



Research Article

Snake-inspired trajectory planning and control for confined pipeline inspection with hyper-redundant manipulators

Junjie Zhu^a, Mingming Su^a, Longchuan Li^b, Yuxuan Xiang^c, Jianming Wang^d, Xuan Xiao^{a,*}^a School of Computer Science and Technology, Tiangong University, Tianjin 300382, China^b College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China^c School of Advanced Science and Technology, Japan Advanced Institute of Science and Technology, Ishikawa 923-1292, Japan^d Tianjin Key Laboratory of Autonomous Intelligence Technology and Systems, Tianjin 300382, China

ARTICLE INFO

Article history:

Received 12 February 2025

Revised 9 May 2025

Accepted 12 May 2025

Available online 3 June 2025

Keywords:

HRM

NMPC

Trajectory planning

Bionics design

Collision avoidance

ABSTRACT

The hyper-redundant manipulator (HRM) can explore narrow and curved pipelines by leveraging its high flexibility and redundancy. However, planning collision-free motion trajectories for HRMs in confined environments remains a significant challenge. To address this issue, a pipeline inspection approach that combines nonlinear model predictive control (NMPC) with the snake-inspired crawling algorithm (SCA) is proposed. The approach consists of three processes: insertion, inspection, and exit. The insertion and exit processes utilize the SCA, inspired by snake motion, to significantly reduce path planning time. The inspection process employs NMPC to generate collision-free motion. The prototype HRM is developed, and inspection experiments are conducted in various complex pipeline scenarios to validate the effectiveness and feasibility of the proposed method. Experimental results demonstrate that the approach effectively minimizes the computational cost of path planning, offering a practical solution for HRM applications in pipeline inspection.

© 2025 The Author(s). Published by Elsevier B.V. on behalf of Shandong University. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

A hyper-redundant manipulator (HRM) refers to a manipulator with significantly more degrees of freedom (DOF) than required to perform a given task. It possesses exceptional flexibility and adaptability, enabling it to navigate around obstacles and reach into narrow spaces [1]. Therefore, HRMs are widely used in complex environments such as space stations and satellites [2–5]. However, the complex environments and high redundancy of HRMs bring significant challenges to path planning [6,7]. In recent years, quickly identify collision-free motion trajectories within constrained spaces, to guide the HRM end-effector to the target inspection location has become a research focus.

The challenge in HRM path planning lies in guiding the end-effector to the target position while avoiding collisions between the manipulator and the surrounding environment. As redundancy increases, exploration capability is strengthened, but this also significantly increases the time and computational cost of path planning [8,9].

Currently, the main path planning methods include graph-based search algorithms and sampling-based approaches. Among

the graph search algorithms, the A* algorithm [10–12] and Dijkstra's algorithm [9] were commonly used. Tang et al. [13] enhanced the A* algorithm with the artificial potential field method. Their approach reduced search nodes, improved efficiency, and optimized the manipulator's obstacle-avoidance posture. Luo et al. [14] extended Dijkstra's algorithm with Delaunay triangulation and planar transformation to better handle complex surfaces, improving path accuracy for both single-robot and multi-robot tasks.

Sampling-based methods are exemplified by rapidly-exploring random trees (RRT) [15]. Due to environmental constraints, several RRT variants have been developed, with RRT* being the most representative. Shen et al. [16] proposed an optimal RRT* approach that introduced two constraints – path length and operational metrics – and employed constrained closed-loop inverse kinematics to optimize obstacle-avoidance paths for industrial robots. Ji et al. [17] proposed the E-RRT* algorithm, which replaced straight-line node connections with elliptical structures. This modification, together with an optimized sampling process, effectively addressed path-planning challenges for hyper-redundant manipulators in narrow spaces. Jia et al. [18] proposed three improved algorithms that considered the maximum joint deflection: MDA-RRT, MDA-RRT*, and MDA-QRRT. They demonstrated these methods' feasibility and optimization performance compared to traditional approaches. These algorithms

* Corresponding author.

E-mail address: xxiao@tiangong.edu.cn (X. Xiao).

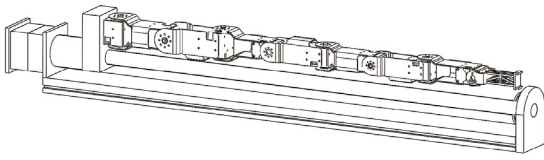


Fig. 1. The mechanical structure design of HRM and sliding rail.

typically require searching through numerous collision-free configurations, leading to high computational costs for finding an optimal solution. However, in space environments, computational resources, energy, and planning time are strictly limited. Therefore, finding a feasible solution that can be quickly planned is a preferable approach.

The slender and flexible body of a snake enables it to effectively explore confined spaces. Inspired by this characteristic, some studies developed modular snake-like structures to construct HRMs [19]. For instance, Qin et al. [20] designed a layered drive system and a composite spool mechanism to achieve synchronized angular motion, simulating the modular structure and locomotion of a snake. Tang et al. [21] utilized a hybrid manipulator structure composed of 12 parallel segments, each including three spherical-prismatic-spherical (SPS) chains and a universal joint chain (3-SPS-U), to replicate serpentine motion. Furthermore, snake movement exhibits a head-following characteristic, where the motion of the entire body depends on the trajectory of the snake's head. This mechanism can be applied to the motion planning of snake-like HRMs. Thus, by planning only the end-effector's target configuration – analogous to the snake's head trajectory – the remaining body joints can automatically follow that path.

Based on this algorithm, we design a pipeline inspection approach tailored for HRMs, which includes three processes: insertion, inspection, and exit. Both the insertion and exit processes utilize the SCA, significantly reducing the planning time for these two phases. The inspection process is controlled using a NMPC approach, which governs the HRM's motion within the pipeline. The contributions are summarized as follows:

- (1) A pipeline inspection approach is designed that can rapidly plan collision-free insertion and exit trajectories for the HRM in curved pipes. This approach generates a feasible motion plan instead of an optimal one. It reduces the number of collision-free configurations that need to be searched, significantly decreasing the planning time.
- (2) A prototype of the HRM is developed, and inspection experiments are conducted in multiple complex narrow pipes, validating the efficiency and feasibility of the algorithm.

The remainder of this article is organized as follows. Section 2 introduces the kinematic modeling of the HRM and discusses the path planning problem. Section 3 provides a detailed explanation of the pipeline control approach, including the insertion, inspection, and exit algorithms. Section 4 focuses on the design and composition of the pipeline inspection system. Section 5 presents the simulation results and experimentally validates the feasibility of the pipeline inspection approach. Section 6 concludes the study and outlines future research directions.

2. Problem formulation

2.1. Modeling

To validate the feasibility of the proposed approach, we design a HRM with 9-DOFs. The system includes a movable base

Table 1
D-H parameters of the snake manipulator.

Link i	θ_i ($^\circ$)	α_i ($^\circ$)	a_i (m)	d_i (m)
$i = 1, 3, 5, 7$	θ_i	-90°	0.1	0
$i = 2, 4, 6, 8$	θ_i	90°	0.1	0

mounted on a sliding rail, eight joints, and interconnecting bodies. The base can move both forward and backward along the rail. The rotational axes of adjacent joints are orthogonal. The specific structure is shown in Fig. 1.

To describe the motion characteristics of the manipulator, its kinematic model is established using Denavit–Hartenberg (DH) parameters and forward kinematics. The DH parameters are shown in Table 1.

Subsequently, the kinematic model is employed to establish the coordinate systems for each link, and the homogeneous transformation matrix between adjacent links is derived. Specifically, ${}^{i-1}_i T$ ($i = 1, 2 \dots n$) represents the homogeneous transformation matrices as shown in Eq. (1).

$${}^{i-1}_i T = \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Where s represents the sine function (sin), and c represents the cosine function (cos).

${}^0_w T$ denotes the transformation matrix from the world coordinate system to the base. ${}^n_w T$ represents the homogeneous transformation matrix between the world coordinate system and the end effector. Then, we can get the forward kinematics formula as follows.

$$\text{fkine}(\mathbf{q}) = {}^n_w T = {}^0_w T {}_0^1 T \dots {}_{n-1}^n T \quad (2)$$

Where $\mathbf{q} = [\theta_1, \theta_2, \dots, \theta_n]$ is the configuration vector of HRM.

2.2. Self-motion

Self-motion enables the end-effector to maintain its position and orientation while other robot links adjust their positions and orientations via redundant joints [22]. The derivation of self-motion is as follows.

The i th Jacobian column \mathbf{j}_i is obtainable through the following vector product calculation.

$$\mathbf{j}_i = \begin{bmatrix} \mathbf{z}_i \times {}^i p_E^0 \\ \mathbf{z}_i \end{bmatrix} \quad (3)$$

Where \mathbf{z}_i is the i th joint axis vector and ${}^i p_E^0$ is the vector from the end-effector to the i th joint. Then the Jacobian matrix can be expressed as:

$$\mathbf{J} = [\mathbf{j}_1 \quad \mathbf{j}_2 \quad \dots \quad \mathbf{j}_n] \quad (4)$$

The core of self-motion control is the Jacobian matrix \mathbf{J} , which relates end-effector velocity vector $\dot{\mathbf{x}}$ to joint velocity vector $\dot{\mathbf{q}}$.

$$\dot{\mathbf{x}} = \mathbf{J}\dot{\mathbf{q}} \quad (5)$$

Due to the redundancy of HRM, the dimension of $\dot{\mathbf{q}}$ is typically higher than that of $\dot{\mathbf{x}}$. To track a desired $\dot{\mathbf{x}}$, we solve for $\dot{\mathbf{q}}$ in the underdetermined system, which admits infinitely many solutions.

To solve this problem, $\dot{\mathbf{q}}$ is decomposed into the minimum norm solution $\dot{\mathbf{q}}_S$ and the general solution of homogeneous equations $\dot{\mathbf{q}}_N$.

$$\begin{aligned} \dot{\mathbf{q}} &= \dot{\mathbf{q}}_S + \dot{\mathbf{q}}_N \\ &= \mathbf{J}^\dagger \dot{\mathbf{x}} + (\mathbf{I} - \mathbf{J}^\dagger \mathbf{J})\mathbf{M} \end{aligned} \quad (6)$$

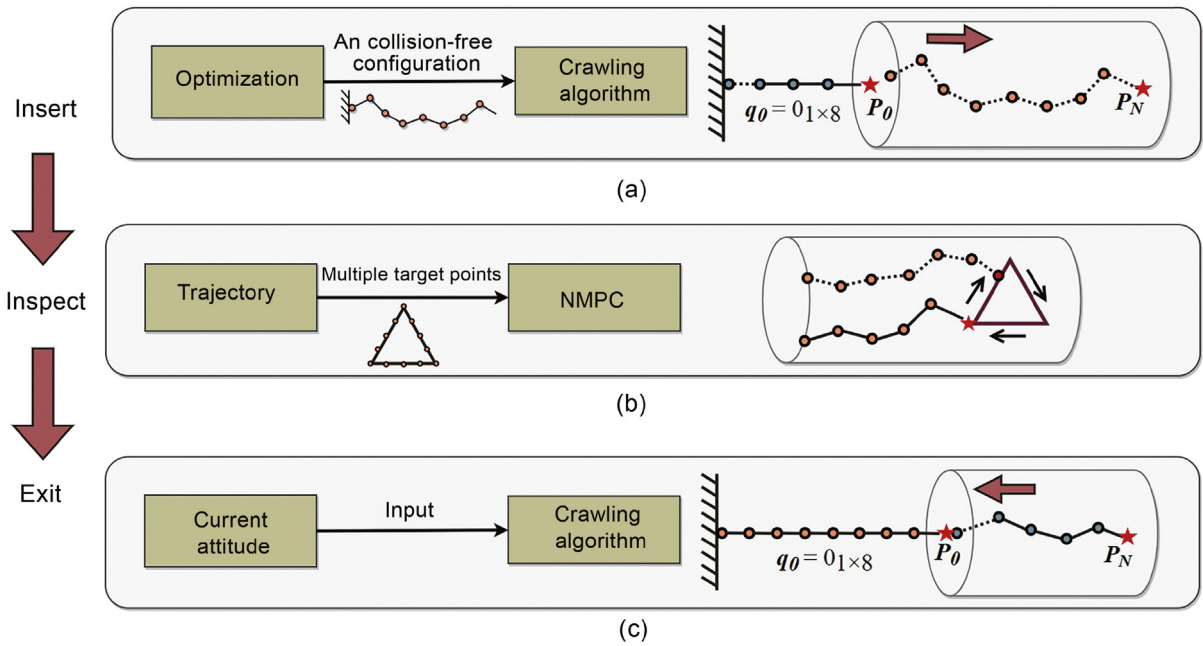


Fig. 2. Flow chart of the pipeline inspection process: q_0 (initial horizontal configuration), P_0 (initial position), P_N (target position).

Where J^\dagger is the pseudo inverse of the Jacobi matrix. $J^\dagger \dot{x}$ is the minimum-norm particular solution (primary task). $(I - J^\dagger J)M$ generates null-space motions for secondary tasks (e.g., obstacle avoidance), with $M = k\nabla H(\mathbf{q})$ (Eq. (7)).

The null space of the Jacobian matrix enables self-motion–reconfiguring joints without affecting the end-effector. This is exploited by the gradient projection method [21] to optimize M .

$$M = k\nabla H(\mathbf{q}) = k \begin{bmatrix} \frac{\partial H(\mathbf{q})}{\partial \theta_1} & \dots & \frac{\partial H(\mathbf{q})}{\partial \theta_n} \end{bmatrix}^T \quad (7)$$

Where k is a scale factor. H is the gradient optimization function, and ∇H is the gradient of H .

Then Eq. (6) can be derived as follows:

$$\dot{\mathbf{q}} = J^\dagger \dot{x} + k(I - J^\dagger J)\nabla H \quad (8)$$

A desired pose vector is achieved by iteratively updating the configuration vector \mathbf{q} using $\dot{\mathbf{q}}$. The following equation describes the update in a small time step Δt .

$$\mathbf{q}(t + \Delta t) = \mathbf{q}(t) + \dot{\mathbf{q}}\Delta t \quad (9)$$

Where $\mathbf{q}(t)$ represents the joint position at time t , and $\mathbf{q}(t + \Delta t)$ represents the updated joint position at time $t + \Delta t$.

Repeatedly conducting the iteration process with Δt will enable a gradual progression towards the optimal configuration.

2.3. Problem formulation

Due to its high redundancy, the HRM is capable of performing flexible and efficient motion in confined spaces. However, this characteristic also increases the complexity of control. Collision-free path planning is crucial to ensuring the safe operation of the HRM during pipeline inspection. Collision-free path planning between the initial and target positions presents the following challenges.

- (1) Kinematic constraints: Precise joint control is required to avoid collisions in confined spaces.
- (2) Complexity of path planning: As the HRM has redundant degrees of freedom, enabling multiple configurations to reach the same position, finding the optimal solution requires significant computational power and time.

To tackle these challenges, we take inspiration from the greedy algorithm, opting for suboptimal rather than optimal solutions. By reducing the number of samples and optimization iterations during the path planning process, the strategy significantly shortens planning time, lowers computational cost, and achieves obstacle avoidance.

3. Control approach

This section outlines the pipeline inspection approach, which consists of three primary stages: pipeline insertion, inspection, and exit. The overall workflow is shown in Fig. 2.

Fig. 2(a) shows the insertion process. In this stage, a method combining NMPC with the SCA is employed, allowing the HRM to insert into the pipeline and reach the target point. The NMPC method finds a collision-free configuration that reaches the target position. Subsequently, the SCA directs the HRM along this configuration within the pipeline, ensuring smooth progression to the target point. This method completes the insertion task by computing only one collision-free configuration, significantly reducing computational overhead.

Fig. 2(b) shows the inspection process, which conducts after the insertion process is complete. NMPC optimization strategies are employed for trajectory tracking to ensure the end-effector of the HRM adheres to the inspection path.

Fig. 2(c) shows the exit process. During the exit process, the HRM utilizes the SCA to slowly exit the pipeline along its current collision-free configuration without re-optimization. This process is essentially the reverse process of the insertion operation.

3.1. Insertion algorithm

3.1.1. Snake-inspired crawling algorithm

Inspired by the serpentine locomotion, we propose the SCA, which allows the HRM to rapidly navigate through pipelines along a collision-free configuration. During insertion, the HRM advances along the sliding rail into the pipeline. The head joints of the HRM are controlled to move along the optimized collision-free configuration, while the body joints follow the trajectory of the preceding joint group.

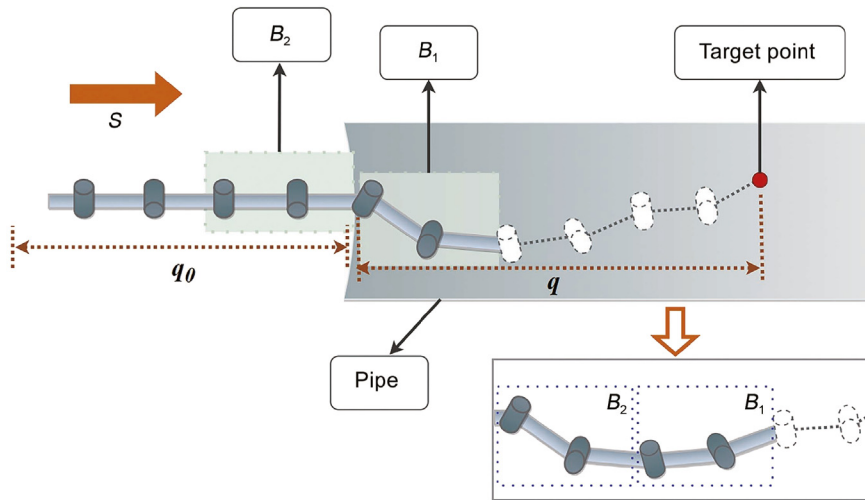


Fig. 3. The specific process diagram of SCA.

To clearly describe the movement process of the HRM, we introduce the following definitions.

- Loop-body: each loop-body B_i comprises two adjacent joints, where $i \in \{1, 2, \dots, \lceil n/2 \rceil\}$. Here, B_1 denotes the end-effector's loop-body and $B_{\lceil n/2 \rceil}$ the base-proximal one.
- S : the length of a loop-body.

During insertion, the SCA operates in loop-body units. As the sliding rail advances, the HRM gradually transitions from the horizontal configuration \mathbf{q}_0 to an optimized collision-free configuration \mathbf{q} .

$$\begin{aligned} \mathbf{q}_0 &= \mathbf{0}_{1 \times n} \\ \mathbf{q} &= [\theta_1, \theta_2, \dots, \theta_n] \end{aligned} \quad (10)$$

The SCA controls the HRM to move along the optimized collision-free configuration, as illustrated in Fig. 3. Initially, the HRM is in configuration \mathbf{q}_0 . It then advances along the sliding rail by a distance S , inserting loop-body B_1 into the pipeline. During this process, loop-body B_1 is precisely adjusted to the corresponding joint angles of the collision-free configuration. Subsequently, the sliding rail advances a distance S , inserting loop-body B_2 into the pipeline and precisely aligning its joint angles with those of B_1 , while B_1 continues to move deeper into the pipeline according to the collision-free configuration. Repeat this process, and the sliding rail continuously supplies new loop-body into the pipeline. The insertion process is completed when the HRM's configuration reaches the target collision-free configuration \mathbf{q} and the end-effector arrives at the target point.

3.1.2. Collision-free configuration planning

The NMPC method is employed to determine the optimal collision-free configuration of the HRM for the SCA. In the problem modeling, the collision constraints are simplified by ensuring that no joint collides with the pipe wall. We use the forward kinematics to calculate the position of each joint \mathbf{p}_i , and further calculate the distance D_{pipe}^i from each joint to the pipeline center based on the pipeline geometry, as follows.

$$\begin{aligned} \mathbf{p}_i &= \text{fkine}(\mathbf{q}) [0 \ 0 \ 0 \ 1]^T \\ D_{\text{pipe}}^i &= \min_{(\mathbf{x}, \mathbf{y}, \mathbf{z}) \in \text{pipe}} \|\mathbf{p}_i - (\mathbf{x}, \mathbf{y}, \mathbf{z})\|_2 \end{aligned} \quad (11)$$

Where \mathbf{p}_0 denotes the position of the base, and the homogeneous coordinate redundancy term of \mathbf{p}_0 is ignored. $(\mathbf{x}, \mathbf{y}, \mathbf{z})$ represents points on the pipeline centerline.

The HRM is ensured to maintain a safe distance D_{safe} from the pipeline. The configuration planning of the HRM aims to determine the configuration with the minimum sum of the absolute joint angles, ensuring smooth motion and avoiding abrupt changes in movement. The inputs include the desired pose vector \mathbf{x}_D of the HRM's end-effector, the safety distance D_{safe} , the pipeline shape and the pipeline radius R . The output is the optimized configuration $\mathbf{q} = [\theta_1, \theta_2, \dots, \theta_n]$. We also consider the rotational angle limits of the HRM's joints. The final path planning method can be expressed as.

$$\begin{aligned} &\text{minimize} \sum_{i=1}^n |\theta_i| \\ &\text{subject to} \begin{cases} \|\mathbf{x}_D - \text{pose}(\text{fkine}(\mathbf{q}))\|_2 = 0 \\ D_{\text{pipe}}^i \leq R - D_{\text{safe}}, i \in [1, n] \\ \theta_i \in [\theta_{lb}, \theta_{ub}], i \in [1, n] \end{cases} \end{aligned} \quad (12)$$

Where the pose function denotes extracting the actual pose vector of the end-effector from the homogeneous transformation matrix returned by the fkine function. θ_{lb}, θ_{ub} denote the upper and lower limits of the joint angles, respectively, and D_{safe} indicates the minimum safe distance between the joint and the pipeline wall.

3.2. Inspection section

After insertion, the inspection process is performed by moving the HRM's end-effector along a trajectory to scan the pipeline's inner wall. The trajectory is discretized into points $M_i (i = 1, \dots, m)$, and the NMPC method computes the collision-free configuration $\mathbf{q}_i \in \mathbb{R}^n$ for each trajectory segment. Here, θ_{ij} denotes the j th joint angle of the i th configuration ($j = 1, \dots, n$).

Compared with Eq. (12), the objective function of collision-free configuration planning in inspection has been changed. The optimization objective is changed to minimize the sum of absolute values of all joint angles and their changes between adjacent configurations. By restricting the variation of the joint angles, it is feasible to prevent violent movements of the HRM and local extreme values near the joint limits. The configuration planning

Table 2
Algorithm performance comparison.

Algorithm	Pipe type	Time-consuming (s)	End-effector error (m)	Minimum distance (m)
RRT	Pipe 1	109.5849	1.0259e-7	0.0841
	Pipe 2	125.6237	1.8367e-7	0.0793
Optimize	Pipe 1	0.4461	1.5474e-7	0.0543
	Pipe 2	0.8863	4.9493e-7	0.0643

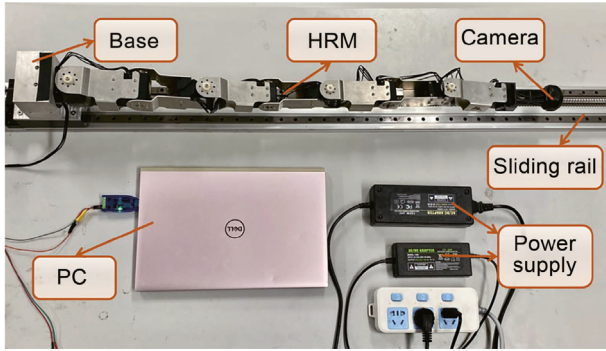


Fig. 4. Experimental platform and equipment display.

method for the i th configuration is as follows.

$$\begin{aligned}
 & \text{minimize } \sum_{j=1}^n |\theta_{ij}| + \sum_{j=1}^n |\theta_{ij} - \theta_{i-1j}| \\
 & \text{subject to } \begin{cases} \|\mathbf{x}_i - \text{pose}(\text{fkine}(\mathbf{q}_i))\|_2 = 0 \\ D_{\text{pipe}}^j \leq R - D_{\text{safe}} \\ \theta_{ij} \in [\theta_{lb}, \theta_{ub}] \\ \Delta\theta_{lb} \leq |\theta_{ij} - \theta_{i-1j}| \leq \Delta\theta_{ub} \end{cases} \quad (13)
 \end{aligned}$$

Where \mathbf{x}_i is the desired pose vector of the i th target point. $\Delta\theta_{lb}$, $\Delta\theta_{ub}$ represent the upper and lower limits of the joint angle variation range. The other parameters follow the same definitions as in Eq. (12). In Eq. (13), the fourth constraint ensures that adjacent configurations meet the configuration similarity criterion. Specifically, the corresponding joint angles between adjacent configurations must lie within a predefined range to reduce oscillations during transitions.

After optimizing and obtaining multiple collision-free configurations, the HRM is controlled to follow the target configurations sequentially, reaching each target point to complete the end-effector inspection task.

3.3. Exit algorithm

After completing the inspection, the system executes the pipeline exit task. The exit process reverses the insertion process and also uses the SCA. The HRM exits the pipeline based on its current collision-free configuration and returns to the initial configuration, in preparation for the next insertion.

4. Experimental platform implementation

To validate the feasibility of the HRM pipeline inspection approach, we developed an experimental pipeline platform. The platform consisted of HRM, sliding rail, detection camera fixed at the end of HRM, and PC-based control system, as shown in Fig. 4.

The HRM adopted a “3 +5” hierarchical design: three XH540-W270-R high-torque motors for the base section, connected via magnesium-aluminum alloy joints, and five XH430-W270-R lightweight motors for the end section, with resin connectors. This hierarchical design ensured a balance between mechanical strength and mobility.

The sliding rail mechanism utilized a ball screw drive system, primarily comprising a stepper motor, a lead screw, and a mounting base for the HRM. This integration significantly expanded the robot’s flexible working space.

The PC control system sent control instructions to the HRM via the TTL protocol and to the sliding rail via the RS485 protocol. Additionally, a detection camera was mounted on the HRM’s end-effector to detect collisions and record pipeline exploration throughout the inspection.

5. Experimental verification

The experiments were conducted to verify the feasibility of the pipeline inspection approach. Firstly, two different collision-free trajectory planning algorithms were compared to evaluate the efficiency of the NMPC algorithm. Secondly, we carried out insertion and exit experiments using two pipelines of different shapes and designed two inspection trajectories to verify that the algorithm can achieve collision-free performance for various pipe shapes.

5.1. Algorithm efficiency test

To evaluate the efficiency of the optimization approach proposed in this article, we utilized the NMPC and RRT algorithms for collision-free configuration optimization of the HRM. In the experiment, both algorithms were tested under identical target poses and obstacle environments. Two different pipes were used in the experiment, as shown in Fig. 6.

MATLAB was used to simulate the process of the HRM inserting into the pipeline, and the simulation results were shown in Fig. 5. Specific results were shown in Table 2, where the minimum distance referred to the shortest distance between the HRM and the pipe wall.

The experimental results indicated that under identical target poses and obstacle environments, the proposed algorithm reduced the computation time compared to the RRT while achieving comparable end-effector accuracy. However, the collision-free configurations obtained through RRT algorithm exhibited better control over the spacing between the HRM body and pipe wall, ensuring safer HRM movement. In summary, although the proposed algorithm improved computational efficiency, this improvement came with a slight trade-off in path planning quality.

5.2. Insert and exit experiment

This section presented insertion and exit experiments in two structurally distinct pipelines, validating the feasibility of the proposed algorithm.

Firstly, we determined the position of the target point (the blue sphere in Fig. 7). The sequential quadratic programming approach was used to find the optimal configuration and ensure it was collision-free for the HRM.

Next, the SCA was employed to move the HRM along this configuration towards the target point. Finally, the HRM exited the pipeline in the current configuration and returned to the horizontal insertion configuration.

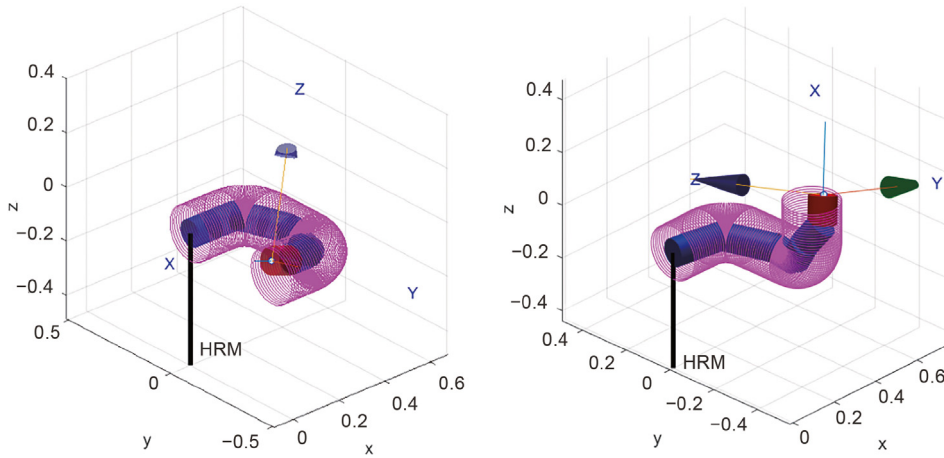


Fig. 5. Obstacle avoidance simulation results for HRM in two distinct pipeline geometries (left: Pipe 1, right: Pipe 2).

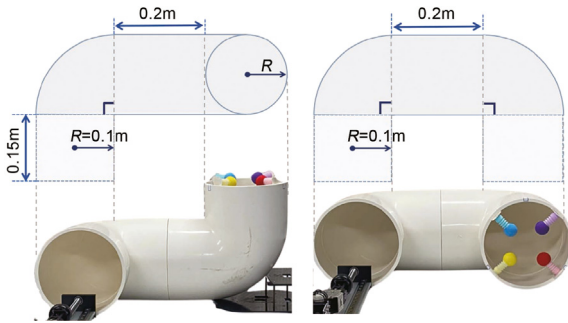


Fig. 6. Structure diagram of two kinds of pipes (right: Pipe 1, left: Pipe 2).

We designed three different target points for two distinct types of pipelines, conducting a total of six experiments to validate the feasibility of the SCA during the insertion process. The experimental process is illustrated in Fig. 7. Based on the experimental results, the average error of the HRM's end position was 0.9533 cm. The detailed results are presented in Table 3.

The experimental results validated the performance of the proposed SCA and configuration planning method.

5.3. Pipeline inspection experiment

In the pipeline inspection experiment, we designed two different inspection routes as follows.

- (1) A circular trajectory with a radius of 5 cm.
- (2) An equilateral triangular trajectory with a side length of 10 cm.

After completing the pipeline insertion process, the HRM moved along the predetermined trajectory. On the inspection path, we selected 3 to 4 points and marked them with spheres, each with a diameter of 3.2 cm, as shown in Fig. 8. During the inspection experiment, the HRM's end-effector was theoretically required to pass through the blue sphere, (purple sphere), red sphere, yellow sphere, and finally the blue sphere again, with the blue sphere serving as both the starting and ending point of the inspection.

Based on the proposed method, target points were selected along the inspection path. The configurations required to reach

Table 3

The collision situation and the error results of each group of experiments.

Pipe	Experiment	Collision (Y/N)	Error (cm)	
			Average	Maximum
Pipe 1	No. 1	N	1.3176	1.5345
	No. 2	N	0.9496	1.1733
	No. 3	N	1.2407	1.6823
Pipe 2	No. 4	N	0.3222	0.6159
	No. 5	N	0.4841	0.5612
	No. 6	N	1.4055	1.5478

these points were computed via sequential quadratic programming (SQP), and the HRM followed the planned trajectory by sequentially achieving each configuration.

Based on the returned motor joint angles, the Euclidean distance between the actual and expected end-effector positions was calculated using forward kinematics as the positioning error. The comparison between the expected and actual trajectories of the HRM's end-effector is shown in Figs. 8 and 9. According to the trajectory comparison diagram, the maximum end-effector tracking error along the planned trajectory was observed to be below 2 cm.

In the same pipeline environment, when executing different trajectories, the HRM demonstrated significantly better tracking performance in straight segments compared to curved ones, with pronounced errors occurring at bends. When executing the same trajectory in different pipeline environments, the HRM exhibited larger tracking errors during lateral movements but demonstrated more stable performance in the upper space. This discrepancy can be attributed to the upper space's proximity to the center of the HRM's dexterous workspace, where motion control achieves higher precision and is less susceptible to interference. In contrast, the lateral space approaches the boundary of the dexterous workspace, resulting in reduced motion accuracy.

6. Conclusion

This article proposed a pipeline inspection approach for exploring target points within confined pipeline environments without collisions for HRMs. The approach combined NMPC with the SCA to handle the insertion, exit, and inspection processes separately. An experimental platform was constructed and validated through tests in two types of pipelines.

Throughout the experiments, the HRM maintained smooth movement along the predetermined trajectory without collisions

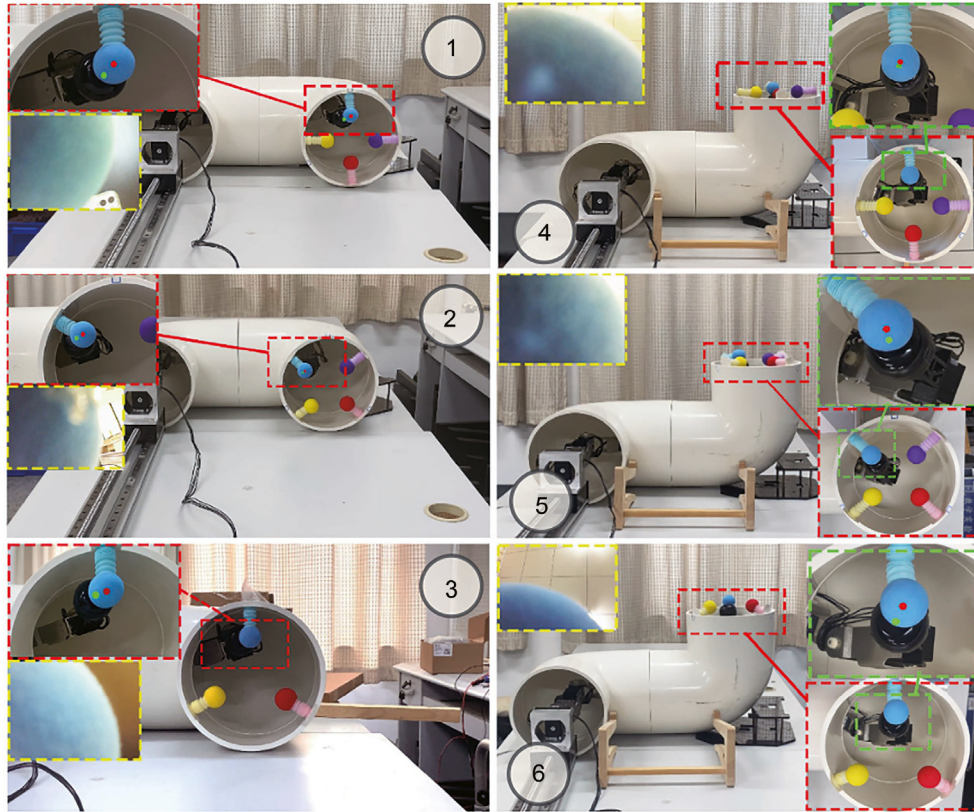


Fig. 7. Diagram of the experimental process of pipeline insertion and exit (Green dots: actual locations; red dots: target locations).

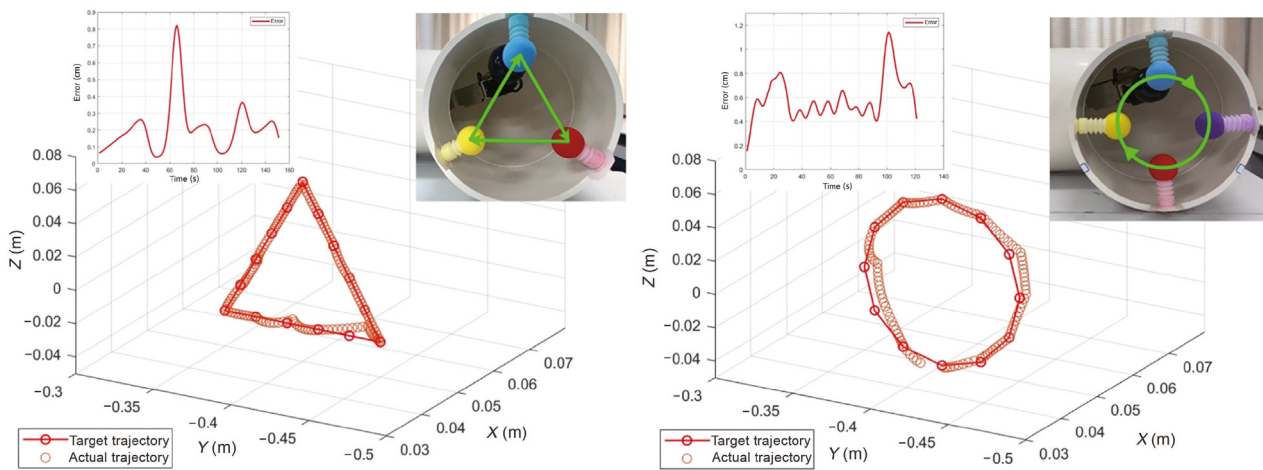


Fig. 8. Inspection experiment and trajectory deviation diagram under different trajectories in Pipe 1.

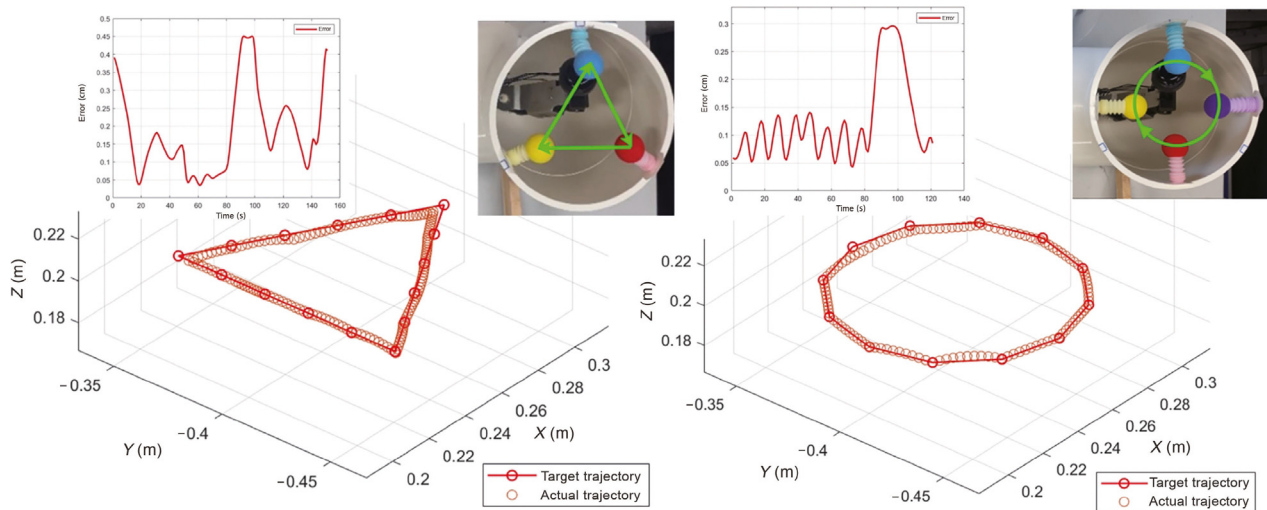


Fig. 9. Inspection experiment and trajectory deviation diagram under different trajectories in Pipe 2.

with the pipeline. During insertion and exit experiments, the HRM efficiently entered and exited the pipeline, reaching the target point with an average positioning error of 0.9533 cm. In the inspection experiments, the end-effector's maximum tracking deviation remained below 2 cm.

Future work will expand the testing and application of the pipeline inspection approach to validate its performance in a wider range of real-world scenarios. Additionally, improving the accuracy of the end-effector and optimizing the approach's performance will remain key challenges. To enhance the versatility of the inspection approach, future research will focus on real-time modeling and dynamic planning for unknown environments.

CRediT authorship contribution statement

Junjie Zhu: Writing – original draft, Data curation. **Mingming Su:** Writing – review & editing, Software. **Longchuan Li:** Writing – review & editing, Methodology. **Yuxuan Xiang:** Supervision, Investigation. **Jianming Wang:** Supervision. **Xuan Xiao:** Writing – review & editing, Supervision, Methodology.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at <https://doi.org/10.1016/j.birob.2025.100245>.

References

- [1] Z. Mu, L. Zhang, L. Yan, Z. Li, R. Dong, C. Wang, N. Ding, Hyper-redundant manipulators for operations in confined space: Typical applications, key technologies, and grand challenges, *IEEE Trans. Aerosp. Electron. Syst.* 58 (6) (2022) 4928–4937, <http://dx.doi.org/10.1109/TAES.2022.3217746>.
- [2] Z. Hu, H. Yuan, W. Xu, T. Yang, B. Liang, Equivalent kinematics and pose-configuration planning of segmented hyper-redundant space manipulators, *Acta Astronaut.* (2021) URL <https://api.semanticscholar.org/CorpusID:234860021>.
- [3] W. Wan, C. Sun, J. Yuan, Adaptive caging configuration design algorithm of hyper-redundant manipulator for dysfunctional satellite pre-capture, *IEEE Access* 8 (2020) 22546–22559, URL <https://api.semanticscholar.org/CorpusID:211059986>.
- [4] Q. Luo, Q. Hu, Y. Zhang, Y. Sun, Segmented hybrid motion-force control for a hyper-redundant space manipulator, *Aerosp. Sci. Technol.* 131 (2022) 107981, <http://dx.doi.org/10.1016/j.ast.2022.107981>, URL <https://www.sciencedirect.com/science/article/pii/S1270963822006551>.
- [5] L. Du, J. Yuan, S. Bao, R. Guan, W. Wan, Robotic replacement for disc cutters in tunnel boring machines, *Autom. Constr.* 140 (2022) 104369, <http://dx.doi.org/10.1016/j.autcon.2022.104369>, URL <https://www.sciencedirect.com/science/article/pii/S0926580522002424>.
- [6] L. Zhang, S. Huang, Z. Du, G. Ouyang, H. Chen, Motion-planning algorithm for a hyper-redundant manipulator in narrow spaces, *Comput. Mater. Contin.* (2022) URL <https://api.semanticscholar.org/CorpusID:248341812>.
- [7] C. Wang, H. Xie, H. Yang, An iterative path-following method for hyper-redundant snake-like manipulator with joint limits, *Ind. Robot* 50 (2023) 505–519.
- [8] H. Ji, H. Xie, H. Yang, A spatial path following method for hyper-redundant manipulators by step-by-step search and calculating, in: 2022 7th International Conference on Robotics and Automation Engineering, ICRAE, 2022, pp. 292–298, <http://dx.doi.org/10.1109/ICRAE56463.2022.10056161>.
- [9] A. Soltani, H. Tawfik, J. Goulermas, T. Fernando, Path planning in construction sites: performance evaluation of the dijkstra, a, and GA search algorithms, *Adv. Eng. Informatics* 16 (4) (2002) 291–303, [http://dx.doi.org/10.1016/S1474-0346\(03\)00018-1](http://dx.doi.org/10.1016/S1474-0346(03)00018-1), URL <https://www.sciencedirect.com/science/article/pii/S1474034603000181>.
- [10] C. Xu, Z. Liu, C. Hu, X. Li, Improved hybrid A* algorithm obstacle avoidance strategy based on reinforcement learning, in: 2023 42nd Chinese Control Conference, CCC, 2023, pp. 4077–4082, <http://dx.doi.org/10.23919/CCC58697.2023.10239886>.
- [11] R. Song, Y. Liu, R. Bucknall, Smoothed A* algorithm for practical unmanned surface vehicle path planning, *Appl. Ocean Res.* 83 (2019) 9–20, <http://dx.doi.org/10.1016/j.apor.2018.12.001>, URL <https://www.sciencedirect.com/science/article/pii/S0141118718302621>.
- [12] D. Yu, M.-I. Roh, Method for anti-collision path planning using velocity obstacle and A* algorithms for maritime autonomous surface ship, *Int. J. Nav. Archit. Ocean. Eng.* 16 (2024) 100586, <http://dx.doi.org/10.1016/j.ijnaoe.2024.100586>, URL <https://www.sciencedirect.com/science/article/pii/S2092678224000050>.
- [13] X. Tang, H. Zhou, T. Xu, Obstacle avoidance path planning of 6-DOF robotic arm based on improved A* algorithm and artificial potential field method, *Robotica* 42 (2) (2024) 457–481, <http://dx.doi.org/10.1017/S0263574723001546>.

- [14] M. Luo, X. Hou, J. Yang, Surface optimal path planning using an extended dijkstra algorithm, *IEEE Access* 8 (2020) 147827–147838, <http://dx.doi.org/10.1109/ACCESS.2020.3015976>.
- [15] H. Wei, Y. Zheng, G. Gu, RRT-based path planning for follow-the-leader motion of hyper-redundant manipulators, in: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2021, pp. 3198–3204, <http://dx.doi.org/10.1109/IROS51168.2021.9635876>.
- [16] H. Shen, W.-F. Xie, J. Tang, T. Zhou, Adaptive manipulability-based path planning strategy for industrial robot manipulators, *IEEE/ASME Trans. Mechatronics* 28 (3) (2023) 1742–1753, <http://dx.doi.org/10.1109/TMECH.2022.3231467>.
- [17] H. Ji, H. Xie, C. Wang, H. Yang, E-RRT*: Path planning for hyper-redundant manipulators, *IEEE Robot. Autom. Lett.* 8 (12) (2023) 8128–8135, <http://dx.doi.org/10.1109/LRA.2023.3325716>.
- [18] L. Jia, Y. Huang, T. Chen, Y. Guo, Y. Yin, J. Chen, Mda+rrt: A general approach for resolving the problem of angle constraint for hyper-redundant manipulator, *Expert Syst. Appl.* 193 (2022) 116379, <http://dx.doi.org/10.1016/j.eswa.2021.116379>, URL <https://www.sciencedirect.com/science/article/pii/S0957417421016717>.
- [19] J. Liu, Y. Tong, J. Liu, Review of snake robots in constrained environments, *Robot. Auton. Syst.* 141 (2021) 103785, <http://dx.doi.org/10.1016/j.robot.2021.103785>, URL <https://www.sciencedirect.com/science/article/pii/S0921889021000701>.
- [20] G. Qin, A. Ji, Y. Cheng, W. Zhao, H. Pan, S. Shi, Y. Song, A snake-inspired layer-driven continuum robot, *Soft Robot.* (2021) URL <https://api.semanticscholar.org/CorpusID:237608110>.
- [21] L. Tang, L.-M. Zhu, X. Zhu, G. Gu, A serpentine curve based motion planning method for cable-driven snake robots, in: 2018 25th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), 2018, pp. 1–6, <http://dx.doi.org/10.1109/M2VIP.2018.8600874>.
- [22] X. Hua, G. Wang, J. Xu, K. Chen, Reinforcement learning-based collision-free path planner for redundant robot in narrow duct, *J. Intell. Manuf.* 32 (2) (2021) 471–482, URL https://EconPapers.repec.org/RePEc:spr:joinma:v:32:y:2021:i:2:d:10.1007_s10845-020-01582-1.