**Journal of Computing and Electronic Information Management**
ISSN: 2413-1660 | Vol. 20, No. 3, 2026

# Collision-Free Trajectory Optimization of Manipulator Based on Black-Winged Kite Algorithm

**Zhenhua Xue**

Academy of Interdisciplinary Studies, Guangdong Polytechnic Normal University, Guangzhou 510665, China

**Abstract:** Aiming at the requirement of global obstacle avoidance for each link of the manipulator, an obstacle avoidance trajectory planning method based on the Black-winged Kite Algorithm (BKA) was proposed. Firstly, the envelope model of the obstacle and the manipulator was simplified, and the collision detection problem was transformed into the problem of calculating whether the distance between the center of the obstacle envelope model and each joint was too large. Secondly, the trajectory planning was carried out by using cubic polynomial, and the trajectory curve was adjusted by adjusting the cubic coefficient to meet the obstacle avoidance demand. Finally, the BKA was used to optimize the time of the motion trajectory, and the running time of the manipulator was reduced under the premise of meeting the working demand. The simulation results of MATLAB showed that this method could plan a trajectory that could meet the collision-free requirements of each link of the robot arm, and the movement time of the robot arm was greatly shortened, which could ensure that the robot arm could complete the task smoothly and efficiently in actual production.

**Keywords:** Robotic arm; Collision-free; Time optimal trajectory planning; Black-winged Kite Algorithm.

## 1. Introduction

With the development of machine vision, sensors and other related technologies, the performance of robotic arms has also been significantly improved, and it has been widely used in many fields such as industry, medical treatment and aerospace. Due to the complex application environment, it is necessary to study the obstacle avoidance trajectory planning in order to realize the safe and stable operation of the manipulator[6]. Nowadays, robotic arms have been applied to automated production in various fields, especially in modern production processes such as assembly, welding, testing, loading and unloading [7]. Trajectory planning [8], as the basis of manipulator control, can design the time, speed and path points of manipulator movement according to task requirements to complete various tasks.

There are many commonly used obstacle avoidance algorithms for robotic arms, such as A* algorithm, Rapidly Exploring Random Tree (RRT) algorithm and artificial potential field method[1]. However, these algorithms mainly focus on obstacle avoidance path planning at the end of the manipulator, and lack constraints on the movement of other linkages of the manipulator. With the development of computer technology, various intelligent optimization algorithms have been widely studied and applied in obstacle avoidance trajectory planning. Wang Xian et al. proposed an obstacle-avoidance trajectory planning method based on genetic algorithm and B-spline curve[2]. The control point position of B-spline curve was optimized by genetic algorithm to achieve curve trajectory adjustment, but the generalization and verification in three-dimensional space were not carried out. Chen et al. proposed an obstacle avoidance method for intermediate points of serial robots. The intermediate points were selected to represent the trajectory of the robot arm in segments, and then the position of the intermediate points was optimized by genetic algorithm to meet the obstacle avoidance requirements and optimize the kinematic performance of the robot arm at the same time[3] .

Aiming at the obstacle avoidance requirements of the robot arm, this paper takes the total running time of the end of the robot arm as short as possible, and the joint Angle, angular velocity and angular acceleration curves of the robot arm are continuously smooth as optimization goals, and proposes a time-optimal trajectory planning algorithm of the robot arm based on the Black-winged Kite Algorithm (BKA) to optimize the running time of the robot arm.

## 2. Problem description

Because AUBO-i5 robot arm is small in size, light in weight, can be installed quickly, and can cooperate with humans or other automation equipment, this paper chooses AUBO-i5 robot arm as a model to study the time optimal trajectory planning algorithm of the robot arm. As shown in Fig 1, the AUBO-i5 robotic arm has 6 joints, each of which is one degree of freedom. Three degrees of freedom (joint 4, joint 5 and joint 6) can confirm the position information of the end of the robotic arm, and the other three degrees of freedom (joint 1, joint 2 and joint 3) can determine the pose information of the robotic arm. The six degrees of freedom of the AUBO-i5 robot arm enable the robot arm to reach any position within the operable range in any position in three-dimensional space. According to the initial parameter Settings of the robot arm, the DH (Denavit Hartenberg) parameters of the AUBO-i5 robot arm are listed in Table 1.

The forward kinematics analysis of the robot arm is a process of obtaining the end-pose matrix by multiplying the transformation matrix between the coordinate systems of the two links[4]. According to the parameters in DH table, the transformation matrix between adjacent coordinate systems can be established by means of coordinate transformation, and then the forward kinematics equation of the manipulator can be obtained. Here is the coordinate transformation expression of the coordinate system between the two adjacent joints.

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

By substituting the DH parameters of the AUBO-i5 manipulator into equation (1), the transformation matrix of the coordinate system of each adjacent joint can be calculated. The rotation transformation matrix from the base to the end actuator of the AUBO-i5 manipulator is obtained by multiplying the transformation matrices of the coordinate systems of the above adjacent joints successively according to equation (2).

$$ {}^{0}_{6}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T\,{}^{5}_{6}T \tag{2} $$

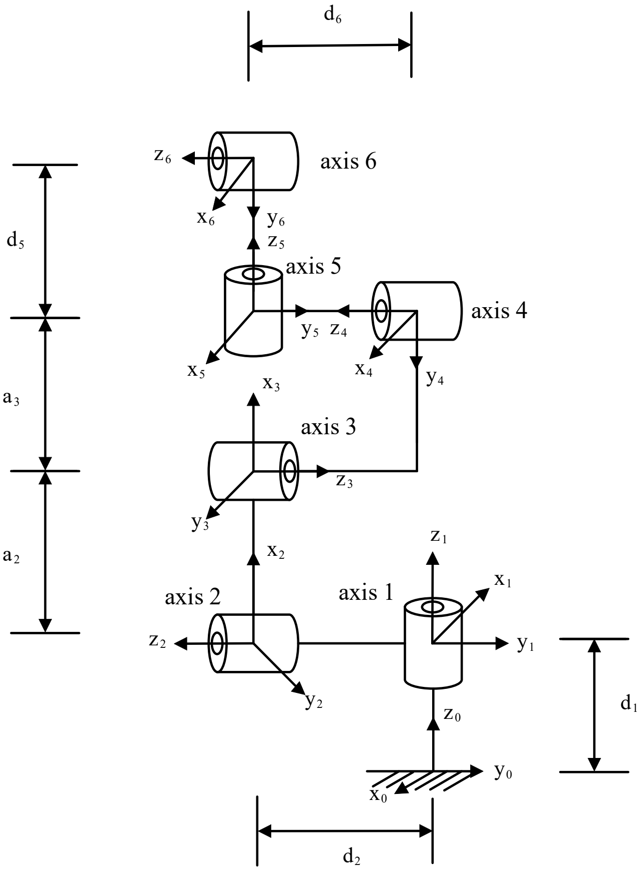

**Fig. 1** Schematic diagram of joints of AUBO-i5 robotic arm

The main purpose of collision detection is to determine whether the collision conditions are satisfied between the joints of the manipulator and the obstacles. Because of the various shapes of obstacles, it is difficult to make concrete modeling expression, so a slightly larger simple geometry can be used to represent the original obstacles completely. Since the joint shape of the manipulator arm is approximately circular, the enveloping sphere envelope is used for it. However, obstacles are difficult to be described by a specific model due to their different structures, so the enveloping sphere envelope is also used for obstacles.

The elements in row 1, 2, and 3 of column 4 of the coordinate transformation matrix (represented by ${}^{0}_{i}T(1,4)$, ${}^{0}_{i}T(2,4)$, ${}^{0}_{i}T(3,4)$) represent the $x$, $y$, and $z$ coordinates of the corresponding joints, respectively. Remember that the coordinates of the center of the sphere surrounded by the obstacle are O$(o_x, o_y, o_z)$, so finding the distance between the joint of the robot arm and the obstacle can be converted to finding the distance between point O and $x$, $y$, and $z$.

When it is detected that the distance between the obstacle and the joint of the manipulator arm is less than the sum of the envelope radii of the two envelope bodies, it is regarded as collision interference; otherwise, it is regarded as no collision interference.

**Table 1.** DH parameters of AUBO-i5 robotic arm

| Link $i$ | Articular Angle $\theta_i$ / (°) | Connecting rod offset $d_i$ /cm | Connecting rod length $a_i$ /cm | Torsion Angle $\alpha_i$ / (°) |
|---|---|---|---|---|
| 1 | 180 | 9.85 | 0 | 0 |
| 2 | -90 | 12.15 | 0 | -90 |
| 3 | 0 | 0 | 40.8 | 180 |
| 4 | -90 | 0 | 37.6 | 180 |
| 5 | 0 | 10.25 | 0 | -90 |
| 6 | 0 | 9.4 | 0 | 90 |

# 3. Black-winged kite algorithm

Black-winged kite optimization algorithm is a simple and effective meta-heuristic optimization method[5]. Based on the attack and migration behavior of black-winged kites, the attack and migration stages of BKA were modeled. BKA algorithm mainly consists of three steps:

Initialize. BKA is initialized randomly:

$$ X_i = BK_{lb} + rand(BK_{ub} - BK_{lb}) \tag{3} $$

Aggressive behavior. As predators of small grassland mammals and insects, black-winged kites Angle their wings and tails according to wind speed in battle, hover silently to observe prey, and then dive quickly to attack. The strategy includes different attack behaviors for global exploration and search.

$$ y_{t+1}^{i,j} = \begin{cases} y_t^{i,j} + n(1+\sin(r)) \times y_t^{i,j}, p < r \\ y_t^{i,j} + n \times (2r-1) \times y_t^{i,j}, else \end{cases} \tag{4} $$

Migration behavior. Bird migration is a complex behavior that is influenced by environmental factors such as climate and food availability

$$ y_{t+1}^{i,j} = \begin{cases} y_t^{i,j} + C(0,1) \times (y_t^{i,j} - L_t^{j}), F_i < F_{ri} \\ y_t^{i,j} + C(0,1) \times (L_t^{j} - m \times y_t^{i,j}), else \end{cases} \tag{5} $$

$$ m = 2 \times \sin(r + \pi/2) \tag{6} $$

The BKA pseudocode is shown in Table 2.

**Table 2.** BKA pseudocode

```
Algorithm: Black-winged kite algorithm
Input: The population size pop, maximum number of iterations T, and variable dimension dim
Algorithm: Black-winged kite algorithm
Output: The best quasi-optimal solution obtained by BKA for a given optimization problem.
Initialization phase
Initialization of the position of Black-winged kites and evaluation of the objective fiction
Calculate the fitness value of each Black-winged kite
    while ($t< T$) do
/* Attacking behavior */
    if $p<r$
        $y_{t+1}^{i,j} = y_t^{i,j} + n(1+\sin(r)) \times y_t^{i,j}$
    else if do
        $y_{t+1}^{i,j} = y_t^{i,j} + n \times (2r-1) \times y_t^{i,j}$
    end if
/* Migration behavior */
    if $F_i < F_{ri}$ do
        $y_{t+1}^{i,j} = y_t^{i,j} + C(0,1) \times (y_t^{i,j} - L_t^{j})$
    else if do
```

Continued **Table 2.** BKA pseudocode

```
            $y_{t+1}^{i,j} = y_t^{i,j} + C(0,1) \times (L_t^j - m \times y_t^{i,j})$
        end if
    /*Select the best individual*/
        if $y_{t+1}^{i,j} < L_t^j$
            $Xbest = y_{t+1}^{i,j}, Fbest = f(y_{t+1}^{i,j})$
        else if do
            $Xbest = L_t^j, Fbest = f(L_t^j)$
        end if
    end while
Return $Xbest$ and $Fbest$
```

## 4. Simulation and discussion

This paper is based on MATLAB platform for simulation verification. Set the cross section radius $R_1$ of the cylinder envelope corresponding to the joint of the manipulator arm as 5 cm and the obstacle envelope radius $R_O$ as 10 cm, then the safety distance$R = R_1 + R_O = 15cm$. Mechanical arm set to initial and final position $\theta_0 = [0\ 0\ 0\ 0\ 0\ 0]$ and $\theta_f = [0\ 0\ 0\ 0\ 0\ 0]$, the obstacles of center set as O (30,70,20).

The robot model was built by MATLAB Robotics Toolbox and verified by simulation. The motion trajectory of the end of the robot arm in three-dimensional space obtained by simulation was shown in Fig 2. It can be seen from Fig 2 that the trajectory of the end of the robot arm is smooth and can better meet the obstacle avoidance requirements.

The Angle, angular velocity and angular acceleration of the robot arm at each interpolation point were recorded, and the parameter change curve of each joint of the robot arm as shown in Fig. 3 could be drawn.

It can be seen from Fig 3 that the curves of Angle, angular velocity and angular acceleration of each joint are smooth without abrupt change, and all indicators are within the kinematic constraint range of the robot arm, which can meet the performance requirements of the robot arm.

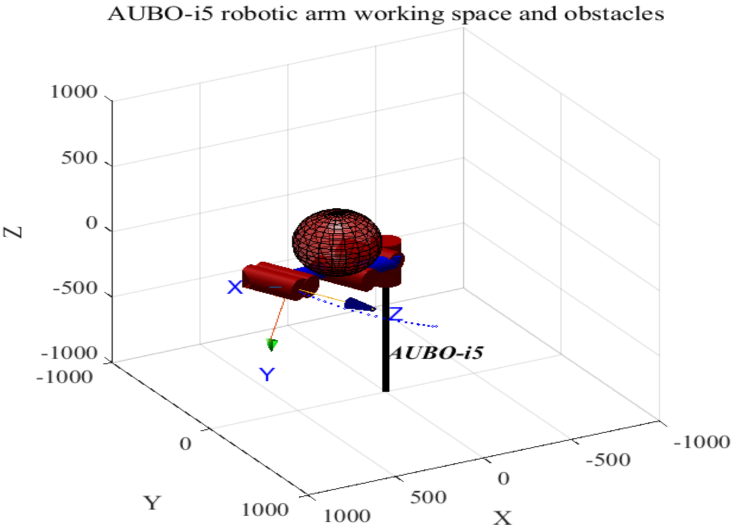

**Fig. 2** The trajectory of the end of the robotic arm

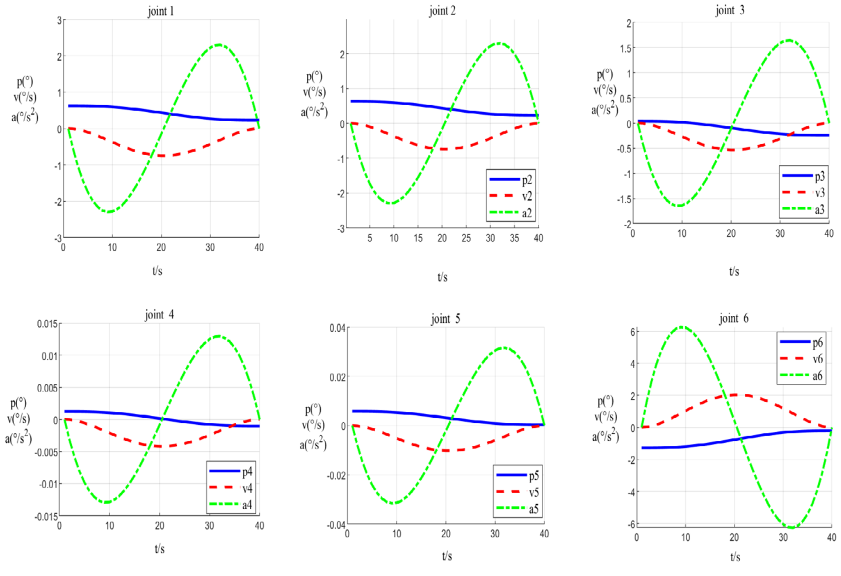

**Fig. 3** Curve of Angle, velocity and acceleration of each joint

## 5. Conclusion

In this paper, based on cubic polynomial trajectory planning, the BKA is used to optimize the trajectory adjustment. The simulation results show that the proposed method can plan a trajectory that meets the collision-free requirements of each joint of the robot arm, and the movement time of the robot arm is greatly shortened, which can ensure that the robot arm can complete the task smoothly and efficiently in actual production.

# References

[1] Yu xia C, Tian yu W, and Xian lun W. Path planning of robotic arm based on velocity potential field method[J]. Machine Tool & Automation Technology, 2022(07):9-12.

[2] Xian W, Wei S, Shulin S, et al. Smooth obstacle avoidance path planning based on genetic

[3] algorithm and B-spline curve [J]. Computer Systems & Applications, 2012, 21(02):65-71.

[4] Chen Z, Su W, Li B, et al. An intermediate point obstacle avoidance algorithm for serial robot[J].Advance in Mechanical Engineering,2018,10(5):1-15

[5] Yuhang H, Shang ying Z, Xiaoyan Z. Simulation Study on the Kinematics of 6R Manipulator Based on MATLAB[J]. Manufacturing Technology and Machine Tools, 2022(11):27-32.

[6] Wang, J, Wang, Wc, Hu, Xx, et al. Black-winged kite algorithm: a nature-inspired meta-heuristic for solving benchmark functions and engineering problems. Artificial Intelligence Review (2024) 57:98

[7] Varshney, Vikas, et al. "Design and development of intellectual robotic arm." 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom). IEEE, 2019.

[8] Yuxuan H. Review of trajectory planning for industrial robots[C]//Journal of Physics: Conference Series. IOP Publishing, 2021, 1906(1): 012041.

[9] Wei B, Shu S H, Zhang Y, et al. Kinematics Analysis and Trajectory Planning Simulation of Manipulator Based on AUBO-I5[C]//2020 4th Annual International Conference on Data Science and Business Analytics (ICDSBA). IEEE, 2020: 72-75.