Research on Optimization of Joint Angle Path for a 6-DoF Robotic Arm Based on Kinematics and Genetic Algorithm

Shuai Xia*, Yibo Wang

International College, Chongqing University of Posts and Telecommunications, Chongqing, China, 400065

* Corresponding Author Email: sx3351658664@outlook.com

Abstract. In industrial applications and automation, robotic manipulators play a crucial role. Several complex robotic systems perform numerous industrial tasks, including painting, welding, assembly, pick-and-place operations, and more. The position of the end-effector and the joint angles are of utmost importance, and their optimization directly impacts the manipulator's motion accuracy and work efficiency. Therefore, this paper proposes an optimization model for the joint angle path of a 6-degree-of-freedom (6-DoF) robotic arm based on forward and inverse kinematics and the Genetic Algorithm (GA). Firstly, using the Denavit-Hartenberg (D-H) parameters and the zero-position state, a simplified diagram of the 6-DoF robotic arm is drawn using the Robotics Toolbox in MATLAB. Then, a single-objective nonlinear optimization model is established with the goal of minimizing both the end-effector error and the total rotated joint angles. Initially, the joint angles and the posture of the robotic arm after grasping are obtained through forward and inverse kinematics analysis. Subsequently, by assessing whether the joint angles of the robotic arm after grasping satisfy the constraint conditions, the Analytic Hierarchy Process (AHP) is employed to determine the weights of minimizing the end-effector error and the total rotated joint angles in the objective function. The GA is then utilized to solve the model. Finally, this paper discusses and analyzes the established model, providing a new perspective for the optimization of the joint angle path of a robotic arm.

Keywords: Forward kinematics, inverse kinematics, GA genetic algorithm

1. Introduction

In recent years, advancements in technology have led to the widespread adoption of robotic arms in industrial production and daily applications. A robotic arm, an automated device composed of multiple links and joints, is extensively utilized in industrial manufacturing, precision operations, hazardous environments, and logistics. Enhancing its accuracy and energy efficiency hinges on optimizing joint-angle trajectories. Through kinematic and dynamic modeling, as well as joint-angle trajectory optimization, the positional precision of the end-effector can be improved, total rotation angles minimized, and operational efficiency ensured in complex tasks.

Current research in robotics focuses intensively on motion control and path planning. Zhao et al. [1] proposed a deep reinforcement learning (DRL)-based method, MAPPO-IK, for inverse kinematics and control of 6-DoF robotic arms. By designing reward mechanisms to minimize end-effector pose errors, this approach addresses the limitations of traditional algorithms in generalization and computational load, achieving real-time solutions and robust obstacle avoidance. However, DRL methods demand substantial training data and resources, exhibit poor interpretability, and rely heavily on reward function design. Nizar Rokbani et al. [2] developed a multi-objective MO-m-PSO algorithm for 5-DOF arms, enhancing particle swarm optimization (PSO) by adapting particle behavior rather than tuning parameters. While effective for educational 3D-printed arms, PSO remains prone to local optima and parameter sensitivity. Research [3] introduced a 3-5-3 polynomial interpolation method with improved PSO for UR5 robotic arm trajectory planning, optimizing motion smoothness and accuracy, though performance depends on interpolation parameters. Another study [4] analyzed Kinova Gen's kinematics through D-H parameters and decoupling methods, validated via simulations, yet lacked practical optimization strategies.

Existing work predominantly addresses kinematics, path planning, and optimization algorithms. Building on these foundations, this paper proposes a 6-DoF robotic arm joint-angle optimization model integrating forward/inverse kinematics with GA. After deriving joint angles and postures via kinematic analysis, AHP determines weightings for minimizing end-effector errors and total joint rotations, forming a single-objective nonlinear model solved by GA. This framework enables multi-objective optimization with adaptability to diverse constraints via AHP.

The study offers a novel approach to joint-angle optimization and provides valuable insights for robotic control in industrial automation, balancing precision and energy efficiency while maintaining operational flexibility.

2. Kinematic analysis of robotic arm

2.1. D-H parameter method

The pose of the end effector of a six-degree-of-freedom robotic arm plays a crucial role in practical applications. To accurately solve for its pose, it is necessary to clarify the geometric relationships between each link. To this end, a method of fixing coordinate systems on the links is adopted, and homogeneous transformation matrices are used to describe the rotation and translation relationships between adjacent link coordinate systems. This standardized modeling method is widely used in the field of robotics and is known as the D-H parameter method.

The D-H parameter method is divided into the Standard D-H parameter method [5] and the Modified D-H parameter method [6] based on differences in parameter definitions. The principles of the two are similar, but their applicable scenarios differ: the Standard D-H parameter method is more suitable for modeling open-chain robotic structures and is less capable of directly handling tree-like or closed-chain systems; the Modified D-H parameter method, on the other hand, extends its applicability and is compatible with the aforementioned complex structural types.

Therefore, in this paper, we have chosen the Modified D-H method [6]. We establish a fixed coordinate system on the preceding joint of each link and comprehensively describe the relationships between the link coordinate systems using four parameters. These parameters are: link twist angle α_{i-1} — the rotation angle around the X_{i-1}-axis, from Z_{i-1} to Z_i. Link length a_{i-1} — the translational distance along the X_{i-1}-axis, from Z_{i-1} to Z_i. Joint angle θ_i — the rotation angle around the Z_i-axis, from X_{i-1} to X_i. Link offset d_i — the translational distance along the Z_i-axis, from X_{i-1} to X_i.

All of the joints in the robotic arm under study in this research are revolving joints, and Table 1 displays the pertinent D-H characteristics

Table 1. D-H parameter table.

Joint i	$a_i/(mm)$	$\alpha_i/(mm)$	$d_i/(mm)$	θ_i Joint transformation range/(°)
1	0	0	600	-160~160
2	300	-90	0	-150~15
3	1200	0	0	-200~80
4	300	-90	1200	-180~180
5	0	-90	0	-120~120
6	0	-90	0	-180~180

The D-H parameter method uses four parameters to represent the joint and the connecting rod constituting the joint: the joint angle θ_j, link offset d_j, link length a_j, and twist angle α_j. During the movement of the robotic arm, these four parameters also change accordingly, and we can use homogeneous transformation matrices to solve the transformation relationship between adjacent coordinate systems.

The transformation process from link coordinate system $\{j-1\}$ to $\{j\}$ is called basic rotation and translation, which can be represented by a 4×4 homogeneous transformation matrix as:

$$
{}^{j-1}A_j\left(\theta_j,d_j,a_j,\alpha_j\right)=T_{Rz}\left(\theta_j\right)T_z\left(d_j\right)T_x\left(a_j\right)T_{Rx}\left(\alpha_j\right)
=\begin{pmatrix}\cos\theta_j & -\sin\theta_j\cos\alpha_j & \sin\theta_j\sin\alpha_j & a_j\cos\theta_j\\ \sin\theta_j & \cos\theta_j\cos\alpha_j & -\cos\theta_j\sin\alpha_j & a_j\sin\theta_j\\ 0 & \sin\alpha_j & \cos\alpha_j & d_j\\ 0 & 0 & 0 & 1\end{pmatrix} \tag{1}
$$

Where, the ${}^{j-1}A_j$ matrix is the homogeneous transformation matrix from coordinate system $\{j-1\}$ to $\{j\}$.

2.2. Forward kinematics analysis of robotic arm

According to reference [7], the forward kinematics of a robotic arm is the problem of transforming from joint space to Cartesian space. When the robotic arm is in motion, we can obtain the final end-effector pose parameters and Cartesian space coordinates through the joint variables θ_j. Using the aforementioned homogeneous transformation matrix, we can derive the forward kinematics equation of the robotic arm:

$$
{}^0_6T={}^0_1T\,{}^1_2T\,{}^2_3T\,{}^3_4T\,{}^4_5T\,{}^5_6T=\begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x\\ r_{21} & r_{22} & r_{23} & p_y\\ r_{31} & r_{32} & r_{33} & p_z\\ 0 & 0 & 0 & 1\end{bmatrix} \tag{2}
$$

Where $(r_{11},r_{21,}r_{31})$ is the approach vector, (r_{12},r_{22},r_{32}) is the orientation vector, (r_{13},r_{23},r_{33}) is the normal vector, and (p_x,p_y,p_z) is the end-effector coordinate.

Here, Co_i represents $\mathrm{Cos}\,\theta_i$, Si_i represents $\mathrm{Sin}\,\theta_i$, Si_{23} represents $\mathrm{Sin}(\theta_2+\theta_3)$, and Co_{23} represents $\mathrm{Cos}(\theta_2+\theta_3)$. The expressions for each parameter in the forward kinematics equation are as follows:

$$
\begin{cases}
r_{11}=co_1\left[co_{23}\left(co_4co_5co_6-si_4si_6\right)-si_{23}si_5co_6\right]+si_1\left(si_4co_5co_6+co_4si_6\right)\\
r_{21}=si_1\left[co_{23}\left(co_4co_5co_6-si_4si_6\right)-si_{23}si_5co_6\right]-co_1\left(si_4co_5co_6+co_4si_6\right)\\
r_{31}=-si_{23}\left(co_4co_5co_6-si_4si_6\right)-co_{23}si_5co_6\\
r_{12}=co_1\left[-co_{23}\left(co_4co_5si_6+si_4co_6\right)+si_{23}si_5si_6\right]+si_1\left(-si_4co_5si_6+co_4co_6\right)\\
r_{22}=si_1\left[-co_{23}\left(co_4co_5si_6+si_4co_6\right)+si_{23}si_5si_6\right]-co_1\left(-si_4co_5si_6+co_4co_6\right)\\
r_{32}=si_{23}\left(co_4co_5si_6+si_4co_6\right)+co_{23}si_5si_6\\
r_{13}=-co_1\left(co_{23}co_4si_5+si_{23}co_5\right)-si_1si_4si_5\\
r_{23}=-si_1\left(co_{23}co_4si_5+si_{23}co_5\right)+co_1si_4si_5\\
r_{33}=si_{23}co_4si_5-co_{23}co_5\\
p_x=co_1\left(co_{23}a_3-si_{23}d_4+co_2a_2+a_1\right)\\
p_y=si_1\left(co_{23}a_3-si_{23}d_4+co_2a_2+a_1\right)\\
p_z=-si_{23}a_3-co_{23}d_4-si_2a_2
\end{cases} \tag{3}
$$

2.3. Inverse kinematics analysis of robotic arm

Based on reference [7], the inverse kinematics problem of a robotic arm involves solving for the joint angles θ_j of the manipulator's joints given the Cartesian space coordinates and orientation of the end-effector, where the following parameters $r_{11},r_{12},r_{13},r_{21},r_{22},r_{23},r_{31},r_{32},r_{33},p_x,p_y,p_z$ are constants. Denoting the joint angles of the six joints as θ_1、θ_2、θ_3、θ_4、θ_5、θ_6, we have:

Denoting the first joint angle as θ_1, there are two solutions for θ_1:

$$\begin{cases} \theta_1 = \arctan 2(p_y, p_x) \\ \theta_1 = \arctan 2(-p_y, -p_x) \end{cases} \tag{4}$$

Denoting the third joint angle as θ_3, θ_3 can be expressed as:

$$\theta_3 = \arctan 2(a_3, d_4) - \arctan 2(k, \pm\sqrt{a_3^2 + d_4^2 - k^2}) \tag{5}$$

Where $k = \dfrac{\left(co_1 p_x + si_1 p_y - a_1\right)^2 + p_z{}^2 - a_2{}^2 - a_3{}^2 - d_4{}^2}{2a_2}$

Denoting the second joint as θ_2, θ_2 can be expressed as:

$$\theta_2 = \theta_{23} - \theta_3 = \arctan 2(m, n) - \theta_3 \tag{6}$$

Where

$$\begin{aligned} m = si_{23} = \frac{\left(-a_3 - a_2 co_3\right) p_z - \left(d_4 - a_2 si_3\right)\left(co_1 p_x + si_1 p_y - a_1\right)}{\left(co_1 p_x + si_1 p_y - a_1\right)^2 + p_z{}^2} \\ n = co_{23} = \frac{\left(a_2 si_3 - d_4\right) p_z + \left(a_3 + a_2 co_3\right)\left(co_1 p_x + si_1 p_y - a_1\right)}{\left(co_1 p_x + si_1 p_y - a_1\right)^2 + p_z{}^2} \end{aligned} \tag{7}$$

Denoting the fourth joint angle as θ_4, θ_4 can be expressed as:

When $s_5 \neq 0$:

$$\theta_4 = \arctan 2(-r_{13} si_1 + r_{23} co_1, -r_{13} co_1 co_{23} - r_{23} si_1 co_{23} + r_{33} si_{23}) \tag{8}$$

Where

$$\begin{cases} r_{13}(co_1 co_{23} co_4 + si_1 si_4) + r_{23}(si_1 co_{23} co_4 - co_1 si_4) - r_{33} si_{23} co_4 = -si_5 \\ -r_{13} co_1 si_{23} - r_{23} si_1 si_{23} - r_{33} co_{23} = co_5 \end{cases} \tag{9}$$

When $s_5 = 0$, θ_4 and θ_6 jointly determine the end-effector orientation. Since the fourth and sixth joints are coaxial, θ_4 can be expressed as:

$$\theta_4' = \theta_4 + 180^\circ \tag{10}$$

Denoting the fifth joint angle as θ_5, θ_5 can be expressed as:

$$\theta_5 = \arctan 2(si_5, co_5) \tag{11}$$

Where

$$\begin{cases} r_{13}(co_1 co_{23} co_4 + si_1 si_4) + r_{23}(si_1 co_{23} co_4 - co_1 si_4) - r_{33} si_{23} co_4 = -si_5 \\ -r_{13} co_1 si_{23} - r_{23} si_1 si_{23} - r_{33} co_{23} = co_5 \end{cases} \tag{12}$$

Corresponding to θ_4', θ_5' can be expressed as:

$$\theta_5' = -\theta_5 \tag{13}$$

Denoting the sixth joint angle as θ_6, θ_6 can be expressed as:

$$\theta_6 = \arctan 2(si_6, co_6) \tag{14}$$

Where

$$\begin{cases} r_{11}(-co_1co_{23}si_4+si_1co_4)+r_{21}(-si_1co_{23}si_4-co_1co_4)+r_{31}si_{23}si_4=si_6 \\ r_{12}(-co_1co_{23}si_4+si_1co_4)+r_{22}(-si_1co_{23}si_4-co_1co_4)+r_{32}si_{23}si_4=co_6 \end{cases} \tag{15}$$

Corresponding to $\theta_4^{'}$, $\theta_6^{'}$ can be expressed as:

$$\theta_6^{'}=\theta_6+180^{\circ} \tag{16}$$

Therefore, there are 8 sets of solutions for the inverse kinematics of the robot. However, in practical motion, the optimal solution needs to be selected. Generally, the principle of minimizing the change in joint rotation is used to complete the actual solution.

3. Robot arm 6DoF model

3.1. Robot arm pose

To calculate the objective function mentioned later, the Cartesian space coordinate point (p_x, p_y, p_z) of the robotic arm's end-effector can be obtained using forward kinematics, given the six joint angles, link torque angles, link lengths, and link offsets of the robotic arm.

$$ {}_6^0T={}_1^0T{}_2^1T{}_3^2T{}_4^3T{}_5^4T{}_6^5T=\begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{17}$$

To determine the weights mentioned later, the six joint angles of the robotic arm can be calculated using inverse kinematics, given the link torque angles, link lengths, joint angles, link offsets, and the Cartesian space coordinate point (p_x, p_y, p_z) of the robotic arm's end-effector.

3.2. Analytic Hierarchy Process to Determine Weights

In the modeling of robotic arm motion, it is necessary to simultaneously optimize the minimization of end-effector error and the planning of joint angle trajectories. Based on inverse kinematics analysis of target points, the target angles for each joint can be calculated when the robotic arm's grasping action is triggered. Due to the constraints on the range of joint angles, optimization strategies need to be developed for different scenarios:

(1) Calculating angles within the constraint range: prioritizing the minimization of the total change in the six joint angles (energy consumption optimization);

(2) Calculating angles outside the constraint range: focusing on minimizing the end-effector positioning error (accuracy optimization).

A weight function is established through the AHP [8] to quantitatively balance the optimization priorities of the total joint angle change and the end-effector error.

The AHP [8] decomposes complex decision-making problems hierarchically, combining qualitative and quantitative analyses. With the total joint angle change (A) and the end-effector error (B) as the criterion layer, a two-dimensional evaluation system is constructed. Under the condition of these two indicators, the judgment matrix naturally satisfies complete consistency, and pairwise comparisons are conducted using a scaling method. If the emphasis is on optimizing the total joint angle change, the judgment matrix is designed as follows:

$$A=\begin{bmatrix} 1 & 3 \\ \frac{1}{3} & 1 \end{bmatrix} \tag{18}$$

If the optimization of the end-effector error is given more attention, the judgment matrix is:

$$B=\begin{bmatrix}1 & \frac{1}{3}\\ 3 & 1\end{bmatrix} \tag{19}$$

Finally, the weights are approximately calculated by solving for the eigenvectors, with the weight of the indicator being λ_1 and the weight of the other indicator being λ_2 .

3.3. Establishment of 6DoF Model Based on Nonlinear Programming

3.3.1. Establishment of objective function

When a robotic arm receives a task to grasp an object, it typically involves adjusting the joint angles while aiming to minimize both the end-effector error after grasping and the sum of the changes in the six joint angles. Therefore, the joint angles $\theta_1 \sim \theta_6$ of the six joints are the decision variables, and the objective function is established to minimize the weighted sum of the end-effector error and the sum of the changes in the six joint angles:

$$\min F=\lambda_1\sum_{i=1}^{6}\left|(\theta_i-\theta_i^{initial})\right|+\lambda_2\sqrt{(p_x(\theta)-x_{end})^2+(p_y(\theta)-y_{end})^2+(p_z(\theta)-z_{end})^2} \tag{20}$$

Where $\theta_i^{initial}$ represents the zero-position state angles of the joints, λ_1、λ_2 represents the weights assigned by the Analytic Hierarchy Process, and x_{end}、y_{end}、z_{end} are the initial coordinates of the target point x、y、z, which can be obtained by substituting $\theta_i^{initial}$ into the forward kinematics. $p_x(\theta), p_y(\theta), p_z(\theta)$ Represents the position of the end-effector, which can be obtained by substituting θ_i into the forward kinematics.

3.3.2. Establishment of constraints

Typically, the six joint angles θ_i have a range of motion, and constraint conditions are established based on the D-H parameter table:

$$s.t.\begin{cases}-160^\circ \le \theta_1 \le 160^\circ\\ -150^\circ \le \theta_2 \le 15^\circ\\ -200^\circ \le \theta_3 \le 80^\circ\\ -180^\circ \le \theta_4 \le 180^\circ\\ -120^\circ \le \theta_5 \le 120^\circ\\ -180^\circ \le \theta_6 \le 180^\circ\end{cases} \tag{21}$$

In summary, the final 6DoF model based on nonlinear programming is obtained as follows:

$$\begin{gathered}\min F=\lambda_1\sum_{i=1}^{6}(\theta_i-\theta_i^{initial})+\lambda_2\sqrt{(p_x(\theta)-x_{end})^2+(p_y(\theta)-y_{end})^2+(p_z(\theta)-z_{end})^2}\\ s.t.\begin{cases}-160^\circ \le \theta_1 \le 160^\circ\\ -150^\circ \le \theta_2 \le 15^\circ\\ -200^\circ \le \theta_3 \le 80^\circ\\ -180^\circ \le \theta_4 \le 180^\circ\\ -120^\circ \le \theta_5 \le 120^\circ\\ -180^\circ \le \theta_6 \le 180^\circ\end{cases}\end{gathered} \tag{22}$$

4. Results

4.1. Drawing of a Six Degree of Freedom Diagram for a Robot Arm

For the depiction of the 6-DoF robotic arm diagram [9], this paper utilizes the Robotic Toolbox in MATLAB. By inputting the D-H parameter table and the zero-state configuration, the 6-DoF robotic arm diagram can be generated, as shown in Figure 1.

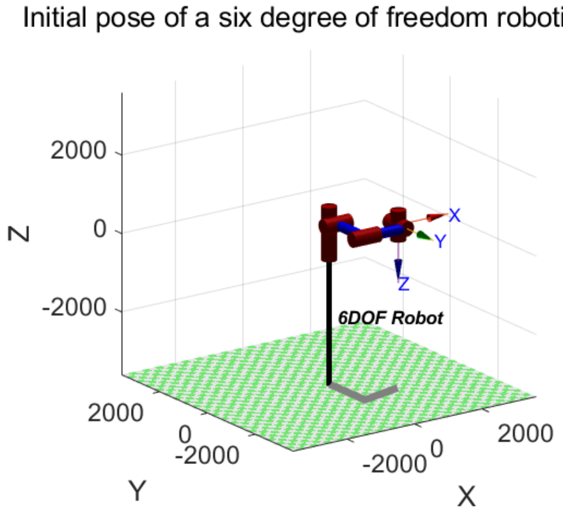

Figure 1. Initial pose of a six degree of freedom robotic arm.

4.2. The solution of weights in Analytic Hierarchy Process

Taking the center of the robotic arm base as the origin and using the target grasping point (1500mm, 1200mm, and 200mm) as an example, this paper conducts an inverse kinematics analysis of the target point. The analytical solution is solved using MATLAB by substituting the coordinates of the target point, and the joint angles obtained are as shown in Table 2:

Table 2. The various joint angles of the six robotic arms

θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
38.6598°	-6.4810°	-3.2787°	0.0000°	9.7597°	-31.6187°

After solving, it is found that all six joint angles of the robotic arm are within the range of joint angle changes required by the problem. Therefore, the focus is on minimizing the total rotation angle of the six joints. Next, the AHP is used to assign weights to these two indicators.

For the AHP weight assignment, the judgment matrix is constructed as follows:

$$A = \begin{Bmatrix} 1 & 3 \\ \frac{1}{3} & 1 \end{Bmatrix} \tag{23}$$

Using MATLAB, the weights for the total rotation angle λ_1、 and the end-effector error λ_2 are solved as shown in Table 3:

Table 3. Indicator weight.

λ_1	λ_2
0.7500	0.2500

4.3. Solution of 6DoF Model

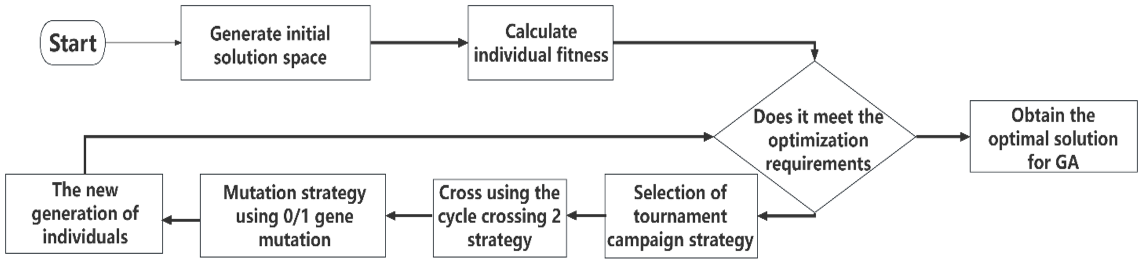

Figure 2. Genetic Algorithm Flowchart.

For the 6DoF model, this paper employs the Genetic Algorithm [10] to solve it, as shown in Figure 2. The specific steps are as follows:

Step 1: Randomly generate initial solutions, encode them, and calculate the fitness of each individual.

Step 2: Select individuals using a tournament selection strategy, which involves randomly selecting N individuals from the population. Due to varying fitness values, the fittest individuals survive and are selected to enter the next generation.

Step 3: Perform crossover between populations using a two-point crossover strategy.

Step 4: Complete population mutation using a 0/1 gene mutation strategy.

Step 5: If the optimization requirements are met at this point, record the optimal solution. If not, repeat Steps 2-4.

Through these steps, the minimized total rotation angle and end-effector error are obtained, as shown in the table below:

Table 4. Minimize the total rotation angle and end error.

Total rotation angle	End error
196.6272	0.8962

From the results in Table 4, the following conclusions can be drawn: The total rotation angle of the joints solved by the 6DoF model is 196.6272°, and the end-effector error is 0.8962 mm.

5. Conclusions

Based on existing research in the field of robotic arms, this paper proposes a new method for optimizing the joint angle path of a 6-degree-of-freedom robotic arm. By utilizing forward and inverse kinematics analysis, the initial and post-grasping poses and joint angles of the robotic arm are obtained. The AHP is then employed to determine the weights of minimizing the end-effector error and total joint rotation in the objective function, thereby establishing a single-objective nonlinear optimization model. Subsequently, the genetic algorithm is used to solve the model. The final solution yields a total rotational angle of the joints of 196.6272° and an end-effector error of 0.8962mm, achieving a comprehensive optimization of both the total rotational angle and the end-effector error. The model presented in this paper is highly accurate, comprehensive, adaptable, and verifiable. It not only provides a new perspective for optimizing the joint angle path of robotic arms but also offers valuable insights into robotic arm control in industrial applications and automation fields.

In the future, we will continue to deepen our research, exploring advanced algorithms and technologies to improve the motion accuracy and efficiency of robotic arms. At the same time, we will closely align with the actual needs of industrial scenarios, continuously optimizing model parameters and adaptability to enhance their response capabilities to complex working conditions and tasks.

References

[1] Zhao C, Wei Y, Xiao J, et al. Inverse kinematics solution and control method of 6-degree-of-freedom manipulator based on deep reinforcement learning[J]. Scientific Reports, 2024, 14(1): 12467...

[2] Rokbani N, Neji B, Slim M, et al. A multi-objective modified PSO for inverse kinematics of a 5-DOF robotic arm [J]. Applied Sciences, 2022, 12(14): 7091.

[3] Zhang J, Wang P. Improved Particle Swarm Optimization for Trajectory Planning in a Six-Degree-of-Freedom Robotic Arm[C]//2024 17th International Conference on Advanced Computer Theory and Engineering (ICACTE). IEEE, 2024: 329-331.

[4] Raghunathan A V, Sivakumar A K, Palaniswamy A M. Kinematic study of curved 6 DOF arm[C]//AIP conference proceedings. AIP Publishing, 2024, 3035(1).

[5] Singh A, Singla A, Soni S. Extension of DH parameter method to hybrid manipulators used in robot-assisted surgery [J]. Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine, 2015, 229(10): 703-712.

[6] Stewart J J P. Optimization of parameters for semiempirical methods VI: more modifications to the NDDO approximations and re-optimization of parameters [J]. Journal of molecular modeling, 2013, 19: 1-32.

[7] Huang Jie. Research on obstacle avoidance path planning of six-degree-of-freedom manipulator [D].Jiangnan University,2023.DOI:10.27169/d.cnki.gwqgu.2023.000349.

[8] Yang C, Zheng X, Dai C, et al. Risk assessment of coal supply chain based on analytic hierarchy process and fuzzy comprehensive evaluation [J]. Heliyon, 2025, 11(4):e42629-e42629.

[9] Lu Jiahao, Ping Xueliang, Li Zhaoyang. Research on motion simulation of articulated robots based on MATLAB Robotic Toolbox [J]. Machine Tool & Hydraulics, 2017, 45(17): 60-62+81.

[10] Eremeev V A. A genetic algorithm with tournament selection as a local search method [J]. Journal of applied and industrial mathematics, 2012, 6(3):286-294.