

Article

Obstacle-Avoidance Planning in C-Space for Continuum Manipulator Based on IRRT-Connect

Yexing Lang ¹, Jiaxin Liu ¹, Quan Xiao ^{2,*}, Jianeng Tang ¹, Yuanke Chen ² and Songyi Dian ²

¹ State Grid Electric Power Research Institute of Liaoning Electric Power Co., Ltd., Shenyang 110000, China; bianyushan_ever@163.com (Y.L.); liujxldk@163.com (J.L.); jialuntang@126.com (J.T.)

² College of Electrical Engineering, Sichuan University, Chengdu 610065, China; chenyanke@stu.scu.edu.cn (Y.C.); scudiansy@scu.edu.cn (S.D.)

* Correspondence: xiaoquan@stu.scu.edu.cn

Abstract: Aiming at the challenge of trajectory planning for a continuum manipulator in the confined spaces of gas-insulated switchgear (GIS) chambers during intelligent operation and maintenance of power equipment, this paper proposes a configuration space (C-space) obstacle-avoidance planning method based on an improved RRT-Connect algorithm. By constructing a virtual joint-space obstacle map, the collision-avoidance problem in Cartesian space is transformed into a joint-space path search problem, significantly reducing the computational burden of frequent inverse kinematics solutions inherent in traditional methods. Compared to the RRT-Connect algorithm, improvements in node expansion strategies and greedy optimization mechanisms effectively minimize redundant nodes and enhance path generation efficiency: the number of iterations is reduced by 68% and convergence speed is improved by 35%. Combined with polynomial-driven trajectory planning, the method successfully resolves and smoothens driving cable length variations, achieving efficient and stable control for both the end-effector and arm configuration of a dual-segment continuum manipulator. Simulation and experimental results demonstrate that the proposed algorithm rapidly generates collision-free arm configuration trajectories with high trajectory coincidence in typical GIS chamber scenarios, verifying its comprehensive advantages in real-time performance, safety, and motion smoothness. This work provides theoretical support for the application of continuum manipulator in precision operation and maintenance of power equipment.

Keywords: continuum manipulator; unstructured environments; bidirectional rapidly exploring random tree; coppeliasim; path planning



Academic Editor: Zheng Chen

Received: 9 April 2025

Revised: 8 May 2025

Accepted: 9 May 2025

Published: 13 May 2025

Citation: Lang, Y.; Liu, J.; Xiao, Q.; Tang, J.; Chen, Y.; Dian, S. Obstacle-Avoidance Planning in C-Space for Continuum Manipulator Based on IRRT-Connect. *Sensors* **2025**, *25*, 3081. <https://doi.org/10.3390/s25103081>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the continuous growth of electrical loads and consumer populations, the power industry has entered a phase of rapid development, accompanied by extensive deployment of generation, transmission, transformation, distribution, and consumption infrastructure. Consequently, the operation, maintenance, and inspection of critical power equipment have become pivotal in ensuring grid stability [1,2].

In confined maintenance environments such as gas-insulated switchgear (GIS), traditional rigid manipulators prove ill-suited for compact configurations due to their mechanical constraints. In contrast, cable-driven continuum manipulators demonstrate exceptional suitability for such inspection scenarios by leveraging their slender profile, redundant degrees of freedom, and enhanced adaptability [3].

Similar to rigid manipulators, investigating methods to minimize temporal costs in path planning and motion expenditure remains critically significant for inspection and maintenance tasks involving continuum manipulators. However, while the redundant degrees of freedom inherent to continuum manipulators enhance their maneuverability in confined environments, they concurrently increase the complexity of motion planning problems [4,5]. This necessitates the development of a robust trajectory planning methodology that explicitly accounts for the structural characteristics and kinematic constraints of continuum manipulators.

For conventional rigid manipulators, extensive research exists on path planning solutions employing genetic algorithms [6], ant colony optimization [7], and artificial potential field methods [8]. However, the redundancy characteristics of continuum manipulators have predominantly focused current investigations on imitation learning [9], optimization-based approaches [10], and reinforcement learning techniques [11]. Recent advancements in sampling-based planners have also introduced improved probabilistic roadmap (PRM) variants for complex multi-agent systems. For instance, Ref. [12] proposed a hierarchical PRM framework for multi-UAV path planning in cluttered 3D environments, demonstrating enhanced computational efficiency through obstacle decomposition, while Ref. [13] developed a congestion-aware multi-robot cooperative PRM optimizer to balance coverage and collision risks in dynamic settings. Despite these innovations, attempts to adapt classical PRM [14] or rapidly exploring random tree (RRT) [15] methods to continuum manipulators (e.g., in minimally invasive surgical robots [16,17]) still suffer from excessive redundant nodes that induce body distortion phenomena during deployment.

Most existing methods primarily conduct planning in Cartesian space with weak interaction between joint configurations and the environment. In recent years, some scholars have chosen to map obstacles from Cartesian space to the configuration space (C-space) [18], thereby providing more efficient solutions for tasks requiring arm-shape collision avoidance, eliminating extensive inverse kinematics computations and improving solving efficiency. However, due to the lack of effective mapping between the configuration space of cable-driven continuum manipulator and obstacle maps, planning efficiency cannot meet operational maintenance requirements.

In summary, this paper focuses on cable-driven continuum manipulators and proposes a C-space obstacle-avoidance trajectory planning method suitable for multi-segment continuum manipulators. The method addresses rapid shape collision detection, combines step-by-step optimization and greedy strategies to effectively improve solving efficiency and reduce node count, and further optimizes and smooths the driving trajectory. By employing MATLAB, CoppeliaSim Edu V4.2.0, and physical hardware co-simulation, both simulations and experiments demonstrate the algorithm's rapidity and the accuracy of arm-shape planning.

2. Machine Design

Currently, existing actuation methods for continuum manipulators include individual cable/tendon-driven systems, shape memory alloy actuation, pneumatic artificial muscle actuation, and hybrid actuation based on electroactive reactants and fluids. Due to the narrow and complex internal environment of GIS chambers, the designed actuation mechanism requires miniaturization to enable the manipulator to better perform various tasks. Compared to other solutions, the cable-driven approach utilizes micromotors to control manipulator motion, which meets miniaturization design requirements, thus selecting cable-driven actuation for continuum manipulators. The designed actuation mechanism, as shown in Figure 1, consists of micro DC motors, gear reducers, precision bearings, couplings, ball-bearing slides, linear guide rails, lead screws, drive cables, and fixed terminals.

The actuation mechanism has dimensions of 140 mm × 50 mm × 42 mm and can be directly mounted onto the mobile platform of GIS chamber maintenance robots.

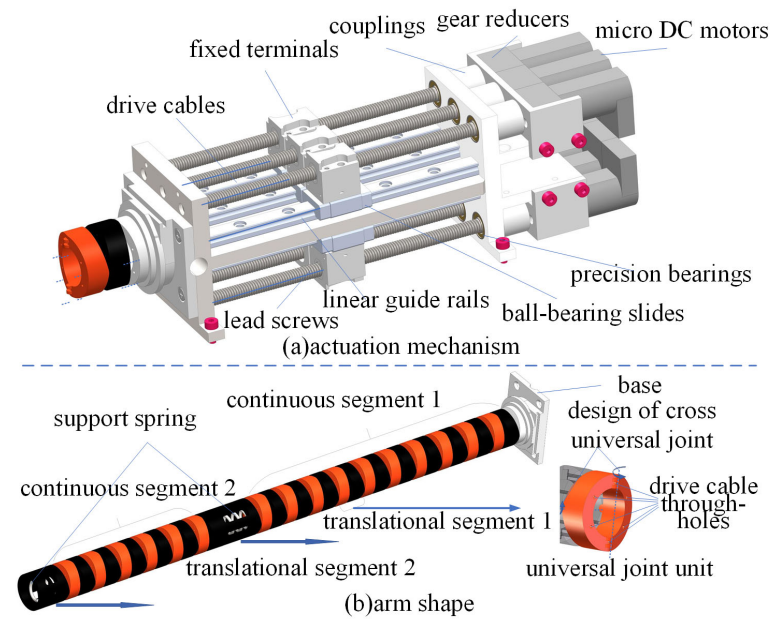


Figure 1. Structural diagram of continuum manipulator.

The continuum body, as the core structure of the manipulator, must be designed to ensure motion in complex environments through bending and rotation. Inspired by the universal joint principle, a continuum manipulator body structure incorporating universal joint units is designed, as shown in Figure 1. This structure features a hollow cylindrical design with an outer diameter of 20 mm, inner diameter of 10 mm, and total length of 304 mm, with an internal spring installed for support. The continuum body comprises two continuous segments and two translational segments. Each continuous segment consists of multiple sets of universal joint units connected in series, enabling bending and rotational motion, while the translational segments are used for mounting actuators or securing drive cables. Drive cable through-holes are reserved on the circumferential cross-section of the continuum body.

3. Problem Description and Kinematics Analysis

3.1. Problem Description

Continuum manipulator motion planning essentially involves generating continuous actuator trajectories from a starting point to a target point within static or dynamic workspaces; these actuator trajectories are typically continuous functions of time.

The entire configuration space C is composed of the free configuration space C_{free} and the obstacle configuration space C_{obs} . Given the initial virtual joint configuration q_{init} ($q_{init} \in C_{free}$) and the target virtual joint configuration q_{goal} ($q_{goal} \in C_{free}$) of continuum manipulator $\mathcal{A}$, the path planning aims to find a configuration sequence Q from q_{init} to q_{goal} that satisfies kinematic collision-free constraints, i.e.,

$$\begin{aligned} Q &= (q_0, q_1, q_2, \dots, q_i), \quad i = 0, 1, 2, \dots, k \\ \text{s.t. } q_0 &= q_{init}, \quad q_k = q_{goal} \\ q_i &\in C_{free} \end{aligned} \quad (1)$$

The point-to-point motion planning problem for continuum manipulators can be described as follows. In complex GIS environments, given the initial configuration q_{init} and

target configuration q_{goal} of the continuum manipulator, a continuous function is obtained mapping $l: [0, t_f] \rightarrow L$ for the drive cable trajectory with respect to time t , such that the joint configuration satisfies $q(L(0)) = q_{init}$, $q(L(t_f)) = q_{goal}$, and for any time $t \in [0, t_f]$, $q(L(t)) \in C_{free}$. The virtual joint configuration $q(L(t))$ at any time t is derived from the two-segment continuum manipulator's forward kinematics relationship FK_1 .

As shown in Figure 2, after obtaining the obstacle-avoidance configuration, the inverse kinematics (IK) from the virtual joint space to drive space is utilized to derive the drive cable variation time sequence $L(t)$, i.e.,

$$L(t) = \{L(q^{t_0}), \dots, L(q^{t_1}), \dots, L(q^{t_f})\} \quad (2)$$

where $L(q^{t_i}) = (\Delta l_1^1(q^{t_i}), \Delta l_1^2(q^{t_i}), \Delta l_1^3(q^{t_i}), \Delta l_2^1(q^{t_i}), \Delta l_2^2(q^{t_i}), \Delta l_2^3(q^{t_i}))$ represents the variation amounts of six driven cables from their original lengths at the corresponding time instances. Here, q^{t_i} denotes the joint configuration q_i reached at time t_i .

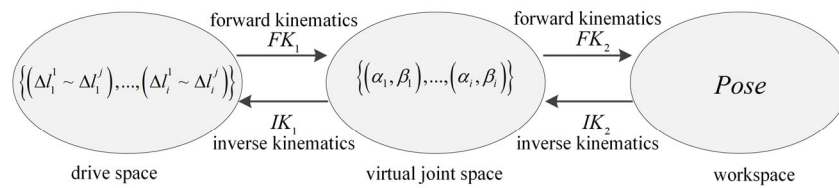


Figure 2. Kinematic mapping relationships of continuum manipulator.

3.2. Kinematics Analysis of Continuum Manipulators

For the cable-driven continuum manipulator, a geometric model based on the PCC (Piecewise Constant Curvature) assumption is established, introducing virtual joint-space variables (α, β) to represent the rotation angle and bending angle of a single segment. Under this framework, the kinematic model of the cable-driven continuum manipulator comprises two parts: the mutual mapping between the drive space (composed of all drive cables) and the artificially introduced virtual joint space, and the mutual mapping between the virtual joint space and the manipulator's end-effector workspace. The forward kinematics relationships FK include the mapping from the drive space to virtual joint space FK_1 and from the virtual joint space to workspace FK_2 . The inverse kinematics relationships IK consist of the mapping from the workspace to virtual joint space IK_2 and from the virtual joint space to drive space IK_1 . As shown in Figure 2, the subscripts i in the first two spaces denote the i -th continuum segment, while the superscript j indicates the number of drive cables required for that segment, which can be 3 or 4.

The geometric model of the two-segment continuum manipulator is shown in Figure 3. According to the designed prototype structure, the manipulator body comprises two continuous segments and two translational segments. The base coordinate system O_0 is established at the center of the base of the first continuous segment. The coordinate system O_1 is established at the center of the end of the first continuous segment. The base coordinate system O_2 of the second continuous segment is established at the end of the translational segment of the first continuous segment. The end coordinate system O_3 of the second continuous segment is established at the end of the second continuous segment, and finally, the workspace coordinate system O_4 of the robot is established at the center of the translational segment at the end of the second continuous segment. Among these, the x-axis positive direction of the base coordinate system O_0 points toward the first drive cable through-hole; the z-axis positive direction is perpendicular to the xy-plane of the base coordinate system and upward, while the y-axis positive direction follows the right-hand rule. The x-y-z-axes of all other coordinate systems align with those of the base coordinate system O_0 . In the diagram, β_1 and β_2 represent the bending angles of the

first and second segments relative to their respective base coordinate system planes, while α_1 and α_2 denote the rotation angles relative to the x-axis of their own base coordinate systems. The two-segment continuum manipulator is driven by three drive cables per segment, resulting in six through-holes arranged at 60-degree intervals counterclockwise from the x-axis in the base coordinate system O_0 . These through-holes drive the motion of the two-segment continuum manipulator, where holes numbered $l_1^1 \sim l_1^3$ correspond to the drive cable through-holes for the first continuous joint, $l_2^1 \sim l_2^3$ correspond to those for the second continuous joint, and r denotes the distance from the base coordinate system center to the drive cable through-holes.

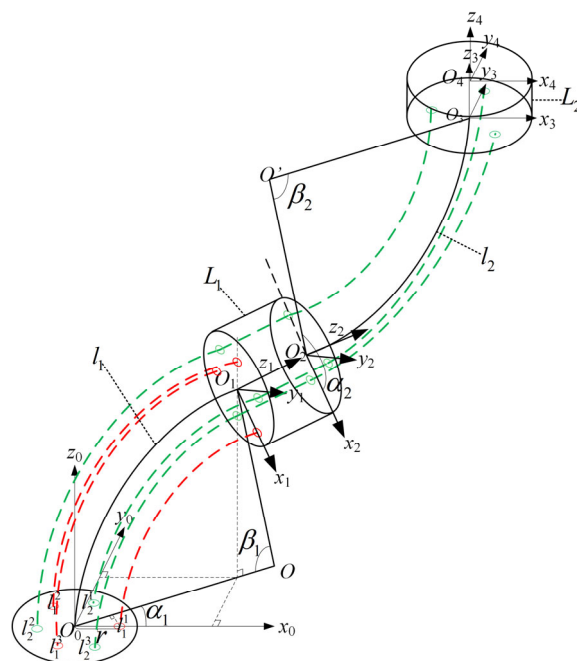


Figure 3. The geometric model of the two-segment continuum manipulator.

4. Inverse Kinematics Solving Based on IRRT-Connect

4.1. Algorithm Theory

Compared to the traditional RRT algorithm, RRT-Connect employs a dual-tree expansion mechanism, initiating from both the start and goal points as root nodes. Each tree extends toward the other's direction through alternating expansion, effectively reducing planning time. Furthermore, the incorporation of heuristic steps significantly improves search speed, particularly demonstrating enhanced performance in navigating narrow passages. The principle is illustrated in Figure 4.

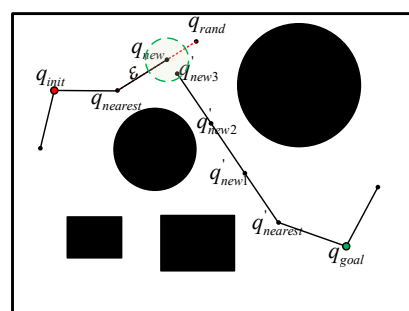


Figure 4. RRT-Connect schematic diagram.

This section proposes an improved algorithm based on RRT-Connect for a two-segment continuum manipulator to address PTP motion planning in GIS complex cavity environments. The path planning operates in the virtual joint configuration space C of the manipulator, where $q = (\alpha_1, \beta_1, \alpha_2, \beta_2)$ represents any configuration in C , and ε denotes the step length during RRT-Connect expansion. To address the computational inefficiencies caused by redundant nodes, the method introduces a maximized step length strategy. This is further enhanced by stepwise optimization to reduce the number of sampling nodes, along with modified pruning and collision detection mechanisms specifically adapted to the kinematic characteristics of the continuum manipulator. These refinements collectively improve memory utilization and algorithmic adaptability in complex cavity environments. The flow of the improved algorithm is shown in Table 1.

Table 1. Algorithm pseudocode.

RRT-Connect Planner for Continuum Manipulator (x_{init}, x_{goal}, Map)	
1	$T_a.init(x_{init}); T_b.init(x_{goal});$
2	for $k = 1$ to K do
3	$x_{rand} \leftarrow SampleNode(Step\ Optimization);$
4	if not($Extend(T_a, x_{rand}) = Trapped$) then
5	if ($Connect(T_b, x_{new}) = Reached$) then
6	Return $PATH(T_a, T_b);$
7	end if
8	end if
9	if $ T_a > T_b $ then
10	SwapTrees(T_a, T_b);
11	end if
12	end for
13	PathPruning()
14	Return Failure

4.2. Step Optimization

When invoking the Connect() function for target tree T_g expanding toward start tree T_i extending only a fixed step length ε may result in multiple sampling points along this direction in T_g . In reality, only the farthest sampling point in that direction is required. This situation creates numerous redundant points in T_g , causing unnecessary memory waste. To resolve this issue, the single collision detection is replaced with multiple collision detections, ensuring that leaf nodes achieve the maximum feasible step length during directional expansion, thereby maximizing the step length.

When determining the new sampling point q_{new} , the single collision detection with step length ε is replaced by four sequential collision detections at 0.5ε , ε , 1.5ε , 2ε . This method effectively reduces the number of leaf nodes in the random expansion tree, thereby mitigating memory consumption.

$$q_{new} = q_{nearest} + \varepsilon \frac{(q_{rand} - q_{nearest})}{\|q_{rand} - q_{nearest}\|} \quad (3)$$

Simultaneously, the specific value of step length ε directly impacts both node quantity and solution efficiency. Therefore, a step-prior selection strategy is incorporated during the four-step greedy expansion: the difference in variation between the initial configuration q_{init} and target configuration q_{goal} is precomputed before algorithm iteration to further

determine the value of ε . The variation difference D_q is defined as the sum of the absolute values of each element of the vector, as follows:

$$D_q = \sum_{i=1}^4 |q_{goal}^i - q_{init}^i| \quad (4)$$

Based on the practical requirements of path planning for a two-segment continuum manipulator in GIS complex cavity environments, the step length ε is coupled with the variation value D_q . The pre-selection scheme for the step length is shown in Equation (5):

$$\varepsilon = \begin{cases} 2 \times \Delta \dots, D_q \geq Threshold_1 \\ 1.6 \times \Delta \dots, Threshold_1 > D_q \geq Threshold_2 \\ \Delta \dots, 0 < D_q < Threshold_2 \end{cases} \quad (5)$$

where ε is the determined step length, Δ is the reference length, and $Threshold_1$ and $Threshold_2$ are the different change thresholds.

4.3. Pruning Strategy

The pruning strategy can be described as follows: Taking the first waypoint (i.e., q_{init}) in the path planning sequence $Q = (q_{init}, q_2, \dots, q_{goal})$ as the fixed point q_{fixed} , we sequentially select a point q in reverse order from $(q_{goal}, \dots, q_2)$ to perform collision detection between q and q_{fixed} . If the actual kinematic arm configuration connecting the two waypoints q and q_{fixed} does not collide with obstacles, these two points can be directly connected. The successfully connected point q is then designated as the new fixed point q'_{fixed} . This process continues iteratively from q_{goal} to the updated q'_{fixed} until the new fixed point can be directly connected to q_{goal} .

Considering the challenges in visualizing pruning strategies in high-dimensional spaces, this paper illustrates the pruning concept using a two-dimensional schematic diagram, as shown in Figure 5.

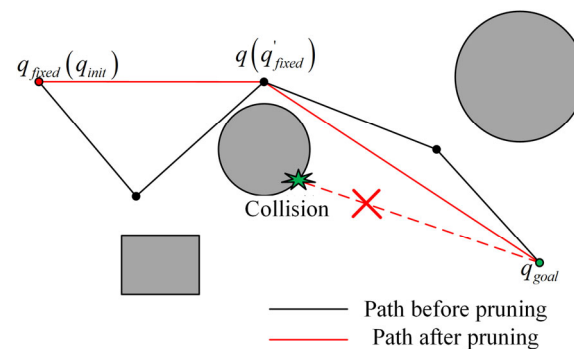


Figure 5. Schematic diagram of pruning strategy.

4.4. Collision Detection

Since the path planning in this study operates within the virtual joint configuration space, the actual arm shape of the continuum manipulator can be derived using the forward kinematics relationship $FK_2(q)$ from the virtual joint space to the workspace. As shown in Figure 6, by equidistantly sampling points along the spine line of the actual arm shape based on bounding sphere positions, the spatial coordinates of all bounding spheres $\{P_1, P_2, P_3, \dots, P_i\}$ are obtained.

The environmental dimension-reduction strategy transfers 3D bounding sphere collision detection to a 2D plane. This strategy implements collision detection between two-segment continuum manipulators and GIS cavity interiors, as specifically shown in Figure 7.

In the figure, R, r represent the GIS metal shell radius and internal conductive column radius, respectively, d_{thr} denotes the dangerous distance threshold, and R_{max}, R_{min} indicate the inner and outer boundary radii of the manipulator's motion safety zone annulus.



Figure 6. Collision model of continuum manipulator.

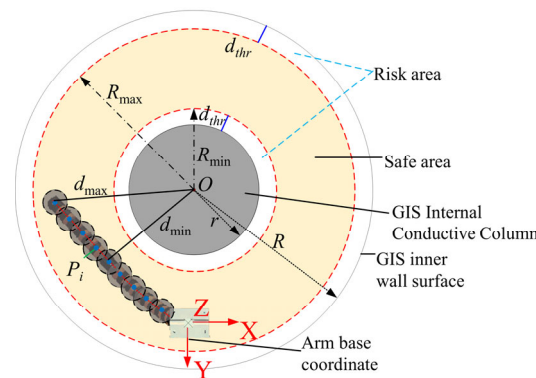


Figure 7. Collision detection based on dimensionality-reduction strategy.

The minimum and maximum distances between the bounding sphere model of the continuous manipulator and the O point of the GIS cavity center are as follows:

$$\begin{cases} d_{\min} = \min\{d_1, d_2, \dots, d_i\} \\ d_{\max} = \max\{d_1, d_2, \dots, d_i\} \end{cases} \quad (6)$$

The distance $d_i = \|O^{xy} - P_i^{xy}\|$ between the bounding spheres and the GIS cavity center must ensure whether both $d_{\min}$ and $d_{\max}$ lie within the annular safety zone formed by $R_{\min}$ to $R_{\max}$.

4.5. Trajectory Planning Based on Driving Space

The IRRT-Connect algorithm obtains point-to-point path planning results for a two-segment continuum manipulator in GIS complex environments. Based on these results, further trajectory planning is required, with the trajectory planning outcomes directly serving as the underlying input for controlling the manipulator's motion.

Driving-line trajectory planning involves two steps: (a) quintic polynomial interpolation within the virtual joint space; (b) deriving driving-line variation trajectories from the virtual joint-space planning results.

The virtual joint angle position change function is expressed as follows:

$$\begin{aligned} q_i(t) &= a_{i0} + a_{i1}t + a_{i2}t^2 + a_{i3}t^3 + a_{i4}t^4 + a_{i5}t^5 \\ \dot{q}_i(t) &= a_{i1} + 2a_{i2}t + 3a_{i3}t^2 + 4a_{i4}t^3 + 5a_{i5}t^4 \\ \ddot{q}_i(t) &= 2a_{i2} + 6a_{i3}t + 12a_{i4}t^2 + 20a_{i5}t^3 \end{aligned} \quad (7)$$

By constraining the virtual joint angular position, velocity, and acceleration corresponding to the starting point at the initial time t_0 and the target point at the end time t_f , the polynomial coefficients can be obtained as follows:

$$\begin{cases} a_{i0} = q_{i0} \\ a_{i1} = \dot{q}_{i0} \\ a_{i2} = \frac{\ddot{q}_{i0}}{2} \\ a_{i3} = \frac{(\ddot{q}_{if} - 3\ddot{q}_{i0})(t_f - t_0)^2 - (8\dot{q}_{if} + 12\dot{q}_{i0})(t_f - t_0) + 20(q_{if} - q_{i0})}{2(t_f - t_0)^3} \\ a_{i4} = \frac{(3\ddot{q}_{i0} - 2\ddot{q}_{if})(t_f - t_0)^2 + (14\dot{q}_{if} + 16\dot{q}_{i0})(t_f - t_0) + 30(q_{if} - q_{i0})}{2(t_f - t_0)^4} \\ a_{i5} = \frac{(\ddot{q}_{if} - \ddot{q}_{i0})(t_f - t_0)^2 - (6\dot{q}_{if} + 6\dot{q}_{i0})(t_f - t_0) + 12(q_{if} - q_{i0})}{2(t_f - t_0)^5} \end{cases} \quad (8)$$

5. Simulation Results and Analysis

The prototype model is set with the parameters of the continuum robot. The length of joint segment 1 is $l_1 = 173$ mm, the length of translation segment 1 is $L_1 = 18$ mm, the length of joint segment 2 is $l_2 = 101$ mm, and the length of translation segment 2 is $L_2 = 12$ mm. The distance r from the center of the base coordinate system to the center of the drive cable through-hole is 7.25 mm. The performance parameters of the computer used in the simulation are as follows: the CPU is Intel® Core™ i7-10700, the memory (RAM) is 16 GB, the graphics card is NVIDIA GTX 1660 SUPER (From ASUS, Suzhou, China), and the MATLAB version is 2018a.

5.1. IRRT-Connect Algorithm Simulation

In the simulation, the data with a large difference between the initial and target configuration changes are selected as example ①, and the data with a small difference are selected as example ②.

The specific data of simulation example ①: Posture from beginning to end $(-195.35, 109.31, 110.14, 1.89, -0.78, -1.57) \rightarrow (-195.35, -109.31, 110.14, -1.89, -0.78, 1.57)$, and corresponding start and end virtual joint variables $(1.57, 2.36, 1.10, 3.93) \rightarrow (1.57, 3.93, 1.10, 2.36)$. The specific data of simulation example ②: Posture from beginning to end $(-39.56, -77.83, 287.64, 0.06, 0.02, 0.33) \rightarrow (-121.57, -156.58, 179.71, -1.23, -0.67, 1.45)$, corresponding start and end virtual joint variables $(0.47, 4.08, 0.31, 0.16) \rightarrow (1.10, 4.32, 0.94, 2.98)$. In the GIS cavity simulation environment, the basic RRT-connect algorithm and the IRRT-connect algorithm are used to carry out 100 planning experiments on two groups of instances, respectively, and the specific data are shown in Table 2. The last experimental planning results are shown in Figure 8. The results obtained from 100 planning experiments are as follows.

Table 2. Comparison of RRT-connect algorithm improvement efficiency.

Example	Algorithm	Average Number of Iterations	Average Number of Leaf Nodes	Average Number of Path Nodes	Average Number of Path Nodes
①	RRT-Connect	150	46	19	0.017 s
	IRRT-Connect	48	12	8	0.011 s
②	RRT-Connect	2	8	8	0.002 s
	IRRT-Connect	5	5	4	0.002 s

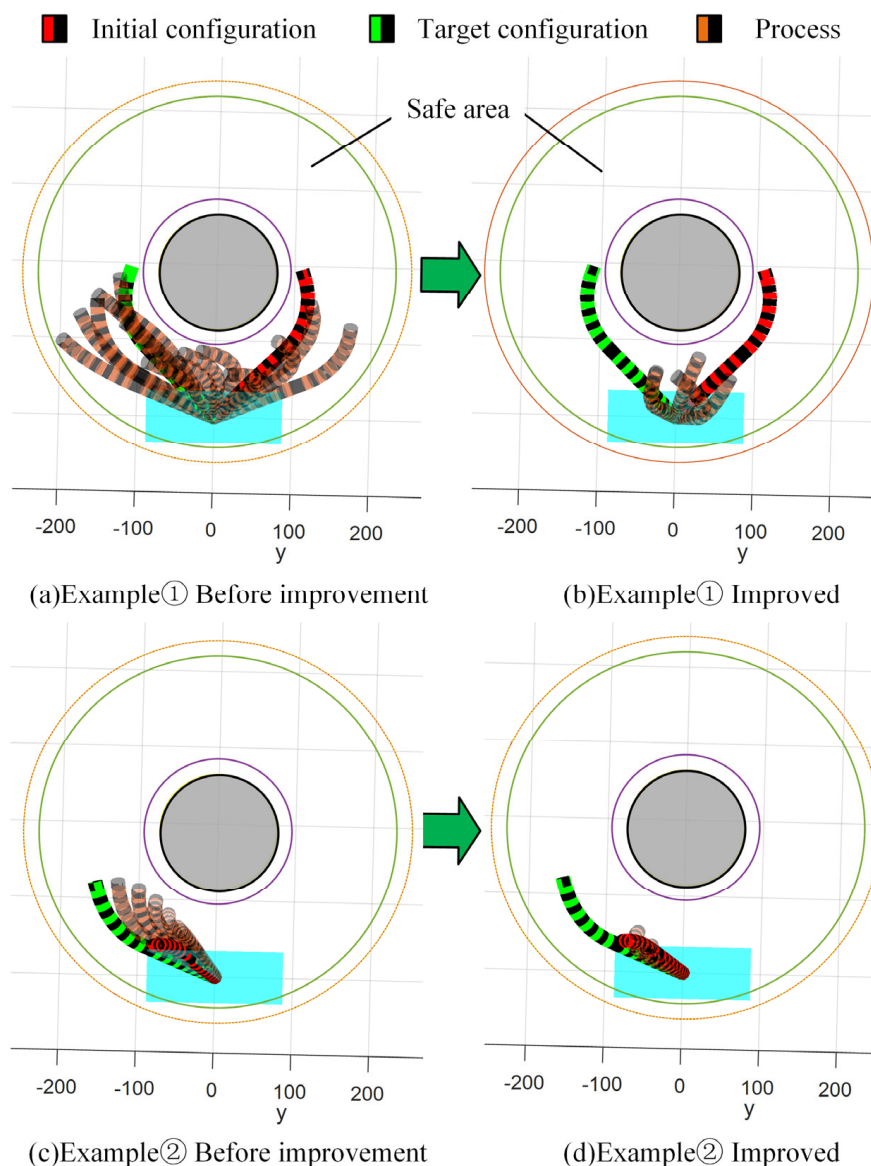


Figure 8. Comparison of planning results before and after RRT-connect algorithm improvement.

5.2. Drive Trajectory Smoothing Test

To validate the practical effectiveness of applying the quintic polynomial trajectory planning method to a two-segment continuum manipulator, a point-to-point motion case is selected for simulation testing, and the example parameters are set as follows. Virtual joint configuration $q_{init} = (0.2, 2, 1.5, 2.5)$ is the starting point of polynomial trajectory planning, $q_{goal} = (1.4, 0.2, 0.15, 1.2)$ is the target point, and the speed constraint $\dot{q}_{init} = 0$, $\dot{q}_{goal} = 0$ and acceleration constraint $\ddot{q}_{init} = 0$, $\ddot{q}_{goal} = 0$ are set at the same time. The time from the starting point to the target point is 5 s, so $t_0 = 0$, $t_f = 5$ s. The results are shown in Figure 9.

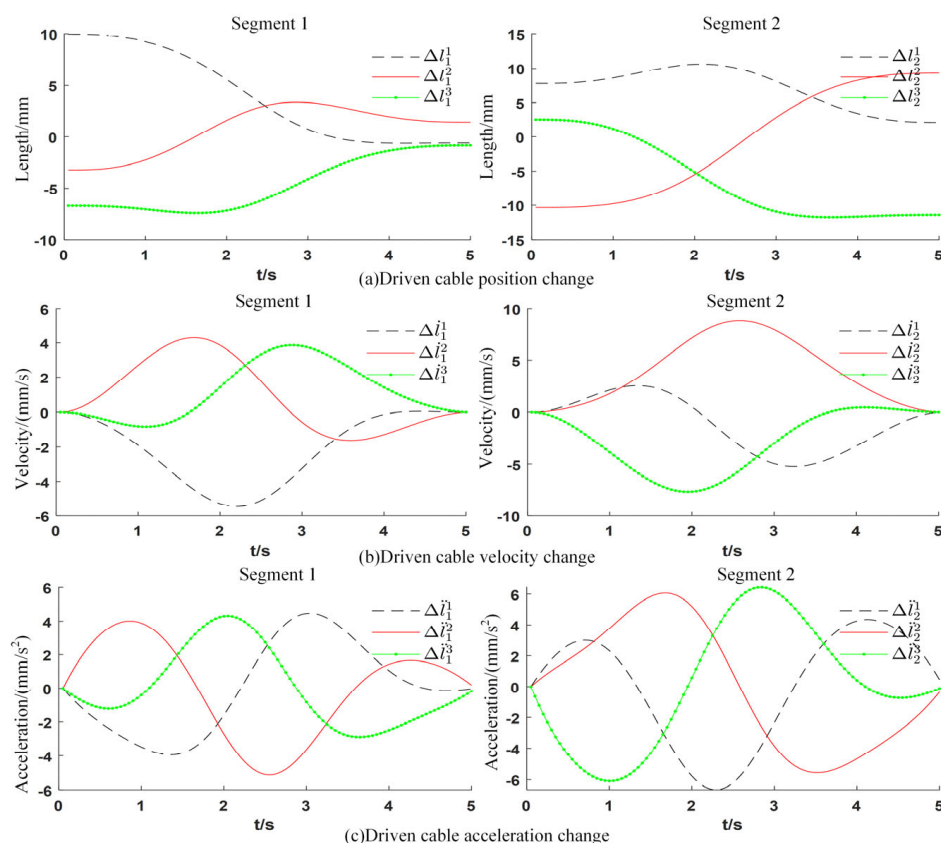


Figure 9. Polynomial trajectory planning.

5.3. Co-Simulation of MATLAB and CoppeliaSim

The motion planning process from initial virtual joint variables (1.57, 2.36, 1.10, 3.93) to target virtual joint variables (1.57, 3.93, 1.10, 2.36) is accomplished using the improved RRT-Connect algorithm and quintic polynomial interpolation. As shown in Figure 10, the planned results (six driving cable sequences Δl_i^j) are transmitted in real time to the CoppeliaSim virtual simulation platform to control the two-segment continuum manipulator's virtual model. To compare virtual simulation motion control effects, a 550 KV GIS cavity and two-segment continuum manipulator numerical simulation model are implemented in MATLAB. During motion control simulations, MATLAB numerical simulations and CoppeliaSim virtual model simulations are synchronized. As shown in Figure 11, the timing comparison between its bitwise numerical and virtual simulation motion control. The end trajectory comparison is shown in Figure 12

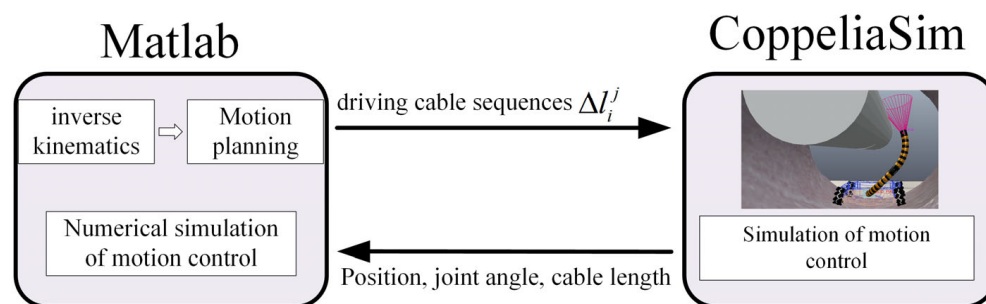


Figure 10. Motion control simulation program.

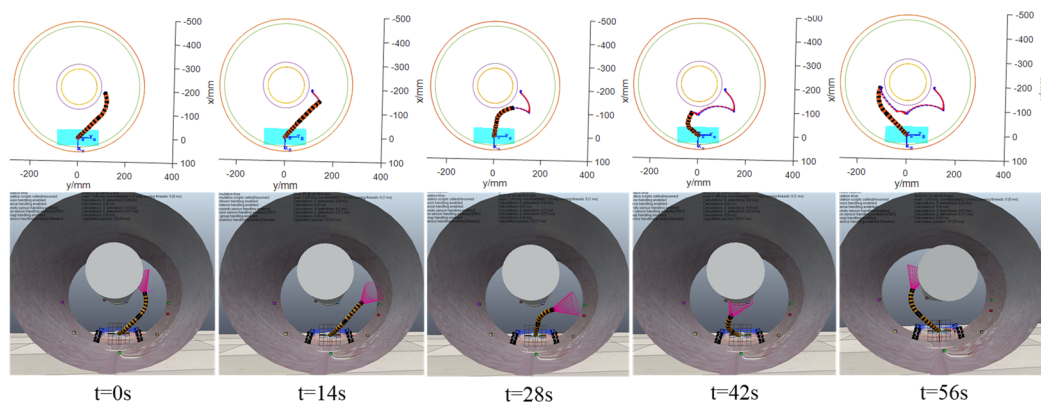


Figure 11. Numerical simulation (**top**) and virtual simulation (**bottom**) motion control timing diagrams.

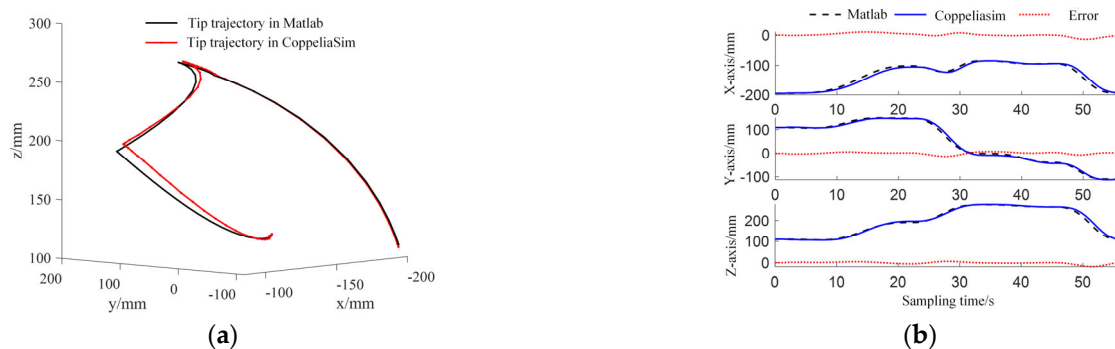


Figure 12. End trajectory comparison. (a) Tip trajectory comparison; (b) comparison of changes in spatial coordinates.

6. Experiment and Result Analysis

6.1. Experiment Preparation and Process

To verify the effectiveness of the proposed obstacle-avoidance planning algorithm, a GIS experimental environment as shown in Figure 13 is built. The experimental environment includes a two-stage continuous manipulator prototype, an electric control system, and a real 500 kV GIS tank. The electric control system uses STM32 (From STMicroelectronics, Shenzhen, China) embedded processor to drive and control six DC motors connected to the driving cables.

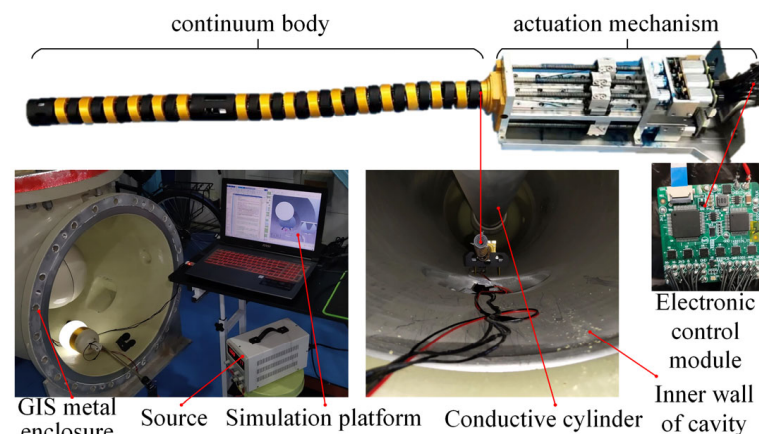


Figure 13. Prototype and experimental environment.

The prototype of the two-stage continuum manipulator is composed of a continuum body, a driving mechanism, and an electronic control module. The continuum body is made of nylon material by 3D printing. The electronic control module uses an STM32 chip as the underlying microcontroller to drive the micro DC motor which can control the expansion and contraction of the drive line. The experimental environment is 550 kV GIS equipment provided by State Grid Corporation.

In the GIS cavity application scenario, the schematic diagram of the point-to-point (PTP) motion control experiment for a two-segment continuum manipulator is shown in Figure 14. The CoppeliaSim platform achieves real-time communication with the underlying microcontroller STM32 (From STMicroelectronics, Shenzhen, China) via serial port. During synchronized simulations between MATLAB and CoppeliaSim, the target driving cable sequence $(\Delta l_1^1, \Delta l_1^2, \Delta l_1^3, \Delta l_2^1, \Delta l_2^2, \Delta l_2^3)$ from CoppeliaSim is transmitted in real time to the STM32 microcontroller through the serial interface, enabling synchronous planning and control among all three systems.

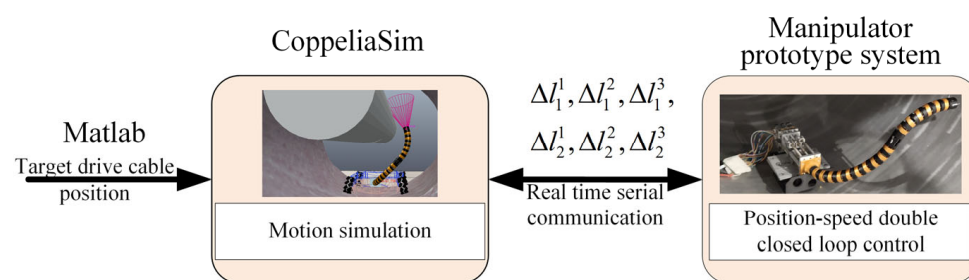


Figure 14. Prototype experimental control block diagram.

The drive commands sent through the serial port reach the underlying microcontroller of the robot with a delay controlled within 35 ms, which is less than the motor execution frequency of 50 ms.

6.2. Analysis of Experimental Results

As shown in Figure 15, the comparative timing diagrams of virtual simulation and physical experiment motion control reveal that the physical continuum manipulator and simulation model exhibit broadly similar motion processes during PTP operations. Due to the narrow and complex internal structure of GIS equipment, existing measurement techniques cannot reliably capture the end-effector trajectory or virtual joint variables during point-to-point motion of the two-segment continuum manipulator. Experimental data acquisition is limited to real-time driving cable positional changes obtained from motor encoder feedback. Comparative analyses between target, simulated, and experimental positional values of the manipulator's six driving cables are shown in Figure 16a,b. The diagrams indicate intermittent positional errors during experimental phases, potentially attributable to mechanical inaccuracies in the continuum body prototype causing motor stalling or inherent encoder errors in DC motors. Despite observed deviations, the results demonstrate the feasibility of deploying two-segment continuum manipulators for GIS cavity inspection and maintenance tasks, confirming that the proposed motion planning methodology achieves effective PTP motion planning and control in complex GIS cavity environments.

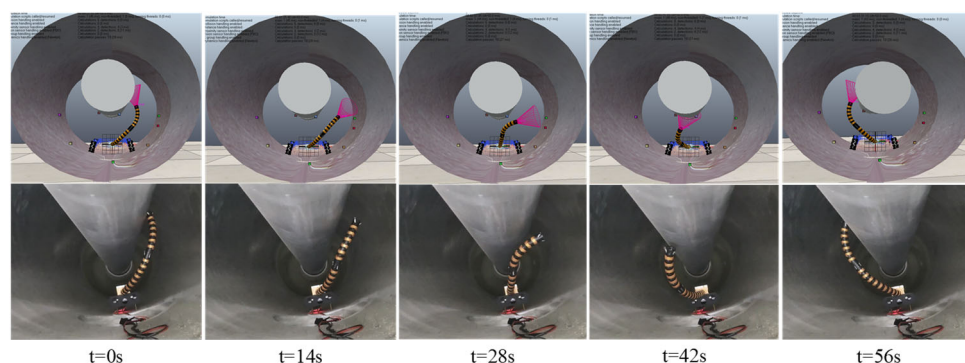


Figure 15. Comparison of motion control between virtual simulation (top) and physical experiments (bottom).

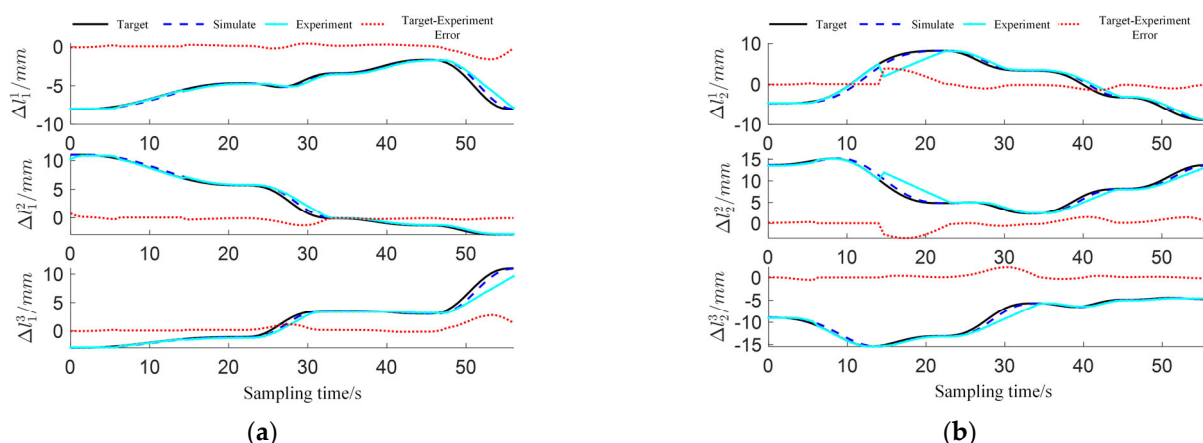


Figure 16. Driven cable comparison: (a) segment 1 driven cable comparison; (b) segment 2 driven cable comparison.

7. Conclusions

To address the collision-avoidance shape planning challenges in key-area fixed-point inspection tasks for GIS equipment maintenance, this study proposes a collision-avoidance planning method for continuum manipulators based on an improved RRT-Connect algorithm. The method transforms Cartesian-space collision avoidance into virtual joint-space obstacle evasion. First, a joint-space obstacle map is established, and the improved RRT-Connect algorithm enhances computational efficiency by reducing node quantity and expansion rate. Furthermore, polynomial-driven spatial trajectory planning calculates smooth driving-line variations to control the dual-segment continuum manipulator's end-effector toward target positions. Simulation and experimental validation on a two-segment continuum manipulator demonstrates the proposed C-space collision-avoidance shape planning method achieves effective and rapid performance, laying the groundwork for real-time manipulator shape planning. For practical implementation, future work will incorporate end-effector motion constraints to extend from point-to-point obstacle-avoidance motion planning to continuous path motion planning, and to achieve wide constraints on end position without significantly increasing the planning time, thus further improving the operational efficiency of the continuum manipulator.

Author Contributions: Conceptualization, Q.X. and Y.C.; methodology, S.D.; software, Q.X.; validation, Y.L., J.L. and J.T.; formal analysis, Y.C.; investigation, J.L.; resources, J.T.; data curation, Y.L.; writing—original draft preparation, Y.L.; writing—review and editing, Q.X.; visualization, Y.L.; supervision, S.D.; project administration, J.L.; funding acquisition, J.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by State Grid Liaoning Electric Power Co., Ltd. Science and Technology Project Grant (Grant number: SGLNDK00SPJS2400108(2024YF-63)).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within this article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: Y.L., J.L. and J.T. are employed by the company State Grid Electric Power Research Institute of Liaoning Electric Power Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

IRRT	Improved rapidly exploring random tree
GIS	gas-insulated switchgear
C-space	configuration space
PRM	probabilistic roadmap

References

1. The National Energy Administration. Releases 2022 National Power Industry Statistics [EB/OL]. 18 January 2023. Available online: https://www.nea.gov.cn/2023-01/18/c_1310691509.htm (accessed on 18 February 2024).
2. IEA. Electricity Market Report 2023 [EB/OL]. 16 February 2023. Available online: <https://www.iea.org/reports/electricity-market-report-2023> (accessed on 18 February 2024).
3. Kolachalama, S.; Lakshmanan, S. Continuum Robots for Manipulation Applications: A Survey. *J. Robot.* **2020**, *2020*, 4187048. [CrossRef]
4. Luo, M.; Li, E.; Zhang, A.; Tan, M.; Liang, Z. A Bioinspired Coiled Cable-Driven Manipulator: Mechatronic Design and Kinematics Planning with Multiconstraints. *IEEE Trans. Mechatron.* **2023**, *28*, 3155–3166. [CrossRef]
5. Ma, C.; Zhao, T.; Xiang, G.; Ren, J.; Chen, Y.; Dian, S. End Positioning Control of Flexible Manipulator Based on Inverse Kinematics. *J. Mech. Eng.* **2021**, *57*, 163–171.
6. Li, J.; Hu, Y.; Yang, S. A Novel Knowledge-Based Genetic Algorithm for Robot Path Planning in Complex Environments. *IEEE Trans. Evol. Comput.* **2025**, *29*, 375–389. [CrossRef]
7. Hu, K.; Dong, Q.; Ma, Z.; Li, B.; Qi, T. Welding robot path planning based on ant colony algorithm. In Proceedings of the 2024 5th International Conference on Artificial Intelligence and Electromechanical Automation (AIEA), Shenzhen, China, 14–16 June 2024; pp. 13–18.
8. Wang, L.; Sun, Z.; Wang, Y.; Wang, J.; Zhao, Z.; Yang, C.; Yan, C. A Pre-Grasping Motion Planning Method Based on Improved Artificial Potential Field for Continuum Robots. *Sensors* **2023**, *23*, 9105. [CrossRef] [PubMed]
9. Mayer, A.; Müller, D.; Raisch, A. Demonstration-based programming of multi-point trajectories for collaborative continuum robots. *IFAC-PapersOnLine* **2019**, *52*, 513–518. [CrossRef]
10. Greigarn, T.; Poirot, N.; Xu, X. Jacobian-based task-space motion planning for MRI-actuated continuum robots. *IEEE Robot. Autom. Lett.* **2018**, *4*, 145–152. [CrossRef] [PubMed]
11. Graule, M.; McCarthy, T.; Teeple, C.; Werfel, J.; Wood, R. SoMoGym: A toolkit for developing and evaluating controllers and reinforcement learning algorithms for soft robots. *IEEE Robot. Autom. Lett.* **2022**, *7*, 4071–4078. [CrossRef]
12. Fu, J.; Sun, G.; Liu, J.; Yao, W.; Wu, L. On hierarchical multi-UAV dubins traveling salesman problem paths in a complex obstacle environment. *IEEE Trans. Cybern.* **2023**, *54*, 123–135. [CrossRef] [PubMed]
13. Fu, J.; Yao, W.; Sun, G.; Ma, Z.; Dong, B.; Ding, J.; Wu, L. Multirobot cooperative path optimization approach for multi-objective coverage in a congestion risk environment. *IEEE Trans. Syst. Man Cybern. Syst.* **2023**, *54*, 1816–1827. [CrossRef]
14. Kavraki, L.; Kolountzakis, M.; Latombe, J. Analysis of probabilistic roadmaps for path planning. *IEEE Trans. Robot. Autom.* **1998**, *14*, 166–171. [CrossRef]
15. Vêras, L.; Medeiros, F.; Guimarães, L. Systematic literature review of sampling process in rapidly-exploring random trees. *IEEE Access* **2019**, *7*, 50933–50953. [CrossRef]

16. Chiluisa, A.; Van Rossum, F.; Gafford, J. Computational optimization of notch spacing for a transnasal ear endoscopy continuum robot. In Proceedings of the 2020 International Symposium on Medical Robotics, Atlanta, GA, USA, 18–20 November 2020; pp. 188–194.
17. Wu, K.; Wu, L.; Ren, H. Motion planning of continuum tubular robots based on centerlines extracted from statistical atlas. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems, Hamburg, Germany, 28 September–2 October 2015; pp. 5512–5517.
18. Qi, R.; Zhou, W.; Wang, T. An Obstacle Avoidance Trajectory Planning Scheme for Space Manipulators Based on Genetic Algorithm. *ROBOT* **2014**, *36*, 263–270.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.