International Journal of Control, Automation and Systems 16(X) (2018) 1-9
http://dx.doi.org/10.1007/s12555-016-0693-3
ISSN:1598-6446 eISSN:2005-4092
http://www.springer.com/12555

Motion Planning of Robot Manipulator Based on Improved NSGA-II

Ying Huang* and Minrui Fei

Abstract: In this paper, the trajectory of a robot manipulator is planned using the non-dominated sorting genetic algorithm II (NSGA-II). Moreover, consumed time, Cartesian trajectory length, and smooth movement are used as the multi-objective to be optimized [1, 2]. The Pareto optimal solution set is obtained through NSGA-II, and simulation is used to obtain and verify the results. In an actual engineering case, the optimal solution of the Pareto solution set can be selected as the optimal path of a robot manipulator. Results show that the relationship between consumed time and joint jerk is a priority solution to practical engineering selection. Moreover, the spatial distribution of the optimal solution set is improved by enhancing the proposed crowding distance mechanism in the conventional NSGA-II algorithm.

Keywords: Crowding distance, joint jerk, manipulator, non-dominated sorting, NSGA-II, Pareto.

1. INTRODUCTION

During the execution of a given task, the smooth movement, reasonable path, speed, and steady performance of a robot manipulator are factors to be concerned about. In their article [1], Bahha *et al*. considered the exceeding torque, joint traveling distance, Cartesian trajectory length, and consumed time as indexes and then integrated these with weighting factors into a fitness function evaluation to achieve optimization. The method uses a weighted objective function to transform the multi-objective optimization problem (MOP) into a single-objective optimization problem. However, the objectives to be optimized usually have different dimensions in the process. Moreover, the weights of these objectives appear subjective and cannot be reasonably allocated.

Similar to the optimization of the robot manipulator [3], most optimization problems in practical engineering are MOPs [4–6]. Coello *et al*. [7] discussed a multi-objective optimization method, including algorithms, methods and diversity, methodological issues, and applications. Zitzler *et al*. [8] analyzed the MOP and proposed a mathematical framework for studying quality assessment methods for MOPs. The approaches were compared directly by evaluating the outcomes using Pareto dominance [9]. Therefore, the absolute optimal solution is difficult to find. However, the Pareto optimal solution set comprises a series of Pareto solutions. Thus, the most satisfactory solution can be selected from the Pareto optimal solution set according to different targets and demands. Madavan [10] compared the Pareto optimal solution set, which is the approach described in the current work and combines the robust and effective differential evolution strategy, with the key elements of non-dominated sorting genetic algorithm-II (NSGA-II). The study of Knowles and Corne [11, 12] suggested that their proposed algorithm produces competitive Pareto fronts according to an applied convergence metric and significantly outperforms three other algorithms. Thus, an approach based on the Pareto multi-objective cooperative co-evolutionary algorithm was proposed in their paper [13].

An evolutionary algorithm (EA) for obtaining a Pareto set approximation is appropriate for the MOP of the robot manipulator because of the complex and nonlinear model. The multi-objective EA (MOEA) has currently become the main method in the aforementioned MOP [14–16]. Over time, many algorithms have been developed for MOEA, of which the representative ones are the strength Pareto evolutionary algorithm, NSGA-II, Pareto archived evolution strategy, and MOEA based on decomposition, which were proposed by Zitzler and Thiele [17], Deb [18], Knowles [19], and Zhang *et al*. [20], respectively. In addition, other evolutionary algorithms, such as the interval multi-objective particle swarm optimization algorithm, was proposed by Na *et al*. [21].

NSGA-II [18] is a main classical algorithm in second-

Manuscript received November 8, 2016; revised July 2, 2017 and September 14, 2017; accepted January 17, 2018. Recommended by Associate Editor Huanqing Wang under the direction of Editor Hyun-Seok Yang. This work was supported by Research Program funded by DianJi University, ShangHai, China.

Ying Huang is with the School of Mechatronic Engineering and Automation, ShangHai University, Yanchang Road No. 149, Zhabei District, ShangHai 200072, China , and he is also with the Electrical School, DianJi University, Ganlan Road No.1350, Pudong New District, ShangHai 201306, China (e-mail: 653498640@qq.com). Minrui Fei is with the School of Mechatronic Engineering and Automation, ShangHai University, Yanchang Road No. 149, Zhabei District, ShangHai 200072, China (e-mail: mrfei@staff.shu.edu.cn).
* Corresponding author.

generation evolutionary multi-objective optimization. This algorithm uses the Pareto dominance relation to classify layers and then calculates the density of the individual in each layer using crowding distance [22, 23], k-nearest neighbor [24], ε-domination [25], and grading. Therefore, NSGA-II is an improved algorithm. This algorithm is superior to other new multi-objective optimization methods [26]. Furthermore, NSGA-II is greatly enhanced by some projects. By improving non-dominated sorting [27], crossover and mutation operator [28], and generated solutions [29], an improved optimization performance is obtained. Consequently, NSGA-II and its enhanced version have been successfully applied in many engineering practices. In the engineering control practice of the robot manipulator, improved trajectory tracking control ensures that the manipulator operates along a well-planned trajectory [30–35]. Thus, the manipulator performs the task and implements multi-objective optimization simultaneously.

This study attempts to improve the NSGA-II and apply it to the motion planning of a robot manipulator. Under the constraints of angular velocity and joint angle, the consumed time, trajectory length, and joint jerk [2, 36] are considered the multi-objective and the Pareto front is obtained through the evolution of the dominant layer and crowding distance. The simulation shows that the optimization is effective. Moreover, the relationship between consumed time and joint jerk is a priority solution to practical engineering selection. By improving the crowding distance with the obtuse angle short distance (OASD) mechanism proposed in this paper in the conventional NSGA-II algorithm [22–25], the distribution of the optimal solution set becomes appropriate.

This paper is organized as follows: The second part introduces robot manipulator motion planning, analyzes multi-objective optimization to a certain extent, and describes several objectives to be optimized in robot motion planning. The third part explains the Pareto dominant mechanism, introduces the dominance layer and sorting of NSGA-II, and explains the improvement mechanism of crowding distance. The fourth part focuses on the application of NSGA-II to the multi-objective optimization of the robot manipulator and the process of the improved algorithm. Finally, the fifth part illustrates the 3-DOF robot manipulator and the parameters of NSGA-II and runs the simulation and the analysis of the robot and the multi-objective optimization.

2. SYSTEM DESCRIPTION

The robot manipulator system aims to optimize objectives, namely, consumed time, Cartesian trajectory length, and smooth movement, while considering the constraint conditions of position, velocity, acceleration, and torque. The trajectory at the end of the robot manipulator is composed of several high-order spline curve segments, and the robot must avoid obstacles in space during its operation. Therefore, the robot should run in a limited space, that is, from the initial point to the final point, and avoid obstacles while achieving optimal performance objectives.

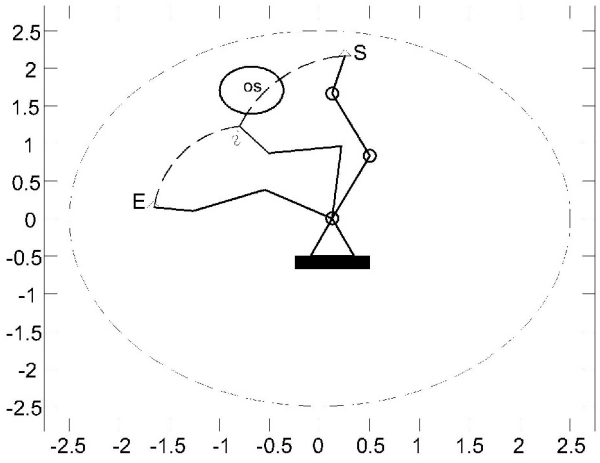

Fig. 1. 3-DOF robot manipulator trajectory operation and obstacle avoidance.

2.1. Multi-objective optimization problems

An MOP has n decision variables and m objective variables, which can be expressed as

$$\begin{cases} \text{minimize } y = F(x) \\ \qquad\qquad = (f_1(x), f_2(x), \cdots, f_m(x))^T, \\ \text{subject to } g_i(x) \le 0, \qquad i = 1,2,\cdots,q, \\ h_j(x) = 0, \qquad j = 1,2,\cdots,p, \end{cases} \tag{1}$$

where $x = (x_1, x_2, \cdots, x_n)^T \in X \subset R^n$ is an n-dimensional decision vector, X is an n-dimensional decision space, $y = (y_1, y_2, \cdots, y_m)^T \in Y \subset R^m$ is an m-dimensional objective vector, and Y is an m-dimensional objective space. The m mapping functions from the decision space to the objective space are defined by the objective function $F(x)$, whereas $g(x)$ and $h(x)$ define q inequality constraints and p equality constraints, respectively.

2.2. Problem description

Fig. 1 shows the 3-DOF robot manipulator operator in the task space from the initial point S to the final point E. The space has a circular obstacle OS.

This study selects two high-order polynomials considering the spatial situation in the manipulator trajectory. Multi-objective optimization aims to determine the interpolation points of the junction and thus constitute the motion trajectory, which is composed of two sections of the polynomial curves.

On the basis of the factors of the order of acceleration and jerk, this paper selects the first four-order polynomial and the second five-order to form the trajectory [1].

From the initial points S to an intermediate point via point i, a quadrinomial is used to describe the point in this segment as follows:

$$\theta_{k,1}(t) = a_{k0} + a_{k1}t_i + a_{k2}t_i^2 + a_{k3}t_i^3 + a_{k4}t_i^4 \quad (i = 0, \cdots, mp-1; \quad k = 1,2,3), \tag{2}$$

where $\theta_{k,1}$ is the set of each time joint angle q_1, q_2, q_3 values in the segment, a is a polynomial coefficient, mp is the intermediate point via point i, and t_i is the ith time segment.

Between the intermediate point i and the final point E, the point in the segment can be described by a quintic polynomial as follows:

$$\theta_{k,2}(t) = b_{k0} + b_{k1}t_j + b_{k2}t_j^2 + b_{k3}t_j^3 + b_{k4}t_j^4 + b_{k5}t_j^5 \quad (j = mp, \cdots, t; \quad k = 1,2,3), \tag{3}$$

where $\theta_{k,2}$ is the set of each time joint angle q_1, q_2, q_3 values in the segment, b is a polynomial coefficient, t_j is the jth time segment, and t is the total consumed time.

In general, the trajectory of the robot manipulator must be able to move from the initial position to the final position while avoiding obstacles. Moreover, the working efficiency of the robot manipulator should be improved to reduce energy consumption and to protect the power component. Therefore, the trajectory is expected to have short consumed time, short length, smooth movement, low energy consumption, small torque change, and other indexes. The sections show the optimization of some of the aforementioned features.

2.3. Objective optimization

2.3.1 Consumed time

$$f_1 = t_1 + t_2, \tag{4}$$

where f_1 is the total motion time, which evaluates the efficiency of the movement of the manipulator, t_1 is the execution time from the initial point S to the intermediate point via point i, and t_2 is the execution time from the intermediate point i to the final point E.

2.3.2 Cartesian trajectory length

$$f_2 = d_1 + d_2 = \sum_{i=0}^{t1} sqrt(x_i'^2 + y_i'^2) + \sum_{j=0}^{t2} sqrt(x_j'^2 + y_j'^2), \tag{5}$$

where i, j are the execution time points of the first and second polynomials, respectively, x, y are the Cartesian space coordinates, and d_1, d_2 are the first and second trajectory lengths, respectively. f_2 represents the total Cartesian trajectory length used to evaluate the reasonableness of the robot trajectory.

2.3.3 Joint jerk

System jerk is used to evaluate the smoothness of the robot motion. Variable acceleration is the main factor of jerk generation [37–39] and can be obtained from the third derivative of the position (2) and (3). Thus, the jerk is calculated as follows:

$$Jq_{k,i+1} = 6a_{k,i3} + 24a_{k,i4}T_{k,i}, \tag{6}$$

$$Jq_{k,j+1} = 6b_{k,j3} + 24b_{k,j4}T_{k,j} + 60b_{k,j5}T_{k,j}^2, \tag{7}$$

$$f_3 = sqrt\left[\sum_{k=1,i=0,j=0}^{k=3,i=t1,j=t2} (Jq_{k,i+1}, Jq_{k,j+1})^2\right], \tag{8}$$

where k is the joint serial, i, j are the execution time points of the first and second polynomials, respectively, a, b are the coefficients of the first and second polynomials, respectively, T is the execution time point in the time segments t_1 and t_2, J is the joint jerk, and f_3 is the total joint jerk of the robot, which is used to evaluate the smoothness of the system motion.

3. PARETO OPTIMAL SOLUTION SET BASED ON IMPROVED NSGA-II

NSGA-II uses a rapid non-dominated sorting mechanism, drives the convergence of the search process to the Pareto optimal front, and ensures the diversity of the Pareto optimal solutions.

3.1. Pareto dominant mechanism

The Pareto dominant assumes that $x_A, x_B \in \Omega$ are two feasible soultions for the multi-objective optimization shown in Eq. (9) and (10). x_B and x_A are compared to determine the Pareto dominant. The solutions are Pareto dominant if and only if

$$\forall i \in \{1,2,\cdots,m\}, f_i(x_A) \le f_i(x_B), \tag{9}$$

$$\exists j \in \{1,2,\cdots,m\}, f_j(x_A) < f_j(x_B), \tag{10}$$

which can be written as $x_A \prec x_B$ (x_A dominates x_B).

This study selects $f_1(p) - f_1 \le 0, f_2(p) - f_2 \le 0, and f_3(p) - f_3 \le 0$ as the Pareto dominant conditions. The dominant figure is simplified and shown in Fig. 2.

Fig. 2 shows that the solid points are composed of consumed time f_1, trajectory length f_2, and joint jerk f_3. Equations (9) and (10) show that layer 1 is more dominant than layer 2, layer 2 is more dominant than layer 3, and layer 1 is used as the Pareto front.

3.2. Interval crowding distance

Crowding distance strategy is used to describe the distribution density of a certain optimal solution set.

Fig. 3 shows a 3D multi-objective crowding distance. The solid points represent individuals of the same non-dominated solution layer on the Pareto front.

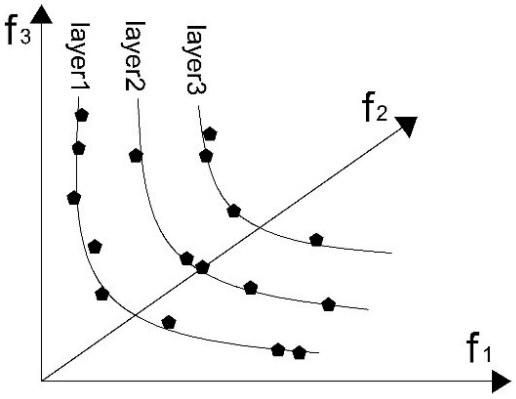

Fig. 2. Pareto dominant layered.

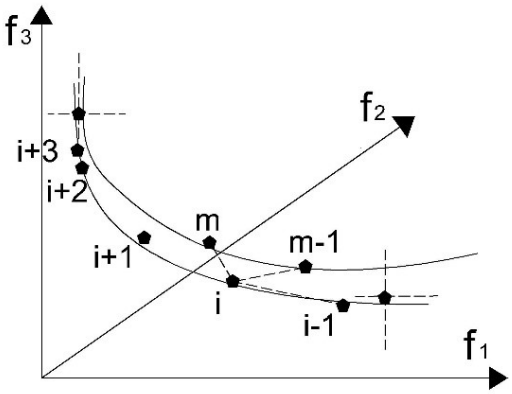

Fig. 4. Complex Pareto crowding distance.

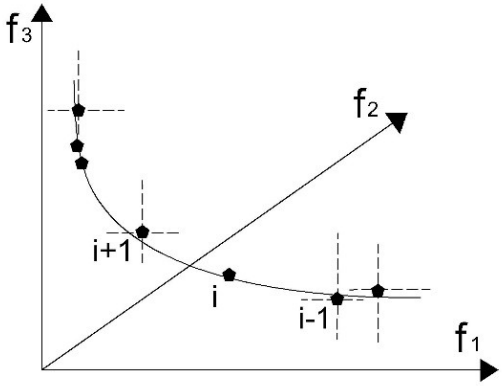

Fig. 3. Pareto crowding distance.

The individual crowding distance of the population is based on the individual layer of a non-dominated solution layer. The formula for the crowding distance of individual i is defined as follows:

$$d_i = \sqrt{\sum_{k=1}^{m} \left(\left| f_k^{i+1} - f_k^{i-1} \right| / (f_k^{max} - f_k^{min}) \right)^2}. \tag{11}$$

For the Pareto front of the robot manipulator system, the crowding distance is

$$\begin{cases} d_{i1} = \left| f_1^{i+1} - f_1^{i-1} \right| / (f_1^{max} - f_1^{min}), \\ d_{i2} = \left| f_2^{i+1} - f_2^{i-1} \right| / (f_2^{max} - f_2^{min}), \\ d_{i3} = \left| f_3^{i+1} - f_3^{i-1} \right| / (f_3^{max} - f_3^{min}), \\ d_i = \sqrt{d_{i1}^2 + d_{i2}^2 + d_{i3}^2}, \end{cases} \tag{12}$$

where d_i is the crowding distance of the Pareto front individual i, f_k^i is the Pareto front individual i function value of objective k, and f_k^{max} and f_k^{min} are the maximum and minimum function values of the Pareto front of objective k, respectively.

An analysis of the aforementioned 3D multi-objective linear curve shows that the crowding distance between the two individuals in the front and at the back is feasible. However, the multi-objective in this paper is not a linear curve; hence, the Pareto front is a nonlinear surface. The two individuals can be well-separated rather than the two supposed individuals in the front and at the back after sorting with the traditional crowding distance mechanism. Therefore, the traditional crowding distance mechanism cannot obtain the space distribution of the Pareto optimal solution. Moreover, the interval of the two individuals cannot be a good response to the crowding situation of the individual.

Fig. 4 shows that according to the traditional processing mode, the crowding distance of individual i is the interval between individual $i+1$ and individual $i-1$. However, in the Pareto front, individual $m-1$ is closer to individual i and individual m is closest to individual i. Evidently, the crowding distance of individual m and $i-1$ is reasonable in the Pareto front surface.

In this study, the crowding distance is obtained using the OASD mechanism. Individual m should be closest to individual i to obtain the crowding distance of individual i. If individual $i-1$ does not exist, then sides mi and $i(m-1)$ form an acute angle. Sides mi and $i(i+1)$ form an acute angle as well. Subsequently, the distance between individual m and individual i is considered the crowding distance of individual i. If individual $i-1$ exists, then the distance between individual m and individual $i-1$ is considered the crowding distance of the individual i because the angle formed by the sides of mi and $m(i-1)$ is an obtuse angle.

Although the traditional crowding distance is the space interval between the front and rear individuals, it cannot reflect the distance from the individual to the front and the individual to the rear. The crowding situation cannot be reflected better. Fig. 4 shows that the crowding distance of individual $i+2$ is large and composed of the front individual $i+1$ and the rear individual $i+3$. However, individuals $i+2$ and $i+3$ are close. Therefore, this measurement aims

to judge the ratio of the shortest distance to the crowding distance of the individual point. If the value is smaller than the value of the ε ratio, which is the minimum possible value, then the crowding distance is replaced with the shortest distance. Thus, the probability for the individual point to be excluded is increased.

4. MULTI-OBJECTIVE OPTIMIZATION BASED ON IMPROVED NSGA-II

To address the multi-objective optimization, the conventional approach is adopted by the process of robot manipulation on the basis of NSGA-II. Fig. 1 shows that population generation and evolution are the evolutionary processes of the intermediate point via point i in the optimization algorithm. The elements of the point include three joint angles (q_1, q_2, q_3), a third-link horizontal angle q_{phif}, three-joint angular velocities $(v_{q_1}, v_{q_2}, v_{q_3})$, and the consumed times of two segments (t_1, t_2). The multi-objective optimization is the optimization process of consumed time, trajectory length, and joint jerk (f_1, f_2, f_3). In the algorithm [18], the population size, mating pool size, crossover probability, mutation probability, and iteration number must be set. The following are main evolutionary improved mechanisms.

4.1. Tournament rule

If the layer series in any two randomly generated populations are unequal, then the smaller of the two populations will normally be selected as the candidate in the competition; if the layer series are equal, then the population with larger crowding distance will be selected.

4.2. New population generation rule

Size N parent population P_t is generated using the tournament rule. The parent population P_t produces size N offspring population by conventional genetic algorithm crossover and mutation. N-size parent population and offspring population are merged to form a size $2N$ population $R_t = P_t \cup Q_t$. Then, a new-generation population is produced through Pareto non-dominated sorting and crowding distance calculation.

4.3. Crowding distance improvement

After layering and sorting the populations, the crowding distance mechanism is illustrated according to OASD, as shown in Fig. 5.

Algorithm process:

1) After layering and sorting the population, the solution individuals of each layer are obtained.
2) After sorting in each dimension, the boundary points are obtained. Subsequently, the characteristic distances of the boundary individuals are set to the maximum.

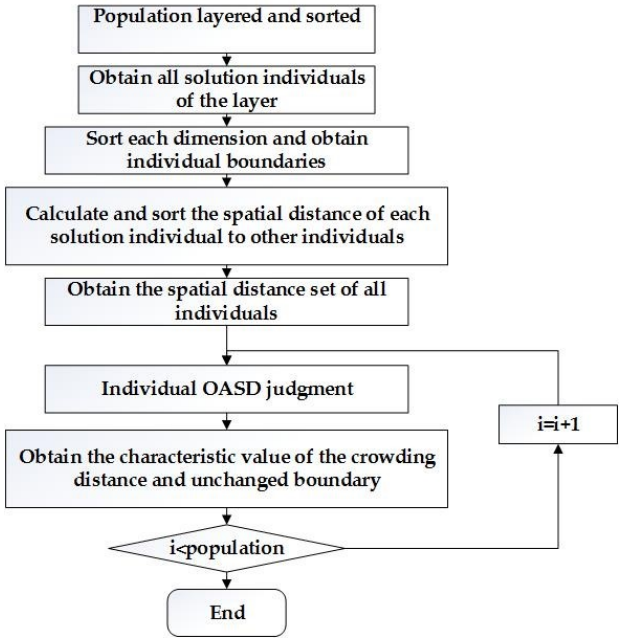

Fig. 5. Crowding distance improvement in a layer.

3) Except the boundary, the spatial distances of each solution individual to other individuals are calculated, and the distance set is obtained. The individuals in the distance set are sorted from small to large. The other solution individuals are treated similarly.
4) OASD algorithm:

   Obtain a solution individual and its distance set, in which $D1$ is the minimum distance between the first and the second individuals, whereas $D2$ is the distance between the first individual and any individual other than the second. If any one individual is selected as in $D2$, then $D3$ refers to the distance between that specific individual and the second individual.

   If $D3 < D1$, $D3 < D2$, then $D1$ is selected as the characteristic distance and judging continues until the end of the set.

   If $D3 > D1$, $D3 < D2$, then $D1$ is selected as the characteristic distance and judging continues until the end of the set.

   If $D3 > D1$, $D3 > D2$, then $D3$ is selected as the characteristic distance and the cycle judgement ends.

   Short distance judgement is used to set an artificially set reference ratio ε first and divide the minimum distance by the characteristic distance. If the obtained value is smaller than ε, then the characteristic distance will be replaced by the minimum distance. This replacement increases the probability of this individual to be eliminated because it is too close to the front or the rear individual; otherwise, the characteristic distance remains unchanged.

5) All other individuals are treated similarly.

4.4. Evaluation criteria

In the study of multi-objective optimization, performance indexes usually have to be used to evaluate whether the algorithm achieves the required goal. The spatial metric *SP* is one of the indexes used to measure the uniformity of the distribution of the Pareto solution.

In this study, the crowding distance mechanism of NSGA-II is improved. Thus, the spatial metric *SP* is used to analyze the distribution of the solution sets before and after improvement.

Average distance:

$$\overline{d} = \frac{1}{n}\sum_{i=1}^{n} d_i. \tag{13}$$

Spatial metric *SP*:

$$Aver_D = \left(\frac{1}{n-1}\sum_{i=1}^{n}\left|d_i - \overline{d}\right|\right), \tag{14}$$

$$SP = Aver_D/\overline{d}, \tag{15}$$

where n is the number of non-dominated solutions obtained by the algorithm and d_i is the Euclidean distance between two coherent points in the target space. In this study, the distance from the nearest individual to each solution individual is obtained. The spatial metric *SP* reflects the degree of the uniformity of the distribution of the optimization solution set in the target space. The smaller the value, the more uniform the distribution.

5. SIMULATION TEST

Fig. 1 shows that a 3-DOF manipulator is used in the plane operation with a circular obstacle. The parameters of the robot manipulator are set, and the parameters of the improved NSGA-II are as follows:

Population size: $pops$=100
Mating pool size: $pool$=50
Tournament scale: $tournament$=2
Crossover probability: $crossprop$=0.8
Mutation probability: $mutprop$=0.05
Iteration number: $maxgen$=80
Short distance ratio: ε=0.1

5.1. Operating result diagram

Fig. 6 shows a 3-DOF manipulator with an improved trajectory length solution in the Pareto front from the initial point S to the final point E with obstacle avoidance. The system with short consumed time or smooth movement is shown in Fig. 7.

The diagram of the effect of the robot arm shows that the Pareto front of the multi-objective optimization results makes up the optimal solution set, from which any solution can be selected.

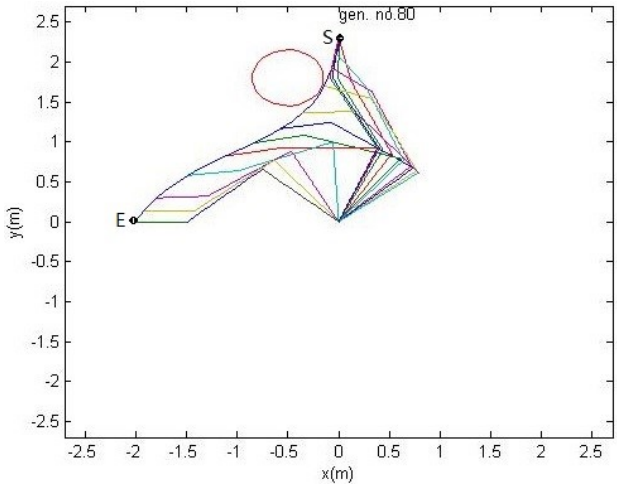

Fig. 6. Improved trajectory length in multi-objective optimization.

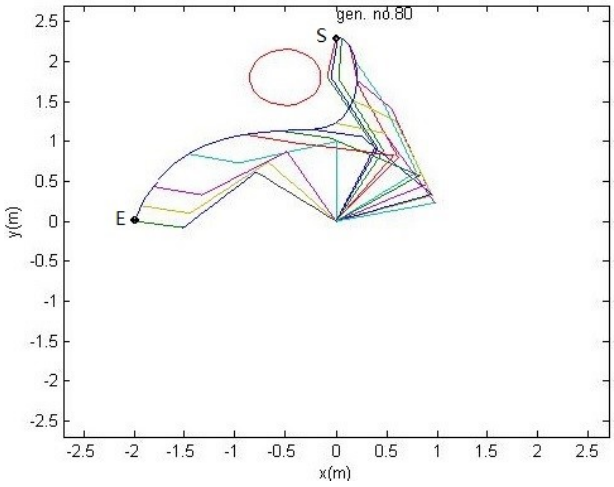

Fig. 7. Short consumed time or smooth movement situation.

5.2. NSGA-II process and crowding distance improvement

In the multi-objective optimization process, relative optimal solutions are obtained by the conventional and improved NSGA-II. The Pareto front solutions of NSGA-II ("o") and improved NSGA-II ("*") are shown in Fig. 8.

In the conventional NSGA-II, the crowding distance d_i can be obtained by (12) by dividing the length between the front individual $i+1$ and the rear individual $i-1$ of the same layer by the total span of the layer. During the improvement, the crowding distance obtained from the OASD mechanism is the distance from the individual i to either the nearest individual (shortest distance $D1$) or the third side ($D3$). The crowding distances of the Pareto front before and after improvement are shown in Table 1.

Without the improvement of the crowding distance in

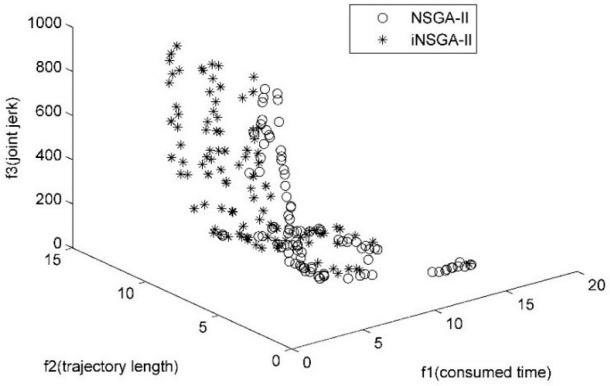

Fig. 8. Pareto front before and after improvement.

Table 1. Conventional crowding distance and improved crowding distance.

Layer	Conventional crowding distance	Improved crowding distance
1	0.0217	0.0564
1	0.0223	0.0583
1	.....	.....
1	0.0492	0.1024
1	0.0500	0.1032
1	.....	.....
1	0.0790	0.1858
1	0.0874	0.1993

Table 2. Average distances of conventional and improved NSGA-II.

Serial	Conventional $\overline{d}$	Improved $\overline{d}$
1.	3.1299e-02	5.4521e-02
2.	3.5976e-02	5.9374e-02
...	.....	.....
11.	2.2849e-02	5.5904e-02
12.	3.4405e-02	6.2507e-02
...	.....	.....
20.	3.9065e-02	5.5349e-02

the conventional NSGA-II, 100 optimal solutions are obtained by NSGA-II, in which the crowding distance is in the range of $0.02 < distance < 0.08$. With the improved crowding distance mechanism OASD in NSGA-II, the crowding distance is in the range of $0.05 < distance < 0.19$, thus showing that the Pareto front solution set of the improved NSGA-II algorithm has a larger and more reasonable distribution than the conventional algorithm. Thus, the optimal individuals are representative.

According to the aforementioned evaluation criteria, the multi-objective optimization operations are conducted 20 times before and after the improvement. The average distance of the Pareto optimization solutions, which is the mean value of all non-dominated solutions to the nearest

Table 3. Spatial metric *SP* of conventional and improved NSGA-II.

Serial	Conventional *SP*	Improved *SP*
1.	0.5395	0.4052
2.	0.5538	0.4044
...	.....	.....
11.	0.5033	0.4241
12.	0.6121	0.3090
...	.....	.....
20.	0.5085	0.3784

individual of the Pareto front, is calculated. Without improvement, the average distance is in the range of $2.28e-02 < d < 3.90e-02$. After improvement, the average distance is in the range of $5.45e-02 < d < 6.25e-02$. The non-dominated solutions after improvement have higher spatial dispersion and more reasonable distribution than before improvement.

After the average distance calculation and comparison, the spatial metric *SP* is further calculated, as shown in Table 3. Without improvement, the spatial metric *SP* is approximately 0.5. After improvement, the spatial metric *SP* is approximately 0.4 and the latter is smaller than the former. After improvement, the deviation between the crowding distance and the average distance of the Pareto front solutions are smaller and the solution distributions are more uniform than before improvement.

5.3. Multi-objective optimization Pareto front figure after improvement

A set of relative optimal solutions is obtained with the aforementioned improvement in the multi-objective optimization, as shown below.

Fig. 9 shows a 3D diagram and three decomposed 2D diagrams. The Pareto optimal solutions show the linear relationship between consumed time f_1 and joint jerk f_3, which significantly declines as time f_1 increases. However, within time f_1, the trajectory length f_2 is likely to be shorter or longer in the figure. Moreover, no clear relationship between trajectory length f_2 and joint jerk f_3 is observed in the optimal solutions. The value of joint jerk f_3 is not determined by the trajectory length f_2. Therefore, in this multi-objective optimization project, the relationship between consumed time f_1 and joint jerk f_3 is a priority solution to practical engineering selection.

6. CONCLUSION

This paper focuses on a robot manipulator and multi-objective optimization in trajectory motion planning. In the obstacle avoidance operation, the consumed time, trajectory length, and joint jerk are used as the multi-objective in optimization. NSGA-II is used to obtain

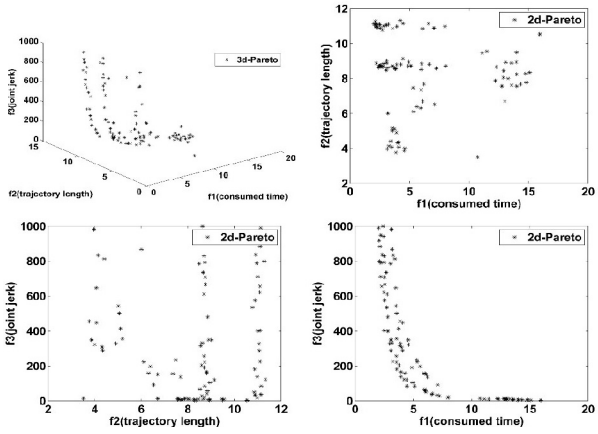

Fig. 9. Three-objective optimization Pareto front.

the Pareto front of the multi-objective optimization. This study attempts to analyze layered sorting and crowding distance and to improve the crowding distance mechanism. Finally, simulation results show the effectiveness of the improved algorithm and find the relationship between consumed time and joint jerk a priority solution to practical engineering selection.

REFERENCES

[1] B. I. Kazem, A. I. Mahdi, and A. T. Oudah, "Motion planning for a robot arm by using genetic algorithm," *Jordan Journal of Mechanical and Industrial Engineering*, vol. 2, no. 3, pp. 131-136, September 2008.

[2] A. Piazzi and A. Visioli, "Global minimum-jerk trajectory planning of robot manipulators," *IEEE Transactions on Industrial Electronics*, vol. 47, no. 1, pp. 140-149, February 2000. [click]

[3] P. K. Sahu, "Optimal trajectory planning of industrial robots using geodesic," *International Journal of Robotics and Automation*, vol. 5, no. 3, September 2016.

[4] B. Ombuki, B. J. Ross, and F. Hanshar, "Multi-objective genetic algorithms for vehicle routing problem with time windows," *Applied Intelligence*, vol. 24, no. 1, pp. 17-30, 2006. [click]

[5] M. A. Abido, "Multiobjective evolutionary algorithm for electric power dispatch problem," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 3, pp. 315-329, June 2006. [click]

[6] K. Sindhya, K. Miettinen, and K. Deb, "A hybrid framework for evolutionary multi-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 4, pp. 495-511, August 2013. [click]

[7] C. A. C. Coello, "Evolutionary multi-objective optimization: a historical view of the field," *IEEE Computational Intelligence Magazine*, vol. 1, no. 1, pp. 28-36, February 2006. [click]

[8] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. Fonseca, "Performance assessment of multiobjective optimizers: an analysis and revies," *IEEE Trans. on Evolutionary Computation*, vol. 7, no. 2, pp. 117-132, April 2003. [click]

[9] E. Zitzler and L. Thiele, "Multi-objective evolutionary algorithms: a comparative case study and the strength Pareto approach," *IEEE Trans. on Evolutionary Computation*, vol. 3, no. 4, pp. 257-271, November 1999. [click]

[10] N. K. Madavan, "Multiobjective optimization using a Pareto differential evolution approach," *Proceedings of the Congress on Evolutionary Computation*, Honolulu, IEEE, vol. 2, pp. 1145-1159, 2002.

[11] J. D. Knowles and D. W. Corne, "Approximating the nondominated front using the Pareto archived evolution strategy," *Evolutionary Computation*, vol. 8, no. 2, pp. 149-172, 2000. [click]

[12] D. W. Corne, J. D. Knowles, and M. J. Oates, "The pareto envelope-based selection algorithm for multiobjecitve optimization," In: Schoenauer M, Deb K, Rudolph G, eds. *Proceedings of the Parallel Problem Solving from Nature VI Conference*. Paris, France, 2000. 839-848.

[13] Z.-Y. Xing, Y. Zhang, Y.-L. Hou, and L.-M. Jia, "On generating fuzzy systems based on Pareto multi-objective cooperative coevolutionary algorithm," *International Journal of Control, Automation, and Systems*, vol. 5, no. 4, pp. 444-455, August 2007.

[14] K. Deb, *Multi-objective Optimization Using Evolutionary Algorithms*, John Wiley & Sons, Chicester, pp. 13-46, 2001.

[15] X. F. Zou, Y. Chen, M. Z. Liu, and L. H. Kang, "A new evolutionary algorithm for solving many-objective optimization problems," *IEEE Trans. on Systems, Man, and Cybernetics, Part B*, vol. 38, no. 5, pp. 1402-1412. 2008. [click]

[16] C. A. C. Coello, A. D. Veldhuizen, and G. B. V. Lamont, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Kluwer Academic, pp. 256-278, 2002.

[17] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: improving the strength Pareto evolutionary algorithm for multiobjective optimization," *EUROGEN 2001-Evolutionary Methods for Design, Optimisation and Control with Applications to Industrial Problems*, September 2001.

[18] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182-197, April 2002. [click]

[19] J. Knowles and D. Corne, "The Pareto archived evolution strategy: a new baseline algorithm for Pareto multiobjective optimisation," *Proceedings of the 1999 Congress on Evolutionary Computation, CEC 99*, vol. 1, July 1999.

[20] Q. Zhang and H. Li. "MOEA/D: a multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, issue. 6, pp. 712-731, December 2007. [click]

[21] G. Na, X. Sun, D. Gong, Y. Zhang, "Solving robot path planning in an environment with terrains based on interval multi-objective PSO," *International Journal of Robotics and Automation*, vol. 31, no. 2, pp. 100-110, January 2016.

[22] F.-A. Fortin and M. Parizeau, "Revisiting the NSGA-II crowding-distance computation," *GECCO' 13*, Amsterdam, The Netherlands, July 6-10, 2013.

[23] C.-S. Tsou, S.-C. Chang, and P.-W. Lai, "Using crowding distance to improve multiobjective PSO with local search," *Swarm Intelligence, Focus on Ant and Particle Swarm Optimization, InTech*, ISBN: 978-3-902613-09-7, pp. 532, December 2007.

[24] M. A. Jabbar, B. L. Deekshatulu, and P. Chandra, "Classification of heart disease using *k*-nearest neighbor and genetic algorithm," *International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA)*, 2013.

[25] J. Yan, M. Li, Z. Xu, and J. Xu, "A simple Pareto adaptive ε-domination differential evolution algorithm for multi-objective optimization," *The Open Automation and Control Systems Journal*, vol. 7, 338-345, 2015. [click]

[26] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II," *IEEE Trans. on Evolutionary Computation*, vol. 12, no. 2, pp. 284-302, April 2009. [click]

[27] R. G. L. D'Souza, K. C. Sekaran, and A. K. Andasamy, "Improved NSGA-II based on a novel ranking scheme," *Journal of Computing*, vol. 2, no. 2, pp. 91-95, February 2010.

[28] Y.-L. Li, W. Shao, J.-T. Wang, and H. Chen, "An improved NSGA-II and its application for reconfigurable pixel antenna design," *Radioengineering*, vol. 23, no. 2, pp. 733-738, June 2014.

[29] D.-Q. Guo, J-P. Wang, J. Huang, *et al*. "Chaotic-NSGA-II: an effective algorithm to solve multi-objective optimization problems," *International Conference on Intelligent Computing and Integrated Systems* (ICISS), Guilin, China, pp. 20-23, 2010.

[30] H. Wang, W. Sun, and P. X. Liu, "Adaptive intelligent control of nonaffine nonlinear time-delay systems with dynamic uncertainties," *IEEE Transactions on systems, Man, and Cybernetics: Systems*, vol. 47, no. 7, pp. 1474-1485, July 2017.

[31] X. Zhao, P. Shi, X. Zheng, and J. Zhang, "Intelligent tracking control for a class of uncertain high-order nonlinear systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 9, pp. 1976-1982, September 2016.

[32] H. Wang, X. Liu, and P. Shi, "Observer-based fuzzy adaptive output-feedback control of stochastic nonlinear multiple time-delay systems," *IEEE Transactions on Cybernetics*, vol. 47, no. 9, pp. 2568-2578, September 2017. [click]

[33] X. Zhao, P. Shi, and X. Zheng, "Fuzzy adaptive control design and discretization for a class of nonlinear uncertain systems," *IEEE Transactions on Cybernetics*, vol. 46, no. 6, pp.1476-1483, June 2016.

[34] X. Zhao, H. Yang, H. R. Karimi, and Y. Zhu, "Adaptive neural control of MIMO nonstrict-feedback nonlinear systems with time delay," *IEEE Transactions on Cybernetics*, vol. 46, no. 6, pp.1337-1349, June 2016.

[35] H. Wang, X. Liu, and K. Liu, "Adaptive fuzzy tracking control for a class of pure-feedback stochastic nonlinear systems with non-lower triangular structure," *Fuzzy Sets and Systems*, vol. 302, pp. 101-120, November 2016. [click]

[36] L. Liu, C. Chen, X. Zhao, and Y. Li, "Smooth trajectory planning for a parallel manipulator with joint friction and jerk constraints," *International Journal of Control, Automation and Systems*, vol. 14, no. 4, pp. 1022-1036, 2016.

[37] H.-I Lin, Y.-C. Liu, "Minimum-jerk robot joint trajectory using particle swarm optimization," *Proc. of 2011 First International Conference on Robot, Vision and Signal Processing*, pp. 118-121, 2011.

[38] P. Huang, K. Chen, J. Yuan, and Y. Xu, "Motion trajectory planning of space manipulator for joint jerk minimization," *Proc. of IEEE International Conference on Mechatronics and Automation*, pp. 3543-3548, 2007. [click]

[39] P. B. Mulik, "Optimal trajectory planning of industrial robot with evolutionary algorithm," *Proc. of 2015 International Conference on Computation of Power, Energy, Information and Communication*, pp. 0256-0263, 2015.

Ying Huang received his B.S. degree in 2001 and his M.S. degree in 2006, both from Fu Zhou University, China. And currently he is a Ph.D. candidate in control science and engineering in Shanghai University. His research interests are in the fields of robot manipulator motion and control, humanoid robot.

Minrui Fei is a Professor and Doctoral Supervisor at Shanghai University, Vice-chairman of Chinese Association for System Simulation, and Standing Director of China Instrument & Control Society. He received his B.S., M.S. and Ph.D. degrees all from Shanghai University, in 1982, 1992 and 1997, respectively. His research interests include networked advanced control and system implementation, distributed and fieldbus control systems, robot manipulator control, modeling and control of flexible arms, and cooperative robots.