



A Path Planning Method Based on Optimal Arm Angle of 7-Degrees-of-Freedom Manipulator

Li Zhang¹ · Bing Han^{1,2} · Jianwei Niu³ · Kun Tong¹ · Xinyang Tian¹ · Xiaolong Yu^{1,4} · Renluan Hou¹ · Yuliang Guo¹

Received: 4 October 2024 / Accepted: 15 March 2025 / Published online: 22 April 2025
© The Author(s) 2025

Abstract

This paper aims to address the issue of explicitly utilizing the joint limit avoidance, singularity avoidance, and self-motion obstacle avoidance properties of a 7-degrees-of-freedom manipulator during path planning, given a cartesian space task path. First, the redundant degree of freedom of the manipulator is represented using arm angle parameterization, and the mapping of joint limits and singular configurations to arm angle space is derived in detail, in conjunction with analytical inverse kinematics solution. Next, based on the cartesian space task path, the arm angle space is discretized to obtain the feasible arm angle range under the self-motion obstacle avoidance condition. The arm angle values determine the joint angles of the manipulator. To minimize task execution time and improve efficiency, the total joint angle displacement of the manipulator is set as the optimization objective, and an improved particle swarm optimization algorithm is employed to iteratively search for the optimal arm angles at each path point. Finally, the effectiveness of the proposed method is validated through both simulation and real machine experiments.

Keywords Redundant manipulator · Inverse kinematics · Path planning · Joint limit avoidance · Self-motion obstacle avoidance

1 Introduction

The 7-degrees-of-freedom (7-DOF) manipulator has become one of the research hotspots in robotics. Compared with the 6-DOF manipulator, the 7-DOF manipulator has higher flexibility and can prevent joints from over-limiting [1], avoid singularity configurations [2], and optimize joint torque [3] while completing established tasks. Therefore, the 7-DOF manipulator is widely used in the fields of medical treatment, welding, narrow space operations, automotive paint

spraying, aerospace, humanoid robots, etc. In operations such as welding and spraying, there are strict requirements on the geometric accuracy of the end path. In this case, the manipulator not only needs to ensure that the motion trajectory of the end effector meets the task requirements, but also needs to realize the functions of obstacle avoidance and singular configuration avoidance at the same time. Given the end path, how to explicitly utilize these characteristics of the 7-DOF manipulator (preventing joint over-limit, avoiding singular configurations, self-motion obstacle avoidance, and optimizing joint torque, etc.) so that the manipulator can complete the task with the optimal path is a very worthy research issue.

To study the joint limit and singularity avoidance characteristics of the 7-DOF manipulator, its inverse kinematics solution needs to be solved first. Scholars have proposed a variety of methods to solve the inverse kinematics solution of the 7-DOF manipulator. The velocity-based method mainly uses the Jacobian matrix for solution. Although this method is widely used, it requires complex calculations, has low solution accuracy, and is numerically unstable at the Jacobian singularity [4–7]. Literatures [8–12] solve the inverse kinematics solution of the manipulator based on

✉ Xiaolong Yu
yxiaolong@buaa.edu.cn

¹ Zhejiang Industrial Big Data And Robot Intelligent System Key Laboratory, Hangzhou Innovation Institute, Beihang University, Hangzhou 310052, China

² School of Mechanical Engineering, Zhejiang University, Hangzhou 310030, China

³ State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing 100191, China

⁴ School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China

intelligent optimization algorithms, but this type of method requires iterative calculations and has poor real-time performance. The position-based method mainly uses the analytical method to solve the inverse kinematics solution [13–15], but this method can only solve the inverse kinematics solution of the manipulator with a specific geometric configuration. Since the 7-DOF manipulator widely adopts the S-R-S configuration, this method still has certain versatility. Lee et al. [16] first proposed a position-based inverse kinematics solution method for redundant manipulators by parameterizing redundant joint. However, this method cannot be used to analyze the limits of manipulator joints. Shimizu et al. [1] used the arm angle to describe the self-motion of the elbow joint, derived the joint angle expression with the arm angle as the independent variable, and briefly discussed the calculation of the arm angle range under joint limit conditions. To solve the singular problem of the algorithm, Yan et al. [17] used two orthogonal vectors to define the reference plane, thus proposing the concept of double arm angle. However, they did not explore the relationship between arm angle and joint angle. Gong et al. [18] conducted an inverse kinematics solution analysis on the discrete path, by introducing arm angle constraint, they successfully established a mapping relationship between the arm angle and joint angle, and obtained an arm angle interval that avoids joint limit. Then they obtained the arm angle range under obstacle avoidance conditions through collision detection. However, they did not discuss the impact of singularity on the arm angle interval, and did not continue to optimize the arm angle to obtain the optimal arm angle that satisfies the motion constraints. Faria [19], Tian [20], and Wang [21] introduced configuration parameters to control the global manifold of the manipulator, and used arm angle to parameterize the redundant DOF of the manipulator. Based on the relationship between arm angle and joint angle, they obtained a feasible arm angle interval under the joint limit and singular conditions. However, they did not provide further detailed calculation methods for the arm angle range. Quan et al. [22] designated the end joint as a redundant joint and found the expressions of the remaining six joints regarding the redundant joint. Then they optimized the path based on the path cost and obtained a collision-free continuous trajectory of the manipulator. However, they did not explore the possible effects of joint limit and singularity on the path.

Benzaoui et al. [23] introduced the self-motion vector into the inverse kinematics solution level of the manipulator based on the Jacobian matrix, and avoided obstacles while achieving trajectory tracking in cartesian space. However, they did not further process and optimize the cartesian space path. Liu et al. [24] proposed a method for collaborative optimization of obstacle avoidance and singularity avoidance by introducing two real-time minimum distance obstacle avoidance parameters and improving the

self-motion obstacle avoidance in the zero space. However, this method has not been verified on a 7-DOF manipulator. Zhu et al. [25] proposed a new self-motion control method that enables the manipulator to complete obstacle avoidance while reaching the task target, and simulated it on a 7-DOF space manipulator. However, they did not jointly study the self-motion obstacle avoidance and joint limit avoidance problems. Li et al. [26] proposed a general motion planning framework. The framework first combined rapidly-exploring random tree (RRT) and deep reinforcement learning (DRL) to solve the length optimal path in cartesian space; then combined DRL with inverse kinematics solution to solve the inverse kinematics problem of redundant manipulator with energy constraints. However, the performance of this method depended on the success rate of inverse kinematics solution (the highest is only 82.8%), and the optimization goal was limited to minimizing energy consumption, which was difficult to expand to other optimization goals. Qiu et al. [27] introduced two parameters, end pose and arm angle, to describe the path of a 7-DOF manipulator in cartesian space to utilize its redundancy. They solved the motion path that satisfies singularity avoidance, obstacle avoidance, and joint limit avoidance based on a graph search algorithm. However, this method took the shortest path as the optimization goal and could not further optimize the overall energy consumption or joint torque of the motion path, nor could it intuitively display the distribution of all possible paths. To ensure the exponential stability of unmanned surface vehicles with time-varying coupling delays, Long et al. [28, 29] proposed a sampled-data control method based on non-volatile memory. However, this paper mainly solves the problem of offline path planning of the manipulator, and the signal delay phenomenon has little effect on offline path planning. Nevertheless, in subsequent real-time path planning research, the impact of signal delay on real-time path planning will be explored.

At present, there is still a lack of dedicated path planning methods for 7-DOF manipulators. As far as we know, few studies have combined factors such as joint limit avoidance, singularity avoidance, and self-motion obstacle avoidance to enable the manipulator to complete the task with the optimal path. To solve this problem, this paper proposes a path planning method for a 7-DOF manipulator based on the optimal arm angle. The structure of this article is as follows. Section 2 analyzes the kinematic model of the 7-DOF manipulator. Section 3 deduces the calculation method of the arm angle range under joint limit avoidance and singular avoidance conditions in detail. Section 4 introduces the obstacle avoidance path planning algorithm of the redundant manipulator. Section 5 introduces the calculation method of the optimal arm angle based on the improved particle swarm optimization algorithm. Section 6 verifies the method proposed in the article.

Fig. 1 The coordinate system of each axis of the Xmate pro 3 manipulator

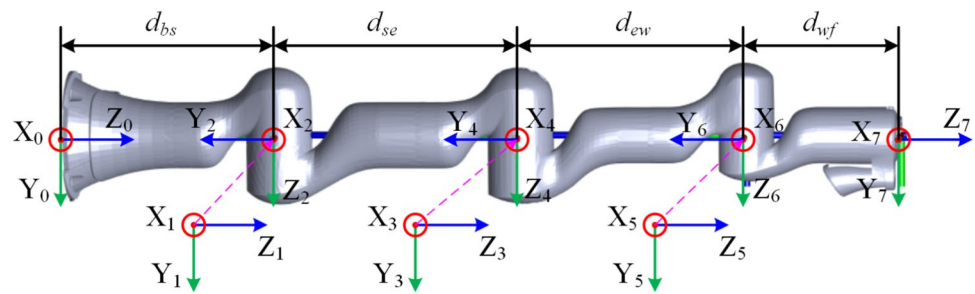


Table 1 The MD-H kinematic parameters of the manipulator

Joint	a_i / mm	$\alpha_i / ^\circ$	d_i / m	$\theta_i / ^\circ$
1	0	0	0.3415	θ_1
2	0	-90	0	θ_2
3	0	90	0.394	θ_3
4	0	-90	0	θ_4
5	0	90	0.366	θ_5
6	0	-90	0	θ_6
7	0	90	0.2503	θ_7

2 Kinematic Analysis

Similar to the human arm structure (shoulder-elbow-wrist), the 7-DOF manipulator usually adopts the S-R-S (spherical-rotational-spherical) configuration. The axes of joints 1–3 and 5–7 in the S-R-S configuration all intersect at one point, forming two spherical structures with 3-DOF. The two spherical structures are connected by joint 4. Figure 1 shows the S-R-S configuration of the 7-DOF manipulator. The Xmate pro3 manipulator is used as a model for illustration. The modified Denavit-Hartenberg (MD-H) method is used to establish the coordinate system of each axis of the manipulator. Table 1 lists the MD-H kinematic parameters of the manipulator.

To better represent the self-motion of the manipulator, the configuration parameters GC_k and the arm angle ψ are introduced. GC_k is used to represent the manifold of the manipulator in space, corresponding to the inverse kinematics solution of the manipulator. The parameter ψ is used to represent the self-motion of the manipulator, that is, the redundant DOF of the manipulator, as shown in Fig. 2. When the angle of joint 3 is zero, the point E^v is located on the plane BSW. The plane BSW is set as the reference plane, and the angle between the plane SEW and the plane BSW is the arm angle ψ .

$$GC_k = \begin{cases} 1\theta_i \geq 0 \\ -1\theta_i < 0 \end{cases}, k = 2, 4, 6 \quad (1)$$

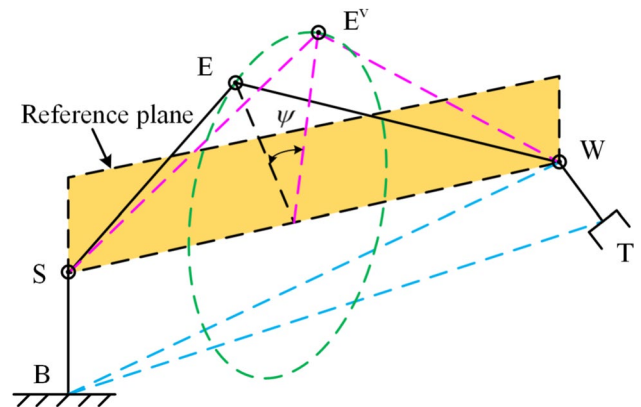


Fig. 2 Arm angle diagram

To better show the contribution of this article, the solution process of the inverse kinematics solution is placed in [Appendix](#).

3 Joint limit and Singularity

To ensure the stable operation of the manipulator in the task space, joint limit avoidance and singularity avoidance are two important conditions. According to Eqs. (33) and (38), the joint angle and singularity can be mapped to the arm angle space, and the joint limit avoidance and singularity avoidance can be achieved by determining the feasible arm angle range. The feasible arm angle range can be expressed as:

$$\Psi_{\text{lim}} = \bigcap_7 \Psi_i \text{ and } \Psi_i = \bigcup_{j=1}^n \Psi_{ij}, i = 1, 2, \dots, 7 \quad (2)$$

where Ψ_i is the arm angle range corresponding to the i -th joint; Ψ_{ij} is the j -th interval of the arm angle range of the i -th joint. By analyzing Eqs. (25), (33), and (38), it can be obtained that, the relationship between the $\theta_1, \theta_3, \theta_5, \theta_7$ and the arm angle ψ can be expressed by the arctangent function (Eq. (3)); the relationship between the θ_2, θ_6 and the arm angle ψ can be expressed by the arccosine function (Eq. (4)); the θ_4 has nothing to do with the arm angle ψ .

$$\theta_i(\psi) = a \tan 2(GC_k(a_n \sin \psi + b_n \cos \psi + c_n), GC_k(a_d \sin \psi + b_d \cos \psi + c_d)) \quad (3)$$

$$\theta_i(\psi) = GC_k \arccos(a \sin \psi + b \cos \psi + c) \quad (4)$$

where a_n , b_n and c_n are the first input coefficients of atan2 of θ_1 , θ_3 , θ_5 , θ_7 in Eqs. (34) and (38), a_d , b_d and c_d are the second input coefficients of atan2 of θ_1 , θ_3 , θ_5 , θ_7 in Eqs. (34) and (38), and a , b and c are the coefficients of $\arccos$ of θ_2 , θ_6 in Eqs. (34) and (38).

3.1 Arctangent Function

Assuming $\mu = GC_k(a_n \sin \psi + b_n \cos \psi + c_n) = GC_k(a_d \sin \psi + b_d \cos \psi + c_d)$ by differentiating both sides of Eq. (3) with respect to ψ , we can get:

$$\frac{d\theta_i}{d\psi} = \frac{1}{1 + (u/v)^2} (u/v)' = \frac{u'v - uv'}{u^2 + v^2} = \frac{a_t \sin \psi + b_t \cos \psi + c_t}{u^2 + v^2} \quad (5)$$

where $a_t = c_n b_d - b_n c_d$, $b_t = a_n c_d - c_n a_d$, $c_t = a_n b_d - b_n a_d$. Using the Weierstrass-substitution method, the stationary point of Eq. (3) can be obtained:

$$\psi_0 = 2 \arctan \left(\frac{a_t \pm \sqrt{a_t^2 + b_t^2 - c_t^2}}{b_t - c_t} \right) \quad (6)$$

According to the relationship between $a_t^2 + b_t^2 - c_t^2$ and 0, Eq. (6) can be divided into three situations.

3.1.1 Two Stationary Points

When $a_t^2 + b_t^2 - c_t^2 > 0$, Eq. (3) has two stationary points. When the joint angle θ_i is between $[-\pi, \pi]$, the function graph changes continuously, as shown in Fig. 3. Because the atan2 function maps the joint angle to $[-\pi, \pi]$, when the joint angle θ_i exceeds $-\pi$ or π , the function graph mutates, as shown in Fig. 5.

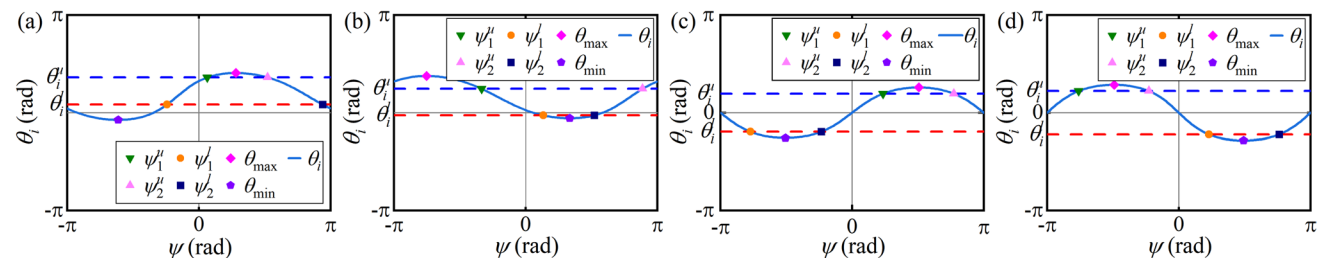


Fig. 3 Arctangent function, two stationary points, the function graph changes continuously, and the upper and lower limits of the joint intersect with the function graph. **a** Case 1. **b** Case 2. **c** Case 3. **d** Case 4

1) Function graph changes continuously

When the function graph of Eq. (3) changes continuously, the function values at the stationary points are the maximum value $\theta_{\max}$ and the minimum value $\theta_{\min}$ respectively. According to the relationship between $\theta_{\max}$, $\theta_{\min}$ and the upper limit θ_i^u , lower limit θ_i^l of joint i , the calculation of the feasible arm angle range Ψ_i of joint i can be divided into the following five situations:

$$(1) \theta_{\min} < \theta_i^l < \theta_{\max} \text{ and } \theta_{\min} < \theta_i^u < \theta_{\max}$$

At this time, both the upper and lower limits of the joint intersect with the function graph. Using the Weierstrass-substitution method for Eq. (3), we can get:

$$\psi(\theta_i) = 2 \arctan \left(\frac{-b_p \pm \sqrt{b_p^2 - 4a_p c_p}}{2a_p} \right) \quad (7)$$

where $a_p = GC_k((c_d - b_d) \tan \theta_i + (b_n - c_n))$, $b_p = 2GC_k(a_d \tan \theta_i - a_n)$, $c_p = GC_k((b_d + c_d) \tan \theta_i - (b_n + c_n))$. Bringing $\theta_i = \theta_i^l$ and $\theta_i = \theta_i^u$ into Eq. (7), we can get the intersection points ψ_1^l and ψ_2^l of the joint lower limit and the function graph, and ψ_1^l is on the left side of ψ_2^l ; and the intersection points ψ_1^u and ψ_2^u of the joint upper limit and the function graph, and ψ_1^u is on the left side of ψ_2^u . Figure 3 shows four possible situations. Equation (8) is the feasible arm angle range corresponding to Fig. 3.

$$\begin{cases} \text{Fig3(a): } \Psi_i = \Psi_{i1} \cup \Psi_{i2} = [\psi_1^l, \psi_1^u] \cup [\psi_2^l, \psi_2^u] \\ \text{Fig3(b): } \Psi_i = \Psi_{i1} \cup \Psi_{i2} = [\psi_1^u, \psi_1^l] \cup [\psi_2^l, \psi_2^u] \\ \text{Fig3(c): } \Psi_i = \Psi_{i1} \cup \Psi_{i2} \cup \Psi_{i3} = [-\pi, \psi_1^l] \cup [\psi_2^l, \psi_1^u] \cup [\psi_2^u, \pi] \\ \text{Fig3(d): } \Psi_i = \Psi_{i1} \cup \Psi_{i2} \cup \Psi_{i3} = [-\pi, \psi_1^u] \cup [\psi_2^l, \psi_1^l] \cup [\psi_2^u, \pi] \end{cases} \quad (8)$$

$$(2) \theta_{\min} < \theta_i^l < \theta_{\max} \text{ and } \theta_i^u > \theta_{\max}$$

At this time, the joint lower limit intersects with the function graph, According to the relationship between the maximum value point $\theta_{\max}$ and the intersection points ψ_1^l, ψ_2^l , the calculation of the feasible arm angle range can be divided into three situations. The feasible arm angle range corresponding to Fig. 4a and b is $\Psi_i = [-\pi\psi_1^l] \cup [\psi_2^l \pi]$, The feasible arm angle range corresponding to Fig. 4(c) is $\Psi_i = [-\psi_1^l\psi_2^l]$.

- (3) $\theta_i^l < \theta_{\min}$ and $\theta_{\min} < \theta_i^u < \theta_{\max}$

The upper limit of the joint intersects with the function graph, and the feasible arm angle range is $\Psi_i = [-\psi_1^u\psi_2^u]$ or $[-\pi\psi_1^u] \cup [\psi_2^u \pi]$.

- (4) $\theta_i^l < \theta_{\min}$ and $\theta_i^u > \theta_{\max}$

The feasible arm angle range is $\Psi_i = [-\pi, \pi]$.

- (5) $\theta_i^l > \theta_{\max}$ or $\theta_i^u < \theta_{\min}$

The feasible arm angle range is an empty set, that is, $\Psi_i = \emptyset$.

2) Function graph mutates

When the function graph of Eq. (3) mutates, the function values at the stationary points are not the maximum value $\theta_{\max}$ and the minimum value $\theta_{\min}$, but the local maximum value $\theta_{\max}^{\text{lo}}$ and the local minimum value $\theta_{\min}^{\text{lo}}$. According to the relationship between $\theta_{\max}^{\text{lo}}, \theta_{\min}^{\text{lo}}$ and the upper limit θ_i^u , lower limit θ_i^l of joint i , the calculation of the feasible arm angle range Ψ_i of joint i can be divided into the following six situations:

- (1) $\theta_i^l < \theta_{\max}^{\text{lo}}$ and $\theta_i^u > \theta_{\min}^{\text{lo}}$

At this time, the upper limit of the joint intersects with the upper half of the function graph, and the lower limit of the joint intersects with the lower half of the function graph. Figure 5 shows six possible situations. Equation (9) is the feasible arm angle range Ψ_i corresponding to Fig. 5. Only one possible situation is shown in Fig. 5b, c, e and f.

$$\begin{aligned} \text{Fig5(a): } \Psi_i &= [\psi_1^l \psi_2^l] \cup [\psi_1^u \psi_2^u] \\ \text{Fig5(b): } \Psi_i &= [\psi_1^l \psi_2^l] \cup [\psi_1^u \psi_2^u] \\ \text{or } \Psi_i &= [-\pi \psi_1^l] \cup [\psi_2^l \pi] \\ \text{Fig5(c): } \Psi_i &= [\psi_1^l \psi_2^l] \cup [\psi_1^u \psi_2^u] \\ \text{or } \Psi_i &= [-\pi \psi_1^u] \cup [\psi_2^u \pi] \\ \text{Fig5(d): } \Psi_i &= [\psi_1^u \psi_2^u] \cup [\psi_1^l \psi_2^l] \\ \text{Fig5(e): } \Psi_i &= [\psi_1^u \psi_2^u] \cup [\psi_1^l \psi_2^l] \\ \text{or } \Psi_i &= [-\pi \psi_1^l] \cup [\psi_2^l \pi] \\ \text{Fig5(f): } \Psi_i &= [\psi_1^u \psi_2^u] \cup [\psi_1^l \psi_2^l] \\ \text{or } \Psi_i &= [-\pi \psi_1^u] \cup [\psi_2^u \pi] \end{aligned} \quad (9)$$

- (2) $\theta_{\max}^{\text{lo}} < \theta_i^l < \theta_{\min}^{\text{lo}}$ and $\theta_i^u > \theta_{\min}^{\text{lo}}$

The upper limit of the joint intersects the upper part of the function graph, and the feasible arm angle range is $\Psi_i = [-\pi\psi_1^l] \cup [\psi_2^u \pi]$ or $[\psi_1^u\psi_2^u]$.

- (3) $\theta_i^l > \theta_{\min}^{\text{lo}}$ and $\theta_i^u > \theta_{\min}^{\text{lo}}$

Both the upper and lower limits of the joint intersect the upper part of the function graph, and the feasible arm angle range is $\Psi_i = [\psi_1^u\psi_2^u] \cup [\psi_2^l \psi_2^u]$ or $[\psi_1^l\psi_1^u] \cup [\psi_2^u \psi_2^l]$ or $[-\pi\psi_1^l] \cup [\psi_2^l \psi_1^u] \cup [\psi_2^u \pi]$ or $[-\pi\psi_1^u] \cup [\psi_2^u \psi_1^l] \cup [\psi_2^l \pi]$.

- (4) $\theta_i^l < \theta_{\max}^{\text{lo}}$ and $\theta_i^u < \theta_{\max}^{\text{lo}}$

Both the upper and lower limits of the joint intersect the lower part of the function graph, and the feasible arm angle range is $\Psi_i = [\psi_1^l\psi_1^u] \cup [\psi_2^u \psi_2^l]$ or $[\psi_1^u\psi_1^l] \cup [\psi_2^l \psi_2^u]$ or $[-\pi\psi_1^l] \cup [\psi_2^l \psi_1^u] \cup [\psi_2^u \pi]$ or $[-\pi\psi_1^u] \cup [\psi_2^u \psi_1^l] \cup [\psi_2^l \pi]$.

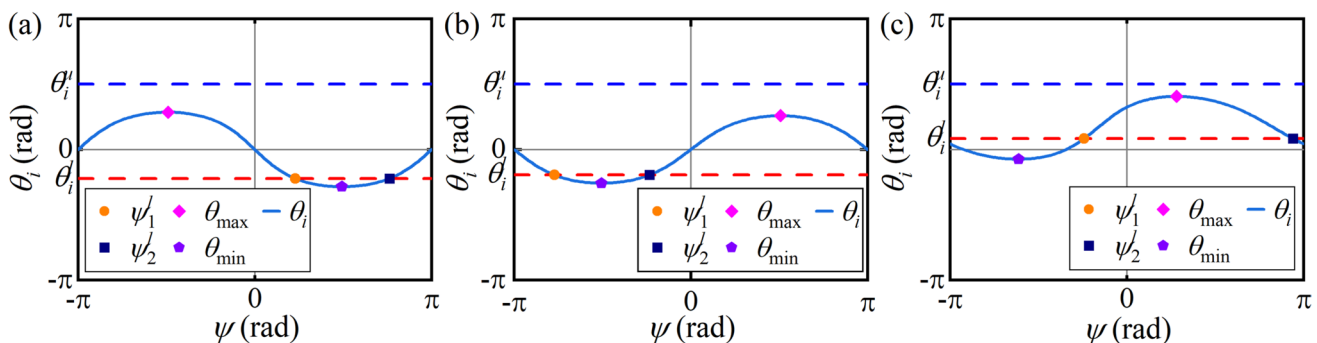


Fig. 4 Arctangent function, two stationary points, the function graph changes continuously, the joint lower limit intersects the function graph. **a** Case 1. **b** Case 2. **c** Case 3

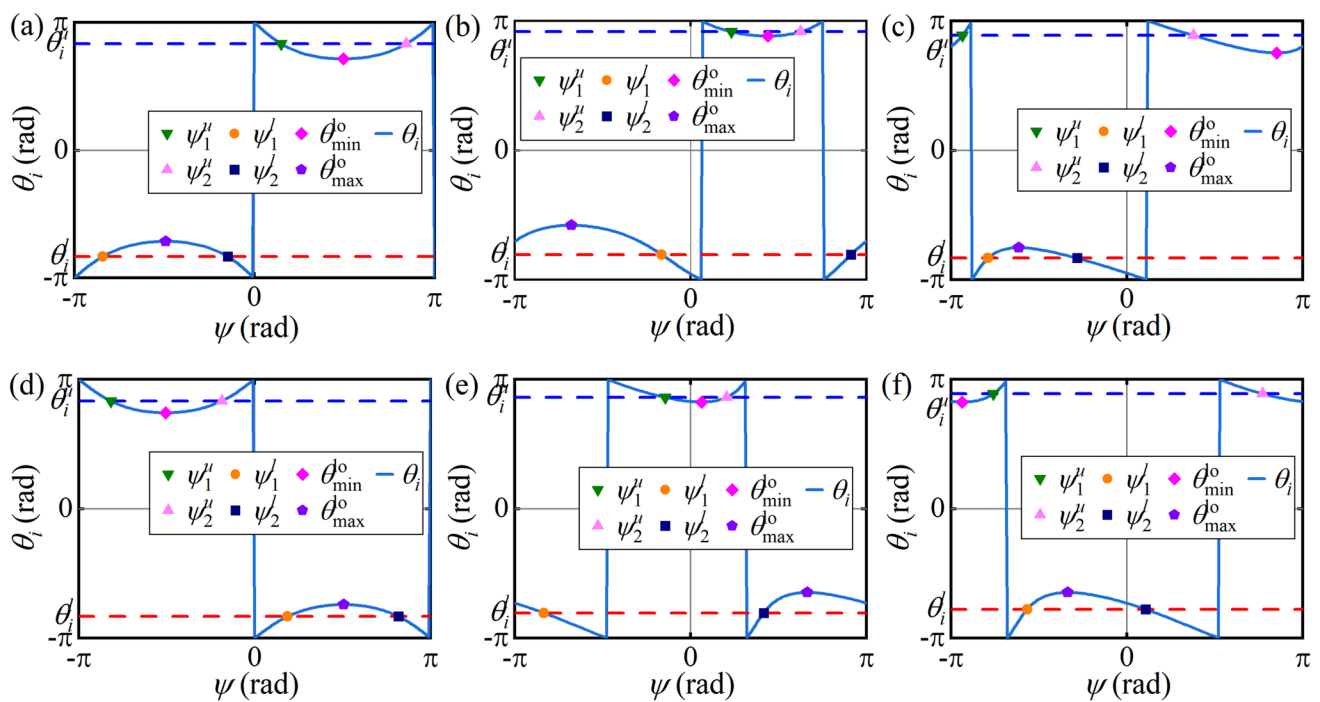


Fig. 5 Arctangent function, two stationary points, the function graph mutates. **a** Case 1. **b** Case 2. **c** Case 3. **d** Case 4. **e** Case 5. **f** Case 6

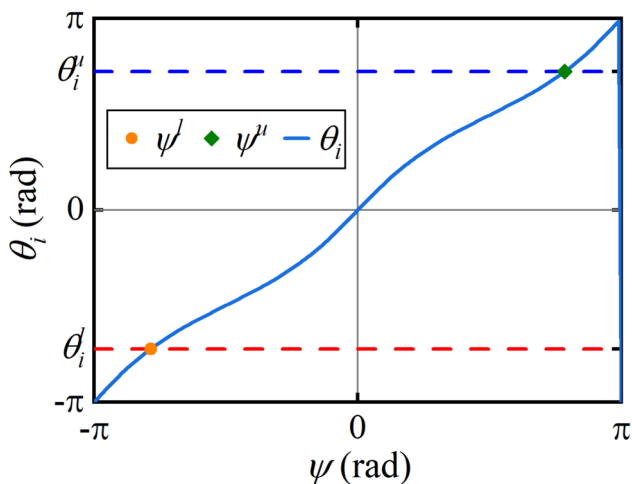


Fig. 6 Arctangent function, no stationary point, the function graph changes continuously

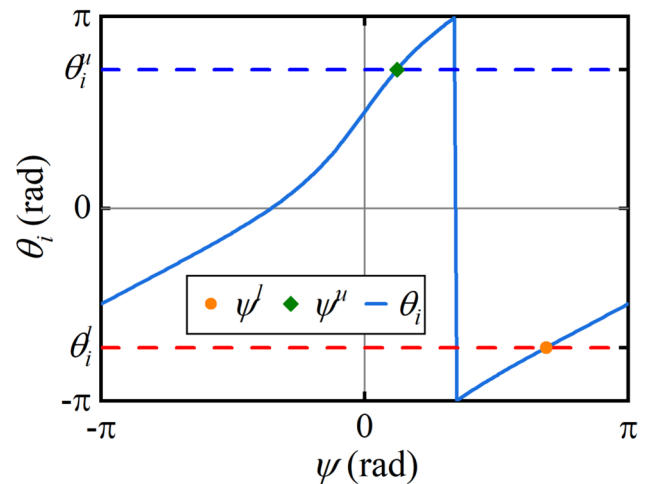


Fig. 7 Arctangent function, no stationary point, the function graph mutates

$$(5) \quad \theta_i^l < \theta_{max}^{lo} \text{ and } \theta_{max}^{lo} < \theta_i^u < \theta_{min}^{lo}$$

The lower limit of the joint intersects the lower half of the function graph, and the feasible arm angle range is $\Psi_i = [\psi_1^l \psi_2^l] \text{ or } [-\pi \psi_1^l] \cup [\psi_2^l \pi]$.

$$(6) \quad \theta_{max}^{lo} < \theta_i^l < \theta_{min}^{lo} \text{ and } \theta_{max}^{lo} < \theta_i^u < \theta_{min}^{lo}$$

The upper and lower limits of the joint do not intersect with the function graph, and the feasible arm angle range is an empty set, that is, $\Psi_i = \emptyset$.

3.1.2 Without Stationary Points

When $a_i^2 + b_i^2 - c_i^2 < 0$, Eq. (3) has no stationary point. When the joint angle θ_i is between $[-\pi, \pi]$, the function

graph changes continuously, as shown in Fig. 6. Because the atan2 function maps the joint angle to $[-\pi, \pi]$, when the joint angle θ_i exceeds $-\pi$ or π , the function graph mutates, as shown in Fig. 7.

1) Function graph changes continuously

When the function graph of Eq. (3) changes continuously, both the upper and lower limits of the joint intersect with the function graph. By bringing $\theta_i = \theta_i^l$ and $\theta_i = \theta_i^u$ into Eq. (7) respectively, we can obtain the intersection points ψ^l and ψ^u of the upper and lower limits of the joint and the function graph. If the function is a monotonically increasing function, the feasible arm angle range is $\Psi_i = [\psi^l, \psi^u]$, as shown in Fig. 6; if the function is a monotonically decreasing function, the feasible arm angle range is $\Psi_i = [\psi^u, \psi^l]$.

2) Function graph mutates

When the function graph of Eq. (3) mutates, both the upper and lower limits of the joint intersect with the function graph. When the functions on the left and right sides of the mutation are monotonically increasing functions, the feasible arm angle range is $\Psi_i = [\psi^l, \psi^u]$ or $[-\pi, \psi^u] \cup [\psi^l, \pi]$, as shown in Fig. 7. When the functions on the left and right sides of the mutation are monotonically decreasing functions, the feasible arm angle range is $\Psi_i = [\psi^u, \psi^l]$ or $[-\pi, \psi^l] \cup [\psi^u, \pi]$.

3.1.3 Singular Arm Angle

When $a_t^2 + b_t^2 - c_t^2 = 0$, Eq. (13) has a singular stationary point, as shown in Fig. 8. At this singular stationary point, the same arm angle corresponds to two joint angles. The value ψ_{sing} of the arm angle at the singular stationary point can be obtained from Eq. (10). The feasible arm angle range must be selected outside $[\psi_1^s, \psi_2^s]$, where $\psi_1^s = \psi_{\text{sing}} - \delta$, $\psi_2^s = \psi_{\text{sing}} + \delta$, $\delta = 1e-6$.

$$\psi_{\text{sing}} = 2 \arctan \left(\frac{a_t}{b_t - c_t} \right) \quad (10)$$

When Eq. (3) is an increasing function and when the singular arm angle is not at $\pm\pi$, depending on whether the function graph intersects with the upper and lower limits of the joint, the calculation of the feasible arm angle range can be divided into the following four situations:

- (1) The mutation of θ_i at singular arm angle is $-\pi$. The feasible arm angle range is $\Psi_i = \emptyset$ or $[\psi^u, \psi_1^s]$ or $[-\pi, \psi_1^s] \cup [\psi^l, \pi]$ or $[-\pi, \psi_1^s] \cup [\psi_2^s, \pi]$ or $[\psi^l, \psi^u]$ or $[-\pi, \psi^u] \cup [\psi^l, \pi]$ or $[-\pi, \psi^u] \cup [\psi_2^s, \pi]$ or $[\psi_2^s, \psi^l]$. Figure 8(a) shows one of the possible situations.
- (2) The mutation of θ_i at singular arm angle is π . The feasible arm angle range is $\Psi_i = [\psi^l, \psi^u]$ or $[\psi_2^s, \psi^u]$ or $[\psi^l, \psi_1^s] \cup [\psi_2^s, \psi^u]$ or $\emptyset$ or $[\psi^l, \psi_1^s]$. Figure 8(b) shows one of the possible situations.

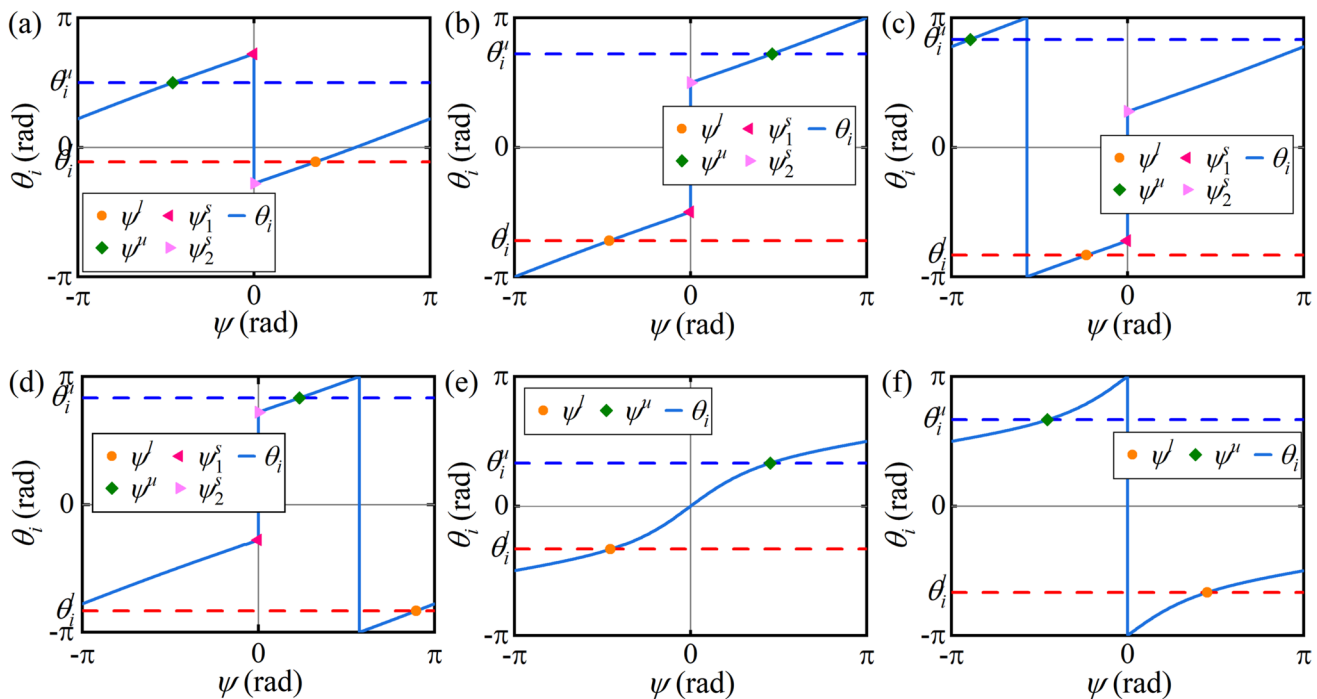


Fig. 8 Arctangent function, singular arm angle, increasing function. **a** Case 1. **b** Case 2. **c** Case 3. **d** Case 4. **e** Case 5. **f** Case 6

- (3) The mutation of θ_i at singular arm angle is π . When θ_i exceeds $-\pi$ or π , the function graph mutates again, and the mutation is to the left of the singular arm angle. The feasible arm angle range is $\Psi_i = [\psi^u \psi_1^s]$ or $[-\pi \psi^u] \cup [\psi^l \pi]$ or $[-\pi \psi^u] \cup [\psi_2^s \pi]$ or $[\psi^l \psi_1^s] \cup [\psi_2^s \psi^u]$ or $[-\pi \psi^u] \cup [\psi^l \psi_1^s] \cup [\psi_2^s \pi]$ or $[\psi_2^s \psi^u]$ or $\emptyset$ or $[\psi^l \psi_1^s]$. Figure 8(c) shows one of the possible situations.
- (4) The mutation of θ_i at singular arm angle is π . When θ_i exceeds $-\pi$ or π , the function graph mutates again, and the mutation is to the right of the singular arm angle. The feasible arm angle range is $\Psi_i = [\psi^u \psi_1^s]$ or $[\psi_2^s \psi^u]$ or $[\psi^l \psi_1^s] \cup [\psi_2^s \psi^u]$ or $[-\pi \psi^u] \cup [\psi_2^s \psi^u] \cup [\psi^l \pi]$ or $\emptyset$ or $[\psi^l \psi_1^s]$ or $[-\pi \psi^u] \cup [\psi^l \pi]$ or $[-\pi \psi^u] \cup [\psi^l \pi]$. Figure 8(d) shows one of the possible situations.

When Eq. (3) is an increasing function and when the singular arm angle is at $\pm\pi$, depending on whether the function graph intersects with the upper and lower limits of the joint, the calculation of the feasible arm angle range can be divided into the following two situations:

- 1) The function graph changes continuously. The feasible arm angle range is $\Psi_i = \emptyset$ or $[\psi^l \pi - \delta]$ or $[-\pi + \delta \pi - \delta]$ or $[\psi^l \psi^u]$ or $[-\pi + \delta \psi^u]$. Figure 8(e) shows one of the possible situations.
- 2) When θ_i exceeds $-\pi$ or π , the function graph mutates. The feasible arm angle range is $\Psi_i = [\psi^l \psi^u]$ or $[-\pi + \delta \psi^u]$ or $[-\pi + \delta \psi^u] \cup [\psi^l \pi - \delta]$ or $\emptyset$ or $[\psi^l \pi - \delta]$. Figure 8(f) shows one of the possible situations.

In the same way, the feasible arm angle range Ψ_i when Eq. (3) is a decreasing function can be obtained.

3.2 Arccosine Function

Assuming $m = a \sin \psi + b \cos \psi + c$, by differentiating ψ on both sides of Eq. (4), we can get:

$$\frac{d\theta_i}{d\psi} = -GC_k \frac{1}{\sqrt{1-m^2}} m' = -GC_k \frac{1}{\sin \theta_i} (a \cos \psi - b \sin \psi) \quad (11)$$

3.2.1 Two Stationary Points

When $\sin \theta_i \neq 0$, Eq. (4) has two stationary points, and the function values at the stationary points are the maximum value $\theta_{\max}$ and the minimum value $\theta_{\min}$ respectively, as shown in Fig. 9. Using the Weierstrass-substitution method, the stationary point calculation formula of Eq. (4) can be obtained:

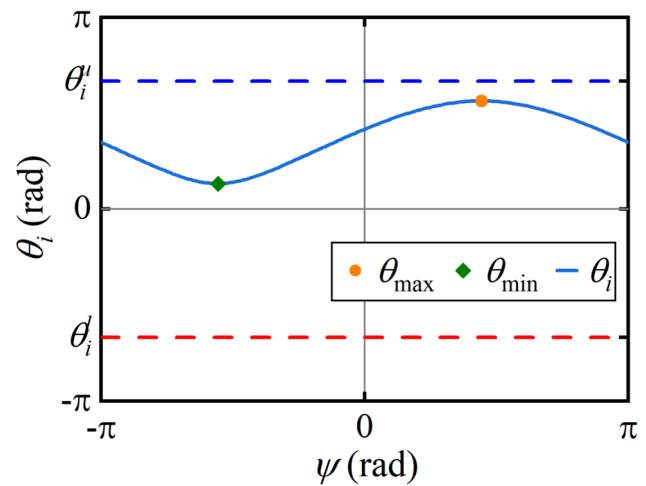


Fig. 9 Arccosine function, two stationary points

$$\psi_0 = 2 \arctan \left(\frac{-b \pm \sqrt{a^2 + b^2}}{a} \right) \quad (12)$$

Using the Weierstrass-substitution method, we get:

$$\psi(\theta_i) = 2 \arctan \left(\frac{a \pm \sqrt{a^2 + b^2 - (c - \cos \theta_i)^2}}{\cos \theta_i + b - c} \right) \quad (13)$$

By bringing $\theta_i = \theta_i^l$ and $\theta_i = \theta_i^u$ into Eq. (13), the intersection points of the upper and lower limits of the joint and the function graph can be obtained. According to the relationship between $\theta_{\max}$, $\theta_{\min}$ and θ_i^l , θ_i^u , the calculation of the feasible arm angle range Ψ_i can be divided into five situations. For specific methods, refer to the calculation method of the feasible arm angle Ψ_i in Section 3.1.1. Figure 9 shows one of the possible situations.

3.2.2 Singular Arm Angle

When $\sin \theta_i = 0$, Eq. (4) has one stationary point and one singular point, and the singular point is a singular configuration of the manipulator. It can be proved that when $\theta_i = 0$, $a^2 + b^2 - (c - 1)^2 = 0$ also satisfies, the value of the arm angle at the singular point can be calculated using Eq. (14), and the value of the arm angle at the stationary point can be calculated according to Eq. (12). When $\theta_i = \pi$, $a^2 + b^2 - (c + 1)^2 = 0$ also satisfies, the value of the arm angle at the singular point can be calculated using Eq. (15), and the value of the arm angle at the stationary point can be calculated according to Eq. (12). By bringing the arm angle value ψ_0 at the stationary point and the arm angle value ψ_{sing} at the singular point into Eq. (4) respectively, the corresponding joint angles θ_0 and θ_{sing} can be

obtained. According to the magnitude, θ_0 and θ_{sing} are set to the maximum value θ_{max} and the minimum value θ_{min} .

$$\psi_{\text{sing}} = 2 \arctan \left(\frac{a}{b - (c - 1)} \right) \quad (14)$$

$$\psi_{\text{sing}} = 2 \arctan \left(\frac{a}{b - (c + 1)} \right) \quad (15)$$

The value of the configuration parameter GC_k determines the distribution of the function graph. When $GC_k > 0$, the graph is distributed between $[0, \pi]$; when $GC_k < 0$, the graph is distributed between $[-\pi, 0]$. Figure 10 shows the function graph for $GC_k > 0$. Because the function graph is continuous at singular point, the influence of singular arm angle does not need to be considered when calculating the feasible arm angle range. According to the relationship between θ_{max} , θ_{min} and θ_i^l , θ_i^u , the calculation of the feasible arm angle range Ψ_i can be divided into five situations. For specific methods, refer to the calculation method of the feasible arm angle Ψ_i in Section 3.1.1.

4 Self-Motion Obstacle Avoidance

After the operation path in cartesian space is given, the operation path can be converted into a joint angle path through inverse kinematics. As shown in Fig. 2, since the self-motion

of the manipulator is a redundant circle rotating around the line SW at point E, the arm angle space is discretized within the range of $[-\pi, \pi]$. Through the inverse kinematics solution, the joint angle corresponding to the discretized arm angle is obtained. By performing collision detection on the entire manipulator, the points E_1 and E_2 where the manipulator begins to collide with the obstacle can be obtained. The corresponding arm angles of E_1 and E_2 are respectively ψ_1^{obs}

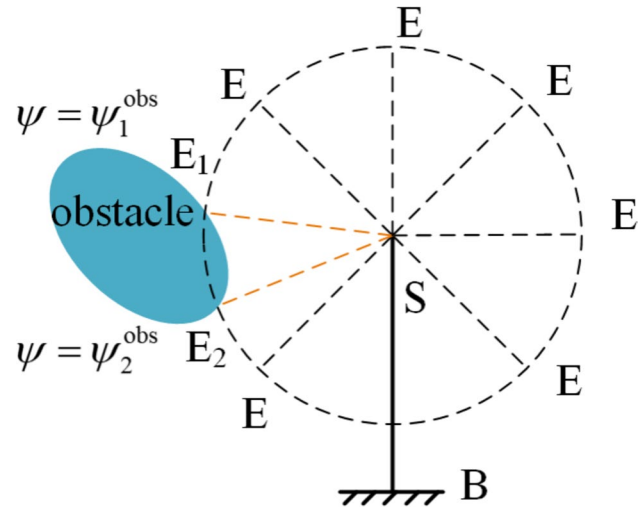


Fig. 11 Arm angle range under self-motion obstacle avoidance conditions

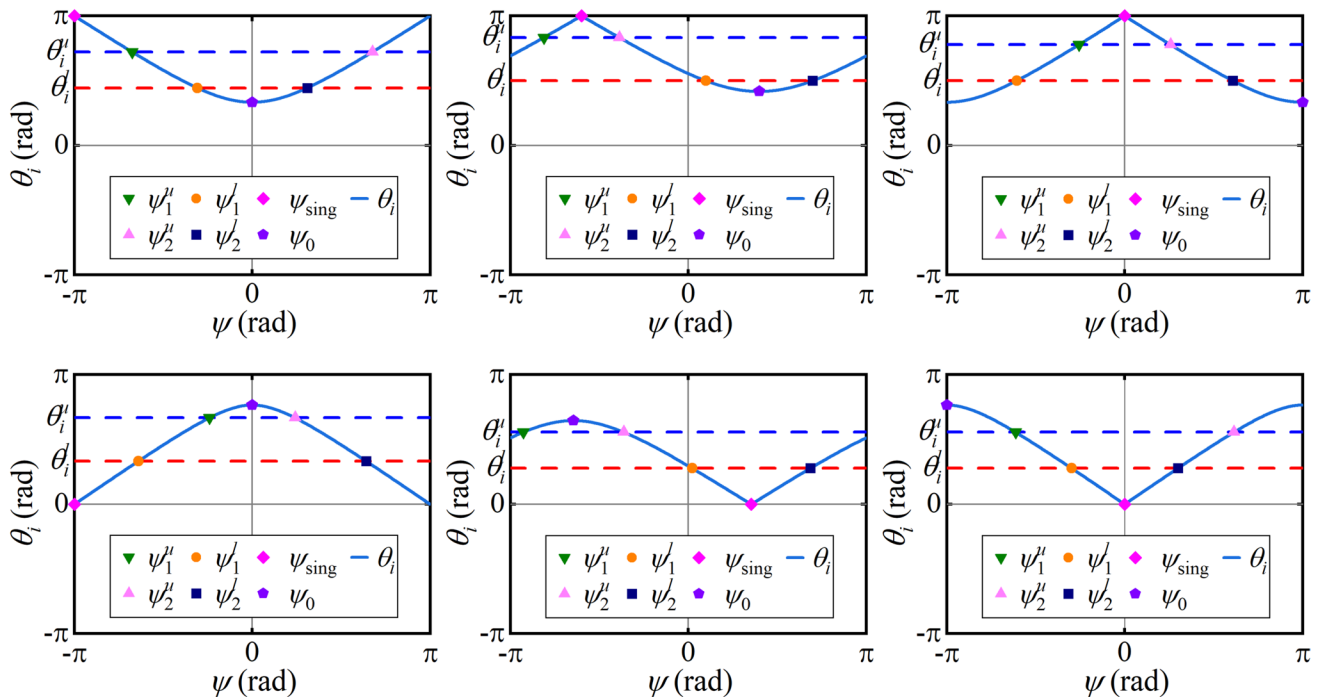


Fig. 10 Arccosine function, singular arm angle, $GC_k > 0$

and ψ_2^{obs} . Then the value range of the arm angle under self-motion obstacle avoidance conditions is $\psi_{\text{obs}} = [\psi_1^{\text{obs}}, \psi_2^{\text{obs}}]$, as shown in Fig. 11.

5 Optimal Arm Angle

The selection of the arm angle is particularly important. When the end pose of the manipulator is determined, the arm angle determines the angles of each joint. Reasonable selection of the arm angle can shorten the movement time of the manipulator and improve the operating efficiency of the manipulator. According to the arm angle range Ψ_{lim} under joint limit avoidance, singularity avoidance conditions and the arm angle range Ψ_{obs} under self-motion obstacle avoidance condition, the arm angle range $\Psi = \Psi_{\text{lim}} \cup \Psi_{\text{obs}}$ can be obtained by intersecting the two arm angle ranges. To improve the operating efficiency of the manipulator, it is necessary to select a set of optimal arm angles corresponding to each path point. This article takes the total angular displacement of the joints of the manipulator as the optimization objective, and the objective function can be expressed as:

$$f = \sum_{i=1}^{n-1} \text{dis}_i^{i+1} \text{ and } \text{dis}_i^{i+1} = \sum_{j=1}^7 (q_{i+1}^j - q_i^j)^2 \quad (16)$$

where n is the number of path points in the manipulator's operating path, dis_i^{i+1} is the angular displacement between two adjacent path points, q_i^j is the angle of the j -th joint corresponding to the i -th path point.

The particle swarm optimization (PSO) algorithm originated from the study of the predatory behavior of bird flocks. Its basic idea is to find the optimal solution through collaboration and information sharing among individuals [30]. The PSO algorithm has the advantages of being insensitive to initial values, fast in convergence speed, and with few control variables. Therefore, the PSO algorithm is used to solve the arm angle corresponding to each path point in the operation path. Since the arm angle range is in the form of a union of multiple subintervals, to prevent the algorithm from converging to a local optimum, the PSO algorithm is improved in this paper. When initializing particle positions, setting particle velocity and position boundaries, and updating particle velocity and position, calculations are performed within the arm angle range in the form of merged subintervals. When calculating fitness, calculations are performed within the arm angle range in the form of separate subintervals.

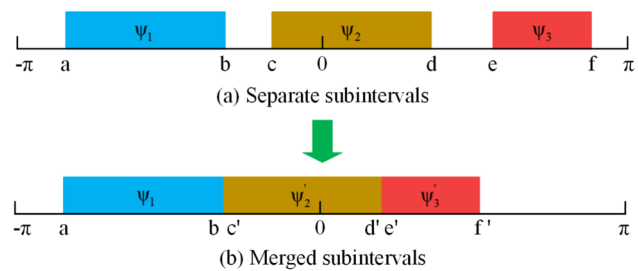


Fig. 12 Arm angle range in the form of merged subintervals and separated subintervals

Figure 12 shows the arm angle ranges in the form of merged subintervals and separate subintervals. Assume that the range of the arm angle is $\psi = \psi_1 \cup \psi_2 \cup \psi_3$, keep the sub-interval ψ_1 unchanged, place the starting point of the sub-interval ψ_2 at the end point b of the sub-interval ψ_1 , and place the starting point of the sub-interval ψ_3 at the end point d' of the sub-interval ψ_2' . The subintervals ψ_1 , ψ_2' , and ψ_3' together form the subinterval $\psi' = \psi_1 \cup \psi_2' \cup \psi_3'$ in the merged form. Set point a to the minimum value $x_{\min}$ of the particle position, and set point f' to the maximum value $x_{\max}$ of the particle position. Set the maximum value $v_{\max}$ and minimum value $v_{\min}$ of particle speed according to Eq. (17), where α is the speed coefficient.

$$\begin{cases} v_{\min} = -\alpha(x_{\max} - x_{\min}) \\ v_{\max} = \alpha(x_{\max} - x_{\min}) \end{cases} \quad (17)$$

When initializing the particle position, to ensure that each subinterval of the arm angle range contains particles, the number n_i of the population contained in the subinterval is determined based on the length of the subinterval, and then the population is randomly generated within each subinterval.

$$n_i = \left(l_i / \sum_{i=1}^k l_i \right) * n \quad (18)$$

where k is the number of subintervals, l_i is the length of the i -th subinterval, and n is the number of populations. The velocity of the particle is initialized using Eq. (19).

$$v_{i,j}(0) = v_{\min} + \text{rand}(* (v_{\max} - v_{\min})) \quad (19)$$

where $v_{i,j}(0)$ is the initial velocity of the i -th particle on the j -th dimension, and $\text{rand}()$ is a random number between 0 and 1. The velocity of the particle is updated using Eq. (20), and if the velocity of the particle exceeds the boundary, the speed needs to be limited.

$$\begin{cases} v_{ij}(t+1) = \omega v_{ij}(t) + c_1 * \text{rand}() * (pbest_{ij} - x_{ij}(t)) + c_2 * \text{rand}() * (gbest_{ij} - x_{ij}(t)) \\ v_{ij}(t+1) = v_{\max}, \text{ if } v_{ij}(t+1) > v_{\max} \\ v_{ij}(t+1) = v_{\min}, \text{ if } v_{ij}(t+1) < v_{\min} \end{cases} \quad (20)$$

where $v_{ij}(t)$, $x_{ij}(t)$ are respectively the speed and position of the i -th particle on the j -th dimension in the t -th iteration, ω is the inertia weight, c_1 and c_2 are the learning factor, $pbest_{ij}$ is the optimal position of the i -th particle, $gbest_{ij}$ is the optimal position of the population. The position of the particle is updated using Eq. (21). If the position of the particle is outside the arm angle range of the merged subinterval form, the out-of-range particle also needs to be processed.

$$\begin{cases} x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \\ x_{ij}(t+1) = x_{\max}, \text{ if } x_{ij}(t+1) > x_{\max} \\ x_{ij}(t+1) = x_{\min}, \text{ if } x_{ij}(t+1) < x_{\min} \end{cases} \quad (21)$$

Table 2 shows the pseudo-code of the arm angle optimization method based on the improved PSO algorithm under the conditions of combining joint limit avoidance, singularity avoidance, and self-motion obstacle avoidance. Steps 01, 02, 05, and 06 are calculated using the arm angle range in the form of subinterval merging. To truly reflect the changes in the objective function, steps 03 and 07 use the arm angle range in the form of subinterval separation for calculation.

Table 2 Arm angle optimization method

Algorithm: Arm angle optimization method based on improved PSO

Input: Feasible arm angle range Ψ

01: Determine n_i according to Eq. (18) and randomly generate the initial position of the particle

02: Initialize the particle speed according to Eq. (19)

03: Calculate the fitness of each particle according to Eq. (16)

04: Determine the population optimal particle $gbest$ and the individual optimal particle $pbest$

While not stop doing

05: Update the particle speed according to Eq. (20)

06: Update the particle position according to Eq. (21)

07: Calculate the fitness of each particle according to Eq. (16)

08: Update the population optimal particle $gbest$ and the individual optimal particle $pbest$

End while

Output: optimal arm angle corresponding to each path point

6 Algorithm Verification

6.1 Verification of Joint Limit Avoidance and Singularity Avoidance Algorithm

To verify the effectiveness of the joint limit avoidance and singularity avoidance algorithm proposed in this article, a set of joint variables $[10^\circ \ 0^\circ \ 30^\circ \ -60^\circ \ 175^\circ \ 80^\circ \ 70^\circ]$ that exceed the joint limit and are in a singular configuration are set. Where joint 5 exceeds the limit, and the angle of joint 2 is 0, the manipulator is 2-axis singular. The manipulator end pose matrix corresponding to this joint variable is:

$${}^0_7\mathbf{T} = \begin{bmatrix} 0.7678 & -0.3349 & -0.5461 & -379.51 \\ -0.5710 & -0.7444 & -0.3462 & -290.40 \\ -0.2906 & 0.5777 & -0.7628 & 727.57 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (22)$$

Set the configuration parameters to $GC_2 = 1$, $GC_4 = 1$, $GC_6 = 1$. According to the algorithm proposed in Section 3, the arm angle range corresponding to each joint can be obtained. By finding the intersection of the feasible arm angle ranges of each joint, the arm angle range that avoids joint limit and avoids singularity in the current pose can be obtained. Table 3 lists the arm angle range corresponding to each joint. By discretizing the arm angle in the interval $[-\pi, \pi]$ and bringing the arm angle into the inverse kinematics solution, the joint angles corresponding to the arm angle

Table 3 Arm angle range corresponding to each joint

Joint	θ_1	θ_3	θ_5	θ_7	θ_2	θ_6
Arm angle range	$[-\pi, -1e-6] \cup [1e-6, 1.527] \cup [2.205, \pi]$	$[-2.835, -1e-6] \cup [1e-6, 2.835]$	$[-\pi, -0.277] \cup [0.092, \pi]$	$[-\pi, \pi]$	$[-\pi, \pi]$	$[-2.072, 1.887]$
Intersection	$[-2.072, -0.277] \cup [0.092, 1.527]$					

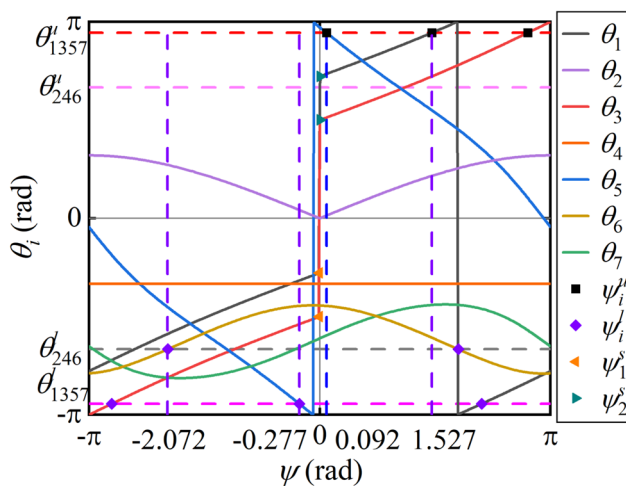


Fig. 13 The change curve of each joint angle θ_i when the arm angle ψ changes

can be obtained. Figure 13 shows the change curve of each joint angle θ_i when the arm angle ψ changes. It can be seen from the figure that in the interval $[-2.063, -1e-6] \cup [1e-6, 1.867]$, the angle values of joints 1, 2, 3, 5, 6, and 7 do not exceed the limits, and at the same time, singular configurations are avoided, which proves that the joint limit avoidance and singularity avoidance algorithm proposed in the article is effective.

6.2 Optimal Path

To further verify the effectiveness of the proposed algorithm, two scenarios, pick-and-place and welding, are designed, as shown in Fig. 14. To maximize utility to the characteristics of the 7-DOF manipulator, the manipulator in Fig. 14 (a) is arranged on the side of the work scene, thereby saving space in front and making the work scene more compact.

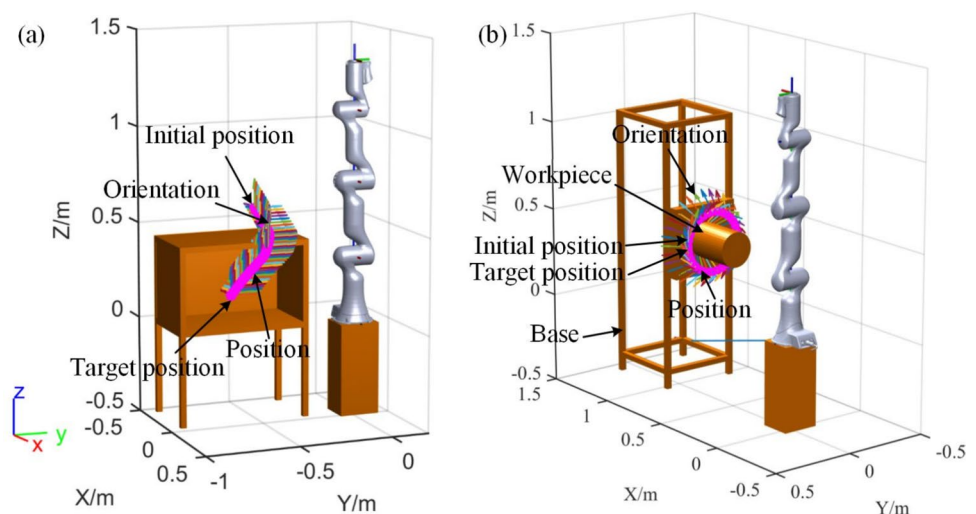
Through its own flexible movement, the 7-DOF manipulator can replace the traditional displacement mechanism and adjust the welding gun angle, thereby reducing hardware investment and cost. Using the 7-DOF manipulator for welding operations can not only improve the reliability of the system, but also enhance the flexibility and automation level of production. To this end, this paper designs the welding scene shown in Fig. 14 (b), in which the workpieces to be welded include plates and cylinders.

OMPL (Open Motion Planning Library) is an open source motion planning library that is mainly used to provide efficient path planning algorithms for robots and other moving objects. It supports a variety of common motion planning algorithms, such as RRT, RRT*, PRM, etc., and can be integrated with other robot frameworks (such as ROS, MoveIt!, etc.). The goal of OMPL design is to provide a flexible and efficient path planning solution and support motion planning in high-dimensional space. The RRT* algorithm in OMPL is used to obtain the path information in the cartesian space in Fig. 14 (a), and the path information in the cartesian space in Fig. 14 (b) is obtained through the computer-aided manufacturing (CAM) method. According to the pose constraints of the end of the manipulator in the task, the orientation matrix corresponding to each position point is generated.

6.2.1 Arm Angle Range Under Joint Limit Avoidance, Singularity Avoidance, and Self-Motion Obstacle Avoidance Conditions

According to Section 3, the arm angle range $\Psi_{\lim}$ under joint limit avoidance and singularity avoidance conditions can be obtained. According to Section 4, the arm angle range Ψ_{obs} under self-motion obstacle avoidance conditions can be obtained. By finding the intersection of $\Psi_{\lim}$ and Ψ_{obs} , the final arm angle range Ψ can be obtained, as shown in Figs. 15 and 17. According to the value of the configuration

Fig. 14 7-DOF manipulator arm angle optimization algorithm verification scene. **a** Pick-and-place scene. (2) Welding scene



parameter GC_k , the arm angle range can be divided into eight cases. Table 4 lists the GC_k values corresponding to the eight cases.

6.2.2 Optimal Arm Angle

Using the improved PSO algorithm proposed in Section 5, the optimal arm angle ψ_{op} corresponding to each path point

of the operation path can be obtained, as shown in Figs. 15 and 17. According to the optimal arm angle ψ_{op} , the configuration parameters GC_k and the pose of each path point, the inverse kinematics solution can be used to obtain the joint angles corresponding to the optimal arm angle, as shown in Figs. 16 and 18. The optimal arm angle distribution in Fig. 15 is stable and there is no sudden change, so the corresponding joint angles in Fig. 16 also remain stable and do

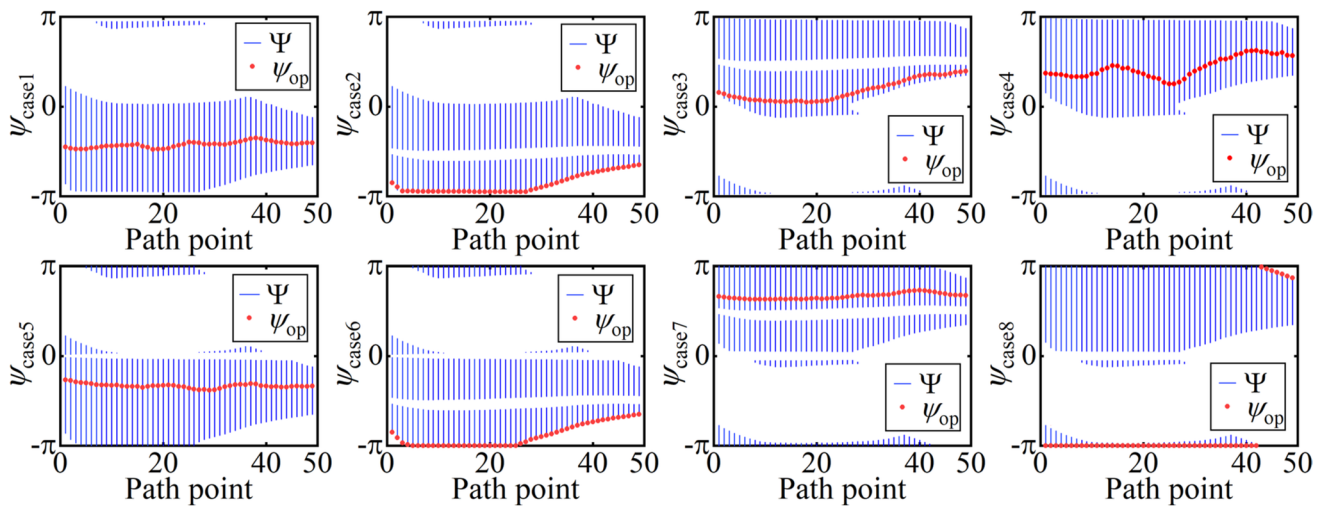


Fig. 15 Optimal arm angle ψ_{op} for each path point of the pick-and-place operation path

Table 4 Values of GC_k corresponding to eight situations

	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7	Case 8
GC_2	1	1	1	1	-1	-1	-1	-1
GC_4	1	1	-1	-1	1	1	-1	-1
GC_6	1	-1	1	-1	1	-1	1	-1

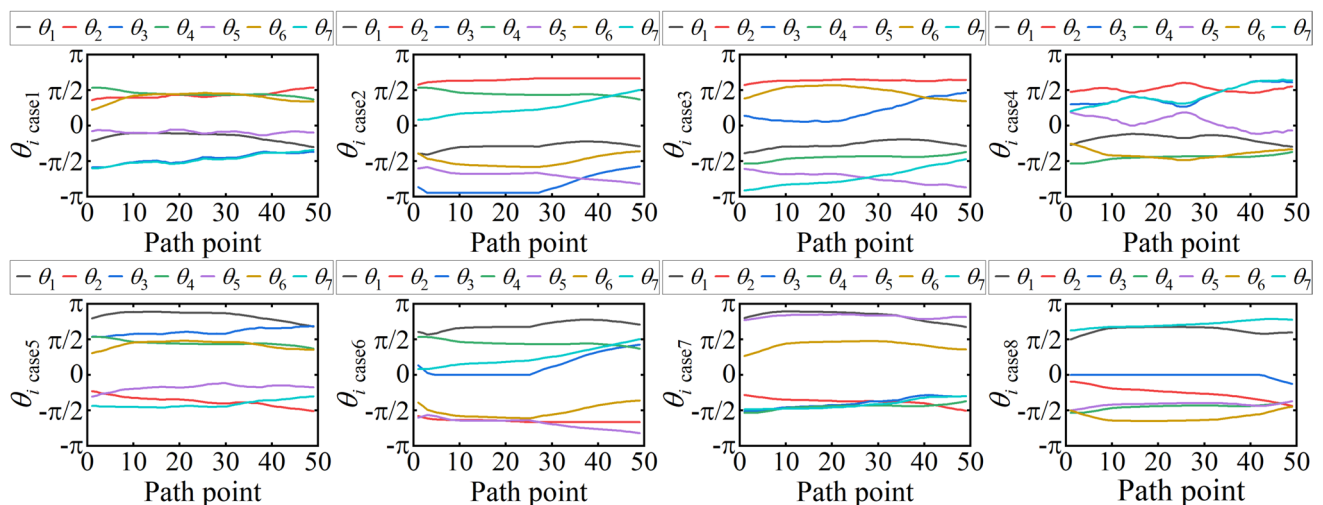


Fig. 16 The joint angles θ_i corresponding to the optimal arm angle ψ_{op} in the pick-and-place operation path

not exceed the upper and lower limits. In contrast, the optimal arm angle of case5, case6, case7 and case8 in Fig. 17 suddenly change, resulting in a sudden change in the corresponding joint angles in Fig. 18, but the angles of each joint are still within the upper and lower limits.

To further verify the superiority of the proposed method, the total angular displacement of the joints of the

manipulator in the pick-and-place experiment is selected for comparative analysis. RRT* is an improved RRT algorithm that can generate asymptotically optimal paths while ensuring computational efficiency. Compared with the standard RRT, RRT* not only expands the tree structure during the sampling process, but also continuously optimizes the path and gradually approaches the optimal solution. To this end,

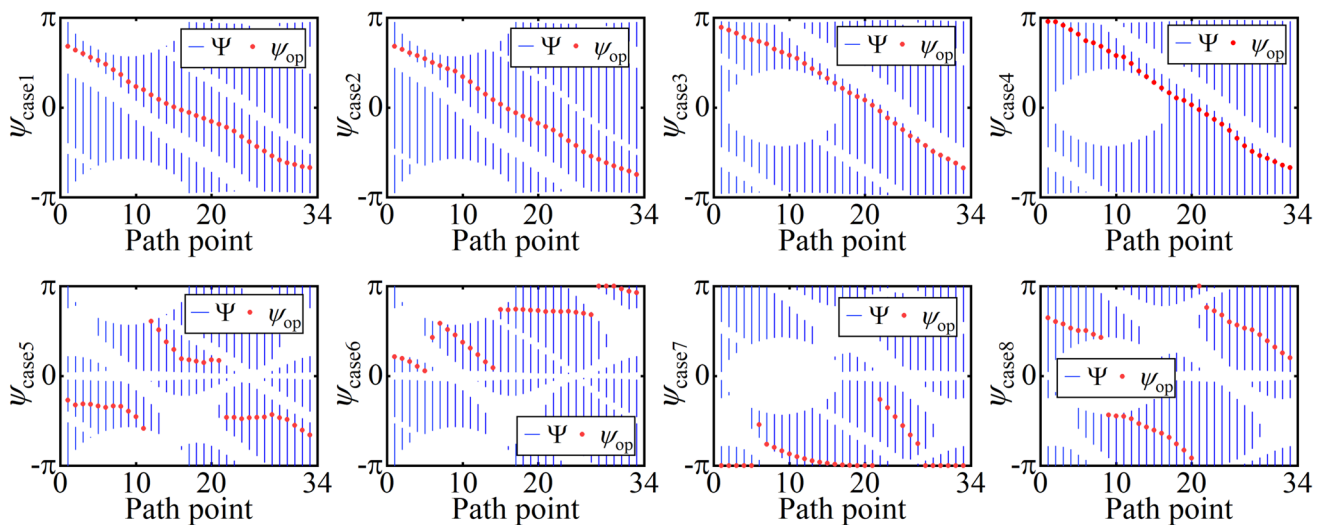


Fig. 17 Optimal arm angle ψ_{op} for each path point of the welding operation path

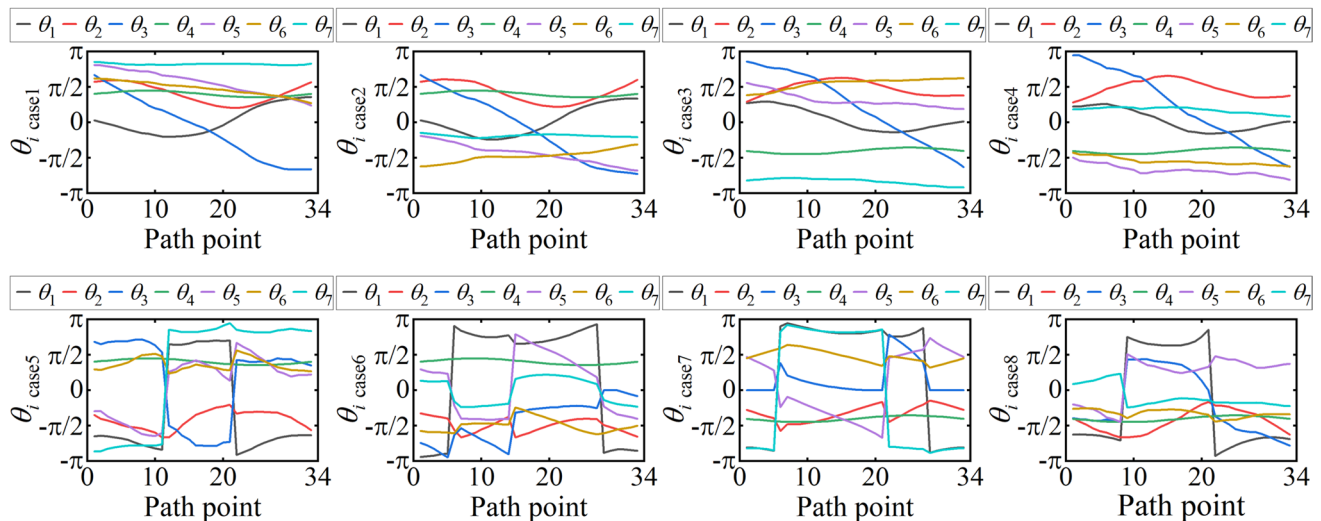


Fig. 18 The joint angles θ_i corresponding to the optimal arm angle ψ_{op} in the welding operation path

Table 5 Total angular displacement of the joint paths generated by the proposed method and the RRT* algorithm

	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7	Case 8
Proposed method/arc	6.99	6.74	7.41	11.51	6.19	7.45	5.44	5.28
RRT/arc	17.63	10.21	10.14	37.69	38.40	11.51	9.94	32.46
RRT*/arc	5.11	6.16	6.26	6.93	5.82	5.72	6.05	6.02

this paper uses the standard RRT and RRT* in OMPL to plan paths for the pick-and-place scene. Table 5 summarizes the total angular displacement of the joint paths generated by the proposed method, RRT, and RRT*. It can be seen from the data in the table that compared with the standard RRT method, the proposed method significantly reduces the total angular displacement of the joints. Compared with the RRT* method, in case 7 and case 8, the paths generated by the proposed method are better than the RRT* algorithm.

To verify the repeatability and accuracy of the improved PSO algorithm proposed in this paper, Case 1 in Fig. 15 is selected for testing. The improved PSO algorithm is run 100 times, and the calculation results are statistically calculated. Figure 19 shows the distribution of optimal arm angle corresponding to each calculation result, where $\psi_{\max}$ is the optimal arm angle corresponding to the maximum fitness, and $\psi_{\min}$ is the optimal arm angle corresponding to the minimum fitness. It can be seen from the figure that the calculation results are distributed more evenly, showing good stability, and the difference in optimal arm angles corresponding to the maximum fitness and the minimum fitness is small. This shows that the improved PSO algorithm proposed in this paper has strong robustness.

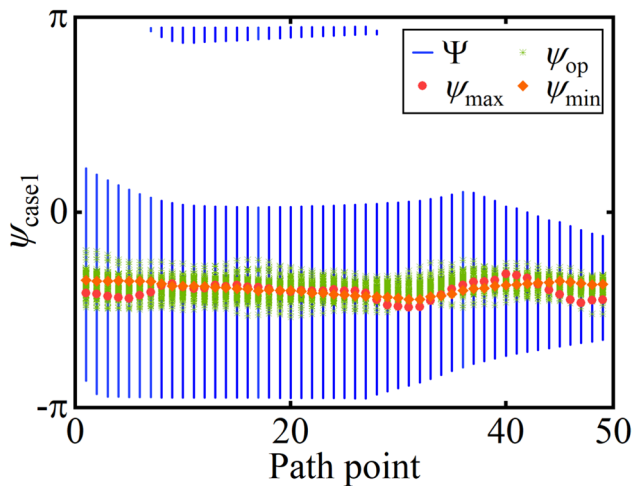


Fig. 19 Statistical results of the optimal arm angle ψ_{op}

If the feasible arm angle range is not processed by merging subintervals, the method proposed in this paper may converge to the local optimal solution, resulting in the failure to adopt the optimal operation path when the task is executed. Taking case 1 in Fig. 17 as an example, the optimal fitness of case 1 is 4465, while the optimal fitness in Fig. 20 is 109,544 and 263,859 respectively. This paper adopts the objective function (Eq. (16)) as the optimal fitness of the PSO algorithm, and the corresponding total joint angular displacements are 66.82° , 330.97° and 513.67° respectively. This shows that by merging the intervals to process the feasible arm angle range, the algorithm can effectively avoid converging to the local optimum, thereby more accurately finding the global optimal arm angle.

By adjusting the optimization target (Eq. (16)), the proposed method can be directly extended to single-objective optimization scenarios, such as minimizing the energy consumption of the operation path (space manipulator) or optimizing the joint torque (grinding process). In addition, combined with an appropriate multi-objective optimization algorithm, this method can also be used to simultaneously optimize multi-objective requirements such as minimum total joint displacement, minimum energy consumption, and optimal joint torque.

6.2.3 Simulation and Real Machine Verification

The operation paths corresponding to the optimal arm angle of Case 1 in Figs. 15 and 17 are selected for verification. First, simulation experiments are carried out in the MATLAB robotics system toolbox, and then real machine experiments are carried out on the manipulator. Figure 21 shows the simulation experiments of the operation paths of the pick-and-place and welding scenarios. Figures 22 and 23 show the real machine experiments of the operation paths of the pick-and-place and welding scenarios, respectively. It can be seen from the figures that the operation paths after the arm angle are optimized can run stably on the manipulator, and no collision occurs at each path point. This shows that the method proposed in this paper selects the arm angle and configuration parameters as the auxiliary control parameters of the manipulator, so that the joints of each path point on

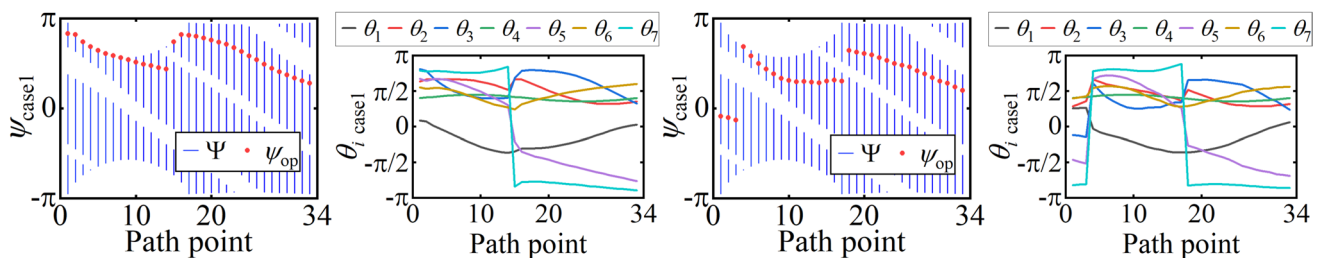


Fig. 20 Optimal arm angle ψ_{op} calculated by the unoptimized PSO algorithm

Fig. 21 Simulation experiments of the operation paths. **a** Pick-and-place scene. (2) Welding scene

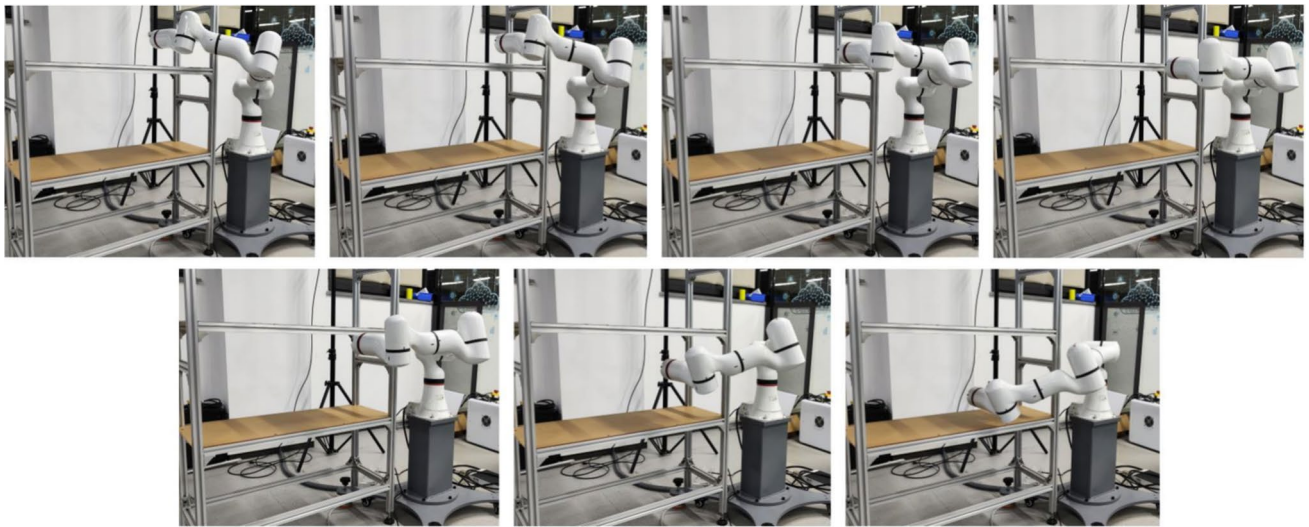
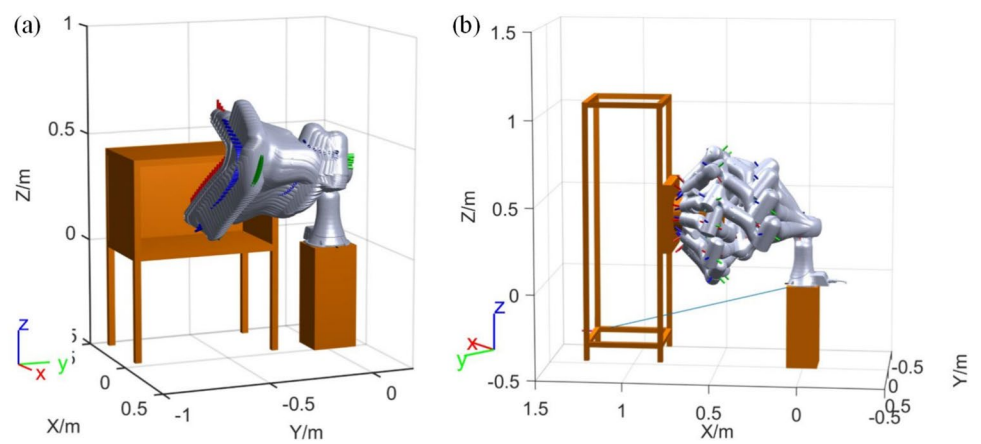


Fig. 22 The real machine experiments of the pick-and-place operation path

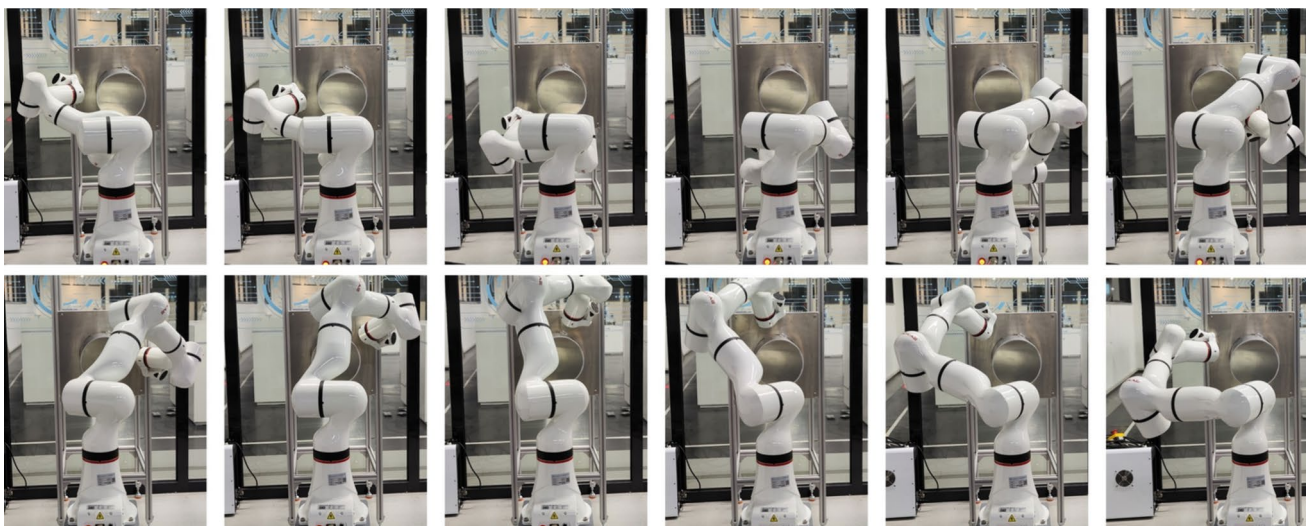


Fig. 23 The real machine experiments of the welding operation path

the operation path meet the requirements of avoiding joint limits, avoiding singularities, and avoiding obstacles by self-motion, and the total angular displacement of the joints of the operation path is the shortest.

7 Conclusion

In this paper, arm angle parameterization is used to characterize the redundant DOF of the S-R-S configuration 7-DOF manipulator. The analytical inverse kinematics solution is combined with the self-motion characteristics to propose a path planning method that explicitly utilizes the redundant characteristics of the manipulator. Based on the initial operation path given in cartesian space, this method takes minimizing the total angular displacement of each joint in the operation path as the optimization goal. Combined with the improved PSO algorithm, the optimal arm angle that meets the requirements of joint limit avoidance, singularity avoidance and self-motion obstacle avoidance is obtained, so that the manipulator can efficiently complete the task with the optimal path.

By adjusting the optimization goal (Eq. (16)), the method in this paper can be directly extended to single-objective optimization scenarios, such as minimizing the energy consumption of the operation path (space manipulator) or optimizing the joint torque (grinding process). In addition, combined with an appropriate multi-objective optimization algorithm, this method can also be used to simultaneously optimize multiple objectives such as minimum total joint displacement, minimum energy consumption and optimal joint torque.

Intelligent optimization algorithms (such as PSO algorithm, genetic algorithm, simulated annealing algorithm) have the advantages of global search capability, strong adaptability and independence from gradient information. However, especially when dealing with large-scale problems, the algorithm converges slowly and cannot guarantee that the global optimal solution is always found. In order to avoid the improved PSO algorithm proposed in this paper from falling into the local optimal solution when dealing with complex paths, subsequent research will use branch and bound method or graph search method to solve the optimal arm angle, and perform path planning based on the optimal arm angle.

According to the principle of bionics, the human arm has seven degrees of freedom and is the most flexible physiological structure. To simulate the function of the human arm, the 7-DOF manipulator and the arm of the humanoid robot are usually designed with the same S-R-S configuration as the human arm. Therefore, this paper takes the 7-DOF manipulator with S-R-S configuration as the research object, explores its analytical inverse kinematics solution,

and uses the arm angle parameterization to represent the redundant DOF of the manipulator. Given an initial working path in Cartesian space, the path planning problem of a 7-DOF manipulator under constraints (including obstacles, joint limits, singular configurations and working efficiency) is studied with the arm angle as the optimization object. Although the research results of this paper are applicable to most 7-DOF manipulators, the configuration change will lead to the change of the analytical inverse kinematics solution, so the proposed method cannot be directly applied to manipulators of other configurations. However, considering the universality of the method and ideas, this study can provide a useful reference for path planning of other types of 7-DOF manipulators.

Appendix: Inverse Kinematics Solution

Forward Kinematics Model

According to the kinematic parameters in Table 1, the forward kinematics model of the manipulator can be obtained:

$${}_0^7\mathbf{T} = {}_0^1\mathbf{T}_1 {}_1^2\mathbf{T}_2 {}_2^3\mathbf{T}_3 {}_3^4\mathbf{T}_4 {}_4^5\mathbf{T}_5 {}_5^6\mathbf{T}_6 {}_6^7\mathbf{T}_7 = \begin{bmatrix} {}_0^7\mathbf{R} & {}_0^7\mathbf{P} \\ 0 & 1 \end{bmatrix} \quad (23)$$

where ${}_0^7\mathbf{R}$ and ${}_0^7\mathbf{P}$ are respectively the orientation and position matrices of the tool coordinate system relative to the base coordinate system, ${}_{i-1}^i\mathbf{T}$ is the pose matrix of coordinate system $\{i\}$ relative to coordinate system $\{i-1\}$, and ${}_{i-1}^i\mathbf{T}$ can be expressed as:

$${}_{i-1}^i\mathbf{T} = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & a_{i-1} \\ \sin \theta_i \cos \alpha_{i-1} & \cos \theta_i \cos \alpha_{i-1} & -\sin \alpha_{i-1} & -\sin \alpha_{i-1} d_i \\ \sin \theta_i \sin \alpha_{i-1} & \cos \theta_i \sin \alpha_{i-1} & \cos \alpha_{i-1} & \cos \alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (24)$$

Virtual Joints

When the angle of joint 3 is zero, the 7-DOF manipulator degenerates into a 6-DOF manipulator. The first four joints of the manipulator at this time are called virtual joints, as shown in Fig. 24.

Given the pose $\mathbf{P}$ of the end of the manipulator, according to the kinematic parameters of the manipulator, we can obtain the coordinate $\mathbf{S} = [0, 0, d_{bs}]^T$ of the shoulder point S and the coordinate $\mathbf{W} = \mathbf{P} * \mathbf{dis}$ of the wrist point W, where $\mathbf{dis} = [0, 0, -d_{wt}]^T$ is the distance moved by the point T in the tool coordinate system along the TW direction. From this, the vector $\mathbf{SW} = \mathbf{W} - \mathbf{S}$ can be obtained. In triangle SE^vW, according to the cosine theorem, we can get θ_4^v :

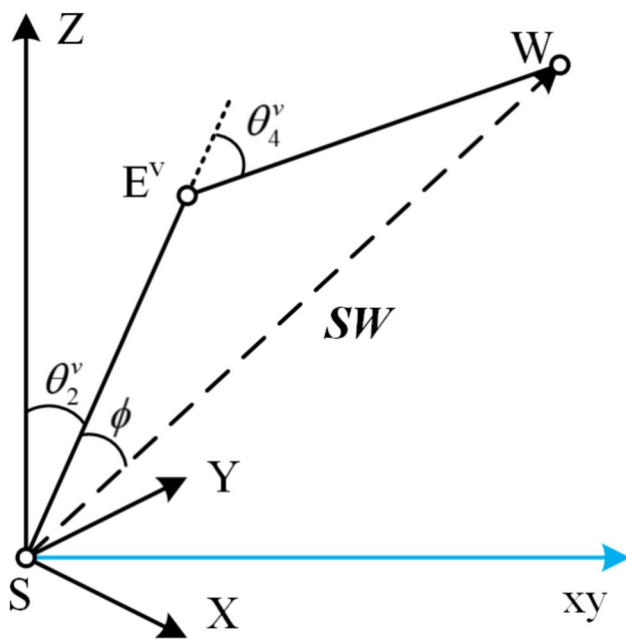


Fig. 24 Virtual joints

$$\theta_4^v = GC_4 * \left(\pi - \arccos \left(\frac{d_{se}^2 + d_{ew}^2 - d_{sw}^2}{2d_{se}d_{ew}} \right) \right) \quad (25)$$

where d_{se} is the module of vector $\mathbf{SW}$. Because $\theta_3^v = 0$, θ_1^v determines the components of the manipulator in the X and Y directions relative to the base coordinate system, it can be obtained:

$$\theta_1^v = a \tan 2(GC_2 * \mathbf{SW}_y, GC_2 * \mathbf{SW}_x) \quad (26)$$

In triangle SE^vW , according to the cosine theorem we can get ϕ :

$$\phi = \arccos \left(\frac{d_{se}^2 + d_{sw}^2 - d_{ew}^2}{2d_{se}d_{sw}} \right) \quad (27)$$

According to Eq. (27), we can get:

$$\theta_2^v = GC_2 * \left(a \tan 2 \left(\sqrt{\mathbf{SW}_x^2 + \mathbf{SW}_y^2}, \mathbf{SW}_z \right) + GC_4 * \phi \right) \quad (28)$$

Angle of Each Joint

According to the value of θ_1^v , θ_2^v , θ_3^v , θ_4^v , we can get:

$${}^0_4\mathbf{R} = {}^0_\psi\mathbf{R}_0^3\mathbf{R}^v \quad (29)$$

And because $\theta_4 = \theta_4^v$, Eq. (29) can be expressed as:

$${}^0_3\mathbf{R} = {}^0_\psi\mathbf{R}_0^3\mathbf{R}^v \quad (30)$$

where

$${}^0_\psi\mathbf{R} = \mathbf{I}_3 + \sin \psi [{}^0\mathbf{sw} \times] + (1 - \cos \psi) [{}^0\mathbf{sw} \times]^2 \quad (31)$$

where ${}^0\mathbf{sw}$ is the unit vector of vector $\mathbf{SW}$, $[{}^0\mathbf{sw} \times]$ is the skew symmetric matrix of ${}^0\mathbf{sw}$. By substituting Eq. (31) into Eq. (30), we can obtain:

$${}^0_3\mathbf{R} = \mathbf{A}_s \sin \psi + \mathbf{B}_s \cos \psi + \mathbf{C}_s \quad (32)$$

where $\mathbf{A}_s = [{}^0\mathbf{sw} \times] {}^3R^v$, $\mathbf{B}_s = -[{}^0\mathbf{sw} \times]^2 {}^3R^v$, $\mathbf{C}_s = [{}^0\mathbf{sw} {}^0\mathbf{sw}^T] {}^3R^v$. According to the forward kinematics model of the manipulator (Eqs. (23) and (24)), we can get:

$${}^0_3R = \begin{bmatrix} * & * & \cos \theta_1 \sin \theta_2 \\ * & * & \sin \theta_1 \sin \theta_2 \\ -\cos \theta_3 \sin \theta_2 & \sin \theta_2 \sin \theta_3 & \cos \theta_2 \end{bmatrix} \quad (33)$$

From this we can get:

$$\begin{cases} \theta_1 = \text{atan2}(GC_2(a_{223}\sin\psi + b_{223}\cos\psi + c_{223}), GC_2(a_{113}\sin\psi + b_{113}\cos\psi + c_{113})) \\ \theta_2 = GC_2\arccos(a_{333}\sin\psi + b_{333}\cos\psi + c_{333}) \\ \theta_3 = \text{atan2}(GC_2(a_{332}\sin\psi + b_{332}\cos\psi + c_{332}), GC_2(-a_{331}\sin\psi - b_{331}\cos\psi - c_{331})) \end{cases} \quad (34)$$

From the forward kinematics model of the manipulator, ${}^0_7\mathbf{R} = {}^0_3\mathbf{R}_3^4\mathbf{R}_4^7\mathbf{R}$ can be obtained, and ${}^7_4\mathbf{R}$ can be expressed as:

$${}^7_4\mathbf{R} = {}^4_3\mathbf{R}^{-1} {}^3_0\mathbf{R}^{-1} {}^7_0\mathbf{R} \quad (35)$$

Similar to ${}^3_0\mathbf{R}$, ${}^7_4\mathbf{R}$ can be expressed as:

$${}^7_4\mathbf{R} = \mathbf{A}_w \sin \psi + \mathbf{B}_w \cos \psi + \mathbf{C}_w \quad (36)$$

where

$$\mathbf{A}_w = {}^4_3\mathbf{R}\mathbf{A}_s^T {}^7_0\mathbf{R}, \mathbf{B}_w = {}^4_3\mathbf{R}\mathbf{B}_s^T {}^7_0\mathbf{R}, \mathbf{C}_w = {}^4_3\mathbf{R}\mathbf{C}_s^T {}^7_0\mathbf{R}.$$

According to the forward kinematics model of the manipulator (Eqs. (23) and (24)), we can get:

$${}^7_4R = \begin{bmatrix} * & * & \cos \theta_5 \sin \theta_6 \\ \cos \theta_7 \sin \theta_6 & -\sin \theta_6 \sin \theta_7 & -\cos \theta_6 \\ * & * & \sin \theta_5 \sin \theta_6 \end{bmatrix} \quad (37)$$

According to Eqs. (36) and (37), we can get:

$$\begin{cases} \theta_5 = \text{atan2}(GC_6(a_{w33}\sin\psi + b_{w33}\cos\psi + c_{w33}), GC_6(a_{w13}\sin\psi + b_{w13}\cos\psi + c_{w13})) \\ \theta_6 = GC_6\arccos(-a_{w23}\sin\psi - b_{w23}\cos\psi - c_{w23}) \\ \theta_7 = \text{atan2}(GC_6(-a_{w22}\sin\psi - b_{w22}\cos\psi - c_{w22}), GC_6(a_{w21}\sin\psi + b_{w21}\cos\psi + c_{w21})) \end{cases} \quad (38)$$

Acknowledgements This research was supported by Zhejiang Province Key R&D Program under Grant 2023C01070, Zhejiang Provincial Natural Science Foundation of China under Grant No. LQ24E050009, Key R&D Program of Zhejiang under Grant 2024C04056(CSJ).

Authors Contribution Li Zhang: Conceptualization, Methodology, Software, Writing – original draft, Writing – review & editing. Bing Han: Methodology, Software, Resources. Jianwei Niu: Supervision, Resources. Kun Tong: Resources, Methodology. Xinyang Tian: Conceptualization, Methodology. Xiaolong Yu: Methodology, Software, Resources, Funding acquisition. Renluan Hou: Funding acquisition, Project administration, Methodology. Ruliang Guo: Methodology, Resources.

Data Availability The datasets used or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Ethical Approval No human involved and all the research is carried out in an ethical manner.

Consent to Participate I hereby give my consent for the participate if any information required.

Consent for Publication I hereby give my consent for the results of this study to be published.

Conflict of Interests The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

References

- Shimizu, M., Kakuya, H., Yoon, W.K., et al.: Analytical Inverse kinematic computation for 7-DOF redundant manipulators with joint limits and its application to redundancy resolution[J]. *IEEE Trans. Rob.* **24**(5), 1131–1142 (2008)
- Nenchev, D.N., Tsumaki, Y., Takahashi, M.: Singularity-consistent kinematic redundancy resolution for the S-R-S manipulator[C]. 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566), vol.4, pp. 3607–3612. (2004)
- Nakamura, Y., Hanafusa, H., Yoshikawa, T.: Task-priority based redundancy control of robot manipulators[J]. *Int. J. Robot. Res.* **6**(2), 3–15 (1987)
- Ortenzi, V., Marturi, N., Rajasekaran, V., et al.: Singularity-robust inverse kinematics solver for tele-manipulation[C]. 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE). (2019). 1821–1828
- Karpińska, J., Tchoń, K.: Performance-oriented design of inverse kinematics algorithms: extended Jacobian approximation of the Jacobian pseudo-inverse[J]. *J. Mech. Robot.* **4**(2), (2012). <https://doi.org/10.1115/1.4006192>
- Simas, H., Di Gregorio, R.: A technique based on adaptive extended Jacobians for improving the robustness of the inverse numerical kinematics of redundant robots[J]. *J. Mech. Robot.* **11**(2), (2019). <https://doi.org/10.1115/1.4042514>
- Kreutz-Delgado, K., Long, M., Seraji, H.: Kinematic analysis of 7-DOF manipulators[J]. *Int. J. Robot. Res.* **11**(5), 469–481 (1992)
- Zeng, Z., Chen, Z., Shu, G., et al.: Optimization of analytical inverse kinematic solution for redundant manipulators using GA-PSO algorithm[C]. Proceedings - 2018 IEEE Industrial Cyber-Physical Systems, ICPS 2018, pp. 446–451. (2018)
- Umar, A., Shi, Z., Wang, W., et al.: A novel mutating PSO based solution for inverse kinematic analysis of multi degree-of-freedom robot manipulators[C]. 2019 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), pp. 459–463. (2019)
- Lee, C.T., Chang, J.Y.: A Workspace-analysis-based genetic algorithm for solving inverse kinematics of a multi-fingered anthropomorphic hand[J]. *Appl. Sci.-Basel.* **11**(6), (2021). <https://doi.org/10.3390/app11062668>
- Mayorga, R.V., Sanongboon, P.: Inverse kinematics and geometrically bounded singularities prevention of redundant manipulators: an artificial neural network approach[J]. *Robot. Auton. Syst.* **53**(3–4), 164–176 (2005)
- Zong, Y.M., Huang, J.Z., Bao, J.H., et al.: Inverse kinematics solution of demolition manipulator based on global mapping[J]. *Proc. Inst. Mech. Eng. Part C-J. Mech. Eng. Sci.* (2023). <https://doi.org/10.1177/09544062231184791>
- An, H.H., Clement, W.I., Reed, B.: Analytical inverse kinematic solution with self-motion constraint for the 7-DOF restore robot arm[C]. 2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pp. 1325–1330. (2014)
- Luo, R.C., Lin, T.W., Tsai, Y.H.: Analytical inverse kinematic solution for modularized 7-DoF redundant manipulators with off-sets at shoulder and wrist[C]. 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 516–521. (2014)
- Singh, G.K., Claassens, J.: An analytical solution for the inverse kinematics of a redundant 7DoF Manipulator with link offsets[C]. 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2976–2982. (2010)
- Lee, S., Bejczy, A.K.: Redundant arm kinematic control based on parameterization[C]. Proceedings. 1991 IEEE International Conference on Robotics and Automation, pp. 458,459,460,461,462,463,464,465. (1991)
- Yan, L., Mu, Z., Xu, W.: Analytical inverse kinematics of a class of redundant manipulator based on dual arm-angle parameterization[C]. 2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 3744–3749. (2014)
- Gong, M., Li, X., Zhang, L.: Analytical inverse kinematics and self-motion application for 7-DOF redundant manipulator[J]. *IEEE Access* **7**, 18662–18674 (2019)
- Faria, C., Ferreira, F., Erhagen, W., et al.: Position-based kinematics for 7-DoF serial manipulators with global configuration control, joint limit and singularity avoidance[J]. *Mech. Mach. Theory* **121**, 317–334 (2018)
- Tian, X., Xu, Q., Zhan, Q.: An analytical inverse kinematics solution with joint limits avoidance of 7-DOF anthropomorphic manipulators without offset[J]. *J. Franklin Inst.* **358**(2), 1252–1272 (2021)

21. Wang, J., Lu, C., Zhang, Y., et al.: A numerically stable algorithm for analytic inverse kinematics of 7-degrees-of-freedom spherical-rotational-spherical manipulators with joint limit avoidance[J]. *J. Mech. Robot.* **14**(5), (2022)
22. Quan, Y., Wang, K., Zhao, C., et al.: Obstacle avoidance method for fixed trajectory of a seven-degree-of-freedom manipulator[J]. *Robotica* **41**(5), 1515–1535 (2023)
23. Benzaoui, M., Chekireb, H., Tadjine, M., et al.: Redundant robot manipulator control with obstacle avoidance using extended Jacobian method[C]. 18th Annual International Mediterranean Conference on Control and Automation (MED), pp. 371–376. (2010)
24. Liu, Y.-B., Dai, Q.: Improvement of obstacle and singularity avoidance path planning algorithm for redundant manipulators[C]. 5th International Conference on Computer Sciences and Automation Engineering (ICCSAE), pp. 959–969. (2015)
25. Zhu, L., Mao, H., Luo, X., et al.: Determining null-space motion to satisfy both task constraints and obstacle avoidance[C]. IEEE International Symposium on Assembly and Manufacturing (ISAM), pp. 112–119. (2016)
26. Li, X., Liu, H., Dong, M.: A general framework of motion planning for redundant robot manipulator based on deep reinforcement learning[J]. *IEEE Trans. Industr. Inf.* **18**(8), 5253–5263 (2022)
27. Qiu, Q., Cao, Q.: Task constrained motion planning for 7-degree of freedom manipulators with parameterized submanifolds[J]. *Ind. Robot Int. J* **45**(3), 363–370 (2018)
28. Long, X., Chen, H., Zheng, J., et al.: Nonfragile memory-based sampled-data control for unmanned surface vehicles with time-varying coupling delay[J]. *Ocean Eng.* **313**, 119480 (2024)
29. Long, X., Chen, H., Zheng, J., et al.: Non-fragile control associated with exponential stability for unmanned surface vessels with the rudder failure and dynamic event-trigger[J]. *Ocean Eng.* **299**, 117393 (2024)
30. Ram, R.V., Pathak, P.M., Junco, S.J.: Inverse kinematics of mobile manipulator using bidirectional particle swarm optimization by manipulator decoupling[J]. *Mech. Mach. Theory* **131**, 385–405 (2019)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Li Zhang received the Ph.D. degree in mechanical manufacturing and automation from Zhejiang University, Hangzhou, China, in 2023. He is currently a postdoctoral researcher with Hangzhou Innovation Institute, Beihang University, Hangzhou, China. His research interests include the analytical inverse kinematics solution, path planning approaches, and trajectory planning techniques for seven-degree-of-freedom manipulator.

Han Bing received the M.S. degree in Control Theory and Control Engineering from Northeastern University, China, in 2017, and is currently a Ph.D. candidate in Mechanical Engineering at Zhejiang University. His research interests include robot operating systems, development of high-performance controllers, coordinated planning and control of dual-arm robots, compliant control, and trajectory planning.

Jianwei Niu received the M.S. and Ph.D. degrees in computer science from Beihang University, Beijing, China, in 1998 and 2002, respectively. He was a visiting scholar with the School of Computer Science, Carnegie Mellon University, USA, from 2010 to 2011. He is currently a professor with the School of Computer Science and Engineering, BUAA. His current research interests include mobile and pervasive computing, and distributed computing.

Kun Tong received the B.E degree in Mechanical Design Manufacturing and Automation from Shenyang Agricultural University and the M.E. and Ph.D. degrees in Mechanical Electronics Engineering from Northeastern University, China, in 2006, 2009 and 2013, respectively. He is mainly engaged in intelligent manufacturing, industrial Internet and other related research work, and his application fields are aerospace, robots, CNC machine tools and other intelligent equipment control and digitalization.

Xinyang Tian received the B.Eng. degree in mechanical engineering from Northeast Agricultural University, Harbin, China, in 2016, and the Ph.D. degree in mechatronic engineering (robot design and control) from Beihang University, Beijing, China, in 2023. He is currently a postdoctoral researcher with Hangzhou Innovation Institute, Beihang University, Hangzhou, China. His research interests include robot design, impedance control, and physical human robot interaction.

Xiaolong Yu received the M.S. degree in Mechanical engineering from Harbin Engineering University, Harbin, China, in 2016. He is now a PhD student in School of Automation Science and Electrical Engineering, Beihang University, Beijing, China. He is currently an associate researcher in Hangzhou Innovation Institute of Beihang University and Zhejiang Key Laboratory of Industrial Big Data and Robot Intelligent Systems. His current research interests include robot perception and motion control.

Renluan Hou received the B.S. degree in material forming and control engineering and M.S. degree in mechanical design and theory from Nanjing Agricultural University, Nanjing, China, in 2010 and 2013, respectively. She received the Ph.D. degree in mechanical manufacturing and automation from Zhejiang University, Hangzhou, China, in 2020. She joined the Hangzhou Innovation Institute, Beihang University, Hangzhou, China in 2020 and is currently an associate researcher of computer science and engineering. Her research interests include trajectory optimization algorithm and motion control algorithm of industrial robot.

Yuliang Guo received the B.E. and Ph.D. degrees in electrical engineering from Zhejiang University, Hangzhou, China, in 2015 and 2020. He is currently an associate research fellow with the Hangzhou Innovation Institute, Beihang University, Hangzhou, China. His research interests include robot dynamics, high-precision control algorithms and physical human-robot interaction.