Int. J. Metrol. Qual. Eng. **10**, 11 (2019)

https://doi.org/10.1051/ijmqe/2019011

International Journal of Metrology and Quality Engineering

Available online at:
www.metrology-journal.org

Research Article
Open Access

Motion planning optimization of trajectory path of space manipulators

Dong Qiao*

College of Architecture and Geomatics Engineering, Shanxi Datong University, Datong, Shanxi 037009, PR China

Received: 11 July 2019 / Accepted: 30 August 2019

Abstract. With the development of the aerospace industry, the work carried out inside and outside the weightless space station is becoming more and more complicated. In order to ensure the safety of astronauts, space manipulators are used for operation, but it will disturb the space station that is a base during work. In order to solve the above problems, in this paper, the planning method of the motion trajectory of manipulators, the motion model of manipulators and the particle swarm optimization (PSO) algorithm used for optimizing the trajectory are briefly introduced, the multi-population co-evolution method is used to improve the PSO algorithm, and the above two algorithms are used to optimize the motion trajectory of the floating pedestal space manipulator with three free degrees of rotation in the same plane by the matrix laboratory (MATLAB) software. It is compared with genetic algorithm. The results show that the improved PSO algorithm can converge to a better global optimal fitness with fewer iterations compared with the traditional PSO algorithm and genetic algorithm. The obtained motion trajectory optimized by the improved PSO algorithm has less disturbances to the pedestal posture, and less time is required to achieve the target motion; moreover the changes of mechanical arm joint are more stable during the motion.

Keywords: Space manipulator / path optimization / particle swarm optimization algorithm / co-evolution

1 Introduction

With the development of the aerospace industry, a variety of aerospace equipment has been launched into space, especially the space station which has realized the extraterrestrial migration [1]. However, the space environment is different from the environment on the Earth, astronauts need to face the harsh working conditions including low temperature, low pressure, weightlessness and dangerous radiation when performing space missions. Once they are negligent, they will be in danger of life, and even some tasks are impossible to complete by manpower in the space environment [2]. Therefore, space robots, for replacing manpower, have been developed, which can perform some tasks in harsh space environment, including sample collection, article fixing, etc., and the above operations need to rely on manipulators of robots and the route plan of manipulators. The reason is that there is weightlessness in space, and the disturbance of mechanical arms to the pedestal during the movement is amplified to be nonnegligible without the constraint of gravity, thereby affecting the whole stability of robots and the completion degree of tasks. Therefore, the motion path of manipulators is needed to be optimized to reduce the disturbance to the pedestal [3]. Chen et al. [4] proposed a method of fault handling strategy and fault-tolerant path planning for the problem of free-swing joint failure of space manipulators, which ensured that after joint failure, the following normal operation of mechanical arms was guaranteed by locking the optimal joint. The simulation results showed that the method was suitable for the robot arms with any free swing joint failure. Zhou et al. [5] proposed a new ant colony algorithm to optimize the path of manipulators. The simulation results showed that the method could effectively plan the optimal path to avoid obstacles. Zhou et al. [6] proposed a singular robust programming method, and the simulation experiment verified the effectiveness of the proposed method in the online adjustment of space robot posture.

In this paper, the planning method of the motion trajectory of manipulators, the motion model of manipulators and the particle swarm optimization (PSO) algorithm used for optimizing the trajectory are briefly introduced, the multi-group co-evolution method is used to improve the PSO algorithm, and the above two algorithms are used to optimize the motion trajectory of the floating pedestal space manipulator with three free degrees of rotation in the same plane by the matrix laboratory (MATLAB) software.

* Corresponding author: qiaodong_dq@163.com

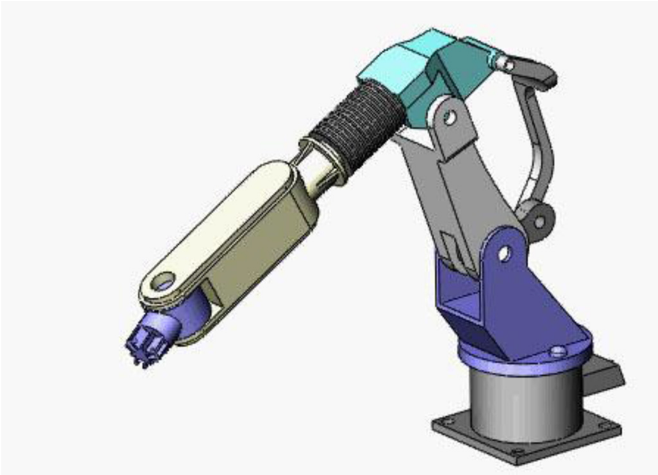

Fig. 1. Space manipulator.

2 Motion planning of manipulators

The basic appearance of the space manipulator is shown in Figure 1. The entire working process of space manipulators can be divided into task division, trajectory optimization and control motion [7]. The task division is an important component of the space manipulator working system. For manipulators, the direct execution of the input work command is complicated, as it needs to consider not only the planning of the path, but also the complicated working environment in the calculation process, which greatly increases the calculation amount and the error rate. The role of task division is to split complicated operational tasks into complex simple operation tasks and then to optimize them separately to get the optimal operation mode. Then manipulators perform task actions, and the constraint condition and the target condition are also needed to be considered in the process. Therefore, there are three task sets in the manipulator working system, i.e., the action set, constraint set and target set. The action set includes operations such as movement, grabbing, release, etc. of mechanical arms; the constraint set contains rules to be followed during the movement of mechanical arms, such as joint angular velocity, range of joint angle variation, obstacles during movement, etc.; the target set is the path goal that needs to be optimized under the guarantee of following the constraint rule in the motion planning of manipulators, including the change of the angular velocity of the joint, etc. Space manipulator can be divided into manual and automatic during work. As the accuracy of manual is not high, automatic planning of trajectory is usually used. In this process, the path planning algorithm is very important for the stability of the manipulator. The PSO algorithm used in this study does not need to set complex parameters, therefore it is not difficult to achieve and the effect is low.

3 Motion equation of space manipulators

As described above, in the working process, space manipulators first divide the tasks of manipulators

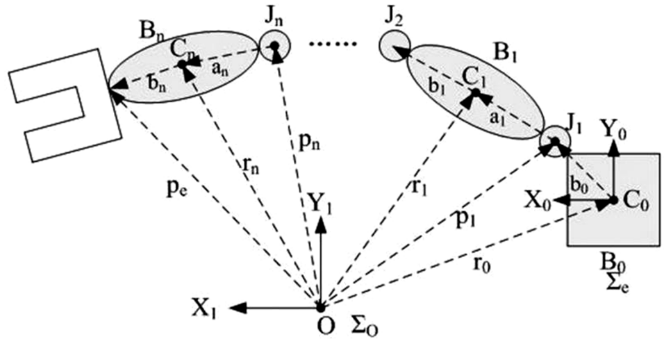

Fig. 2. Schematic diagram of the working model of space manipulators.

according to the input task target, then plan the movement trajectory of manipulators according to the constraint set and action set and use the relevant optimization algorithm to optimize the motion trajectory of manipulators according to the requirement of the target set. In the process of trajectory optimization, the manipulator model is simplified to some versatility in order to facilitate calculation [8]. As shown in Figure 2, the complex manipulator structure is simplified into a structure with the composition of a rigid body and a joint: B_0 represents the rigid body of the manipulator pedestal, $B_1 \sim B_n$ represent the links of the manipulator, $J_1 \sim J_n$ represent the joints of the manipulator, C_0 represents the centroid of the pedestal, $C_1 \sim C_n$ represent the centroids of the corresponding link, Σ_O represents an inertial coordinate system that assumes absolutely stationary, Σ_e represents the coordinate system of the manipulator pedestal, r_0 represents the position vector of the pedestal in the inertial coordinate system, $r_1 \sim r_n$ represent the position vectors of the centroid of the corresponding link in the inertial coordinate system, $p_1 \sim p_n$ represent the position vectors of the joint centroid in the inertial coordinate system, p_e represents the position vectors of the end of the manipulator in the inertial coordinate system, a_i represents the vector of J_i pointing to C_i, $i \in (1,\ 2,\ \ldots,\ n)$, and b_i represents the vector of C_i pointing to $J_{i+1},\ i \in (0,\ 1,\ \ldots,\ n)$. According to Figure 1 and the rule of vector calculation, the position vector of the end of the manipulator in the inertial coordinate system can be obtained:

$$p_e = r_0 + b_0 + \sum_{k=1}^{n} (a_k + b_k) \tag{1}$$

The linear velocity and angular velocity at the end of the manipulator can be obtained by differentiating equation (1), and the matrix form [9] is:

$$\begin{cases} \begin{bmatrix} v_e \\ \omega_e \end{bmatrix} = J_0 \begin{bmatrix} v_0 \\ \omega_0 \end{bmatrix} + J_m \theta \\ J_0 = \begin{bmatrix} E & r_0 - p_e \\ 0 & E \end{bmatrix} \\ J_m = \begin{bmatrix} k_1 \cdot (p_e - p_1) & \cdots & k_n \cdot (p_e - p_n) \\ k_1 & \cdots & k_n \end{bmatrix}, \end{cases} \tag{2}$$

where v_e, v_0 represent the linear velocity of the end of the manipulator and the pedestal respectively, ω_e, ω_0 represent the angular velocity of the end of the manipulator and the pedestal respectively, J_0, J_m represent the Jacobian matrix of the pedestal and the whole mechanical arm, θ represents the joint angle matrix, E represents the unit matrix, and k_i represents an expression of the Z-axis of the joint coordinate system relative to the inertial coordinate system. In this study, it is assumed that the space manipulator is in a free floating state while working, that is, the whole body maintains momentum conservation, and the dynamic equation of the manipulator can be obtained by combining equation (2):

$$\begin{bmatrix} v_e \\ \omega_e \end{bmatrix} = J_g(\varphi_0, \theta, m_i, I_i)[\theta] = \begin{bmatrix} J_{gv} \\ J_{g\omega} \end{bmatrix}[\theta], \qquad (3)$$

where $J_g(\bullet)$ represents the generalized Jacobian matrix of the space manipulator, which is also the function of φ_0, θ, m_i, I_i, the pedestal posture, joint angle, rigid body mass, and rigid body inertia, and J_{gv}, $J_{g\omega}$ represent the line speed and angular velocity terms of $J_g(\bullet)$, respectively.

In order to transform the problem of manipulator path planning into a mathematical problem that can be solved, in this study, the position vector at the end of the manipulator and the equivalent rotation angle are used to form a four-dimensional system variable, $X_i(t)$, and it is parameterized [10]:

$$X_i(t) = X_{i0} + \cos\left(\frac{2\pi t}{T_i} + \varphi_i\right)\cdot(a_{i5}t^5 + a_{i4}t^4 + a_{i3}t^3 + a_{i2}t^2 + a_{i1}t + a_{i0}), \qquad (4)$$

where $i = 1, 2, 3, 4$ represents the dimension, t represents a moment of the motion, T_i represents the motion cycle, φ_i represents the initial angle, $a_{i5} \sim a_{i0}$ represent the coefficients of the corresponding parameter item, and X_{i0} represents the initial system state. The boundary conditions of the motion are:

$$\begin{cases} X_i(0) = X_{i0} \\ X_i'(0) = X_i''(0) = 0 \\ X_i(t_f) = X_{ie} \\ X_i'(t_f) = v_i \\ X_i''(t_f) = \alpha_i, \end{cases} \qquad (5)$$

where t_f represents the final moment of the motion, X_{ie} represents the final position of the motion, and v_i, α_i represents the final speed and acceleration of the motion. In the motion path planning of the manipulator, environmental information is collected by the sensor, that is, the end target position of the manipulator, p_e, end equivalent angle, $\phi(t)$, and v_i, α_i are all known. Based on equations (4) and (5), it can be seen that the path of the manipulator can be planned by simply adjusting the coefficients of the parameter item, the motion cycle and the initial angle in equation (4). In this study, the optimization problem of the manipulator path can be transformed into the optimal combination solution of calculation of the coefficients of the manipulator parameter, the motion cycle and the initial angle.

4 Motion planning optimization based on the improved PSO algorithm

The PSO algorithm which is derived from the study on bird flocking and migration behavior is also known as the "bird flocking algorithm". The formula of calculation iteration [11] is:

$$\begin{cases} V_{i+1} = \varpi V_i + c_1 \cdot x_1 \cdot (pbest_i - P_i) + c_2 \cdot x_2 \cdot (gbest_i - P_i) \\ P_{i+1} = P_i + V_i, \end{cases} \qquad (6)$$

where P_i, V_i represent the position and velocity of particle i, respectively, $pbest_i$, $gbest_i$ represent the individual optimal position and global optimal position, respectively, c_1, c_2 are learning factors, x_1, x_2 are random numbers, which are evenly distributed between 0 and 1, and ϖrepresents the inertia weight. The iteration stops when the optimal solution is found or the set maximum number of iteration is reached. The optimization steps in this study are as follows.

① The parameters that need to be optimized are input into the particles as dimension vectors, and the particle group is initialized, including the number of particle groups, the velocity and position of the particles.

② The value of corresponding objective function is calculated according to the parameters in each particle to select the optimal particle as the temporary global optimal solution. The calculation formula of the objective function [12] is:

$$J_\omega = \lambda_1 B_{\omega f} + \lambda_2 B_{\omega \max}, \qquad (7)$$

where J_ω represents the disturbance angle of the pedestal posture, $B_{\omega f}$ represents the final offset of the pedestal posture, $B_{\omega \max}$ represents the maximum offset of the pedestal posture, and λ_1, λ_2 represent the weight coefficients, which is used to adjust the proportion of the final offset and the maximum offset in the disturbance.

③ The position and velocity of the particle is updated according to equation (6), and the value of objective function is calculated to compare with the previous value of objective function. If it is better, the corresponding particle (including individual and global) will be replaced, otherwise the original value will be retained. This step is cycled until the preset condition is reached, and then iteration is stopped.

The above is the traditional PSO algorithm, which is easy to operate, but it is prone to falling into the local optimal solution in the calculation process. In order to solve this problem, in this study, the parasitic behavior in nature is applied into the PSO algorithm to improve it. The improved algorithm uses three populations for co-evolution in order to avoid the local optimal solution. The main principle is to replace the worst particles of the next population with the optimal particles of the previous

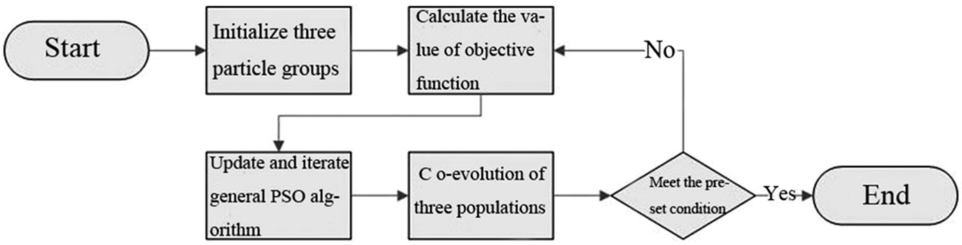

Fig. 3. Flow of the improved PSO algorithm.

population to form a cycle in the three populations and finally to select the optimal solution from the population consisting of the three populations.

The operation flow of the improved PSO algorithm is shown in Figure 3 [13].

① Firstly, similar to the traditional PSO algorithm, the parameters that need to be optimized are input into the particle as dimension vectors, and three particle groups are initialized.

② The value of objective function of each particle is calculated to select the optimal particle as the temporary global optimal solution.

③ The particles in each population are updated according to equation (6), and the value of the objective function before and after the update is compared to determine the retained particles.

④ Three populations co-evolution: the updated populations are arranged in ascending order of the values of objective function, half of the worst particle swarms are replaced with half of the best particle swarms in the previous population, and then the best particle is selected to replace the worst particle in each population.

⑤ The above steps of ②, ③ and ④ are cycled until the preset condition is met, and iteration stops to output the optimal solution.

5 Simulation analysis

5.1 Experimental environment

In this study, the MATLAB software [14] is used to simulate the motion path optimization of manipulators. The experiment is carried out on a laboratory server with configuration of Windows7 system, I7 processor and 16 G memory.

5.2 Experimental parameters

In this study, a floating pedestal space manipulator with three free degrees of rotation lying in the same plane is simulated. The schematic diagram of the model is obtained after the number of links in Figure 1 is changed to three. The parameters of the model are shown in Table 1, the initial posture angle of the pedestal is set as 0, the initial position of the end of the manipulator in the inertial coordinate system is (1.77, −0.95, 2.50), the initial joint angle is (30°, 70°, 50°), the target position of the end of the manipulator is (9.41, 1.66, −1.75), the target speed and acceleration are 0, the motion time of the manipulator is set as 40 s, and the rotation range of the three joint angles is [−270°, 270°].

Table 1. Parameters of space manipulator model.

	Quality/kg	Inertia/kg m^2	a_i/m	b_i/m
Pedestal	400	6.68	–	0.6
Link 1	40	4.51	0.6	0.6
Link 2	30	3.25	0.6	0.6
Link 3	20	2.15	0.6	0.6

The parameters of the traditional PSO algorithm are as follows: the number of initial population is 50, the inertia weight is 0.5, the learning factors are 2, and the maximum number of iteration is 1000.

The parameters of the improved PSO algorithm are as follows: the number of initial population is 50, the inertia weight is 0.5, the learning factors are 2, and the maximum number of iteration is 1000.

Genetic algorithm also has excellent performance in path planning. Therefore, genetic algorithm is selected to compare with the PSO algorithm in this study. Genetic algorithm is a simulation of population evolution in the nature, which takes the rotation angle of the joint angle as the factor in the inherited chromosome and obtains the optimal solution after crossover and mutation operations. The parameters of the genetic algorithm are as follows: the number of initial population is 50, the crossover probability is 0.3, the mutation probability is 0.2, and the maximum time of iterations is 1000.

The three algorithms operate independently for 30 times, and the average values were taken.

5.3 Experimental results

As shown in Figure 4, as the number of iterations increases, the global optimal fitness values calculated by the three algorithms decrease rapidly and converge to a stable state. The initial global optimal fitness value of the traditional PSO algorithm is close to the initial global optimal solution of the improved PSO algorithm. The reason is that the generation of populations of the both is random, after taking the average of 30 times of experiments, the randomness is greatly reduced, and moreover the initial conditions of manipulators are the same in the two algorithms, finally making the initial global optimal fitness values similar. The genetic algorithm converges to stability after iterating 700 times. The traditional PSO algorithm

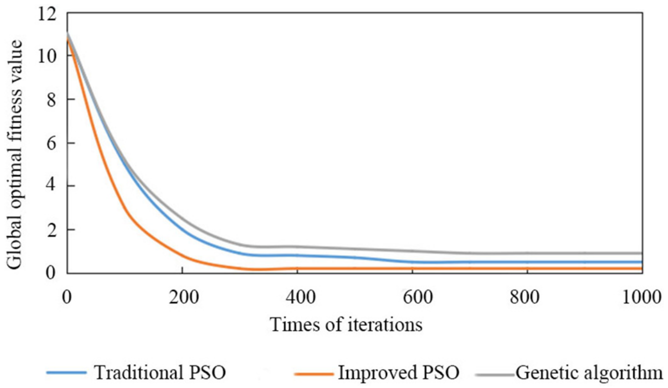

Fig. 4. Convergence curves of three algorithms in the iterative process.

Table 2. Optimization effects of three algorithms on the motion trajectory of the manipulator.

	Final pedestal offset/°	Maximum pedestal offset/°	Motion time/s
Traditional PSO algorithm	0.55	2.2	30.15
Improved PSO algorithm	0.21	1.6	20.51
Genetic algorithm	0.62	2.5	31.22

converges to stability after iterating 600 times, and the improved PSO algorithm converges to stability after iterating 300 times. Moreover, it can be seen intuitively from Figure 3 that the global optimal fitness value obtained by the convergence of the improved PSO algorithm is smaller than that by the traditional PSO algorithm, while the global optimal fitness value obtained by the convergence of the traditional PSO algorithm is smaller than that by the convergence of genetic algorithm, that is, the improved PSO algorithm can obtain better motion planning trajectory.

After optimization of the three algorithms, the optimization effect of the motion trajectory of the manipulator is shown in Table 2. After the motion trajectory of the manipulator optimized by the traditional PSO algorithm reaches the target position, the pedestal posture is shifted by 0.55°, the maximum pedestal offset is 2.2° in the motion process, and finally the time consumed for reaching the target position is 30.15 s. After the motion trajectory of the manipulator optimized by the improved PSO algorithm reaches the target position, the pedestal posture is shifted by 0.21°, the maximum pedestal posture offset is 1.6° during the motion process, and finally the time consumed for reaching the target position is 20.51 s. After the motion trajectory of the manipulator optimized by the genetic algorithm reaches the target position, the pedestal posture is shifted by 0.62°, the maximum pedestal posture offset is 2.5°, and the time consumed for reaching the target position is 31.22 s. It can be seen that the motion trajectory of the manipulator optimized by the improved PSO algorithm not only has less disturbance to the pedestal, but also requires less time to complete the specified target.

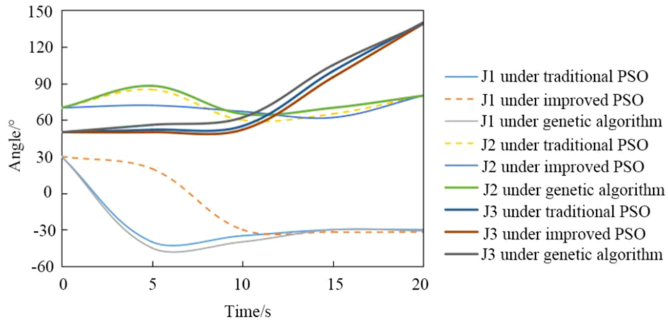

Fig. 5. Variation of the joint angle of the manipulator under the optimization of three algorithms.

The changes of the three joint angles during motion of the manipulator motion trajectory optimized by the three algorithms are shown in Figure 4 (J1, J2 and J3 are joints 1, 2 and 3, respectively). Since the required time obtained by the three algorithms for achieving the motion trajectory is different, only the changes of the joint angle within 0–20 s are intercepted to compare the changes of the joint angle of the both. It can be seen from Figure 5 that joints 1 and 2 change more severely under the planning of the traditional PSO algorithm and genetic algorithm, while the changes of joints 1 and 2 under the planning of improved PSO algorithm are relatively more stable. Joint 3 changes similarly under the planning of the three algorithms, which is because that joint 3 is close to the end of the manipulator, basically acts only at the end target position of the fine adjustment and does not require a large change. In general, the changes of the joint in the motion trajectory of the manipulator under the planning of the improved PSO algorithm are more stable, and the disturbance to the pedestal is smaller.

6 Discussion

From the experimental results above, it is seen that the traditional PSO, improved PSO and genetic algorithms can plan the trajectory of space manipulator, and the planning effect of the improved PSO algorithm is obviously better than that of traditional PSO and genetic algorithms, while the traditional PSO algorithm is slightly better than genetic algorithm. The advantages of the improved PSO algorithm include: (1) in the aspect of convergence, the improved PSO algorithm is the fastest, followed by the traditional PSO algorithm and genetic algorithm; as to the fitness value during convergence stability, the improved PSO algorithm is the smallest, followed by the traditional PSO algorithm and genetic algorithm; (2) in the final optimization result of trajectory, the disturbance of the planning result obtained by the improved PSO algorithm is the smallest, and the time consumed in completing trajectory is the shortest; (3) the variation of the joint angle of the manipulator is more stable under the trajectory planned by the improved PSO algorithm. The reason is that the optimization effect of genetic algorithm greatly depends on the probabilities of crossover and mutations in the iterative process through setting of special parameter are not needed, and moreover the initial population is generated randomly, leading to unsatisfactory optimization effect. The reason of the traditional PSO algorithm is similar to genetic algorithm; though the design is simple and the optimization effect depends on the selection of initial population and learning factor, the improper learning factor will make the algorithm not converge or fall into the local optimal solution. The improved PSO algorithm adopts population co-evolution, which not only expands the range of the initial population, but also eliminates some poor particles in the process of co-evolution, so that the convergence accelerates and the local optimal solution is avoided.

Space manipulators are more widely used in the space industry for experimental operations such as external repair or sample collection of space stations or shuttles. Due to the weightlessness of the space environment, disturbances in the motion of space manipulators will significantly affect the stability of the connection base, thereby increasing the risk of accidents. Therefore, the optimization of the trajectory of space manipulator can not only improve work efficiency, but also reduce the disturbance to the pedestal and improve safety.

7 Conclusion

In this paper, the planning method of the motion trajectory of manipulators, the motion model of manipulators and the PSO algorithm used for optimizing the trajectory are briefly introduced, the multi-population co-evolution method is used to improve the PSO algorithm to make up for the shortcomings of the PSO algorithm, and moreover simulation experiment was carried out on MATLAB software. It is also compared with genetic algorithm.

The results are as follows.

- The improved PSO algorithm converges to stability faster, and the global optimal fitness after stabilization is the smallest.
- The motion trajectory optimized by the improved PSO algorithm has less disturbance to the pedestal posture of the manipulator and requires less time to complete the target action.
- After the improved PSO algorithm optimizes the motion trajectory, the joint changes of the manipulator are more stable, reducing the disturbance to the pedestal.

The research in this paper was supported by Datong Industrial Science and Technology Project: Development and Research on the Intelligent Three-dimensional Rescue Special Operation Equipment (NO. 2016024).

References

1. X. Huang, Y. Jia, S. Xu, S. Lu, J. Beijing Univ. Aeron. Astron. **43**, 488–496 (2017)
2. S.K. Wang, L. Zhu, J.Z. Wang, Trans. Beijing Inst. Tech. **35**, 186–191 (2015)
3. G. Chen, P.C. Ye, Q.X. Jia, H.X. Sun, Control Decis. **30**, 156–160 (2015)
4. G. Chen, W. Guo, Q.X. Jia, X. Wang, Y.Z. Fu, Chin. J. Aeron. **31**, 109–124 (2018)
5. D. Zhou, L. Wang, Q. Zhang, SpringerPlus **5**, 509 (2016)
6. C. Zhou, M.H. Jin, Y.C. Liu, Z. Zhang, Y. Liu, H.S. Liu, Int. J. Autom. Comput. **14**, 169–178 (2017)
7. X. Ge, K. Chen, Chin. J. Theor. Appl. Mech. **48**, 4 (2016)
8. M. Wang, J. Luo, J. Fang, J.P. Yuan, Adv. Space Res. **61**, S0273117718300346 (2018)
9. G. Chen, C. Wei, Q.X. Jia, H.X. Sun, B.Y. Yu, Appl. Mech. Mater. **713–715**, 800–804 (2015)
10. G. Misra, X. Bai, J. Guid. Control Dyn. **40**, 1–8 (2017)
11. H. Mo, L. Xu, Neurocomputing **148**, 91–99 (2015)
12. X. Wang, Y. Shi, D. Ding, X.S. Gu, Eng. Optim. **48**, 299–316 (2016)
13. J.J. Kim, J.J. Lee, IEEE Trans. Industr. Inform. **11**, 620–631 (2017)
14. Z. Ehsan, F. Saeid, Modares Mech. Eng. **14**, 199–210 (2015)

Cite this article as: Dong Qiao, Motion planning optimization of trajectory path of space manipulators, Int. J. Metrol. Qual. Eng. **10**, 11 (2019)