations, the average number of nodes generated by the three algorithms is approximately 102, 604, and 1401, respectively. Compared to the RRT* and RRT algorithms, the continuum-RRT* algorithm yields an average path length reduction of 2.21% and 11.27%, respectively, along with an average iteration number reduction of 67.76% and 69.75%, respectively. The continuum-RRT* algorithm in this paper demonstrates a reduced path length and a decreased number of iterations, effectively mitigating the randomness and lack of guidance in tree structure expansion. By circumventing exhaustive global searches and curbing the iteration count, the algorithm achieves faster convergence. Furthermore, the feasible path generated by the continuum-RRT* algorithm exhibits increased proximity to the optimal path, aligning with the motion prerequisites of the continuum robot. Nevertheless, the continuum-RRT* algorithm exhibits a longer runtime primarily because the heuristic bias sampling necessitates assigning sampling probabilities to all points in the configuration space, resulting in computational overhead.

3.3. Robot trajectory generation

Path smoothing is necessary to convert the output path in Figure 12 into a suitable trajectory. The B-spline curve [44] is used to smooth the path generated by the continuum-RRT* algorithm. Figure 14 shows the result of a polyline path smoothed by the clamped B-spline curve.

In order to ensure that the curve trajectory does not interfere with obstacles, local nodes are inserted into the initial path, and the B-spline curve trajectory is made to fit the initial path more closely by adding control points, as shown in Figure 15.

In Figure 15, $A_{k\text{-}1}$, A_k, and A_{k+1} are the nodes in the initial path, and local nodes B_k and B_{k+1} are generated at the two ends of A_k. Since the angle of each line segment in the initial path and the distance between each node and the obstacles in the initial path are different, the selection of local node positions adopts an adaptive method. The horizontal coordinates of the local node B_k satisfy:

$$|x(B_k) - x(A_k)| = \frac{1}{m_k} |x(A_k) - x(A_{k-1})| \tag{12}$$

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

Table II. *Average data of simulation results.*

Algorithm	Average path length (mm)	Average iteration number	Average running time (s)
Continuum-RRT*	1506.56	533	296.16
RRT*	1540.62	1653	108.208
RRT	1697.88	1762	49.64

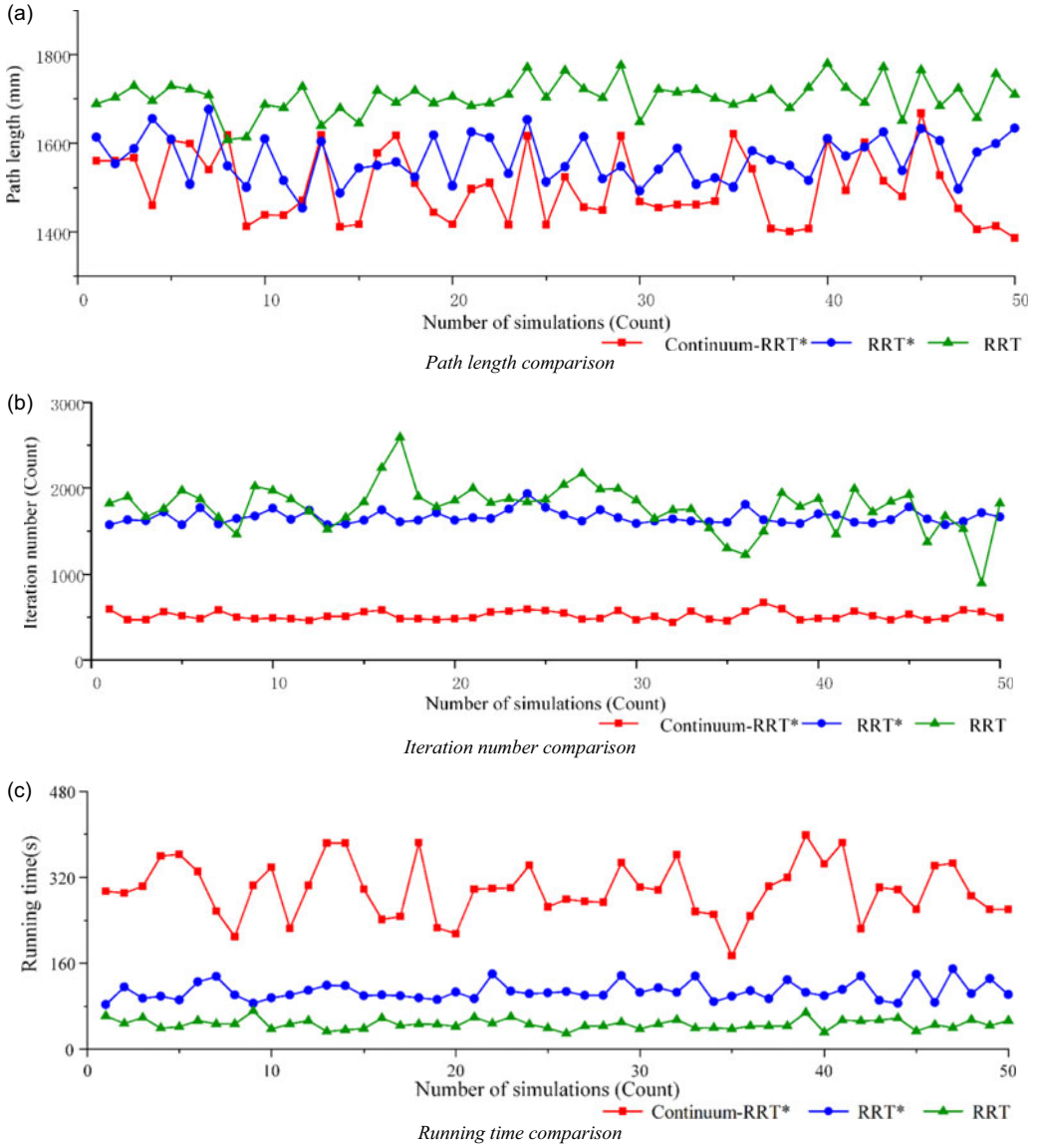


Figure 13. *Comparison analysis of continuum-RRT* and basic RRT/RRT*.*

where m_k is the adaptive coefficient, which is determined according to the positional relationship between each path segment and the obstacles. If the distance between the path and obstacles is short, choose a larger value for m_k; otherwise, opt for a smaller m_k. The coordinates of each local node can be obtained according to the linear equations of different path segments and the adaptive coefficient m_k.

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

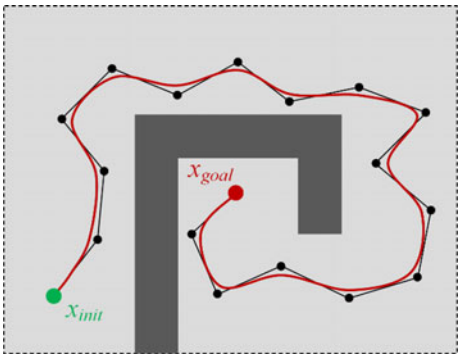


Figure 14. *The result after the polyline path is processed by the clamped B-spline curve.*

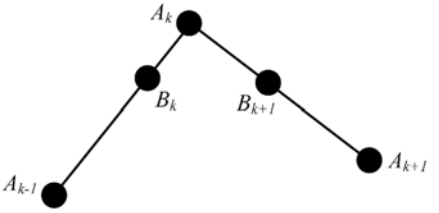


Figure 15. *Insert local nodes.*

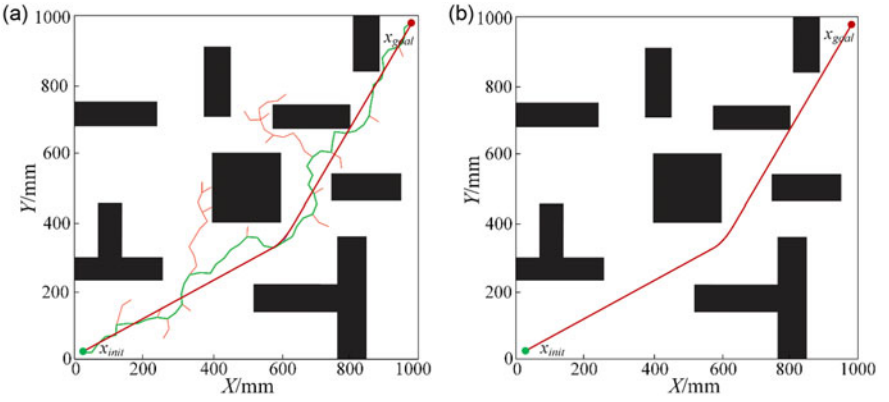


Figure 16. *Trajectory generation based on the B-spline curve.*

The local nodes B_i ($i = 0, 1, 2, \ldots, n$) are introduced into the node set depicted in Figure 12(b), serving as control points for the clamped B-spline curve. This facilitates the smoothing of the initial path. The simulation outcome is demonstrated in Figure 16, where $m_1 = m_2 = 17$ according to simulation comparisons.

3.4. Path-tracking for continuum robot

3.4.1. Discretized representation of robot posture

The flexible element is represented by 65 discrete points, and the distance between each discrete point is 5 mm. According to the constant curvature kinematic, the homogeneous transformation matrix ${}^{A}T_j$

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

from the base coordinate system origin oA to the end coordinate system origin o_j in section j can be expressed as:

$$ {}^{A}T_j = \begin{bmatrix} R_j & N_j \\ 0 & 1 \end{bmatrix} \tag{13} $$

where R_j and N_j denote the pose matrix and position matrix of the end coordinate system of section j with respect to the base coordinate system, respectively. According to Eq. (2), the position of the k-th discrete point of the section j in the coordinate system $\{x_{j\text{-}1}, y_{j\text{-}1}, z_{j\text{-}1}\}$ can be expressed as:

$$ N_j^k = \begin{bmatrix} Nx_j^k \\ Ny_j^k \\ Nz_j^k \end{bmatrix} = \begin{cases} (0, 0, 5\,(k-1))^T, & k \le 4 \\ \begin{cases} \dfrac{L}{\theta_j}\cos\varphi_j\left(1-\cos\left(\dfrac{\theta_j}{52}\,(k-2)\right)\right) \\ \dfrac{L}{\theta_j}\sin\varphi_j\left(1-\cos\left(\dfrac{\theta_j}{52}\,(k-2)\right)\right) \\ \dfrac{L}{\theta_j}\sin\varphi_j\left(\dfrac{\theta_j}{52}(k-2)\right) \end{cases}, & k \le 64 \\ N_j, & k = 65 \end{cases} \tag{14} $$

The position of the k-th discrete point in the section j in the base coordinate system $\{x_0, y_0, z_0\}$ can be calculated by the following expression:

$$ \begin{bmatrix} x_j^k \\ y_j^k \\ z_j^k \end{bmatrix} = R_{j-1} \begin{bmatrix} Nx_j^k \\ Ny_j^k \\ Nz_j^k \end{bmatrix} + N_{j-1} \tag{15} $$

When the posture parameters $[\theta_j, \varphi_j]$ $(j = 1, 2, 3, 4)$ of the four-section continuum robot arm are obtained, the coordinates of its discrete points can be found by Eqs. (13) and (14).

3.4.2. Solving for posture parameters

Figure 17(a) and Figure 17(b) show the path deviation and tip deviation between the manipulator and its feasible path, respectively, where $P(n)$ and $Q(n)$ are the manipulator discrete points and the feasible path discrete points series, respectively. Path deviation and tip deviation can be represented by the Euclidean distance:

$$ E_{path} = \sqrt{\frac{\sum_{i=1}^{n}\left(P_x^n - Q_x^n\right)^2 + \left(P_y^n - Q_y^n\right)^2}{n}} \tag{16} $$

$$ E_{tip} = \sqrt{\left(P_x^n - Q_x^n\right)^2 + \left(P_y^n - Q_y^n\right)^2} \tag{17} $$

Given that the continuum robot in this paper primarily moves in a planar manner, both path deviation and tip deviation are calculated within a 2D space. Establish objective function according to Eq. (16) and Eq. (17)

$$ F(P, Q) = E_{path} + \lambda E_{tip} \tag{18} $$

where λ is a weighting factor to regulate the share of the tip deviation in the objective function.

The standard particle swarm algorithm (PSO) was used to solve the objective function $F(P, Q)$ optimally. The feasible path Q and the objective function $F(P, Q)$ of the four-section continuum manipulator are used as inputs to the PSO algorithm, and the output is the optimal posture parameter at each moment.

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

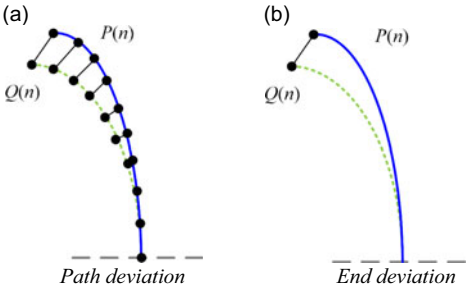


Figure 17. *Error in path-tracking motion.*

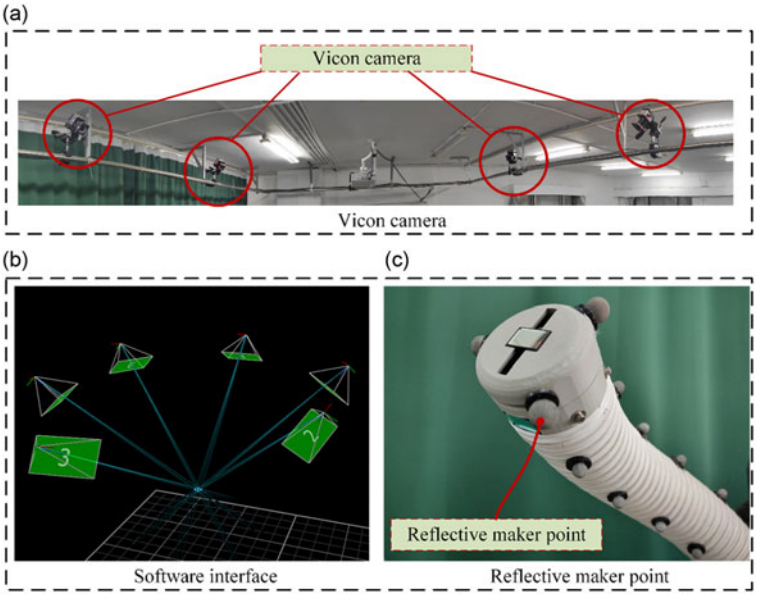


Figure 18. *VICON 3D motion capture system.*

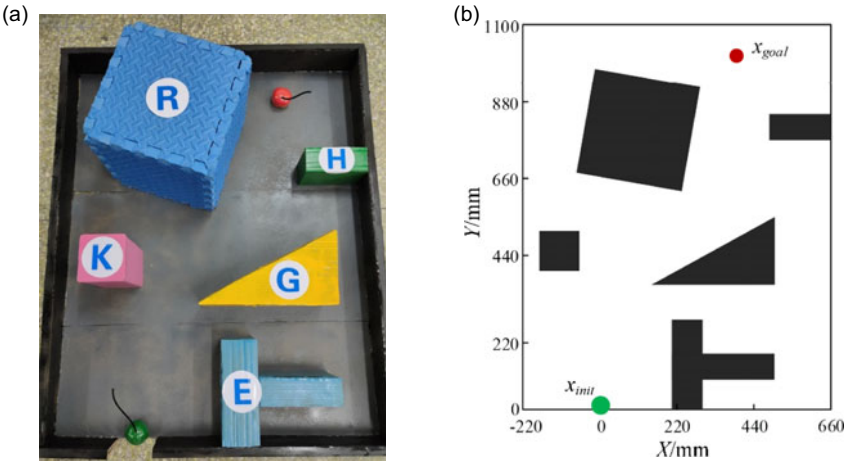
4. Experiment

In the previous section, the path-planning algorithm proposed is introduced and simulated. Whether the smoothed trajectory generation based on the B-spline curve meets the motion requirements of the continuum robot needs to be evaluated by the path-tracking experiment of the continuum robot. Herein, an experimental scenario is constructed as an input to the continuum-RRT* algorithm, and the output path of the algorithm, after B-spline processing, is used as the trajectory for the path-tracking experiment of the continuum robot. The Vicon 3D motion capture system is used to evaluate the attitude changes of the continuum robot along the planned trajectory.

4.1. Construction of experimental platform

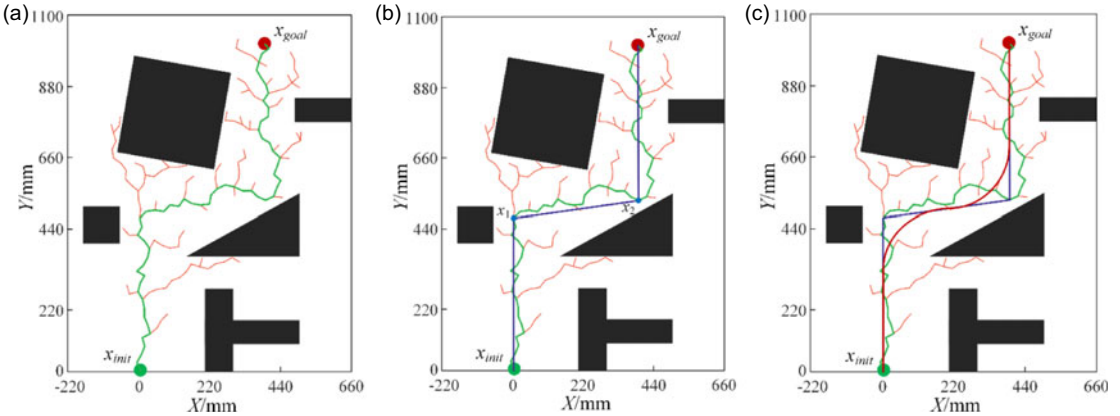
The experimental platform built is shown in Figure 18. The spatial pose of the continuum robot is collected through the Vicon 3D motion capture system, the acquisition accuracy of this device is 0.001 mm, and the limit acquisition frequency is 330 HZ. Its software can output the position of the geometric center of the captured rigid body and the attitude rotation angle of the rigid body coordinate system relative to the calibration coordinate system. As shown in Figure 18, by arranging the reflective marker points evenly around the robot arm, the capture system can collect the central axis posture of the robot arm.

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press



Experimental scene *Grid map representation*

Figure 19. *Experimental scene design and its map representation.*



Path search *Node optimization* *Path generation*

Figure 20. *Path planning for an experimental scene.*

4.2. Path-tracking experiment

The experimental scene of the path-tracking of the continuum robot is shown in Figure 19(a). The size of the experimental scene is 1100 mm*880 mm, where the red sphere is the target node x_{goal}, the green sphere is the initial node x_{init}, and the irregularly shaped objects are obstacles. The above experimental scene is represented as a two-dimensional grid map, as shown in Figure 19(b), with 1 mm map resolution. This two-dimensional raster map serves as an input to the continuum-RRT* algorithm, and the algorithm simulation process is shown in Figure 20.

The motion of the four-section continuum robot arm following the target path in the experimental map is shown in Figure 21, where the spatial position of the robot arm at six different step sizes is recorded.

Throughout the movement process, there are no collisions between any parts of the continuum robot and the obstacles, affirming that the robot successfully navigates from the starting node to the target node of the experimental map.

While collecting data on the motion of the physical prototype, simulations of path-tracking motions under the target path are conducted simultaneously. The tip error and path deviation of the robot obtained

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

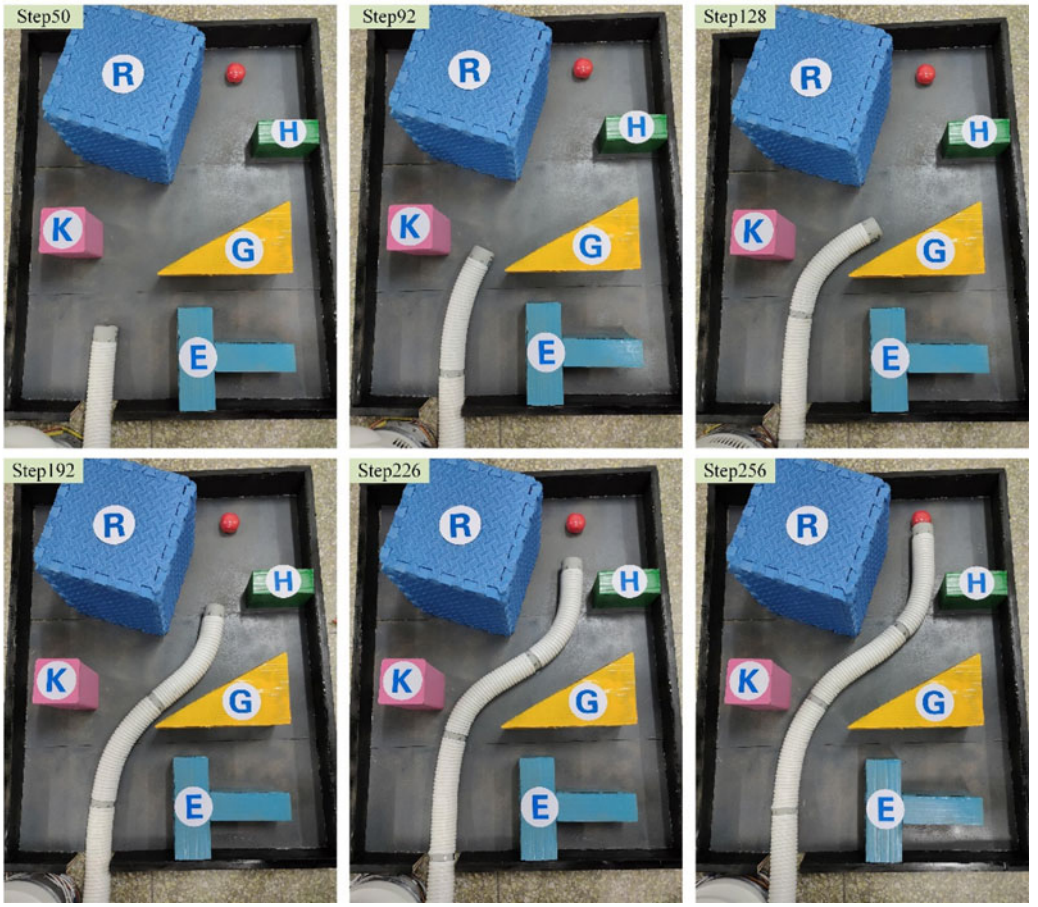
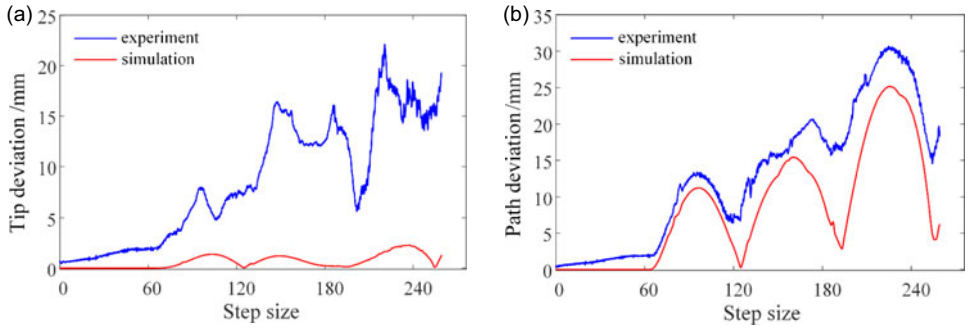


Figure 21. *Robot path-tracking movement along the target path.*



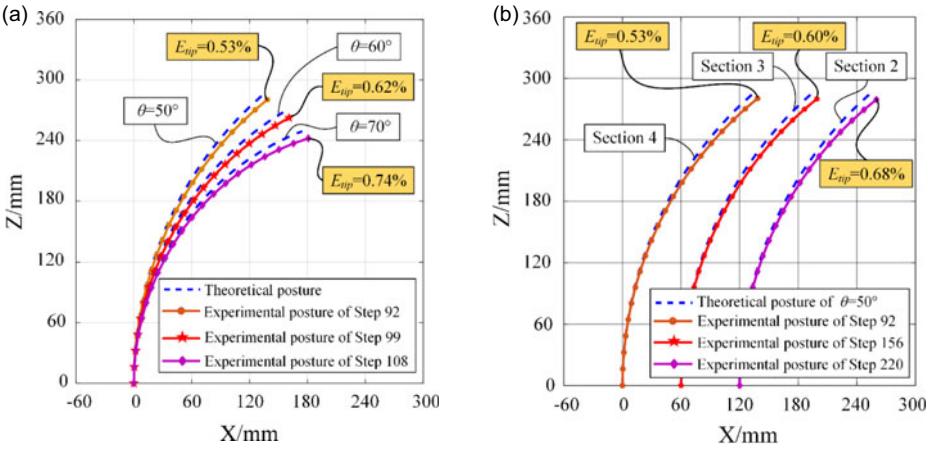
Tip error *Path deviation*

Figure 22. *Error comparison between experiment and simulation at each step.*

at each step length in the experiment are then compared with the simulation results, as shown in Figure 22.

Through the comparison of experimental and simulation errors, it is evident that the actual path deviation slightly exceeds the values obtained in simulation, and exhibits a similar trend. This validates the effectiveness of the path-tracking method and kinematic model. The maximum path deviation

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press



Posture of section 4 at different steps *Posture of different sections at the same posture*

Figure 23. *Analysis of error sources in path-tracking motion.*

observed in the experiment is 30.701 mm. Analysis of the path deviation curve reveals periodic fluctuations, attributed to the target path being composed of several uniformly curved segments. As the robot transitions between different curved segments, significant deviations occur due to variations in path curvature.

Regarding tip error, experimental measurements indicate values notably higher than those obtained in simulation, with a maximum tip error of 22.127 mm. Combining this with the accuracy analysis of the kinematic model from the previous section, the primary sources of robot motion error are as follows: the constant curvature assumption in kinematic modeling represents an idealized scenario, whereas in reality, the drive tendons within the robot do not form continuous curves but rather polygons segmented by segments; the influence of factors such as gravity and friction has not been considered in the motion control process of the robot, leading to a lack of corresponding compensation; elastic deformation of robot materials and manufacturing and assembly errors of components can all contribute to motion errors in the robot. Moreover, the curves of the tip error and path deviation measured from the experiments indicate that as the path length increases, there is an overall trend of increasing motion error in the robot.

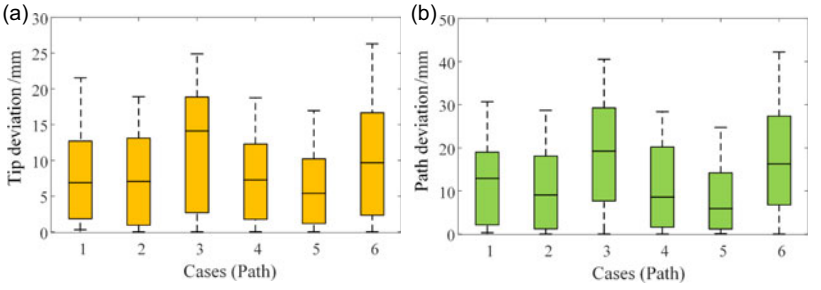
To further validate the above analysis of the error sources, we first extracted the pose measurement data of Section 4 of the continuum robot at different steps from the experiments and analyzed the actual pose of its central axis. The comparison between the actual and theoretical bending postures of Section 4 at steps 92, 99, and 108 is shown in Figure 23(a). The actual bending degree of the robot exceeds the expected posture in the target path, and as the bending angle increases, the tip error gradually increases. When the theoretical bending angle of the robot is 70°, the tip error is 9.864 mm (0.74% relative to the total length of the robot, which is 1333 mm). This indicates that the robot is affected by friction between it and the ground during actual movement, and the thrust from the rear sections causes an excessive bending angle.

Next, we extracted the pose parameters of different sections of the continuum robot from the experimental data as they passed through the same bending path and compared them with the theoretical poses, as shown in Figure 23(b). The tip error of Section 2 reaches 9.064 mm, which is larger than that of Sections 3 and 4. This is because the continuum robot follows a head-following strategy, and the presence of the first two sections causes the tip error to accumulate continuously during path-following movement, making the tip error of Section 2 more pronounced. Additionally, since the continuum robot is composed of serial segments, the deformation errors of each segment accumulate forward, resulting in an overall increase in motion error as the moving distance increases. This is consistent with the trend of the error curve shown in Figure 22.

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

Table III. *Relative error of each target path.*

Error type	Path 1	Path 2	Path 3	Path 4	Path 5	Path 6
Tip error	2.04%	1.75%	2.31%	1.74%	1.57%	2.46%
Path error	2.84%	2.66%	3.75%	2.63%	2.51%	3.91%



Tip error *Path deviation*

Figure 24. *Boxplot of experimental results of target path in different scenes.*

Furthermore, six different experimental scenarios were created by randomly placing obstacles. Each experimental scenario was represented as a grid map, and the continuum-RRT* algorithm was employed to search for paths. After smoothing, six feasible paths for continuum robots were generated. Motion control experiments for path tracking were conducted on these feasible paths within the experimental scenarios. The path error and tip error for the six experiments are determined by computing the ratios of the path deviation E_{path} and tip deviation E_{tip} to the length of the continuum robot arm. The results, including the tip error and path deviation of all target paths, were summarized in box plots in Figure 24. The maximum relative errors for each target path are listed in Table 3.

No outliers were observed in the experimental data, indicating that the robot maintained a consistent posture throughout the experiments. Similar to the simulation results, paths with uneven curvature exhibited more pronounced tip errors and path deviations. The experimental results revealed a maximum path deviation of 3.91%, occurring on path 6, while path 5 exhibited the smallest maximum path deviation at only 2.51%. Regarding tip error, the maximum tip errors ranged from 1.57% to 2.46% across the six target paths.

Comprehensive simulation and experimental results demonstrate that the path-planning method and path-tracking motion control strategy proposed in this paper effectively search feasible paths for the continuum robot and enable it to execute navigation motion along the planned path. Presently, the motion accuracy of the continuum robot is constrained by its kinematic modeling method, as well as factors such as the manufacturing and assembly of its flexible body and the influence of gravity. In future research, potential improvements can be made to the structure of the robot, and new modeling methods or closed-loop control strategies can be introduced to enhance its motion accuracy. It is important to highlight that in the simulations and experiments described above, considering the specific application scenarios addressed in this paper, the simulations were conducted within a two-dimensional environment. Nevertheless, the fundamental principle of this algorithm is to account for the motion constraints of the continuum robot, thus rendering it applicable to three-dimensional scenarios as well. Subsequent efforts will focus on incorporating the mechanical aspects of the robot to provide more accurate predictions of the posture of continuum robots in three-dimensional space. This will facilitate the extension of the path-planning algorithm presented in this paper to three-dimensional scenarios.

5. Conclusion

This paper proposes the continuum-RRT* algorithm for continuum robots, achieved by refining the RRT* algorithm. It capitalizes on the constant curvature kinematics inherent to continuum robots, enabling

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

it to identify paths that align with the specific kinematic attributes of the robots, even in complex and multi-obstacle environments. To enhance the path output from the continuum-RRT* algorithm, B-spline curves are applied for smoothing. Moreover, a four-section continuum robot is constructed, along with corresponding environments for path-tracking experiments, providing validation for the efficacy of the proposed algorithm. Results from these experiments affirm that the four-section continuum robot adeptly accomplishes navigational tasks along the planned paths in multi-obstacle settings. Thus, the paths generated by the continuum-RRT* algorithm aptly satisfy the motion requirements of continuum robots within multi-obstacle environments.

Furthermore, adapting the multi-section continuum robot for three-dimensional scenarios is crucial. Future work will concentrate on developing compensation algorithms for gravity and friction to enhance the accuracy of posture predictions in 3D space. By further refining the calculation processes, we aim to extend the path-planning algorithm proposed in this paper to effectively address three-dimensional applications.

Author contributions. Guohua Gao and Kai Liu originated the design and path planning for a multi-section continuum robot. Prototype preparation, path-planning algorithm, and analysis were performed by Dongjian Li and Chunxu Song under the supervision of Guohua Gao and Kai Liu. Dongjian Li, Yuxin Ge, and Kai Liu wrote the article and all authors read and approved the final manuscript.

Financial support. This work was supported by the National Natural Science Foundation of China (Grant No. 52305003), the Great Scholars Program (Grant No. CIT&TCD20190309), and the China Postdoctoral Science Foundation (Grant No. 2023M730144).

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

Ethical approval. The research does not involve human participants, their data, or biological material, and it does not involve animals.

References

[1] A. D. Luca, G. Oriolo and M. Vendittelli, "Control of Wheeled Mobile Robots: An Experimental Overview," **In:** *Articulated and Mobile Robots for Services and Technology* B. S. Nicosia, A. Bicchi and P. Valigi, eds.) (Springer-Verlag, U.K., 2001) pp. 181–223.

[2] F. Rubio, F. Valero and C. Llopis-Albert, "A review of mobile robots: Concepts, methods, theoretical framework, and applications," *Int J Adv Robot Syst* **16**(2), 1729881419839596 (2019).

[3] N. Ding, C. Peng, M. Lin and C. Wu, "A comprehensive review on automatic mobile robots: Applications, perception, communication and control," *J Circuits Syst Comput* **31**(08), 2250153 (2022).

[4] I. D. Walker, "Continuous backbone "continuum" robot manipulators," *Int Scholar Res Notice* **2023**(1), 726506 (2013).

[5] S. Kim, C. Laschi and B. Trimmer, "Soft robotics: A bioinspired evolution in robotics," *Trends Biotechnol* **31**(5), 287–294 (2013).

[6] J. K. Hopkins, B. W. Spranklin and S. K. Gupta, "A survey of snake-inspired robot designs," *Bioinspir Biomim* **4**(2), 021001 (2009).

[7] C. Yang, S. Geng, I. Walker, D. T. Branson, J. Liu, J. S. Dai and R. Kang, "Geometric constraint-based modeling and analysis of a novel continuum robot with shape memory alloy initiated variable stiffness," *Int J Robot Res* **39**(14), 1620–1634 (2020).

[8] J. Burgner-Kahrs, D. C. Rucker and H. Choset, "Continuum robots for medical applications: A survey," *IEEE Trans Robot* **31**(6), 1261–1280 (2015).

[9] C. Shi, X. Luo, P. Qi, T. Li, S. Song, Z. Najdovski, T. Fukuda and H. Ren, "Shape sensing techniques for continuum robots in minimally invasive surgery: A survey," *IEEE Trans Bio Med Eng* **64**(8), 1665–1678 (2016).

[10] P. E. Dupont, N. Simaan, H. Choset and C. Rucker, "Continuum robots for medical interventions," *Proceed IEEE* **110**(7), 847–870 (2022).

[11] K. Xu, J. Zhao and X. Zheng, "Configuration comparison among kinematically optimized continuum manipulators for robotic surgeries through a single access port," *Robotica* **33**(10), 2025–2044 (2015).

[12] R. Zhang, D. Xie, C. Qian, X. Duan and C. Li, "Design of a flexible robot toward transbronchial lung biopsy," *Robotica* **41**(3), 1055–1065 (2023).

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

[13] L. Angrisani, S. Grazioso, G. D. Gironimo, D. Panariello and A. Tedesco, "On the Use of Soft Continuum Robots for Remote Measurement Tasks in Constrained Environments: A Brief Overview of Applications," **In:** *IEEE International Symposium on Measurements & Networking (M& N)*, (IEEE, 2019) pp. 1–5.
[14] S. Kolachalama and S. Lakshmanan, "Continuum robots for manipulation applications: A survey," *J Robot* **2020**(1), 4187048 (2020).
[15] P. Wang, S. Guo, X. Wang and Y. Wu, "Design and analysis of a novel variable stiffness continuum robot with built-in winding-styled ropes," *IEEE Robot Autom Lett* **7**(3), 6375–6382 (2022).
[16] M. Russo, S. M. H. Sadati, X. Dong, A. Mohammad, I. D. Walker, C. Bergeles, K. Xu and D. A. Axinte, "Continuum robots: An overview," *Adv Intell Syst* **5**(5), 2200367 (2023).
[17] G. Qin, A. Ji, Y. Cheng, W. Zhao, H. Pan, S. Shi and Y. Song, "Design and motion control of an under-actuated snake arm maintainer," *Robotica* **40**(6), 1763–1782 (2022).
[18] S. Cobos-Guzman, D. Palmer and D. Axinte, "Kinematic model to control the end-effector of a continuum robot for multi-axis processing," *Robotica* **35**(1), 224–240 (2017).
[19] N. Sarli and N. Simaan, "Minimal Visual Occlusion Redundancy Resolution of Continuum Robots in Confined Spaces," **In:** *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (IEEE, 2017) pp. 6448–6454.
[20] T. Kato, I. Okumura, S. E. Song, A. J. Golby and N. Hata, "Tendon-driven continuum robot for endoscopic surgery: Preclinical development and validation of a tension propagation model," *IEEE/ASME Trans Mechatron* **20**(5), 2252–2263 (2014).
[21] J. Lai, K. Huang, B. Lu, Q. Zhao and H. K. Chu, "Verticalized-tip trajectory tracking of a 3D-printable soft continuum robot: Enabling surgical blood suction automation," *IEEE/ASME Trans Mechatron* **27**(3), 1545–1556 (2021).
[22] L. Dupourqué, F. Masaki, Y. L. Colson, T. Kato and N. Hata, "Transbronchial biopsy catheter enhanced by a multisection continuum robot with follow-the-leader motion," *Int J Comput Assist Radiol Surg* **14**(11), 2021–2029 (2019).
[23] 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," *Robot Com Int Manuf* **44**, 218–229 (2017).
[24] E. W. Hawkes, L. H. Blumenschein, J. D. Greer and A. M. Okanura, "A soft robot that navigates its environment through growth," *Sci Robot* **2**(8), eaan3028 (2017).
[25] B. T. Phillips, K. P. Becker, S. Kurumaya, K. C. Galloway, G. Whittredge, D. M. Vogt, C. B. Teeple, M. H. Rosen, V. A. Pieribone, D. F. Gruber and R. J. Wood, "A dexterous, glove-based teleoperable low-power soft robotic arm for delicate deep-sea biological exploration," *Sci Rep* **8**(1), 14779 (2018).
[26] A. A. A. Rasheed, A. S. Al-Araji and M. N. Abdullah, "Static and dynamic path planning algorithms design for a wheeled mobile robot based on a hybrid technique," *Int J Intell Eng Syst* **15**(5), 167–181 (2022).
[27] F. Duchoň, A. Babinec, M. Kajan, P. Beňo, M. Florek, T. Fico and L. Jurišica, "Path planning with modified a star algorithm for a mobile robot," *Procedia Engineer* **96**, 59–69 (2014).
[28] M. Elbanhawi and M. Simic, "Sampling-based robot motion planning: A review," *IEEE Access* **2**, 56–77 (2014).
[29] M. G. Park, J. H. Jeon and M. C. Lee, "Obstacle Avoidance for Mobile Robots using Artificial Potential Field Approach with Simulated Annealing," **In:** *IEEE International Symposium on Industrial Electronics Proceedings*, (IEEE, 2001) pp. 1530–1535.
[30] D. Palmer, S. Cobos-Guzman and D. Axinte, "Real-time method for tip following navigation of continuum snake arm robots," *Robot Auton Syst* **62**(10), 1478–1485 (2014).
[31] I. S. Godage, D. T. Branson, E. Guglielmino and D. G. Caldwell, "Path Planning for Multisection Continuum Arms," **In:** *IEEE International Conference on Mechatronics and Automation*, (IEEE, 2012) pp. 1208–1213.
[32] A. Ataka, P. Qi, H. Liu and K. Althoefer, "Real-Time Planner for Multi-Segment Continuum Manipulator in Dynamic Environments," **In:** *IEEE International Conference on Robotics and Automation (ICRA)*, (IEEE, 2016) pp. 4080–4085.
[33] B. Ouyang, Y. Liu, H.-Y. Tam and D. Sun, "Design of an interactive control system for a multisection continuum robot," *IEEE/ASME Trans Mechatron* **23**(5), 2379–2389 (2018).
[34] H. R. Wu, J. J. Yu, J. Pan and P. Xu, "A novel obstacle avoidance heuristic algorithm of continuum robot based on FABRIK," *Sci China Technol Sci* **65**(12), 2952–2966 (2022).
[35] L. G. Torres, A. Kuntz, H. B. Gilbert, P. J. Swaney, R. J. Hendrick, R. J. Webster III and R. Alterovit, "A Motion Planning Approach to Automatic Obstacle Avoidance During Concentric Tube Robot Teleoperation," **In:** *IEEE international conference on robotics and automation (ICRA)*, (IEEE, 2015) pp. 2361–2367.
[36] T. Greigarn, N. L. Poirot, X. Xu and M. C. Cavusoglu, "Jacobian-based task-space motion planning for MRI-actuated continuum robots," *IEEE Robot Autom Lett* **4**(1), 145–152 (2018).
[37] I. A. Seleem, S. F. M. Assal, H. Ishii and H. El-Hussieny, "Demonstration-guided pose planning and tracking for multi-section continuum robots considering robot dynamics," *IEEE Access* **7**, 166690–166703 (2019).
[38] B. H. Meng, I. S. Godage and I. Kanj, "RRT*-based path planning for continuum arms," *IEEE Robot Autom Lett* **7**(3), 6830–6837 (2022).
[39] S. Aggarwal and N. Kumar, "Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges," *Comput Commun* **149**, 270–299 (2020).
[40] Z. Y. Zhao, M. C. Zhou and S. X. Liu, "Iterated greedy algorithms for flow-shop scheduling problems: A tutorial," *IEEE Trans Autom Sci Eng* **19**(3), 1941–1959 (2021).
[41] D. Palmer and D. Axinte, "Active uncoiling and feeding of a continuum arm robot," *Robot Com Int Manuf* **56**, 107–116 (2019).

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press

[42] R. J. Webster III and B. A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *Int J Robot Res* **29**(13), 1661–1683 (2010).
[43] H. Choset and W. Henning, "A follow-the-leader approach to serpentine robot motion planning," *J Aerospace Eng* **12**(2), 65–73 (1999).
[44] M. Elbanhawi, M. Simic and R. N. Jazar, "Continuous path smoothing for car-like robots using B-spline curves," *J Intell Robot Syst* **80**(S1), 23–56 (2015).

Cite this article: G. Gao, D. Li, K. Liu, Y. Ge and C. Song (2024). "A study on path-planning algorithm for a multi-section continuum robot in confined multi-obstacle environments", Robotica **42**, 3324–3347. https://doi.org/10.1017/S0263574724001383

https://doi.org/10.1017/S0263574724001383 Published online by Cambridge University Press