WeiXin Chou
YiWei Liu ✉

https://doi.org/10.21278/TOF.481055523
ISSN 1333-1124
eISSN 1849-1391

AN ANALYTICAL INVERSE KINEMATICS SOLUTION WITH THE AVOIDANCE OF JOINT LIMITS, SINGULARITY AND THE SIMULATION OF 7-DOF ANTHROPOMORPHIC MANIPULATORS

Summary

A novel analytical method for solving the inverse kinematics of seven-degrees-of-freedom manipulators is presented in this paper. The method avoids joint limits and singularities by involving the arm angle (φ) to parametrize elbow redundancy. In addition, this method does not need to calculate the complicated parameter ${}^{0}R_{3}^{0}$. After discretizing the workspace of the robotic arm, it becomes possible to intuitively observe how the manipulability is distributed throughout its workspace. The relations between the arm angle and the joint angles are derived. The feasible intervals for the arm angles can be determined for the specified end posture. Finally, to compare the novel method with existing methods, we tested the proposed algorithm on Windows 10 and Ubuntu 18.04. Simulation statistics indicate that when implemented in C++, the proposed algorithm has significant advantages in both convergence time and stability compared to MATLAB; the algorithm achieves an average computing time of 70.12 μs with a standard deviation of 8.45 μs. Besides, we also conducted a comparative analysis with KDL and IKFast, revealing that the proposed algorithm consistently achieves a computing time of 28.54 μs with a standard deviation of 5.45 μs. This finding can provide a theoretical basis for further research on manipulators.

Key words: *redundant manipulator; inverse kinematics solution; arm angle; joint limits; simulation*

1. Introduction

With the advancement of industrial robotics technology and the diversification of application scenarios, industrial robots are becoming increasingly visible in our surroundings. Compared to the six-degrees-of-freedom (6-DOF) manipulators, the seven-degrees-of-freedom (7-DOF) anthropomorphic manipulators have higher motion flexibility, which enables the completion of most tasks while avoiding obstacles, joint limits, and singularity [1-4]. As a result, there are constraints on the inverse kinematics solution of the manipulator, which increases the complexity of the problem. Therefore, 7-DOF robotic arms have been gaining more attention in recent years, and many researchers in China and abroad have carried out relevant studies in various fields, such as industrial robots, service robots, and space robots [5-9].

For the inverse kinematics problem of the 7-DOF manipulator, several solutions have been proposed, including the most widely used Jacobian method [10-12] and damped least squares [13]. However, the convergence and stability of the algorithm cannot be guaranteed when joint limits are considered [14]. Burdick [15], as well as Lee and Bejczy [16], obtained the inverse solution by converting the redundant arm to a 6-DOF arm by fixing the third joint, but this method sacrificed the self-motion properties of the redundant arms and did not consider the limits of joints. Jingdong Zhao [17] proposed the algorithm that takes the set of joint Cartesian positions as the intermediary. The algorithm divides the inverse solution process into two mapping processes within joint limits to avoid joint drift and elbow swing during the repetitive motion.

In order to solve the problem of avoiding singular points, Kreutz-Delgado et al. [18]. proposed the concept of self-motion, represented by the parameters of arm angle, for the first time. However, this method does not propose how to precisely determine the reference plane. Ortenzi et al. [19] and XJ Huo et al. [11] redefined the reference plane for arm angle by fixing the angle value of the third joint to zero, but this method does not consider the singular position. Yan Lei et al. [20] proposed the concept of double arm angle, which uses two vectors to determine the position of the reference plane. The authors did not provide an explanation for the relationship between the arm angle and the joint angle. Stefan Scherzinger [21] uses the forward dynamics compliance control algorithm to ensure stability in the vicinity of singularities. However, this method was only applied to 6-DOF robotic arms. Hamid Toshani et al., Sorić et al., and Vijavan et al. [22-24] employed the neural network algorithm for finding the inverse solution, which is an efficient algorithm, but it relies on a large amount of data.

This paper focuses on the 7-DOF redundant manipulator developed by Harbin Institute of Technology (HIT). We apply the arm angle principle to obtain the inverse kinematics solution of the manipulator and to present a visualization of its reachable space. Unlike the traditional arm angle method, our approach does not require complicated parameter calculations. Furthermore, the proposed method has high calculation speed, exceptional accuracy, and high stability.

2. Robot model establishment

2.1 Establishment of the HIT robotic arm model

The HIT manipulators have kinematic structure characterized as S-R-S (spherical-rotational-spherical) with all joints being revolute. The shoulder joint comprises 1-3 joints, with the joint axes intersecting at point *S*; joint 4 constitutes the elbow joint, with its ,centre point located at *E*; joints 5-7 constitute the wrist joint, with the joint axes intersecting at point *W*; the tool coordinate system is denoted as T. The 7-DOF manipulator model is shown in Fig. 1, while the Denavit-Hartenberg (D-H) parameters are show in Table 1.

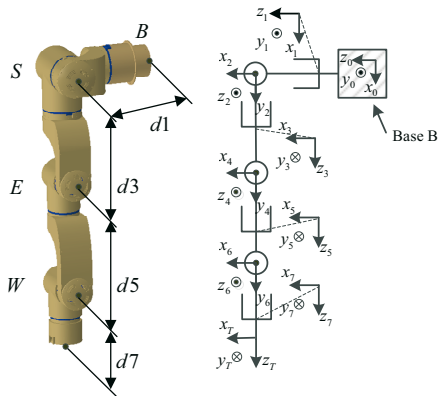

Fig. 1 7-DOF manipulator model

Table 1 D-H parameters

Joint	θ_i /rad	α_i /rad	a_i /m	d_i /m
1	θ1	-π/2	0	0.25
2	θ2	π/2	0	0
3	θ3	-π/2	0	0.40
4	θ4	π/2	0	0
5	θ5	π/2	0	0.40
6	θ6	π/2	0	0
7	θ7	0	0	0.26248

2.2 Forward kinematics of the robotic arm

Using the parameters shown in Table 1, the homogeneous transformations can be established as shown below:

$$ {}^{i-1}_{i}T = \begin{bmatrix} \mathrm{c}\theta_i & -\mathrm{s}\theta_i & 0 & \alpha_{i-1} \\ \mathrm{s}\theta_i \mathrm{c}\alpha_{i-1} & \mathrm{c}\theta_i \mathrm{c}\alpha_{i-1} & -\mathrm{s}\alpha_{i-1} & -d_i \mathrm{s}\alpha_{i-1} \\ -\mathrm{s}\theta_i \mathrm{s}\alpha_{i-1} & \mathrm{c}\theta_i \mathrm{s}\alpha_{i-1} & \mathrm{c}\alpha_{i-1} & d_i \mathrm{c}\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1} $$

where $\sin(\theta_i)$ can be denoted as $\mathrm{s}\theta_i$, $\cos(\theta_i)$ can be denoted as $\mathrm{c}\theta_i$, for i = 1 , 2, ..., 7. In the following text, we apply the same notation.

The posture of the manipulator end-effector relative to the base frame can be expressed as follows:

$$ {}^{0}_{7}T = \prod_{i=1}^{7} {}^{i-1}_{i}T \tag{2} $$

3. Inverse kinematics of redundant manipulators

During the motion of the manipulators, the wrist joint, elbow joint, and shoulder joint always lie in the same plane; that plane is referred to as the arm configuration plane. Self-motion is expressed as the rotational motion of joint 4 around the vector *SW*, and the corresponding rotation circle is referred to as the redundant circle. As shown in Fig. 2, the plane *SEW* is the reference plane. Setting the joint 3 value at 0 is the most common method used to determine the reference plane. Now, it can be observed that the axes of joint 2 and joint 4 are parallel to each other and perpendicular to the plane *BSW* formed by the vectors *BS* and *SW*. However, this method cannot determine the reference plane when the manipulator is in a singularity configuration. From Fig. 2, we can obtain the following expressions:

$$ SE = BE - BS = \begin{bmatrix} -d_{SE}\, \mathrm{c}\theta_1\, \mathrm{s}\theta_2 \\ -d_{SE}\, \mathrm{s}\theta_1\, \mathrm{s}\theta_2 \\ d_{SE}\, \mathrm{c}\theta_2 \end{bmatrix} \tag{3} $$

$$ SW = BT - BS - WT = \begin{bmatrix} d_{EW}\ (\mathrm{s}\theta_4\ (\ \mathrm{s}\theta_1 \mathrm{s}\theta_3 0 - \mathrm{c}\theta_1 \mathrm{c}\theta_2 \mathrm{c}\theta_3) - \mathrm{c}\theta_1 \mathrm{c}\theta_4 \mathrm{s}\theta_2) - d_{SE} \mathrm{c}\theta_1 \mathrm{s}\theta_2 \\ d_{EW} (\mathrm{s}\theta_4\ (\mathrm{c}\theta_1 \mathrm{s}\theta_3 + \mathrm{c}\theta_2 \mathrm{c}\theta_3 \mathrm{s}\theta_1) + \mathrm{c}\theta_4 \mathrm{s}\theta_1 \mathrm{s}\theta_2) - d_{SE} \mathrm{s}\theta_1 \mathrm{s}\theta_2 \\ d_{SE} \mathrm{c}\theta_2 + d_{EW} \mathrm{c}\theta_2 \mathrm{c}\theta_4 - \mathrm{c}\theta_3 \mathrm{s}\theta_2 \mathrm{s}\theta_4 \end{bmatrix} \tag{4} $$

where d_{SE} is the length between the wrist frame and elbow frame and d_{EW} is the length between the elbow frame and wrist frame.

With the use of MATLAB, the following equation can be obtained:

$$ (SE \times SW) BS = 0 \tag{5} $$

When joint 3 is set at 0, the plane *BSW* will remain unchanged. The plane *SE'W* is called the arm plane. A clear definition of the reference plane can be provided. According to the right-hand

rule, the angle between the reference plane and the arm plane can be referred to as the arm angle, denoted as φ. The range of the arm angle is $\varphi \in [-\pi \ \ \pi]$ if the joint limits are not considered.

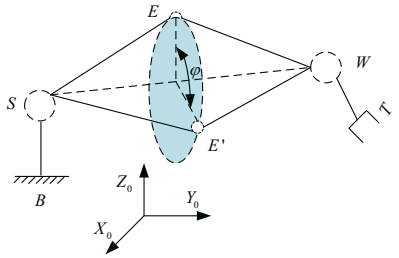

Fig. 2 Definition of the arm angle

3.1 Calculation of the parametric closed solution of the manipulator

Let ${}^0X_7 \in R^3$, ${}^0R_7 \in SO(3)$ denote the posture of the end-effector in the base frame and let us define the vector MN as ${}^KX_{MN}$ in the frame $O_K xyz$. The value d_i (i = 1, 3, 5, 7) describes the manipulator link length, d_{BS} represents the length between the base frame and shoulder frame, and d_{WT} represents the length between the wrist frame and tool frame. As shown in Figs.1 and 2, we can see that $d_{BS} = d_1$, $d_{SE} = d_3$, $d_{EW} = d_5$ and $d_{WT} = d_7$.

From the joint position vector, we can get ${}^0X_7$ and ${}^0_7R$ of the manipulator, which can be expressed as follows:

$${}^0X_7 = d_{BS} + {}^0_3R\left\{{}^3d_{SE} + {}^3_4R({}^4d_{EW} + {}^4_7R\,{}^7d_{WT})\right\} \tag{6}$$

$${}^0_7R = {}^0_3R\,{}^3_4R\,{}^4_7R \tag{7}$$

The point S in the base frame can be represented as follows:

$$PS = [0,\ 0,\ d_{BS}] \tag{8}$$

The point W in the base frame can be represented as follows:

$${}^0X_{S'W} = {}^0X_{BT} - {}^0X_{BS'} - {}^0R_7\,{}^7X_{WT} \tag{9}$$

$$\begin{bmatrix} PW \\ 1 \end{bmatrix} = Td \times \begin{bmatrix} 0 \\ 0 \\ -d_{WT} \\ 1 \end{bmatrix} \tag{10}$$

Given the posture of the manipulator end-effector, due to the existence of self-motion characteristics, the manipulator can rotate around the SW axis. The points S, W and E always form a triangle during the rotation process, where d_{SE} and d_{EW} are already known. Based on the above analysis, we can calculate $d_{SW} = \|PW - PS\|$; the value of the $\cos \angle SEW$ can be calculated by applying the cosine theorem.

$$\cos \angle SEW = \frac{d_{SE}^2 + d_{EW}^2 - d_{SW}^2}{2 \times d_{SE} \times d_{EW}} \tag{11}$$

The value of the elbow joint can be distributed symmetrically above and below the line SW, and the value of θ_4 can be calculated as follows:

$$\theta_4 = \pm(\pi - \arccos \angle SEW) \tag{12}$$

It can be seen that θ_4 is uniquely determined if the end-effector posture is given, and it is unrelated to φ.

Due to the presence of self-motion, the position of joint 6 is not affected by the movement of joint 5, as their axes intersect at one point. Therefore, if the posture of the manipulator end-effector is given, θ_4 is a constant value. In the process of self-motion, the position of point W is only uniquely determined with joints 1-3. A frame Σ can be established in the plane SEW; the point S is the origin, the X^{Σ} axis is along the vector SW, pointing towards W, and the Y^{Σ} axis is perpendicular to the plane SEW. Thus, the frame Σ can be expressed as follows:

$$\begin{cases} X^{\Sigma} = \dfrac{V_{SW}}{\|V_{SW}\|} \\ Y^{\Sigma} = \dfrac{Z_0 \times X^{\Sigma}}{\|Z_0 \times X^{\Sigma}\|} \\ Z^{\Sigma} = X^{\Sigma} \times Y^{\Sigma} \end{cases} \tag{13}$$

When the values of θ_1, θ_2 and θ_3 are set to 0, the frame Σ can be expressed as Σ_0, which is obtained from the above analysis.

$$P_W = \begin{bmatrix} d_{EW} \times \mathrm{s}\theta_4 \\ 0 \\ d_{BS} + d_{SE} + d_{EW} \times \mathrm{c}\theta_4 \end{bmatrix} \tag{14}$$

$$V_{SW} = P_W - P_S \tag{15}$$

We can set $Z_0 = [0,\ 0,\ 1]^T$. Putting Eqs. (14) and (15) into Eq. (13), we can obtain X^{Σ_O}, Y^{Σ_O} and Z^{Σ_O}. Thus, the frame Σ_o can be expressed as follows:

$$\Sigma_O = [X^{\Sigma_O},\ Y^{\Sigma_O},\ Z^{\Sigma_O}] \tag{16}$$

When the values of θ_1, θ_2 and θ_3 are all 0, the plane SEW coincides with $SE'W$, so $\varphi = 0$, and the frame Σ can be expressed as $\Sigma_D^{\varphi=0}$. By substituting Eqs. (8) and (10) into (13), we can obtain $\Sigma_D^{\varphi=0}$ as follows:

$$\Sigma_D^{\varphi=0} = [X^{\Sigma_D^0},\ Y^{\Sigma_D^0},\ Z^{\Sigma_D^0}] \tag{17}$$

When the values of θ_1, θ_2 and θ_3 are not 0, the frame Σ can be expressed as Σ_D, which can be obtained by rotating $\Sigma_D^{\varphi=0}$ about the x-axis by an angle φ. We can get the frame Σ_D as follows:

$$\Sigma_D = \Sigma_D^{\varphi=0} \times \mathrm{Rot}(x,\ \varphi) \tag{18}$$

After Td is known, the value of φ can be changed to obtain a series of continuous joint values using Rodrigues' rotation formula, which is expressed as follows:

$$\mathrm{Rot}(x,\varphi) = I_3 + \mathrm{s}\varphi \times [x\times] + (1 - \mathrm{c}\varphi) \times [x\times]^2 \tag{19}$$

where I_3 is the identity matrix; x is a unit vector, which can be set [1, 0, 0]; $[x\times]$ is the skew symmetric matrix of vector x.

When joint 1 rotates around the Z_1 axis by an angle of θ_1, joint 2 rotates around the Z_2 axis by an angle of θ_2, and joint 3 rotates around the Z_3 axis by an angle of θ_3. This process can be regarded as a spherical joint rotating around the Z axis by an angle of θ_1, followed by the spherical joint rotating around the Y axis by an angle of θ_2, and, finally, rotating around the X axis by an angel of θ_3. Thus, we can obtain the following expression:

$$ {}_{s}^{0}R = \Sigma_D \times \Sigma_O^T = A_s \, \mathrm{s}\varphi + B_s \, \mathrm{c}\varphi + C_s \tag{20} $$

The coefficient matrices A_s, B_s and C_s satisfy the following equations, respectively: $A_s = \Sigma_D^{\varphi=0} \times [x\times]^2 \times \Sigma_o^T$, $B_s = -1 \times \Sigma_D^{\varphi=0} \times [x\times]^2 \times \Sigma_o^T$, and $C_s = \Sigma_D^{\varphi=0} \times (I_3 + [x\times]^2) \Sigma_o^T$.

Since ${}_{3}^{0}R$ can be expressed in the following form,

$$ {}_{3}^{0}R = \mathrm{Rot}(Z_1, \theta_1) \times \mathrm{Rot}(Z_2, \theta_2) \times \mathrm{Rot}(Z_3, \theta_3) \tag{21} $$

$$ {}_{3}^{0}R = \begin{bmatrix} * & * & \mathrm{c}\theta_1 \, \mathrm{s}\theta_2 \\ * & * & \mathrm{s}\theta_1 \, \mathrm{s}\theta_2 \\ -\mathrm{s}\theta_2 \, \mathrm{c}\theta_3 & \mathrm{s}\theta_2 \, \mathrm{s}\theta_3 & \mathrm{c}\theta_2 \end{bmatrix} \tag{22} $$

the solutions for the shoulder joint can be expressed as functional forms of φ, which are:

$$ \mathrm{c}\theta_2 = -a_{s33} \, \mathrm{s}\varphi - b_{s33} \, \mathrm{c}\varphi - c_{s33} \tag{23} $$

$$ \tan(\theta_1) = \frac{a_{s23} \, \mathrm{s}\varphi + b_{s23} \, \mathrm{c}\varphi + c_{s23}}{a_{s13} \, \mathrm{s}\varphi + b_{s13} \, \mathrm{c}\varphi + c_{s13}} \tag{24} $$

$$ \tan(\theta_3) = \frac{a_{s32} \, \mathrm{s}\varphi + b_{s32} \, \mathrm{c}\varphi + c_{s32}}{\left(a_{s31} \, \mathrm{s}\varphi + b_{s31} \, \mathrm{c}\varphi + c_{s31}\right)} \tag{25} $$

where $a_{s'ij}$, $b_{s'ij}$, $c_{s'ij}$ are the (i, j) elements of A_s, B_s and C_s, respectively.

Based on the above analysis, we can substitute Eq. (17) into (7) to obtain

$$ {}_{7}^{4}R = A_w \, \mathrm{s}\varphi + B_w \, \mathrm{c}\varphi + C_w \tag{26} $$

Since ${}_{7}^{4}R$ can be expressed as

$$ {}_{7}^{4}R = \begin{bmatrix} * & * & \mathrm{c}\theta_5 \, \mathrm{s}\theta_6 \\ \mathrm{s}\theta_6 \, \mathrm{c}\theta_7 & \mathrm{s}\theta_6 \, \mathrm{s}\theta_7 & -\mathrm{c}\theta_6 \\ * & * & \mathrm{s}\theta_5 \, \mathrm{s}\theta_6 \end{bmatrix} \tag{27} $$

the coefficient matrices A_w, B_w and C_w satisfy the following equations, respectively: $A_w = {}_{4}^{3}R^{\mathrm{T}} A_s^{\mathrm{T}} \, {}_{7}^{0}R$, $B_w = {}_{4}^{3}R^{\mathrm{T}} B_s^{\mathrm{T}} \, {}_{7}^{0}R$, and $C_w = {}_{4}^{3}R^{\mathrm{T}} C_s^{\mathrm{T}} \, {}_{7}^{0}R$.

The solutions for the wrist joint can be expressed as functional forms of φ, which are:

$$ \tan(\theta_5) = \frac{a_{w33} \, \mathrm{s}\varphi + b_{w33} \, \mathrm{c}\varphi + c_{w33}}{a_{w13} \, \mathrm{s}\varphi + b_{w13} \, \mathrm{c}\varphi + c_{w13}} \tag{28} $$

$$ \mathrm{c}\theta_6 = -a_{w23} \, \mathrm{s}\varphi - b_{w23} \, \mathrm{c}\varphi - c_{w23} \tag{29} $$

$$ \tan(\theta_7) = \frac{a_{w22} \, \mathrm{s}\varphi + b_{w22} \, \mathrm{c}\varphi + c_{w22}}{a_{w21} \, \mathrm{s}\varphi + b_{w21} \, \mathrm{c}\varphi + c_{w21}} \tag{30} $$

To sum up, Eqs. (12), (23)-(25), (28)-(30) provide a closed solution parametric for the 7-DOF manipulator based on φ. Given Td and φ, the value of each joint of the robotic arm can be determined quickly using this method.

3.2 Manipulator workspace analysis

Inverse kinematics only considers the solvability, without taking into account the reachability of the manipulators. In order to determine whether the trajectory point of the end

of the manipulator can be reached, it is necessary to analyse the workspace and manipulability measure within the range of each joint limits.

The limits of each joint are presented in Table 2; the unit used is degree (°).

Table 2 Limits of joint angles

joint 1(°)	joint 2(°)	joint 3(°)	joint 4(°)	joint 5(°)	joint 6(°)	joint 7(°)
±170	±120	±170	±120	±170	±120	±170

"Manipulability" refers to the ability of a robotic manipulator to move and exert force in various directions. It is defined as a measure of the manipulator operability by the determinant of the product of its Jacobian matrix and its transpose. The joint angle values are randomly selected within the upper and lower limits provided in Table 2. Thus, we can obtain

$$w = \sqrt{\det(J * J^T)} = \sqrt{\lambda_1 \lambda_2 ... \lambda_m} \tag{31}$$

where λ_i represents the eigenvalues of the Jacobian matrix and m is the dimension of the operational space.

When the robotic arm is in a singular configuration (w=0). It is easy to see that the manipulability index is always greater than zero, except when the robotic arm is in a singular configuration. This metrics can be used to measure how close or far the robotic arm is from a singular configuration and, therefore, can be used to assess the flexibility of the robotic arm.

After discretizing the workspace, we visualize the manipulability at discrete points. Now, we can describe the manipulability distribution within the workspace of the robotic arm, and it becomes possible to intuitively observe how the manipulability is distributed throughout the workspace of the robotic arm. In order to facilitate the observation of the distribution of relative manipulability in the robotic arm, a cross-sectional view of the spherical representation of the manipulability map is shown in Fig. 3a. If we exclude the points with a manipulability of 0, the projection can be obtained, as shown in Fig. 3b-d. The validation of the reachable end-effector postures of the robotic arm aligns with the actual scenarios, laying an experimental foundation for future research.

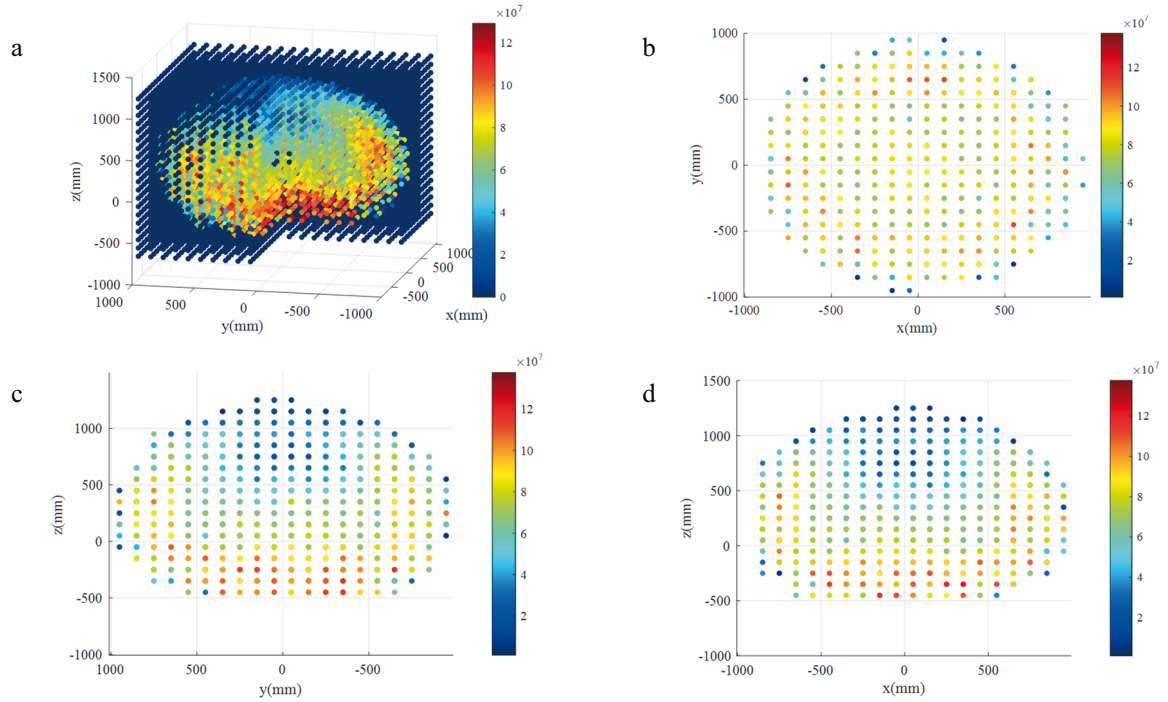

Fig. 3 The spherical representation of the global relative manipulability map: (a) a cross-sectional view, (b) the projection in the XOY plane, (c) the projection in the YOZ plane. (d) the projection in the XOZ plane

3.3 Singularity analyses

The singularity of the manipulators is a kinematic property that exists independent of the chosen frame. As described in Section 2.1, we divide the joints of the manipulator into shoulder, elbow and wrist joints. Accordingly, the singularity can also be divided into shoulder, elbow, and wrist singularities.

3.3.1 Shoulder singularity

Figure 2 shows that when the vectors BS and SW are collinear, the manipulator is in a shoulder singularity, and the motion of the wrist is independent of joint 1 rotation. To detect whether the manipulator has reached a singular position, the singular parameter λ is introduced, which can be calculated using the following formula:

$$\lambda = \sqrt{(BW \times BS)^T (BW \times BS)} \tag{32}$$

When $\lambda < 0$, the shoulder singularity occurs.

3.3.2 Elbow singularity

The singularity for the elbow joint occurs when $\theta_4 = 0$ or $\pm\pi$, which corresponds to the end-effector reaching the limit of its workspace. To avoid this situation, it is important to use reasonable motion planning techniques.

3.3.3 Wrist singularity

When the value of $\theta_6 = 0$ or $\pm\pi$, the axes of joints 5 and 7 become collinear, resulting in no rotation of the wrist, even if the values of θ_5 and θ_7 are equal but in opposite directions. However, this singularity can be avoided.

3.4 Analysis of the value range of arm angle parameters

When the transformation matrix Td of the manipulator is given, a series of joint vectors can be obtained for different values of φ. However, each joint of the manipulator is limited, resulting in motion in a partial space in the continuous self-motion of the manipulator around SW. Thus, determining the effective range of φ is crucial for ensuring the feasibility of the joint solution, which is affected by the joint limits and singularities of manipulators.

The effective range of φ is denoted by Ψ_i. Different equations to analyse the mapping relationship between the joint limits and the effective range of φ are given below. From Eq. (12), one can see that θ_4 is not affected by φ. From the shoulder joint and wrist joint, one can obtain the relationship between θ_1, θ_3, θ_5, θ_7 and φ, and the functional relationship between θ_2, θ_6, and φ.

The relationship between θ_1, θ_3, θ_5, θ_7 and φ can be expressed as follows:

$$\tan\theta_i = \frac{a_n \,\mathrm{s}\varphi + b_n \,\mathrm{c}\varphi + c_n}{a_d \,\mathrm{s}\varphi + b_d \,\mathrm{c}\varphi + c_d} \tag{33}$$

and

$$\tan\theta_i = \frac{f_n(\varphi)}{f_d(\varphi)} \tag{34}$$

We have

$$\begin{cases} f_n(\varphi) = a_n \,\mathrm{s}\varphi + b_n \,\mathrm{c}\varphi + c_n \\ f_d(\varphi) = a_d \,\mathrm{s}\varphi + b_d \,\mathrm{c}\varphi + c_d \end{cases} \tag{35}$$

Taking the derivative on both sides of Eq. (34), we can get

$$\frac{\mathrm{d}\theta_i}{\mathrm{d}\varphi} = \frac{a_t \,\mathrm{s}\varphi + b_t \,\mathrm{c}\varphi + c_t}{f_n^2(\varphi) + f_d^2(\varphi)} \tag{36}$$

where

$a_t = b_d c_n - b_n c_d$, $b_t = a_n c_d - a_d c_n$, $c_t = a_n b_d - a_d b_n$.

When $a_t^2 + b_t^2 - c_t^2 > 0$ is satisfied, it can be observed that there are two stagnation points of θ_i. The stagnation point of θ_i can be obtained from Eq. (35), as shown in Fig. 4.

$$\varphi_0 = 2\arctan\left(\frac{a_t \pm \sqrt{a_t^2 + b_t^2 - c_t^2}}{b_t - c_t}\right) \tag{37}$$

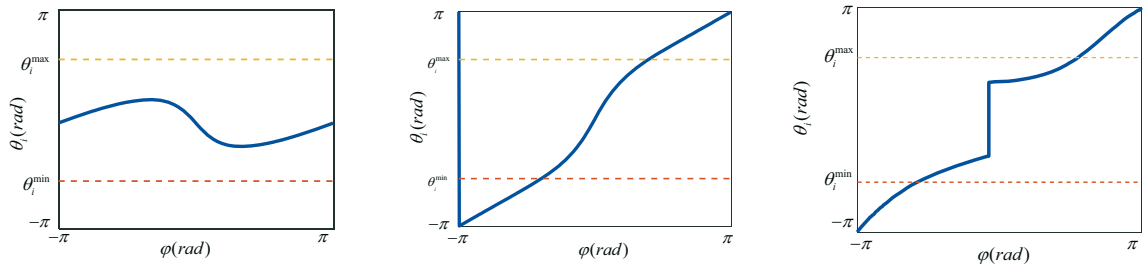

Fig. 4 Changes in the joint angle with changes in the arm angle

In Fig. 4a one can see that when $a_t^2 + b_t^2 - c_t^2 > 0$, the effective range of φ is not affected by the limits of joints. Therefore, the range can be expressed as $\Psi_i \in [-\pi\ \pi]$.

If $a_t^2 + b_t^2 - c_t^2 < 0$, there are no stagnation points for θ_i; thus, Eq. (31) changes monotonically, as shown in Fig. 4b.

From Fig. 4b, it is evident that the joint angle changes monotonically with the change of φ. The effective range of φ depends on the $\theta_i^{\max}$ and $\theta_i^{\min}$ of the joint i; the effective range of φ can be obtained when the intersection point of the lines of $\theta_i^{\min}$ and $\theta_i^{\max}$ is known.

If $a_t^2 + b_t^2 - c_t^2 = 0$, there is one stagnation point for θ_i, which can be obtained from Eq. (37), as depicted in Fig. 4c.

$$\varphi_0 = 2\arctan\left(\frac{a_t}{b_t - c_t}\right) \tag{38}$$

Figure 4c shows the effective range of φ, which depends on the $\theta_i^{\max}$ and $\theta_i^{\min}$ of the joint i. The effective range of φ can be obtained when the intersection point of the lines of $\theta_i^{\min}$ and $\theta_i^{\max}$ is known.

Based on Eqs. (23) and (29), the relationship between θ_2, θ_6 and φ can be expressed as follows:

$$\cos\theta_i = a\,\mathrm{s}\varphi + b\,\mathrm{c}\varphi + c \tag{39}$$

If $\sqrt{a^2+b^2} \neq 0$, then Eq. (37) can be transformed into

$$\sin(\phi+\varphi) = \frac{\cos q - c}{\sqrt{a^2+b^2}} \in [K_{\min}, K_{\max}] \tag{40}$$

where

$\phi = \arctan 2(b,a)$, $K_{\max}$ and $K_{\min}$ can be determined by $\theta_i^{\max}$ and $\theta_i^{\min}$.

On this basis, the problem of determining the effective range of φ can be transformed into solving the following inequality:

$$K_{\min} \leq \mathrm{s}(\phi+\varphi) \leq K_{\max} \tag{41}$$

Let $\mathrm{k}\phi = \phi + \varphi$, for which the maximum and minimum values can be expressed as follows:

$$k\phi_{\min} = \begin{cases} \operatorname{asin}(K_{\min}), & |K_{\min}| \leq 1 \\ \pi/2, & K_{\min} > 1 \\ -\pi/2, & K_{\min} < -1 \end{cases} \tag{42}$$

$$k\phi_{\max} = \begin{cases} \operatorname{asin}(K_{\max}), & |K_{\max}| \leq 1 \\ \pi/2, & K_{\max} > 1 \\ -\pi/2, & K_{\max} < -1 \end{cases} \tag{43}$$

Equation (41) has the following conditions:

(a) $K_{\min} > 1$ or $K_{\max} < -1$, then $\Psi_i = \varnothing$;

(b) $K_{\min} = -1$ and $K_{\max} = 1$, then $\Psi_i = [-\pi \ \ \pi]$;

(c) $K_{\min} \geq 0$ and $K_{\max} \geq 0$, then $k\phi \in [k\phi_{\min} \ \ k\phi_{\max} \ ] \cup [\pi - k\phi_{\max} \ \ \pi - k\phi_{\min} \ ]$;

(d) $K_{\min} \leq 0$ and $K_{\max} \geq 0$, then $k\phi \in [k\phi_{\min} \ k\phi_{\max} \ ] \cup [\pi - k\phi_{\max} \ \ \pi \ ] \cup [-\pi \ \ -\pi - k\phi_{\min} \ ]$;

(e) $K_{\min} < 0$ and $K_{\max} < 0$, then $k\phi \in [k\phi_{\min} \ \ k\phi_{\max} \ ] \cup [-\pi - k\phi_{\max} \ \ -\pi - k\phi_{\min} \ ]$.

Therefore, the range Ψ_i under the joint motion constraint can be calculated based on the range of $k\phi$.

If $\sqrt{a^2+b^2} = 0$, then $\mathrm{c}\theta_i = c$, and the θ_i is constant. Now the manipulator is in a singular configuration, and we can know $\Psi_i \in [-\pi \ \ \pi]$.

The intersection range of the feasible range of the φ with the joint limits of all joints of the manipulator is the feasible range of the joint. The value of θ_4 is not affected by φ. The effective range of the shoulder joint Ψ_S can be written as follows:

$$\Psi_S = \bigcap_{i=1}^{3} \Psi_i \tag{44}$$

The effective range of the wrist joint Ψ_W can be written as follows:

$$\Psi_W = \bigcap_{i=5}^{7} \Psi_i \tag{45}$$

Therefore, the effective range of the manipulator Ψ can be written as follows:

$$\Psi = \Psi_S \cup \Psi_W \tag{46}$$

4. Verification of the proposed method

4.1 Inverse kinematics verification

Taking the 7-DOF manipulator as an example, the dimensions of each link are: $d_{BS} = 0.25\,\text{m}$, $d_{SE} = 0.40\,\text{m}$, $d_{EW} = 0.40\,\text{m}$, and $d_{WT} = 0.26248\,\text{m}$, the values of ${}^0P_7$ and ${}^0R_7$ for the manipulators can be determined from the forward kinematics presented in Section 2.2. However, in this paper, the end-effector posture can be directly given within the reachable space.

$$ {}^0P_7 = [0.4750,\ 0.3653,\ 0.3912]/\,\text{m} $$

$$ {}^0R_7 = \begin{bmatrix} 0.5235 & -0.8862 & -0.01854 \\ -0.4852 & -0.4094 & 0.9112 \\ -0.5654 & -0.4066 & -0.5455 \end{bmatrix} $$

If we disregard the joint limit and set $\phi = \pi/10$ rad, we can obtain 8 groups of feasible solutions from the closed-form inverse kinematics solution, as shown in Table 3. The absolute errors between the robot end-postures corresponding to the 8 sets of solutions and the desired postures are all less than 1×10^{-15}, which illustrates the effectiveness of the proposed inverse kinematics algorithm.

Table 3 Inverse kinematics solution

joint 1 (rad)	joint 2 (rad)	joint 3 (rad)	joint 4 (rad)	joint 5 (rad)	joint 6 (rad)	joint 7 (rad)
2.7575	2.7720	2.3053	1.5488	-0.7959	1.0292	-1.1754
2.7575	2.7720	2.3054	1.5488	2.3457	-1.0292	-1.9662
2.7575	2.7720	-0.8363	-1.5488	-0.7959	-1.0292	-1.9662
2.7575	2.7220	-0.8363	-1.5488	2.3457	1.0292	1.1754
-0.3842	-2.7220	-0.8362	1.5488	-0.7959	1.0292	11754
-0.3842	-2.7220	-0.8362	1.5488	2.3457	-1.0292	-1.9662
-0.3842	-2.7220	2.3053	-1.5488	0.7959	-1.0292	-1.9662
-0.3842	-2.7220	2.3053	-1.5488	-2.3274	1.0292	1.754

4.2 Robotic arm trajectory tracking

Simulation software was used to perform trajectory tracking simulation verification based on the existing HIT redundant manipulator configuration, as shown in Fig. 5. The simulation results are shown in Fig. 6, where the starting point is (-0.8025, 0, 0.2767) and the ending point is (-0.6551, 0.2, 0.6037); all the points of the trajectory are within the workspace. The Cartesian space linear interpolation algorithm was used to discretize the trajectory, and each joint value can be obtained using the proposed inverse kinematics algorithm. The results reveal that the angles of each joint change continuously and smoothly without sudden changes, as shown in Fig. 7.

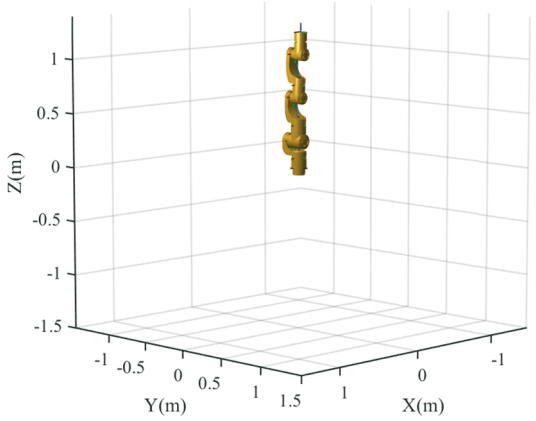

Fig. 5 A URDF model of the robot (joint angles are all 0)

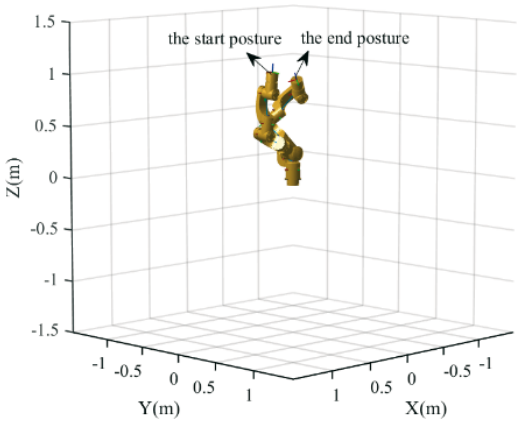

Fig. 6 Robot terminal trajectory simulation

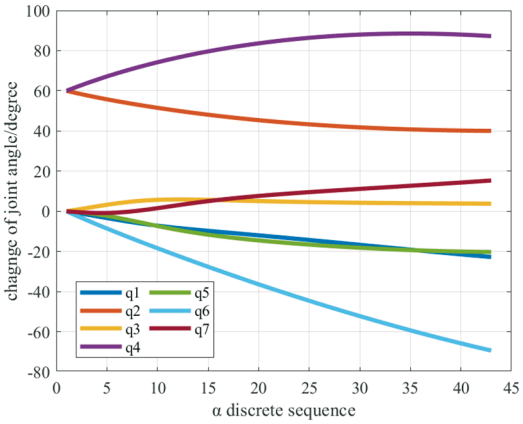

Fig. 7 Changes in joint angles

4.3 Arm angle range verification

The limits of each joint are presented in Table 2. The ${}^{0}X_{7}$ and ${}^{0}R_{7}$ of the manipulator can be obtained based on the $\theta_i^{\max}$ and $\theta_i^{\min}$ of each joint. Combined with the above analysis, the effective range of arm angle φ is shown in Figs. 8 and 9.

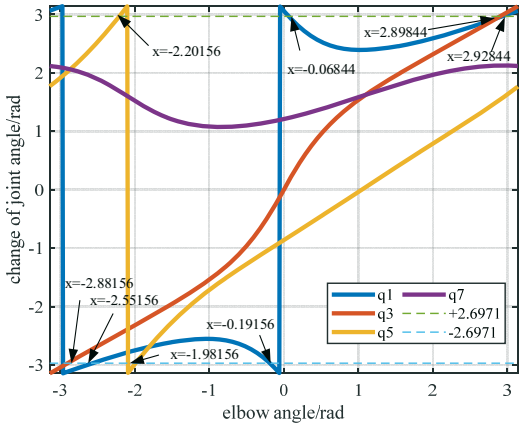

Fig. 8 Change of joint angle with arm angle (joints 1, 3, 5, and 7)

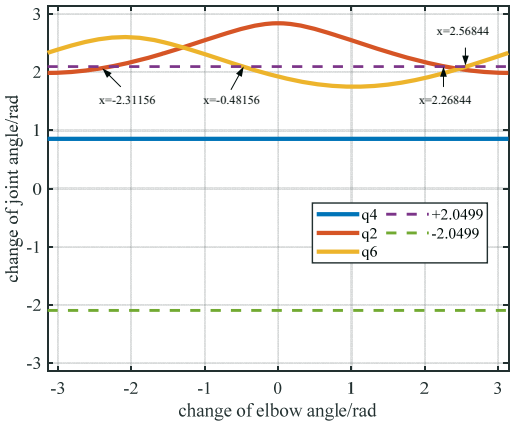

Fig. 9 Change of joint angle with arm angle (joints 2 and 6)

The effective ranges of arm angle φ for joints 1, 3, 5 and 7 are shown in Table 4.

Table 4 The effective ranges of arm angle (for joints 1, 3, 5 and 7)

Ψ_i	Arm angle feasible space
Ψ_1	[-2.5516 - 0.19156] ∪ [0.06844 2.92844]
Ψ_3	[-2.88156 2.89844]
Ψ_5	[-3.1416 -2.20156] ∪ [-1.98156 3.1416]
Ψ_7	[-3.1416 3.1416]

The effective ranges of arm angle φ for joints 2 and 6 are shown in Table 5.

Table 5 The effective ranges of arm angle (for joints 2 and 6)

Ψi	Arm angle feasible space
Ψ_2	[-3.1416 - 2.31156] ∪ [2.26844 3.1416]
Ψ_6	[-0.48156 2.56844]

By calculating the intersection of the above arm angles, the effective range of the overall arm angle of the manipulator can be obtained as follows:

$$\Psi = \Psi_S \cup \Psi_W = [-0.48156 \quad -0.19156] \cup [-1.98156 \quad 2.56844]$$

4.4 Real-time tests

The real-time performance of various operating systems is presented in Table 6. We conducted tests on the proposed algorithm using Windows 10 (Inter i5–11400 at 2.7 GHz, 16.0 GB of RAM) and Ubuntu 18.04 (ROS melodic version). The algorithm, using 10,000 samples of random and feasible postures, was implemented in MATLAB and Visual Studio Code with C++, respectively. Simulation statistics indicate that using Ubuntu 18.04 has significant advantages in both convergence time and stability, compared to Windows10. When implemented in C++, the algorithm achieves an average computing time of 28.54 μs with a standard deviation of 5.45 μs.

Besides, we also conducted a comparative analysis with Kinematics and Dynamics Library (KDL) and IKFast, revealing that our proposed algorithm is faster than KDL, but slower than IKFast. In many cases, however, the IKFast method is generally used as a plug-in package for ROS, but many robotic arms do not support ROS, which means that many robotic arms cannot utilize the IKFast method. In contrast, the proposed algorithm consistently achieves a computing time of 5.45 μs. These results make it well-suited for real-time control and inverse kinematics for 7-DOF anthropomorphic manipulators.

Table 6 Computing times of different inverse kinematics methods

Computing Platform	Test Algorithm	Windows10		Ubuntu18.04	
		Mean /μs	Std.deviation /μs	Mean /μs	Std.deviation /μs
MATLAB	Proposed inverse kinematics	24100	756.3	612.3	60.5
C++	Proposed inverse kinematics	70.12	8.45	28.54	5.45
ROS Melodic	KDL	--	--	982.4	88.4
	IKFast	--	--	8.23	2.78

5. Conclusion

(1) This paper takes the self-made 7-DOF redundant arm as the aim of research and introduces the arm angle parameterization method to realize the inverse kinematics, which simplifies the calculation process. All feasible solutions can be obtained when the arm angle is given. Then, we can use the arm angle parameterization method to solve the arm angle interval satisfying the joint limit and joint singularity, and we do not need to calculate the complicated parameter ${}^{0}R_{3}^{0}$.

(2) After discretizing the workspace, we visualize the manipulability at discrete points, and the manipulability distribution within the workspace of the robotic arm can be described. It becomes possible to intuitively observe how the manipulability is distributed throughout the workspace of the robotic arm.

(3) For the continuous end-effector path, an inverse kinematics algorithm is presented. We tested the proposed algorithm on Windows 10 and Ubuntu 18.04. Simulation statistics indicate that using C++ has significant advantages in both convergence time and stability. Compared to MATLAB, the algorithm achieves an average computing time of 70.12 μs with a standard deviation of 8.45 μs. Besides, we also conducted a comparative analysis with KDL (Kinematics and Dynamics Library) and IKFast. The analysis revealed that the proposed algorithm consistently achieves a computing time of 28.54 μs with a standard deviation of 5.45 μs.

REFERENCES

[1] Hollerbach, J.M. Optimum Kinematic Design for a Seven Degree of Freedom Manipulator, *International Symposium of Robotics Research* **2002**.

[2] Alimchik, K.; Alexandre, A.; Sebastien, G.; Benoit, P. Efficiency evaluation of robots in machining applications using industrial performance measure, *Rob. Comput. Integ. Manuf* **2017**, 48:12–29. https://doi.org/10.1016/j.rcim.2016.12.005

[3] Atawnih, A.; Eorgiou, D.; Doulgeri, Z. Kinematic control of redundant robot with guaranteed joint limit avoidance, *ROBOT AUTON SYST* **2016**, 79:122–131. https://doi.org/10.1016/j.robot.2016.01.006

[4] Chen, Z.; Zeng, Z.;Shu, G.; Chen, Q. Kinematic solution and singularity analysis for 7-dof redundant manipulators with offsets at the elbow, *IEEE Ind. Cyber Phys. Syst* **2018**, 422–427. https://doi.org/10.1109/ICPHYS.2018.8390702

[5] Artemiadis, P. Closed-form inverse kinematic solution for anthropomorphic motion in redundant robot arms, *Adv. Robot, Autom* **2013**, 02 (3):163–171.

[6] Gong, MD.; Li, LD.; Zhang, L. Analytical inverse kinematics and self-motion application for 7-DOF redundant manipulator, *IEEE Access* **2019**, 7:18662-18674. https://doi.org/10.1109/ACCESS.2019.2895741

[7] Hentout, A.; Aouache M.; Maoudj, A.; Akli, I. Human-robot interaction in industrial collaborative robotics: a literature review of the decade 2008-2017, *Adv. Robot* **2019**, 33:764–799. https://doi.org/10.1080/01691864.2019.1636714

[8] Villani, V.; Pini, F.; Leali, F.; Secchi, C.; Fantuzzi, C. Survey on human-robot collaboration in industrial settings: safety, intuitive interfaces and applications, *MECHATRONICS* **2018**, 55:248–266. https://doi.org/10.1016/j.mechatronics.2018.02.009

[9] Elanchezhian, S.; Anitha, G. Software in-loop simulation of a quad tilt rotor unmanned aerial vehicle for transition control, *FAMENA* **2023**, 47(1):21-36. https://doi.org/10.21278/TOF.471033221

[10] Chiacchio, P.; Siciliano, B. A closed-loop Jacobian transpose scheme for solving the inverse kinematics of nonredundant and redundant wrists, *J. Robot. Syst.* **1989**, 6(5): 601–630. https://doi.org/10.1002/rob.4620060507

[11] Huo, XJ., Liu, YW., Jiang, L. Inverse kinematics and control of a 7-DOF redundant manipulator based on the closed-loop algorithm, *Int. J. Adv. Robot. Syst.* **2010**, 7 (4):37–47. https://doi.org/10.5772/10495

[12] Wampler, CW. Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods, *IEEE Trans. Syst. Man Cybern* **1986**, 16(1): 93–101. https://doi.org/10.1109/TSMC.1986.289285

[13] Nakamura, Y.; Hanafusa, H. Inverse kinematic solutions with singularity robustness for robot manipulator control, *ASME J. Dyn. Syst. Meas. Cont* **1986**, 108 (3):163–171. https://doi.org/10.1115/1.3143764

[14] Ortenzi, V.; Marturi, N.; Rajasekaran, V.; Adjigble, M.; Stolkin, R. Singularity-robust inverse kinematics solver for tele-manipulation, *In: Proceedings of the IEEE 15th International Conference on Automation Science and Engineering, Vancouver, Canada*, 25 Aug-28 Aug **2019**,1821–1828. https://doi.org/10.1109/COASE.2019.8842871

[15] Burdick, JW. On the inverse kinematics of redundant manipulators: characterization of the self-motion manifolds. *Adv. Robot* **1989**, 3:25–34. https://doi.org/10.1007/978-3-642-83957-3_3

[16] Lee, S.; Bejczy, A.K.;. Redundant arm kinematic control based on parameterization**.** *IEEE International Conference on Robotics and Automation, Sacramento, CA, USA, **1991**, pp 458-465.*

[17] Zhao, J.D., Xu, Z.C., Zhao, L.L., Li, Y,T., Ma, L.Y., Liu,H. A novel inverse kinematics for solving repetitive motion planning of 7-DOF SRS manipulator. *Robotica* **2023**, 41 (1):392-409. https://doi.org/10.1017/S0263574722001370

[18] Kreutz-Delgado, K., Long, M., Seraji, H. Kinematics analysis of 7-DOF manipulators, *Int. J. Robot. Res* **1992**, 11:469–481. https://doi.org/10.1177/027836499201100504

[19] Ortenzi, V.; Marturi, N.; Rajasekaran, V.;Adjigble, M.; Stolkin, R. Singularity-robust inverse kinematics solver for tele-manipulation, *In: Proceedings of the IEEE 15th International Conference on Automation Science and Engineering, Vancouver, Canada*, 25 Aug-28 Aug **2019**,1821–1828. https://doi.org/10.1109/COASE.2019.8842871

[20] Yan L.; Mu Z.; Xu W.; Analytical inverse kinematics of a class of redundant manipulator based on dual arm-angle parameterization. *IEEE International Conference on Systems, Man, and Cybernetics (SMC). Washington , D.C. ,* **2014**, pp 3744-3749. https://doi.org/10.1109/SMC.2014.6974513

[21] Scherzinger, S.; Roennau, A.; Dillmann, R. Inverse Kinematics with Forward Dynamics Solvers for Sampled Motion Tracking. *In:19th International Conference on Advanced Robotics (ICAR). IEEE, Brazil*, 2 Dec-6 Dec **2020**. https://doi.org/10.1109/ICAR46387.2019.8981554

[22] Toshani, H.; Farrokhi, M. Real-time inverse kinematics of redundant manipulators using neural networks and quadratic programming: A Lyapunov-based approach. *Robot, Auton. Syst* **2014**, 62.6:766-781. https://doi.org/10.1016/j.robot.2014.02.005

[23] Soric, J.; Stanic, M.; Lesicar, T. On neural network application in solid mechanics, *T FAMENA* **2023,** 47(2):45-66. https://doi.org/10.21278/TOF.472053023

[24] Vijavan, S.; Pillai, TP. Application of a machine learning algorithm in a multi stage production system, *T FAMENA* **2023**, 46(1):91-102. https://doi.org/10.21278/TOF.461033121

Submitted: 08.06.2023

Accepted: 29.10.2023

Ph.D. student. WeiXin Chou
Prof. YiWei Liu*
State Key Laboratory of Robotics and Systems, Harbin Institute of Technology, Harbin, Heilongjiang 150080, Harbin, Heilongjiang, P.R.China
*Corresponding author:
lyw_hit@163.com