

RESEARCH ARTICLE

# Shape-controllable inverse kinematics of hyper-redundant robots based on the improved FABRIK method

Pingan Niu<sup>1</sup> , Liang Han<sup>1</sup> , Yunzhi Huang<sup>1</sup> and Lei Yan<sup>2</sup>

<sup>1</sup>School of Electrical Engineering and Automation, Hefei University of Technology, Hefei, China and <sup>2</sup>School of Mechanical Engineering and Automation, Harbin Institute of Technology, Shenzhen, China

**Corresponding author:** Liang Han; Email: [lianghan@hfut.edu.cn](mailto:lianghan@hfut.edu.cn)

**Received:** 15 July 2023; **Revised:** 15 September 2023; **Accepted:** 10 October 2023; **First published online:** 7 November 2023

**Keywords:** hyper-redundant robot; inverse kinematics; improved FABRIK; controllable shape

## Abstract

Hyper-redundant robots have good prospects for applications in confined space due to their high flexibility and slim body size. However, the super-redundant structure brings great challenges for its inverse kinematics with shape constraints. Unfortunately, traditional Jacobian pseudo-inverse-based inverse kinematics method and forward and backward reaching inverse kinematics (FABRIK) method are difficult to constrain the arm shape and realize trajectory tracking in confined spaces. To solve this problem, we propose a shape-controllable FABRIK method to satisfy the given path and shape constraints. Firstly, the kinematic model of the hyper-redundant robot is established, and the canonical FABRIK method is introduced. Based on the preliminary works, the single-layer improved FABRIK method is developed to solve the position and pointing inverse kinematics considering path environment and joint angle constraints instead of two-layer geometric iterations. For tracking the desired end roll angles, the polygonal virtual arm is designed. The real arm roll angle is achieved by controlling its winding on the virtual arm. In this way, the shape can be controlled. Finally, we compare the proposed method with other three approaches by simulations. Results show that the proposed method is more efficient and the arm shape is controllable.

## 1. Introduction

Traditional rigid robots are difficult to perform tasks in narrow spaces. Hyper-redundant robots have great potential for the application in confined space due to their high flexibility and slim body size. Hyper-redundant robots have been widely studied, such as serpentine robot, continuum and soft robot. Inverse kinematics of the hyper-redundant robot is a challenging hot spot in the field. However, high efficient and shape-controllable inverse kinematics of hyper-redundant robots have not yet been solved effectively.

Serpentine robot is snake-like robot composed of multiple small links connected by Hooke's hinge, such as the segmented linkage robot designed by Tianliang Liu [1]. Continuum robots [2–4] are a jointless robot arm with a biological structure similar to an elephant trunk such as the elephant trunk-like robot developed by Walker [5], the BHA [6]. If the material of the continuum robot is soft, it can be classified as an emerging class of bionic soft robots in recent years.

Kinematic modeling: The difficulty lies in the planning problem brought by super-redundant degrees of freedom. For multi-joint continuum robots, which are composed of rigid joints sequentially connected in series, the traditional Denavit-Hartenberg (D-H) [7, 8] modeling method can be used for kinematic modeling. Tianliang Liu [9] conducts kinematic modeling according to the traditional D-H method for a cable-driven continuum robot with multiple rigid links connected by Hooke's hinge. For jointless continuum robots, the equivalent joint method is often used for kinematics modeling. The equivalent joint method equates the motion of a continuum robot without joints to virtual joints and links, trying to find the kinematics and inverse kinematics using the traditional method. Wang [10] used the virtual

joint method to equivalently model a continuum robot with constant curvature and derived its kinematic model using the D-H method. The continuum robot is equated to three rotating and one moving subsets.

Therefore, for continuum and soft robots, scholars have proposed several effective kinematic modeling methods such as the continuous curvature model, the beam model and the Cosserat rod theory. The kinematic modeling of a continuum robot with continuous curvature mainly includes constant curvature methods and non-constant curvature (also known as variable curvature (VC)) methods. Nuelle [11] models the kinematics of a continuum robot using the equivalent arc method based on the assumption of constant curvature. Huang [12] simplified the variable curvature forward kinematic model of a continuum robot arm to reduce the computational complexity. Commonly used constant curvature curves include Bessel curves in addition to circular curves. The beam model and the Cosserat rod theory develop a new theory for the characteristics of the continuum robots by equating them to deformed beams or rods to build kinematic models. Dalvand [13] uses the beam model to model the kinematics of a tendon-driven continuum robot and develops quantitative relationships between the number of tendons, tendon loads, and bending angles. Neumann [14] performs kinematic modeling of a continuum robot using the Cosserat rod model and describes the deformation of the rod using a constant curvature arc and a B spline curve respectively.

**Inverse kinematics:** A common method for solving the inverse kinematics for redundant robotic arms is the numerical iterative method. In the linkage redundant manipulator designed by Tianliang Liu [15], the kinematics of the manipulator is established using the D-H method. Considering the linkage characteristics of the arm, the Jacobi matrix is simplified and the iterative method is used for inverse kinematics. Mu Zonggao [16] proposes a segmented geometry method for the spatial hyper-redundant robot, which geometrically divided the arm into three parts. Then, the kinematics is solved separately according to the segments, thus simplifying the computational process of the inverse kinematics. However, the traditional numerical iterative method based on D-H coordinate system has the outstanding problems of low efficiency and uncontrollable arm shape and is prone to singularity problems, which limit its wide application. The higher the number of joints, the more complex the Jacobi matrix [17] is and the less efficient the solution is.

In contrast, the geometric method is more efficient and easy to constrain the arm shape, thus widely studied for inverse kinematics of hyper-redundant robots. Chirikjian [18] proposed the concept of backbone curve to describe the configuration of the super-redundant manipulator. The proposed method is computationally less expensive and is suitable for online path planning and real-time control. However, the determination and solution of the backbone function depends on experience and skill. In ref. [19], Chirikjian defines the backbone curve as a series of mode functions that can be selected as needed. Once the backbone curve is determined based on the end-effector position, a certain fitting algorithm is used to determine the joint angle. Fahimi [20, 21] used different mode functions to fit the curves, and he applied a recursive fitting algorithm to the gimbal joint to avoid solving a large number of nonlinear equations. Li Sheng [22] proposed an alternative geometric approach that allows the solution of a large number of nonlinear equations. Mohamed [23] proposed another geometric method that can find a solution from the infinite solution of the super-redundant robotic arm. This method sets the angles of adjacent joints to be equal. It is very effective for simple planar robotic arms with equal link lengths and the singularity problem is avoided. Andreas Aristidou [24] introduced the FABRIK geometric iterative method for solving the inverse kinematics. Compared with the traditional method, FABRIK is more the efficient for finding the inverse solution. The traditional FABRIK algorithm cannot solve the inverse kinematics including end roll degrees of freedom. To solve this limitation, Liu Tianliang [25] proposed the double-layer FABRIK algorithm to solve the inverse kinematics of a cable-driven hyper-redundant robot. Joint angle constraints and environmental constraints are considered to constraint the arm shape. Geometric iterative methods are more efficient in solving the inverse kinematics of hyper-redundant robot. However, it still needs further improvement in multiple constraints and efficiency. Recently, artificial neural networks have been used to solve the inverse kinematics problem for robotic arms [26–28]. The size of the training set will vary significantly with the number of degrees of freedom, and the solution is very inefficient on a continuum robotic arm with super-redundant degrees of freedom.

**Table I.** Comparisons among the state-of-the-art materials.

Method	Motivation	Feature	Limitation
Jacobian-based numerical iterative method [13]	Velocity-level kinematics	Versatile	Inefficient, not shape-controllable
Mode functions method [15–20]	Convert kinematics to a system of algebraic equations	Fast computation	Complex calculation, Limited accuracy
FABRIK geometric iterative method [21]	Geometric iterative method	High speed and high precision	The pose cannot be solved, not shape-controllable
Two-layer FABRIK method [22]	Add the attitude iterative in a new outer layer	Six-axis end pose inverse kinematics is solved	Low efficiency, not shape-controllable
The improved FABRIK method in this paper	Solve the pose inverse kinematics in one layer and control shape	Six-axis end pose inverse kinematics is solved, shape-controllable, high efficiency	Need complex virtual model

Contributions: In summary, the geometric iterative method has the advantage of high efficiency and is suitable for the inverse kinematics of hyper-redundant robots. However, the traditional geometric iterative method is different to solve the end roll angle inverse kinematics. At present, only ref. [25] uses two-layer nested iterations to solve end roll angle inverse kinematics. However, it results in large arm shape changes and low efficiency. The above-referenced state-of-the-art materials are compared in Table I. Unlike the presented approaches, we provide an efficient and improved FABRIK algorithm to solve full-degree-of-freedom inverse kinematics problems. It mainly has the following contributions.

- (1) The single-layer improved FABRIK method was developed to solve the position and pointing inverse kinematics considering path, environment and joint angle constraints instead of two-layer method. The efficiency is improved.
- (2) The virtual arm was established as the constraint space of the real arm based on the geometrical constraints of the environment. By adjusting the parameters of the virtual arm, the planning of the roll angle is achieved. The arm shape can be avoided in large changes.
- (3) The improved FABRIK method was verified by inverse kinematic planning simulations in narrow space. Results show that the efficiency is greatly improved and the shape of the arm is constrained.
- (4) The improved FABRIK method was used to realize trajectory tracking and joint angle avoidance limit in narrow space. It has the advantages of simple algorithm and high efficiency.

The rest of this paper is organized as follows. Section 1 presents the preliminary works about kinematics modeling and traditional FABRIK iterative method. In Section 3, we introduce the improved FABRIK method and present the virtual arm to solve planning of roll angle. In Section 4, we introduced trajectory tracking and joint angle avoidance limits in narrow spaces. In Section 5, simulations are carried out to verify the method. Section 6 gives the conclusion.

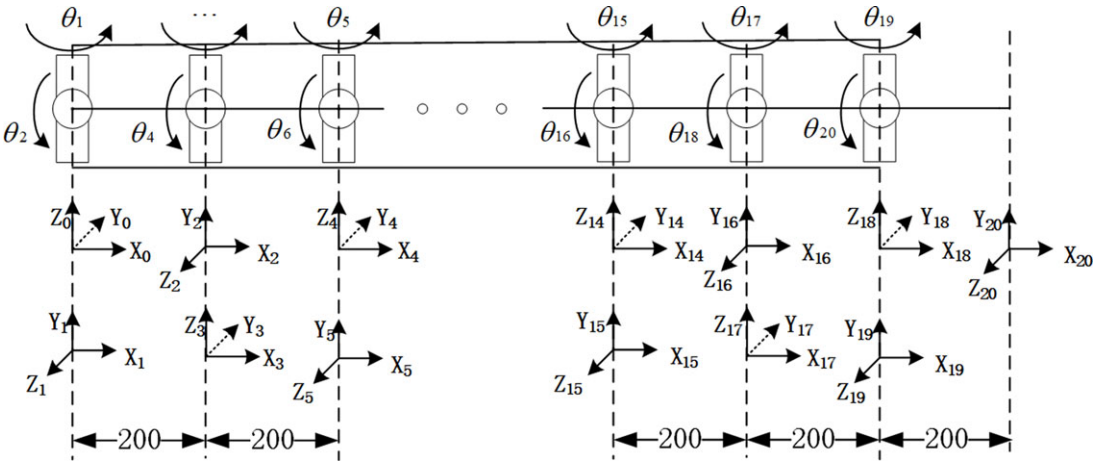
## 2. Preliminaries

### 2.1. Kinematic modeling

For a multi-joint hyper-redundant robot composed of rigid joints connected sequentially, this paper used a modified D-H coordinate system method to establish a D-H coordinate system with 20 degrees of

**Table II.** Multi-joint continuum robot arm D-H parameters.

Link <i>i</i>	$\Theta$ (°)	$\alpha$ (°)	<i>a</i> (mm)	<i>d</i> (mm)
1	0	90	0	0
2	0	0	200	0
3	0	-90	0	0
4	0	0	200	0
...	...	...	...	...
10	0	0	200	0



**Figure 1.** D-H coordinate system of the continuum robot arm.

freedom for 10 modular joints as shown in Fig. 1. The D-H parameters of each joint of the continuum robot arm are obtained according to the rules established in the D-H parameter table, as shown in Table II.

According to the D-H parameter table of the robotic arm system and the rules for the establishment of the D-H coordinate system, the Homogeneous transformation matrix of each joint of the robotic arm  ${}^{i-1}T_i$  is obtained, and its forward kinematic equations are presented as follows.

$$T_{sn} = {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 \cdots {}^{19}T_{20} \tag{1}$$

where  ${}^{i-1}T_i$  is the homogeneous transformation matrix between two adjacent coordinate systems of joint segment *i*.

**2.2. Traditional FABRIK method**

For the inverse kinematic solution problem of multi-joint continuous robot, some scholars have proposed a geometric-based FABRIK method with high efficiency, zero-error solution and no singularity problem. The conventional FABRIK is based on the geometric iterations of forward and backward iterations to quickly find the end position but not the pose, and its algorithm flow chart is shown in Alg. 1. The algorithm process for a four-joint continuum robot is shown in Fig. 2.

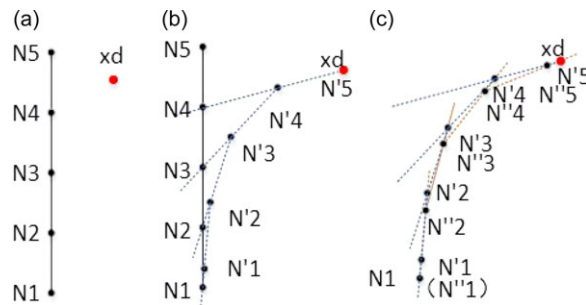
Forward arrival iteration: As shown in Fig. 2(a), *N<sub>i</sub>* is the initial joint point, and the end joint point is coincident with the target point *N<sub>5</sub>* when forward iteration is performed, and *N<sub>4</sub>* is determined according to the desired end target and the length of the joint segment. The length of *N<sub>5</sub> N<sub>4</sub>* is equal to *N<sub>5</sub> N<sub>4</sub>*, and then, the remaining points are determined as shown in Fig. 2(b), such that *N<sub>3</sub>* is on the line of *N<sub>4</sub> N<sub>3</sub>* and the length of *N<sub>4</sub> N<sub>3</sub>* is equal to *N<sub>4</sub> N<sub>3</sub>*, and the remaining points are determined in the same order. Because the forward iteration of the arm joint point *N<sub>1</sub>* is out of the initial point, the backward iteration is needed.

**Algorithm 1. Traditional FABRIK algorithm**

**Input:**  $x_d$ : Desired end position;  $\theta_0$ : initial state;  $i_{max}$ : maximum number of iterations;

**Output:**  $\theta_i$ : The desired joint positions;

- 1: **for**  $i = 1 : i_{max}$  **do**
- 2:     **if**  $\text{norm}(\Delta X > \epsilon) \&\& i < i_{max}$  **then**
- 3:         Forward iteration();
- 4:         Backward iteration();
- 5:     **end if**
- 6: **end for**
- 7: Shape2angle();



**Figure 2.** Traditional FABRIK iteration for position. (a) initial position (b) forward iteration (c) backward iteration.

Backward arrival iteration: As shown in Fig. 2(c), similar to the forward iteration, the target point N'5 is taken as the starting point, and N1 is taken as the target point of N'1 in reverse order to find the position of each joint point.

The above process is one FABRIK iteration, and the zero-error solution can be achieved by several iterations.

**2.3. Two-layer FABRIK method**

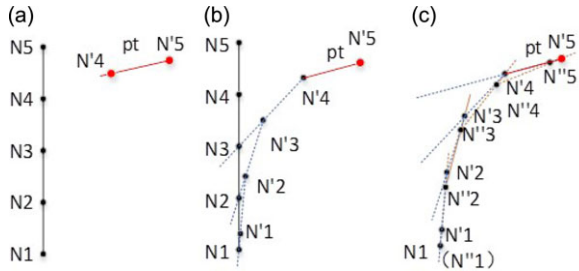
The traditional FABRIK method is difficult to solve the pose tracking, so Tianliang Liu applied two-layer FABRIK method to solve the complete inverse kinematic problem in ref. [22]. The inner loop was used to solve the position and pointing, and the outer loop was used to solve the roll angle, and segmented FABRIK method was used to solve the pointing in inner loop. The segmented FABRIK method divided the end joint and the remaining joint into two parts, so end pointing determines the new position of Ni-1 based on the desired position and pointing Pt and then solves the position of other part using the traditional FABRIK method. However, the end link is directly determined along the desired direction, the end joint angle is uncontrollable and the arm shape is not smooth. Moreover, Two-layer iterations have low efficiency. The segmented FABRIK method for position and pointing is shown in Fig. 3

**3. Shape-controllable FABRIK method**

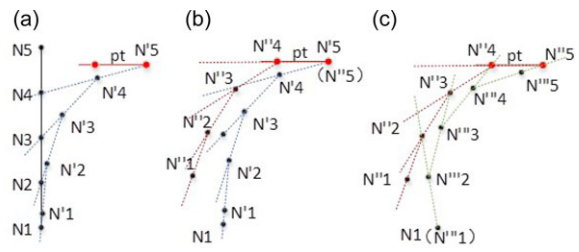
The traditional method of solving the roll angle cannot control the arm shape, which leads to irregular shape change of the arm and makes it impossible to work properly in extreme environments such as pipelines and other narrow spaces. In this section, we proposed a shape-controllable forward and backward reaching inverse kinematics (SCFABRIK) method to solve the rolling angle, which can control the

**Algorithm 2. Improved FABRIK algorithm**

**Input:**  $x_d$ : Desired end position;  $\theta_0$ : initial state;  $i_{max}$ : maximum number of iterations;  
**Output:**  $\theta_d$ : The desired joint positions;  
**for**  $i = 1 : i_{max}$  **do**  
 2: **if**  $\text{norm}(\Delta X > \varepsilon) \&\& i < i_{max}$  **then**  
     Forward iteration();  
 4:     pointing Rotation iteration ();  
     Backward iteration();  
 6: **end if**  
**end for**  
 8: Shape2angle();



**Figure 3.** Segmented FABRIK method for position and pointing. (a) initial position (b) forward iteration (c) backward iteration.



**Figure 4.** Improved FABRIK geometry iteration method. (a) forward iteration (b) pointing rotation iteration (c) backward iteration.

arm shape within a specific geometric range. The solution efficiency of single-layer iteration is greatly improved compared to traditional double-layer iteration.

**3.1. Improved FABRIK method for position and pointing inverse kinematics**

The traditional FABRIK method is a geometric iterative method to find the position. In this paper, we proposed an improved FABRIK geometric iterative method that can solve the position and pose simultaneously in a single-layer iteration. Compared with the traditional FABRIK method, we added a pointing rotation iteration after the forward position iteration. Compared with the two-layer iterative algorithm which directly forces the end link to point to the desired direction, the planning joints are smoother. The improved FABRIK method for position and pointing inverse kinematics is shown as Alg. 2.

The principle of the algorithm is shown in Fig. 4.

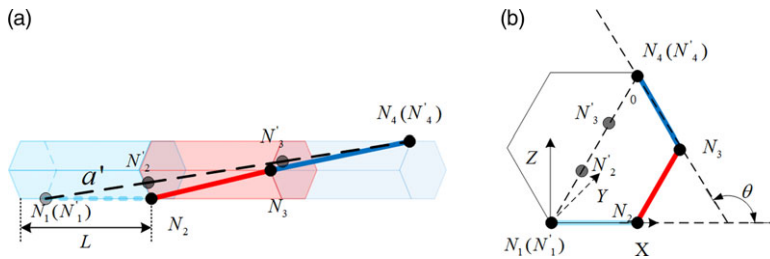


Figure 5. (a) The position of real joint points and virtual joint points (b) Geometry cross-section.

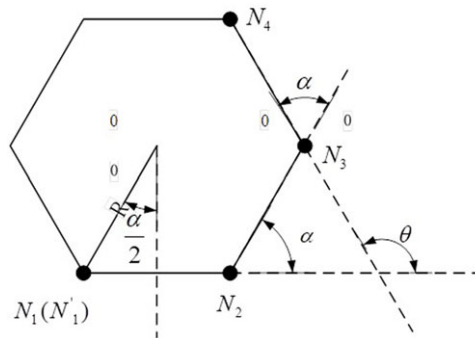


Figure 6. A cross-section of a geometry model.

Unlike the traditional FABRIK forward-backward method, the improved FABRIK method adds a rotation iteration as shown in Fig. 4(b). The 5-degree-of-freedom inverse kinematics solution of the position and pointing can be found by adding this rotation iteration.

3.2. Improved FABRIK method for roll angle inverse kinematics

The design of the super-redundant robot arm is inspired by mollusks such as snakes. Therefore, in solving the inverse solution of the desired roll angle, one can learn from the way a snake wraps around a prism. The end position, pointing and cross-roll angle are constantly changing and conform to certain geometric relationships as the snake wraps around the prism. Moreover, the super-redundant robot can be restricted to the prism.

The positions of virtual and real joint points in the geometry are determined by the following rules.

First, the prism is determined based on the external environment to ensure that the geometry does not come into contact with the external environment.

Then, determine the winding angle  $\theta$  based on the desired roll angle and the current roll angle, and the virtual joint points  $N'i$  in each segment of the geometry are selected based on the straight line connecting the base point and end point as shown in Fig. 5.

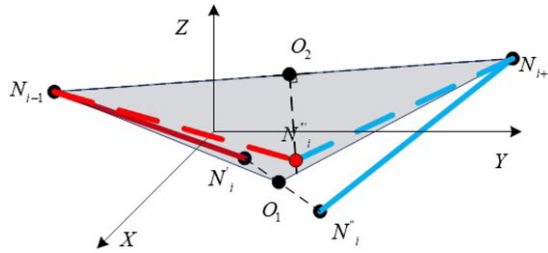
After, we solve the position and pointing inverse kinematics of the virtual arm.

Finally, the position  $N_i$  of real joint points on the geometry can be obtained by wrapping around the prism at the same angle as shown in Fig. 5. The geometric relationship between the real joint points and the geometry is shown in Fig. 6.

The input roll angle is  $\theta$

$$\theta = \theta_t + \Delta\theta \tag{2}$$

$$\Delta\theta = \theta_d - \theta_t \tag{3}$$



**Figure 7.** Adjusted joint points, the red solid line and the blue solid line are separated joint segments, the red dashed line and the blue dotted line are the adjusted joint segments.

The length of the single-segment virtual arm is

$$a' = \sqrt{a^2 - (2R \sin\left(\frac{\alpha}{2}\right))^2} \tag{4}$$

$$\alpha = \theta / (n - 1) \tag{5}$$

where  $\theta$  is the angle of the entire arm winding prism,  $\theta$ , is the current roll angle,  $\alpha$  is the angle of projection of adjacent link on the cross-section of the geometry. roll angle of single joint is  $\alpha$ ,  $a$  is the length of a single link,  $a'$  is a virtual link length,  $L$  is the length of the single-segment geometry, and the length of the single-segment virtual arm is  $a'$ .

After obtaining the model of the virtual arm, the improved FABRIK can be used for virtual arm to solve position and pointing. So we can control the motion of the prism by controlling the position and pointing of the virtual arm and then calculate the real joint points based on the position of the real arm on the geometry.

However, the motion of the geometry will lead to the separation of the real joint points. Thus, separated joint points need to be adjusted to the same point and the adjustment method in this paper is shown in Fig. 7.

If  $N_i$  is separated into two points  $N'_i, N''_i$  among the three joint points of  $N_{i-1}N_iN_{i+1}$ , take the midpoint of  $N'_iN''_iO_1$  to form the plane  $N_{i-1}O_1N_{i+1}$  and make a line perpendicular to  $N_{i-1}N_{i+1}$  on the plane  $N_{i-1}O_1N_{i+1}N''_i$ , where the length of  $O_2N''_i$  is equal to real arm length. This method can effectively adjust the separated joint points, and the adjusted points will be as close to the range of the geometry as possible.

The length of the virtual robotic arm proposed in this method will change with the data of the geometry. So the arm length needs to be updated in iterations. The algorithm is shown as Alg. 3.

## 4. Joint limit avoidance and trajectory planning

### 4.1. Joint limit avoidance

We improved a geometric method for limit avoidance of joint angles based on improved FABRIK as Fig. 8.

Traditional geometric limit avoidance methods only adjust joint points where the joint angle exceeds the limit angle. In some cases, this reduces the continuity of the entire arm, resulting in non-convergence in the process of avoiding the limit, and thus no solution.

The method we proposed is to start at the joint point that exceeds the limit angle of the joint and rotate the remaining arm as the joint point rotates, as shown in Fig. 8. This guarantees continuity throughout the arm and does not diverge over the course of iteration in some cases.



**Algorithm 3. Virtual arm method to find the pose**

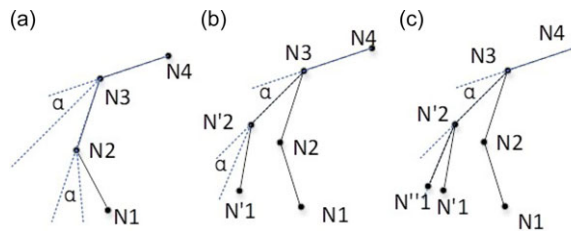
**Input:**  $x_d$ : Desired end position;  $\theta_0$ : initial state;  $i_{max}$ : maximum number of iterations;  $R$ : radius of the outer circle of the geometry based on the outer constraint space;

**Output:** The remaining roll angle error();  $\theta_d$ : The desired joint positions;

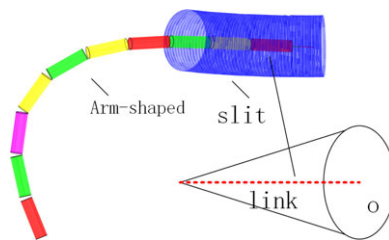
```

1:  for  $i = 1 : i_{max}$  do
2:    if  $\text{norm}(\Delta X > \varepsilon) \&\& i < i_{max}$  then
3:      The remaining roll angle error():  $\Delta\theta_{roll} = \text{acos}(Y_c \cdot Y_d)$ ;
4:      Establishing a geometric model();
5:      solving desired end position pose of virtual :  $X'_d$ 
6:      Forward iteration();
7:      pointing Rotation iteration ();
8:      Backward iteration();
9:    end if
10:  end for
11:  Shape2angle();

```



**Figure 8.** Joint angle avoidance limit. (a) Initial arm angle. (b) First step iteration of joint avoidance limit. (c) Joint Avoidance Limit Step 2 Iteration.



**Figure 9.** Trajectory tracking in confined spaces and the cone is the range of end joint deflection angles.

**4.2. Trajectory planning**

The robotic arm works in a narrow space and it is necessary to realize that the arm tracks the trajectory. During this process, the arm must not touch the obstacle. Figure 9 shows the trajectory tracking process in confined spaces. We proposed a geometric method based on improved FABRIK that can achieve the goal well.

Taking advantage of the continuity of the improved FABRIB method, the arm shape is automatically adjusted according to the change of the path trajectory during the tracking trajectory, and when the arm touches an obstacle, the influence of the end pose on the arm shape is used to adjust the arm shape. When the end pose is determined or cannot meet the obstacle avoidance, the pose of the previous link is adjusted, so as to achieve the goal of fast obstacle avoidance in trajectory tracking. The algorithm flow is

**Algorithm 4. Trajectory tracking**

**Input:**  $x_i$ : trajectory;  $\theta_0$ : initial state;  $i_{\max}$ : maximum number of iterations;  $R$ : radius of the outer circle of the geometry based on the outer constraint space;

**Output:** The remaining roll angle error();  $\theta_d$ : The desired joint positions;

```

1:  for  $j = 1 : n$  do
2:     $X_d = X_j$ 
3:    for  $i = 1 : i_{\max}$  do
4:      if  $\text{norm}(\Delta X > \varepsilon) \&\& i < i_{\max}$  then
5:        The remaining roll angle error();  $\Delta\theta_{roll} = \text{acos}(Y_c \cdot Y_d)$ ;
6:        Forward iteration();
7:        Angle avoidance limit();
8:        if  $d_i > d'$  then
9:          Pointing rotate iteration2();
10:       end if
11:       Pointing rotate iteration();
12:       Backward iteration();
13:       Angle avoidance limit();
14:     end if
15:   end for
16:   Shape2angle();
17: end for

```

shown in Alg. 4 where  $d_i$  is the distance from each joint point to the path trajectory,  $d'$  is the maximum distance from a joint point to a path trajectory.

## 5. Results

### 5.1. Simulations of improved FABRIK method

We carried out inverse kinematic simulations based on the traditional method and the proposed method in this paper respectively and compared the maximum joint angle changes. The proposed method has better continuity and less variation, as shown in Fig. 10.

From Fig. 11, it is easy to find that the traditional method does not take into account the pointing iteration, but directly aligns the end link to the end desired direction. Therefore, the traditional method leads to a large variation of the joint angle, which is close to  $180^\circ$ . In contrast, the joint angle of the method in this paper does not exceed  $80^\circ$ .

We made a table to compare the advantages of the method in this article as shown in Table III, where  $N$  is the number of iterations and the time only counts the iteration time.

In summary, the improved FABRIK geometric iterative method inherits and retains the advantages of the FABRIK geometric method in terms of high efficiency, zero-error solution and no singularity problem; it also solves the problem that the FABRIK method cannot find the pose and the end joint angle of the improved FABRIK method is too large.

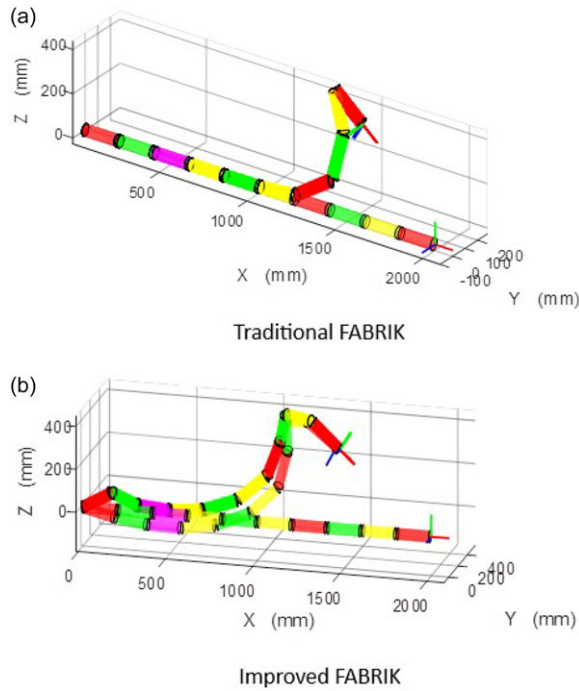
### 5.2. Simulations of joint angle avoidance limit

Select random desired target points and use the improved FABRIK algorithm to solve the desired joint position. The simulation is shown in Fig. 12(a). The simulation after adding joint limit avoidance is shown in Fig. 12(b). The joint angles are shown in Fig. 13.

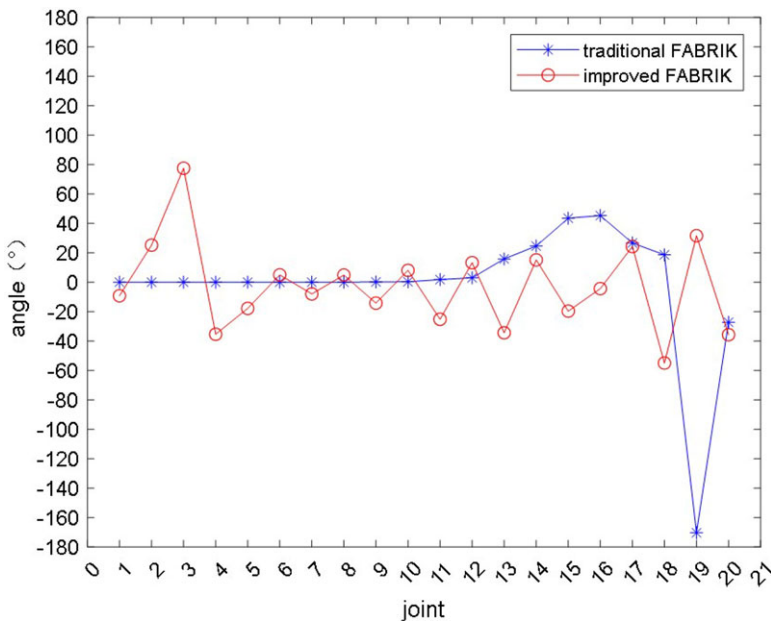
From Fig. 13, it is easy to find that the joint angle obtained by the limit avoidance method in this paper does not exceed  $65^\circ$  instead of the original close to  $140^\circ$ .

**Table III.** The results of a comparison of the two methods.

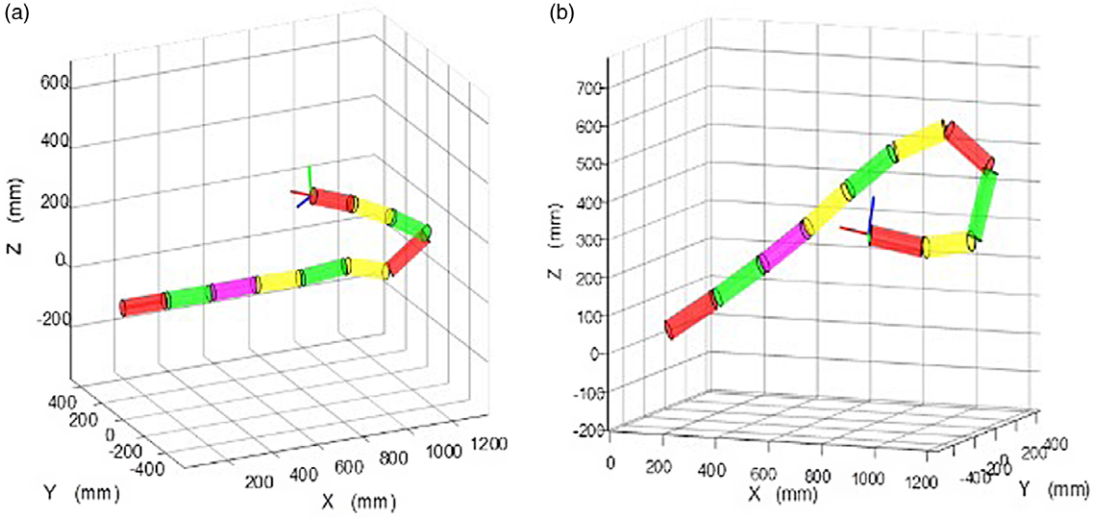
	Pose error N = 15	Time	Maximum joint angle
Ref. [22]	0.01 mm, 0.01°	0.0243 s	-170.3464
The proposed method	0.01 mm, 0.01°	0.0269 s	-54.9675



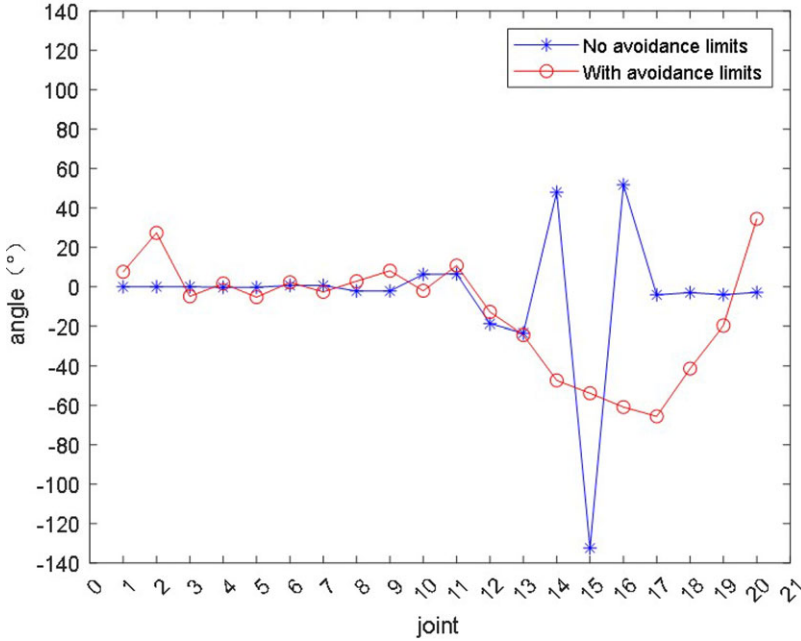
**Figure 10.** Inverse kinematics simulations of traditional FABRIK and improved FABRIK method.



**Figure 11.** Comparison of joint angle. The blue curve shows the joint angles using the tradition FRBRIK. The red curve shows the joint angles using the improved FRBRIK proposed in this paper.



**Figure 12.** Comparative simulations of joint angle avoidance. (a) Joint position without limit avoidance. (b) Joint position with limit avoidance.



**Figure 13.** Comparison of joint avoidance limit. The blue curve shows the joint angles without joint angle avoidance. The red curve shows the joint angles with joint angle avoidance.

**5.3. Simulations of arm shape planning**

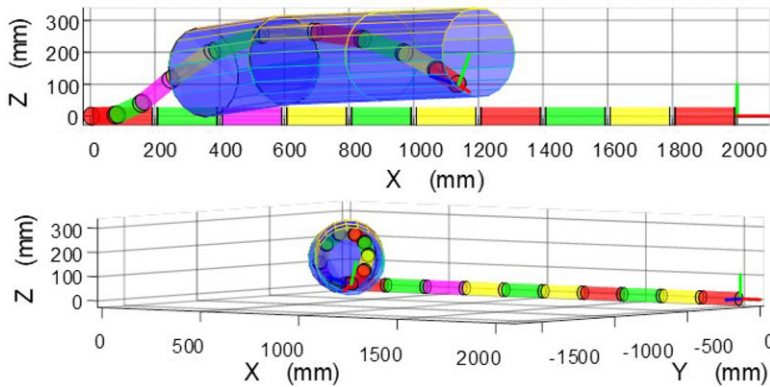
We used pipes to simulate obstacles to observe the arm shape. As shown in Fig. 14, the robotic arm can complete the inverse solution under the condition of obstacle constraints. The table to compare the advantages of the method in this article is shown in Table IV.

The sample variance is calculated as Eq. (6).

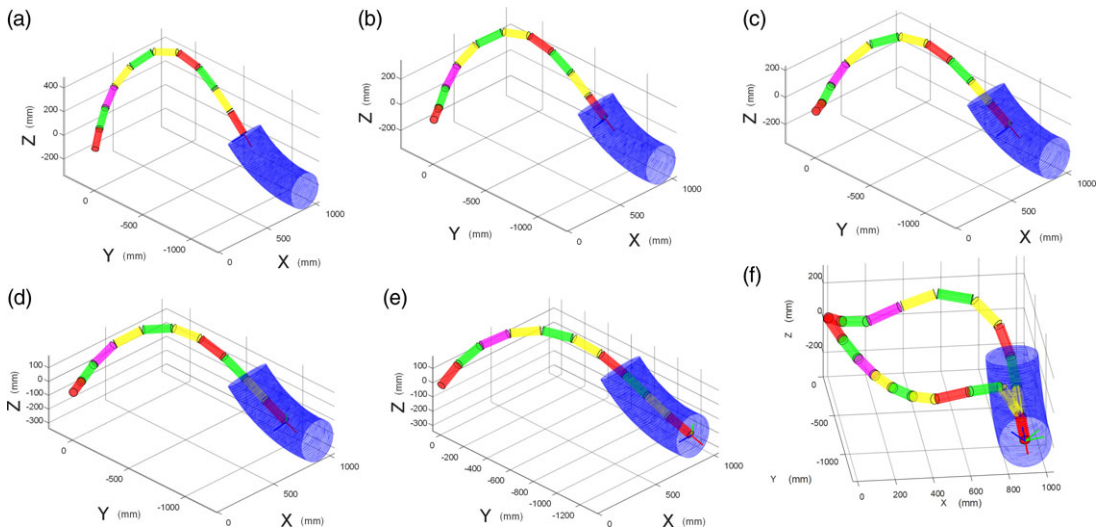
$$S^2 = \sum_{i=1}^n (\theta_i - \bar{\theta})^2 / n \tag{6}$$

**Table IV.** The results of a comparison of the two methods.

	Pose error	Joint angle sample variance	Time
Ref. [22]	1% mm, 1% 1°	24.0217	0.2067 s
The proposed method	1% mm, 1% 1°	13.9553	0.0436 s



**Figure 14.** Different perspectives of simulation of geometric constraints for inverse solution.



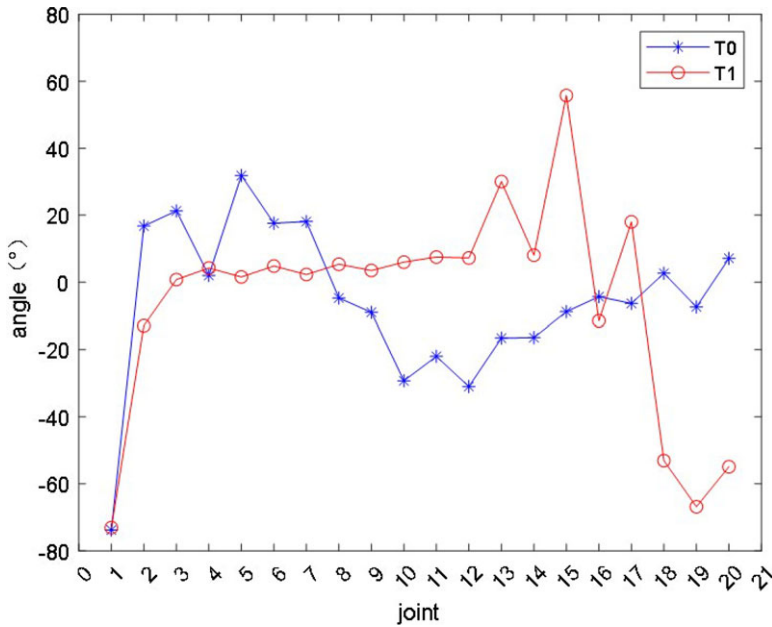
**Figure 15.** Trajectory tracking based on the improved FABRIK.

From the above simulation, it is easy to see that in the process of solving the inverse kinematics of the manipulator, the arm shape can be well constrained within a certain range.

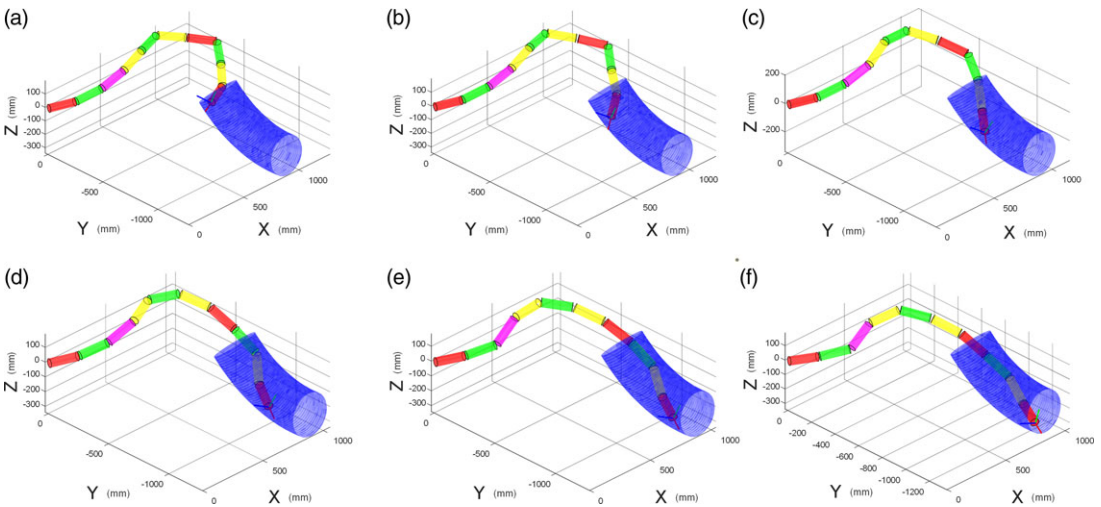
**5.4. Simulations of trajectory tracking**

We implement trajectory tracking based on the improved FABRIK method. Only the solution for the end 5 degrees of freedom can be achieved as shown in Fig. 15(a)–(e). However, traditional methods fail to solve the roll angle problem as shown in Fig. 15(f). The arm hits the pipe.

Figure 16 shows the change of joint angle based on traditional methods. From Fig. 17, we can find the end three angles of the red curve are big which causes the collision with the pipe.



**Figure 16.** The change of joint angle during roll angle solution based on traditional methods. The blue curve is the joint angles in the simulation of 5-degree-of-freedom inverse kinematics at T0. The red curve is the joint angles in the simulation of 6-degree-of-freedom inverse kinematics at T1.

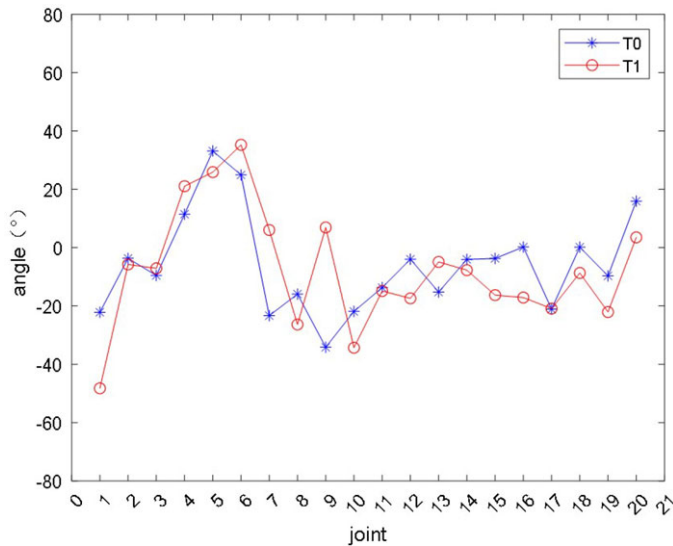


**Figure 17.** Trajectory tracking based on the SCFABRIK.

We implemented trajectory tracking for the terminal 6-degree-of-freedom solution based on the SCFABRIK, and in the process of solving the roll angle, the arm shape is well controlled in a narrow space. The trajectory tracing based on the SCFABRIK method is shown in Fig. 17.

Figure 18 shows the change of joint angle based on SCFABRIK methods and we can find the angles of the red curve are close to the blue curve which provides good continuity in the shape of the arm.

From the above simulation results, it can be seen that the proposed method can effectively and quickly realize the end 6-degree-of-freedom inverse solution in narrow space and the arm shape is controlled, achieving a good continuity effect.



**Figure 18.** The change of joint angle during roll angle solution based on SCFABRIK methods. The blue curve is the joint angles in the simulation of 5-degree-of-freedom inverse kinematics at T0. The red curve is the joint angles in the simulation of 6-degree-of-freedom inverse kinematics at T1.

## 6. Conclusion

In this paper, we improved the FABRIK method, proposed a SCFABRIK inverse kinematics solution method based on geometric constraints and analyzed kinematics modeling, geometric model and virtual manipulator; then, trajectory tracking in narrow space and joint avoidance limit were introduced; finally, we carried out MATLAB simulation and shared the code to the readers for ease of use [29]. The main research results achieved are as follows:

- (1) We analyzed the limitation of FABRIK that only solves position and improved the method and proposed improved FABRIK so that it can solve the position and pointing; it also solves the problem of end joint angle being too big caused by the traditional method in the process of re-finding the pointing.
- (2) In view of the working requirements of ultra-redundant robotic arms in narrow environments and requiring rolling angles, we established a geometric model to constrain the arm shape of the robotic arm within a controllable range, realizing the need to work in a narrow space.
- (3) Based on the improved method, we did trajectory tracking and joint angle limit avoidance to realize the work of the robotic arm in a narrow space.

The limitation of the proposed method is complex. The proposed method needs to build virtual arm, derive the relationship between the virtual arm and the real arm, and solve the joint position of the actual joint point on the virtual arm.

As a special robot, the ultra-redundant robotic arm involves knowledge in many disciplines and fields knowledge, and its research has certain prospect significance. In the future, further research should include:

- (1) The research on the effective obstacle avoidance planning algorithm of the robotic arm in three-dimensional space is helpful to carry out efficient planning and control of different tasks and improve its environmental adaptability in different environments.

- (2) Research on the autonomy and intelligence of robotic arm system. In-depth study of multi-sensor fusion technology of robotic arm system to realize its autonomous intelligent control in unknown and complex environment to ensure the completion of tasks in small unknown environment.

**Author contributions.** Pingan Niu and Liang Han designed the study. Pingan Niu carried out the experiments. Liang Han and Yunzhi Huang calibrated English expression. Pingan Niu wrote the article. Liang Han and Lei Yan did the review and guidance.

**Financial support.** This work was supported by National Natural Science Foundation of China (Grant No. 62273129, Grant No. 62203147), Anhui Provincial Key Research and Development Project (Grant No. 2022a05020025), the Fundamental Research Funds for the Central Universities of China (Grant No. JZ2022HGTA0339).

**Competing interests.** There are no competing interests.

## References

- [1] T. Liu, W. Xu, T. Yang and Y. Li, "A hybrid active and passive cable-driven segmented redundant manipulator: Design, kinematics, and planning," *IEEE/ASME Trans. Mechatron.* **26**(2), 930–942 (2021).
- [2] C. Yang, H. Xu, X. Li and F. Yu, "Kinematic modeling and solution of rigid-flexible and variable-diameter underwater continuous manipulator with load," *Robotica* **40**(4), 1020–1035 (2022).
- [3] D. Y. Kolpashchikov, N. V. Laptev, V. V. Danilov, I. P. Skirnevskiy, R. A. Manakov and O. M. Gerget, "Fabrik-based Inverse Kinematics for Multi-section Continuum Robots," *2018 18th International Conference on Mechatronics - Mechatronika (ME)* (2018) pp. 1–8.
- [4] S. Kolachalama, S. Lakshmanan, "Continuum robots for manipulation applications: A survey," *J. Robot.* **2020**, 19 (2020). <https://doi.org/10.1155/2020/4187048>.
- [5] I. D. Walker, "Continuous backbone "continuum" robot manipulators," *Int. Scholarly Res. Not.* **2013**, 1–19 (2013). <https://doi.org/10.5402/2013/726506>.
- [6] M. Rolf and J. J. Steil, "Constant Curvature Continuum Kinematics as Fast Approximate Model for the Bionic Handling Assistant," *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems* (2012) pp. 3440–3446.
- [7] L.-S. Huang and R.-K. Jiang, "A New Method of Inverse Kinematics Solution for Industrial 7dof Robot," *Proceedings of the 32nd Chinese Control Conference* (2013) pp. 6063–6065.
- [8] J. Yang, L. Jin, Z. Han, D. Zhao and M. Hu, "Sensitivity analysis of factors affecting motion reliability of manipulator and fault diagnosis based on kernel principal component analysis," *Robotica* **40**(8), 2547–2566 (2022).
- [9] W. Xu, T. Liu and Y. Li, "Kinematics, dynamics, and control of a cable-driven hyper-redundant manipulator," *IEEE/ASME Trans. Mechatron.* **23**(4), 1693–1704 (2018).
- [10] C. Wang, C. G. Frazelle, J. R. Wagner and I. D. Walker, "Dynamic control of multisection three-dimensional continuum manipulators based on virtual discrete-jointed robot models," *IEEE/ASME Trans. Mechatron.* **26**(2), 777–788 (2020).
- [11] K. Nuelle, T. Sterneck, S. Lilje, D. Xiong, J. Burgner-Kahrs and T. Ortmaier, "Modeling, calibration, and evaluation of a tendon-actuated planar parallel continuum robot," *IEEE Robot. Autom. Lett.* **5**(4), 5811–5818 (2020).
- [12] X. Huang, J. Zou and G. Gu, "Kinematic modeling and control of variable curvature soft continuum robots," *IEEE/ASME Trans. Mechatron.* **26**(6), 3175–3185 (2021).
- [13] M. M. Dalvand, S. Nahavandi and R. D. Howe, "An analytical loading model for  $n$ -tendon continuum robots," *IEEE Trans. Robot.* **34**(5), 1215–1225 (2018).
- [14] M. Neumann and J. Burgner-Kahrs, "Considerations for Follow-the-Leader Motion of Extensible Tendon-Driven Continuum Robots," *2016 IEEE International Conference on Robotics and Automation (ICRA)* (2016) pp. 917–923.
- [15] T. Liu, Z. Mu, W. Xu, T. Yang, K. You, H. Fu and Y. Li, "Improved Mechanical Design and Simplified Motion Planning of Hybrid Active and Passive Cable-Driven Segmented Manipulator with Coupled Motion," *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2019) pp. 5978–5983.
- [16] Z. Mu, H. Yuan, W. Xu, T. Liu and B. Liang, "A segmented geometry method for kinematics and configuration planning of spatial hyper-redundant manipulators," *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **50**(5), 1746–1756 (2020).
- [17] S. Kleff and N. Li, "Robust motion planning in dynamic environments based on sampled-data hamilton–jacobi reachability," *Robotica* **38**(12), 2151–2172 (2020).
- [18] G. S. Chirikjian, "A Continuum Approach to Hyper-Redundant Manipulator Dynamics," *Proceedings of 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'93)*, vol. 2 (1993) pp. 1059–1066.
- [19] G. S. Chirikjian and J. W. Burdick, "A modal approach to hyper-redundant manipulator kinematics," *IEEE Trans. Robot. Autom.* **10**(3), 343–354 (1994).
- [20] F. Fahimi, H. Ashrafiuon and C. Nataraj, "An improved inverse kinematic and velocity solution for spatial hyper-redundant robots," *IEEE Trans. Robot. Autom.* **18**(1), 103–107 (2002).
- [21] F. Fahimi, H. Ashrafiuon and C. Nataraj, "Obstacle avoidance for spatial hyper-redundant manipulators using harmonic potential functions and the mode shape technique," *J. Field Robot.* **20**(1), 23–33 (2003).



- [22] L. Sheng, W. Yiqing, C. Qingwei and H. Weili, “A New Geometrical Method for the Inverse Kinematics of the Hyper-Redundant Manipulators,” *2006 IEEE International Conference on Robotics and Biomimetics (2006)* pp. 1356–1359.
- [23] H. A. F. Mohamed, M. M. Samer Yahya and S. S. Yang, “A New Inverse Kinematics Method for Three Dimensional Redundant Manipulators,” *2009 ICCAS-SICE (2009)* pp. 1557–1562.
- [24] A. Aristidou and J. Lasenby, “Fabrik: A fast, iterative solver for the inverse kinematics problem,” *Graph Models* **73**(5), 243–260 (2011).
- [25] T. Liu, T. Yang, W. Xu, G. Mylonas and B. Liang, “Efficient inverse kinematics and planning of a hybrid active and passive cable-driven segmented manipulator,” *IEEE Trans. Syst. Man Cybern. Syst.* **52**(7), 4233–4246 (2022).
- [26] W. Yuqi, C. Jinjiang, G. Ranran, Z. Lei and W. Lei, “Study on the design and control method of a wire-driven waist rehabilitation training parallel robot,” *Robotica* **40**(10), 3499–3513 (2022).
- [27] P.-Y. Zhang, T.-S. Lü and L.-B. Song, “RBF networks-based inverse kinematics of 6r manipulator,” *Int. J. Adv. Manuf. Technol.* **26**(1-2), 144–147 (2005).
- [28] Z. Bin, X. Rong and W. Jun, “Kinematics Analysis of a Novel 7-dof Humanoid Manipulator for Table Tennis,” *2011 International Conference on Electronics, Communications and Control (ICECC)* (2011) pp. 1524–1528.
- [29] P. Niu, online. <https://github.com/niupingan/Improved-FB.git>.