

# A Multi-objective Trajectory Planning Method Based on the Improved Elitist Non-dominated Sorting Genetic Algorithm

Zesheng Wang

Zhejiang University of Technology

Yanbiao Li (✉ [lybrory@zjut.edu.cn](mailto:lybrory@zjut.edu.cn))

College of Mechanical Engineering <https://orcid.org/0000-0001-9768-0687>

Kun Shuai

Zhejiang University of Technology

Wentao Zhu

Zhejiang University of Technology

Bo Chen

Zhejiang University of Technology

Ke Chen

Zhejiang University of Technology

---

## Original Article

**Keywords:** hybrid manipulator, bezier curve, improved optimization algorithm, trajectory planning, multi-objective optimization

**Posted Date:** December 4th, 2020

**DOI:** <https://doi.org/10.21203/rs.3.rs-118340/v1>

**License:** © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

---

**Version of Record:** A version of this preprint was published at Chinese Journal of Mechanical Engineering on February 12th, 2022. See the published version at <https://doi.org/10.1186/s10033-021-00669-x>.

# A multi-objective trajectory planning method based on the improved elitist non-dominated sorting genetic algorithm

Zesheng Wang<sup>1,2</sup>, Yanbiao Li<sup>1,2,\*</sup>, Kun Shuai<sup>1,2</sup>, Wentao Zhu<sup>1,2</sup>, Bo Chen<sup>1,2</sup>, Ke Chen<sup>1,2</sup>

## Abstract

A multi-objective trajectory planning approach based on an improved elitist non-dominated sorting genetic algorithm (INSGA-II) is proposed. Trajectory function is planned with a new composite polynomial that by combining of quintic polynomials with cubic Bezier curves. Then, an INSGA-II, by introducing three genetic operators: ranking group selection (RGS), direction-based crossover (DBX) and adaptive precision-controllable mutation (APCM), is developed to optimize travelling time and torque fluctuation. Inverted generational distance, hypervolume and optimizer overhead are selected to evaluate the convergence, diversity and computational effort of algorithms. The optimal solution is determined via fuzzy comprehensive evaluation to obtain the optimal trajectory. Taking a serial-parallel hybrid manipulator as instance, the velocity and acceleration profiles obtained using this composite polynomial are compared with those obtained using a quintic B-spline method. The effectiveness and practicability of the proposed method are verified by simulation results.

**Keywords:** hybrid manipulator; bezier curve; improved optimization algorithm; trajectory planning; multi-objective optimization

## 1 Introduction

With the advancement of the times, robotics technology is also developing rapidly, which makes manipulators widely applied in industrial filed. Responding to many practical robotic applications (such as palletizing, labeling, spot welding), the trajectory planning of the manipulator is very importance for accomplishing tasks, which generally involves two key problems, namely trajectory generation and trajectory optimization. The former provides the precondition for the trajectory planning. Meanwhile, the latter is an efficient way to improve the performance of the trajectory and get the most of the manipulator [1].

Trajectory generation is usually to establish a smooth trajectory of a manipulator by means of interpolating between any two given poses. Common interpolation functions include polynomial, spline, Bezier, and NURBS, etc. Wang et al. [2] divided the joint motion of an industrial robot into accelerated part, constant velocity part and decelerated part, where the accelerated and decelerated trajectories were planned with fourth-order polynomials formed with the property of the root multiplicity. In [3], the trajectory must pass through a number of given discrete characteristic points. The time-optimal and jerk-continuous trajectory planning has been implemented under kinematic constraints by combining cubic splines in Cartesian space and septuple B-splines in joint space. Shi et al. [4] adopted a quintic non-uniform rational B-spline (NURBS) to construct a flexible trajectory of a 6-degree-of-freedom

(DOF) robot, which can guarantee jerk continuous and also velocity and acceleration of initial and final point both can be specified. Dinçer and Çevik [5] designed a composite polynomial composed of quadratic Bezier curves and cubic polynomials for the trajectory planning of a 2-DOF parallel mechanism. The composite polynomials provide a smoother transition at the starting and ending points compared to Bezier curves, namely velocities are zero at the endpoints. Motivated by Ref [5], we develop a composite polynomial by combining quintic polynomials with cubic Bezier curves. The polynomial part can ensure that the velocities and accelerations of the actuators of a manipulator at initial and final movement are zero to achieve the stability of start and stop. The bezier curve part can improve the adjustability of the trajectory through optimization algorithm to obtain better objective performance.

In the processing of trajectory optimization, many different technical criteria have been defined to meet the requirements of the task [6-8]. For instance, the execution time and jerk are intended for improving the productivity and keeping the trajectories smooth. The energy and torque are aimed at reducing the energy consumption and the load on the actuator of the robot. In addition, with respect to trajectory optimization techniques, evolutionary algorithms, which have high efficiency, robustness and adaptability, have been widely applied to resolve minimization problems for objective trajectory functions. In [9], GA was applied to the trajectory planning problem with

---

\*Correspondence: lybrory@zjut.edu.cn

<sup>1</sup> Key Laboratory of E & M, Ministry of Education & Zhejiang Province, Zhejiang University of Technology, Hangzhou 310032, China  
Full list of author information is available at the end of the article.

nonlinear constraints and obstacles to minimize the joint rotation angles of a 2-DOF robot. Lin [10] employed particle swarm optimization (PSO) with K-means clustering to solve the near optimal solution of a minimum-jerk joint trajectory. Only considering a single-objective function may not be suitable for meeting multiple requirements in real-world applications. Currently, in terms of the multi-objective optimization, multi-objective evolutionary algorithms (MOEA) typically utilize non-dominated sorting to provide a number of Pareto solutions for decision-makers rather than converting all objectives into a single-objective function. Thus, this optimization technique has become more popular with researchers. In [11], the time-jerk trajectory of a robotic manipulator was interpolated in the joint space by means of 5th-order B-splines and then optimized by NSGA-II. In [12], a multi-objective function, including motion time, dynamic disturbance, and jerk, was addressed by using multi-objective particle swarm optimization (MOPSO) to obtain the high efficiency and safe motion trajectory of a space robot. Marcos et al. [13] combined the closed-loop pseudoinverse method with a multi-objective genetic algorithm (MOGA) to minimize the joint displacement and the positional error of the end-effector. Ramabalan et al. [14] adopted B-spline functions to define the trajectory of a robot manipulator, and the trajectories optimized by NSGA-II and multi-objective differential evolution (MODE) were compared. The results showed that the efficiency of the MODE technique is higher, and the diversity of the Pareto solution of NSGA-II is richer.

The main differences among the above trajectory planning methods lie in the processing of the interpolation functions and trajectory optimization models, as well as the selection of interpolation functions and optimization algorithms [15]. Due to the complexity of the trajectory problem of manipulators, which needs to be studied in this article, there is still improvement space in the accuracy and efficiency of the solution method. Therefore, this study first adopts the proposed composite polynomial to construct a point-to-point trajectory. Then, to improve the convergence and diversity of the Pareto optimal front and also the computational efficiency of the traditional NSGA-II, this study proposes an INSGA-II to obtain the time and torque fluctuation optimal trajectories.

This article is organized as follows. Section 2 presents a composite polynomial curve, by combining cubic bezier with quintic polynomial, for establishing trajectory optimization model. Section 3 proposes the three improved genetic operators and INSGA-II. Section 4 delineates two performance measures for the Pareto front and the computational efficiency of the algorithms in detail. Section 5 simulates on the trajectory planning problem of the manipulator and makes a discussion about the simulation

results. Finally, Section 6 outlines the conclusions.

## 2 Trajectory optimization modelling

The trajectory planning is generally carried out in operating space and in joint space. In terms of the trajectory planning in joint space, it can avoid singular configurations for the robotic arm, but its application has been limited due to the nonlinear relationship between operating space and joint space [16, 17]. Moreover, the analytical expressions for forward kinematic solutions of most parallel mechanisms are hard to obtain and only can find the numerical solutions. Another method is to perform the trajectory planning in operating space; it is intuitive to avoid obstacles and easy to track the end-effector position and posture [18, 19], but the problem of kinematic singularity is difficult to address using such a method. To facilitate analyzing the dynamic performances of a hybrid manipulator, the trajectory planning problem is handled in the space of the output angle of joint moving platform (OAJ) [33]. It is noted that the mapping of the trajectory between the joint space and the OAJ space can be obtained by applying the inverse kinematics transformation of each joint of the hybrid manipulator, and the OAJ space can be transferred to the operating space through the forward kinematics analysis.

### 2.1 Objective function

In light of the manipulator, it is expected that the joint trajectory is smooth enough to avoid overlarge mechanical vibration, and reduce travelling time as much as possible to improve productivity. In Fang and Huang's works [11, 20], the time integral was set to be a term of the objective function. Li and Wang [21, 22] further took the minimum absolute value of torque fluctuation into account. In this study, two objective functions are involved, namely, the travelling time and the torque fluctuation. The objective functions can be mathematically defined as follows.

Minimize:

$$\begin{cases} f_1(t) = T = \int_0^T dt \\ f_2(t) = \sum_{i=1}^n \int_0^T |\tau_i(t) - \tau_i(t-1)| dt \end{cases} \quad (1)$$

where  $f_1$  denotes the total travelling time,  $f_2$  is the variance of the actuator torque, which is to ensure the stability of the manipulator.  $\tau_i(t)$  and  $\tau_i(t-1)$  denote the torque of the actuator at former and current movement, respectively.  $n$  denotes the number of the robotic joint.

It is obvious that the two objective functions constrain each other because of the opposite effects. The reduction in travelling time would lead to the larger torque fluctuation and less smooth trajectory, while reducing the torque fluctuation would result into the longer execution time and lower work efficiency. The trajectory planning inevitably encounters a trade-off between these two objective

functions. Therefore, by solving the optimization problem with a multi-objective optimization technique, a set of Pareto solutions can be obtained and provided for decision-makers to select.

## 2.2 Constraint conditions

The kinematic constraints include the limits of angular velocity and acceleration, and the dynamic constraint is mainly the actuator torque. To guarantee the starting and stopping stability of the manipulator, the velocity and acceleration of the actuator are both zero at the endpoints. The trajectory of each actuator needs to meet the following equality and inequality constraints.

$$\begin{cases} c_v(t_0, t_f) = 0, \quad c_a(t_0, t_f) = 0, \quad i = 1, 2, \dots, n_d \\ c_d(t) = \frac{\max}{i=1, 2, \dots, n_d} |\theta_i(t)| \leq \sup |\theta_{im}| \\ c_v(t) = \frac{\max}{i=1, 2, \dots, n_d} |\dot{\theta}_i(t)| \leq \sup |\dot{\theta}_{im}| \\ c_a(t) = \frac{\max}{i=1, 2, \dots, n_d} |\ddot{\theta}_i(t)| \leq \sup |\ddot{\theta}_{im}| \\ c_m(t) = \frac{\max}{i=1, 2, \dots, n_d} |\tau_i(t)| \leq \sup |\tau_{im}| \end{cases} \quad (2)$$

where  $t_0$  and  $t_f$  represent initial and final moment.  $c_d$ ,  $c_v$ ,  $c_a$ , and  $c_m$  denote the maximum angle, angular velocity, angular acceleration and torque of each actuator during the entire motion. The equalities describe the initial and final state required for the manipulator, and the inequalities describe the performance of each actuator.

The maximum value of each OAJ velocity and acceleration can be obtained based on the search method [23] by combining the actuator velocity and acceleration boundaries in Eq. (2) with the workspace of the manipulator [21].

$$\begin{cases} \psi_{j\max} = J_{o\max} \dot{\theta}_{i\max} \\ \ddot{\psi}_{j\max} = J_{o\max} \ddot{\theta}_{i\max} + \ddot{J}_{o\max} \dot{\theta}_{i\max} \end{cases} \quad (3)$$

where  $\psi$  denotes the vector of OAJ,  $j$  denotes each OAJ.  $J_{o\max}$  represents Jacobian matrix of the manipulator corresponding to the maximum value of each OAJ velocity.

The maximum value of each OAJ velocity and acceleration in Eq. (3) can be satisfied through determination of the travelling time by the following formula

$$T \geq \max \left( \frac{|\dot{\psi}_{j\max}|}{\dot{\psi}_{j\max}}, \sqrt{\frac{|\ddot{\psi}_{j\max}|}{\ddot{\psi}_{j\max}}} \right) \quad (4)$$

where  $\dot{\psi} = d\psi/d\lambda$  and  $\ddot{\psi} = d^2\psi/d\lambda^2$ .

## 2.3 Composite curve

In the Ref [5], a new composite polynomial is generated by combining cubic polynomials with Bezier curves based on quadratic Bernstein polynomials. This trajectory planning provides a much lower jerky motion that decreases unwanted vibration. However, the acceleration of the

mechanism is not zero at initial and final points by applying the composite polynomial into constructing trajectory, which is unfavorable for the start and stop of the manipulator. Therefore, we develop a new composite polynomial by combining quintic polynomials with Bezier curves based on cubic Bernstein polynomials. The bezier curve part can improve the adjustability of the trajectory through an optimization algorithm to obtain better objective performances. The polynomial part can make that the velocities and accelerations of the actuators at initial and final movement are zero to improve the stability of start and stop.

A Bezier curve of degree  $n$  can be defined in parametric form as

$$y(x) = \sum_{i=0}^n B_i^n(x) P_i = \sum_{i=0}^n \binom{n}{i} x^i (1-x)^{n-i} P_i, \quad x \in [0, 1] \quad (5)$$

where the polynomials  $B_i^n(\lambda)$  are known as Bernstein basis polynomial of order  $n$ ,  $\binom{n}{i}$  is a binomial coefficient.  $P_i$  is the given control point to construct the Bezier curve.

To further improve the start-stop stability of the manipulator, the quintic polynomial can be designed as

$$g(\lambda) = 10\lambda^3 - 15\lambda^4 + 6\lambda^5, \quad \lambda \in [0, 1] \quad (6)$$

Hereupon, the composite polynomial can be obtained by substituting the quintic polynomials into Eq. (6) such that  $x = g(\lambda)$ . It can be expressed as

$$y(\lambda) = \sum_{i=0}^n \binom{n}{i} g^i(\lambda) (1-g(\lambda))^{n-i} P_i \quad (7)$$

In Eq. (7),  $\lambda$  denotes the normalized time, for the travelling time  $T = t_f - t_s$ , if we define  $t = \lambda \cdot T$ , the trajectory of the OAJ can be expressed as

$$\begin{cases} \psi_j(t) = \sum_{i=0}^n \binom{n}{i} g^i\left(\frac{t}{T}\right) g\left(1 - \frac{t}{T}\right)^{n-i} P_{i,j} \\ \dot{\psi}_j(t) = \frac{n}{T} \sum_{i=0}^2 B_i^{n-1}(t) g\left(\frac{t}{T}\right) (P_{i+1,j} - P_{i,j}) \end{cases} \quad (8)$$

The trajectory of each actuator can be obtained by applying the inverse kinematics transformation into Eq. (8).

The bezier curve of Eq. (5) provides a better convergence to the starting and ending points, while the polynomial of Eq. (6) provides a smooth transition in the vicinity of the endpoints. By this method, we tried to exploit the advantages of each polynomial, and the corresponding results would be presented in Section 4.

## 3 Proposed method

The non-dominated sorting genetic algorithm has established itself as a benchmark algorithm for multi-objective optimization, which was first proposed by Deb et al [24]. The main contribution is to obtain the Pareto solutions by sorting the dominated relationship among individuals. However the basic algorithm suffers from a high order of complexity and highly depends on shared parameters. Hereto, NSGA-II [25] replaces the shared

parameters with the crowding degree and introduced elite strategy to retain the excellent individuals in the iterative processing. It, adopting the fast non-dominant sorting method to reduce the computational complexity, has demonstrated the ability to find a good spread of solutions and converge close to the true Pareto-optimal front. Subsequently, to solve the insufficiency of NSGA-II in dealing with the four or many more objectives optimization problems, NSGA-III [26] utilizes the reference point method to substitute the crowding degree method in the replace operation, which can perform better in balancing the diversity and convergence of the algorithm.

There are only two objective functions in the trajectory optimization problem of this article, so we consider NSGA-II as the basic algorithm. However, NSGA-II adopts the selection, crossover and mutation operators of traditional GA, which leads to the loss of population diversity and the poor search ability of the algorithm [27]. Moreover, the manipulator is a nonlinear multivariable and strong-coupling system with extremely complex kinematic and dynamic models. To avoid problems such as premature convergence and low convergence speed in the processing of trajectory optimization of the manipulator using the conventional NSGA-II, INSGA-II, integrating three specially designed operators, is proposed to quickly and accurately obtain the optimal trajectory.

### 3.1 Ranking group selection

The selection of GA commonly adopts the roulette wheel selection and tournament selection. Although their operation mechanism is simple, the process is complicated and requires repeated comparison of the fitness [25, 27]. Motivated by Ref [27], a ranking group selection is used to replace the conventional selection.

The procedure of the RGS is shown in Fig. 1. First, the parent population  $P_0$  of size  $N$  is randomly initialized based on the constraints of the designed variables, where  $N$  is set to a multiple of four. Then the initialized population is sorted into several ranks based on the non-domination sorting. The solutions are assigned fitness equal to the corresponding non-domination levels. Individuals in the first front are assigned fitness value of 1 and individuals in second are given a fitness value 2 and so on. Afterward, the sorted population is uniformly divided into 4 elements in sequence, namely  $X_1$ ,  $X_2$ ,  $X_3$ , and  $X_4$ . Using the basic concept of combinatorics, there are six cases in which two elements are selected from four elements for pairing, including  $(X_1, X_2)$ ,  $(X_1, X_3)$ ,  $(X_1, X_4)$ ,  $(X_2, X_3)$ ,  $(X_2, X_4)$ , and  $(X_3, X_4)$ . The paired groups of individuals formed by RGS are  $\mathbf{I}^A = (X_1, X_1, X_1, X_2, X_2, X_3)$  and  $\mathbf{I}^B = (X_2, X_3, X_4, X_3, X_4, X_4)$ . In the iterative process,  $\mathbf{I}^A$  is responsible for guiding the population towards the optimal region while  $\mathbf{I}^B$  is responsible for increasing the population diversity.

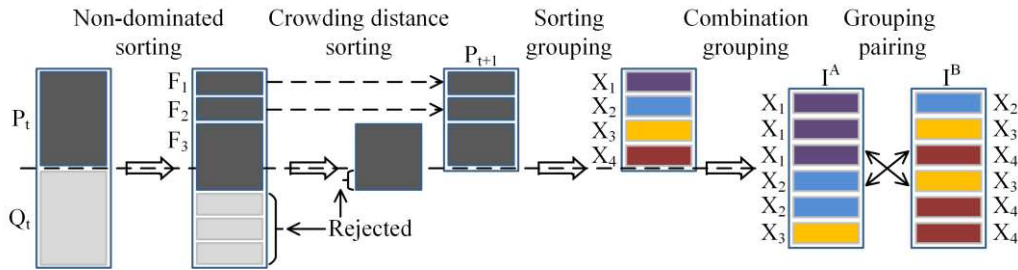


Figure 1 Schematic of the RGS operation

In Fig. 1,  $P_t$  and  $Q_t$  represent parent population and offspring population, and  $P_{t+1}$  represents the parent population in next generation.  $F_1$ ,  $F_2$  and  $F_3$  denote the different ranks of the population.

The individuals in  $\mathbf{I}^A$  and  $\mathbf{I}^B$  that are paired in turn to participate in the crossover can improve the gene diversity of the population and avoid inbreeding, which can promote the generation of high-quality individuals in the procession of gene recombination. Additionally, the RGS is a way to directly calculate the values of the objective functions instead of contrasting looping manner, so the time complexity of the method is small and such the method is easy to implement.

### 3.2 Direction-based crossover

The crossover of GA commonly adopts the simulated binary crossover, which uses a random way to carry out the gene exchange between individuals. Although the operation

mechanism is simple, the method causes degree of blindness. Based on the principle that the better the objective function is, the closer the individual is to the optimal region, and a DBX operator is designed.

Taking two objective functions and two-dimensional variables as an example is shown in Fig. 2. The distribution of the Pareto solution set is obtained according to Section 3.1, and the corresponding individuals are assumed to be  $X_1$  and  $X_2$ . The DBX takes  $X_1$  as the center and uses the direction vector  $d_{11}$  or  $d_{12}$  as the crossover direction to generate new individuals along random steps. The DBX operator can be mathematically expressed as

$$\begin{cases} X_i^* = I_i^A + r_{ij} \times d_{ij}^r \\ d_{ij}^r = I_{ij}^A - I_{ij}^B \quad i = 1, 2, \dots, 3n/2, j = 1, 2, \dots, m \end{cases} \quad (9)$$

where  $i$  denotes the  $i$ -th individual,  $j$  denote the variable dimension, the parameter  $r_{ij}$  is a uniformly distributed

random number in the interval  $[-1, 1]$ . Different from the traditional fixed step crossover, the step size in DBX is randomly generated by the parameter  $r_{ij}$ , which expands the search range of the algorithm.

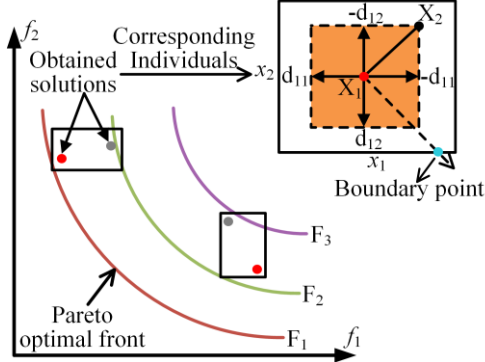


Figure 2 Schematic of the DBX operation

Meanwhile, different paired individuals produce different rectangular areas, which indicate that the difference between paired individuals can increase the population diversity to improve the search ability of the algorithm and increase the generation probability of high-quality individuals. It is noted that if the population generated by Eq. (9) crosses the boundary, it will be limited to the boundary to ensure the rationality of the population genes.

$$p_{it} = \begin{cases} p_{i\min} & \text{if } p_{it} < p_{i\min} \\ p_{i\max} & \text{if } p_{it} > p_{i\max} \\ p_{it} & \text{others} \end{cases} \quad (10)$$

where  $p_{it}$  denotes the value of the  $i$ -th individual in  $t$ -th iteration.  $p_{i\min}$  and  $p_{i\max}$  represent the minimum and maximum value of designed variables.

### 3.3 Adaptive precision-controllable mutation

The purpose of introducing mutation in GA is twofold: One is to make genetic algorithm have local random search ability. The second is to maintain population diversity of the algorithm to avoid immature convergence. In Ref [28], a simple and efficient precision-controllable mutation (PCM) operator is proposed for exploration and exploitation. On the basis of the Ref [28], a self-adaptive mechanism is incorporated into the PCM to improve the convergence speed of the algorithm in this article.

The exploration and exploitation of the PCM can be expressed as Eqs. (11) ~ (16) in Ref [28].

$$X'_i = X_i + \Delta\alpha \quad (11)$$

$$X'_i = X_i - \Delta\alpha \quad (12)$$

where

$$\Delta\alpha = \frac{1}{10^{\text{Random}(p)+1}} \times (\text{Random}(9)+1)$$

$$X'_i = X_i + \Delta\beta \quad (13)$$

$$X'_i = X_i - \Delta\beta \quad (14)$$

where  $\Delta\beta = X_i \times \Delta\alpha - X_i$

$$X'_i = X_i + \Delta\gamma \quad (15)$$

$$X'_i = X_i - \Delta\gamma \quad (16)$$

where  $\Delta\gamma = X_i \div \Delta\alpha - X_i$

The variable  $p$  is the parameter to control the precision in decision space. Function Random ( $p$ ) can generate a pseudorandom number in the range of 0 to  $p - 1$ . If the required search precision is 0.001, the parameter  $p$  can be set to be 3. The value of random number Random (3) should be in the set of  $\{0, 1, 2\}$ , then the corresponding value ranges of  $\Delta\alpha$  from 0.001 to 0.9.

Eqs. (11) and (12) are intended for exploitation, while Eqs. (13) to (16) are designed for exploration. The operator can effectively explore and exploit the decision space, and its computation process is simple and precision is controllable. However, the mutation operator in Ref [28] has not sufficiently utilized the potential information of the contemporary population which can be used for the adaptive selection of exploitation or exploration.

A common adaptive adjustment method is to use the information of the objective function value to adjust the mutation strategy. For the trajectory planning problem of the manipulator, the real Pareto front cannot be obtained in reality. Therefore, the distribution value of population instead of objective function can be used as a parameter to select exploitation and exploration. In this study, the ratio of contemporary population space to decision space is used to determine the mutation strategy of the individual, which can promote rapid and stable evolution of the population.

In the early iterations, the difference between the individuals is large, so the exploration is selected to ensure the diversity of the population and avoid the algorithm falling into local optimality. In the later iterations, the population gradually tends toward the region of the optimum, and the difference between the individuals is small, hence the exploration is selected to keep the excellent individuals to improve the search effective. The APCM operator can be expressed as

$$\xi_i = \frac{x_{i\max}^T - x_{i\min}^T}{x_{i\max} - x_{i\min}} \quad (17)$$

$$\begin{cases} \text{if } \xi_i \geq a, a = \text{rand}(1) \\ \text{exploitation} \begin{cases} X_{\text{Temp}} = X_i + \Delta\alpha & r=1, r = \text{rand}(2) \\ X_{\text{Temp}} = X_i - \Delta\alpha & r=2 \end{cases} \\ \text{if } 1 - \xi_i > a \\ \text{exploration} \begin{cases} X_{\text{Temp}} = X_i + \Delta\beta & r=1, r = \text{rand}(4) \\ X_{\text{Temp}} = X_i - \Delta\beta & r=2 \\ X_{\text{Temp}} = X_i + \Delta\gamma & r=3 \\ X_{\text{Temp}} = X_i - \Delta\gamma & r=4 \end{cases} \end{cases} \quad (18)$$

where  $x_{i\max}^t$  and  $x_{i\min}^t$  are maximum and minimum value of the individuals in contemporary population,  $x_{i\max} - x_{i\min}$

denotes the magnitude of decision space.

Compared with the mutation operator in Ref [28], adding the adaptive adjustment of the mutation operator can promote the balance of local search and global exploration capabilities, thereby making the Pareto boundary distribution better.

### 3.4 Overall algorithm

The flowchart used the INSGA-II for the trajectory planning of the manipulator is shown in Fig. 3. Initializing randomly the parameters within the threshold value gains the initial trajectory curves (Eq. (8)) and the storage of these trajectory is performed, then calculating the objective functions for each chromosome. The first generation population performs non-dominated sorting to find a set of Pareto front (PF), and the population is sorted by the crowding distance. Afterwards, a new parent population is generated by RGS, DBX, and APCM operator, and the parents and offspring are combined to form a population of  $N$  individuals according to the elite strategy. It is continuously judged whether it reaches the number of iterations, and the objective function of each trajectory is compared. Finally, the Pareto solution of the objective functions is obtained after the iteration and the corresponding designed parameters are output.

Compared with conventional NSGA-II, the proposed INSGA-II uses the combination principle to construct a selection operator, which can avoid repeated comparison of the fitness between individuals to improve the convergence speed of the algorithm. DBX can expand the search space and increase the generation probability of high-quality individuals, thereby improving the search ability and convergence speed of the algorithm. The local random search ability of the APCM can accelerate the convergence to the optimal solution, and the exploration strategy of the operator can expand the search space to keep the population diversity.

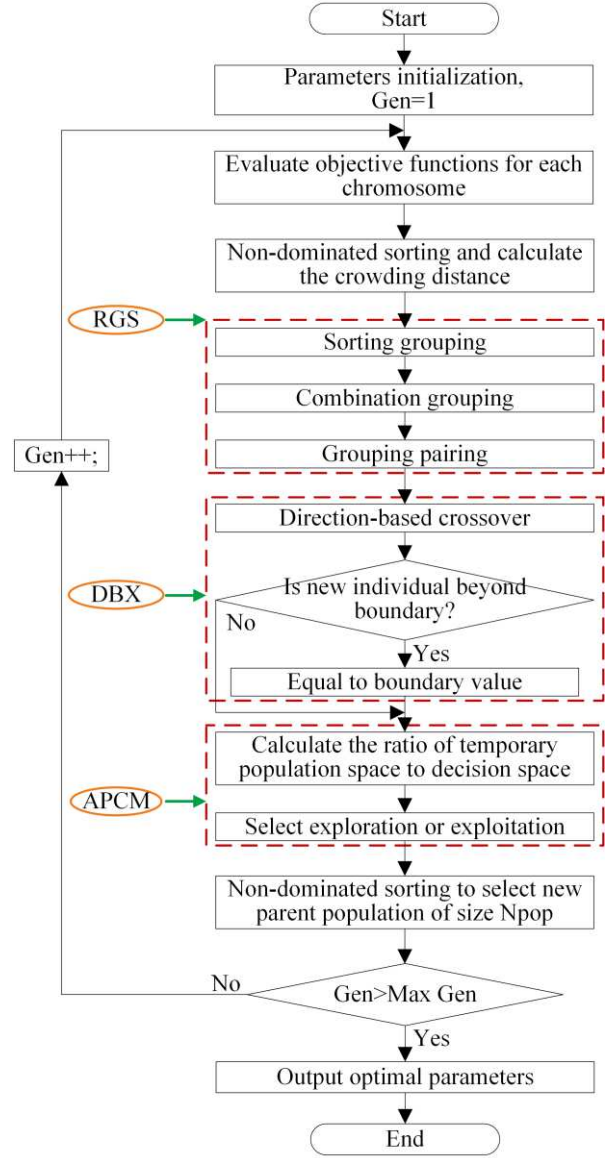


Figure 3 Flowchart of INSGA-II algorithm

## 4 Performance measures for INSGA-II

Given a set of solutions by Section 3, but in some cases, the weight cannot be determined by a decision maker due to insufficient information related to the different criteria. In that situation, we offer a strategy. First, the performances of the INSGA-II are evaluated according to the convergence, diversity and computational efficiency, and then the fuzzy comprehensive evaluation of the solution set is adopted to determine the optimum solution for decision markers.

### 4.1 Performance evaluation index

As for the performances of the multi-objective optimization algorithms, inverted generational distance (IGD) and hypervolume (HV) are very popular for comprehensively measuring the convergence and diversity of algorithms [29]. Meanwhile, the proportion relation between total number of evaluations and total CPU time is used to test the algorithm efficiency [30]. The three metrics can be mathematically expressed as follows



IGD calculates an average minimum distance from each point in the true PF to those obtained by an MOEA. In the processing of calculating IGD, since without any priori PF shape knowledge, all the non-dominated solutions are used as the reference points [31].

$$IGD(S, P^*) = \frac{\sum_{x \in P^*} \text{dist}(x, S)}{|P^*|} \quad (19)$$

where  $P^*$  indicates a set of points uniformly sampled over the true PF, and  $S$  is the set of solutions obtained by an MOEA.  $\text{dist}(x, S)$  denotes the Euclidean distance between the closest individual from  $x$  to  $S$ .  $|P^*|$  is the cardinality of set  $P^*$ . The smaller IGD value indicates that the set  $S$  is closer to the entire PF, and thereby the convergence and diversity are better.

$$HV(S) = \text{VOL}(\bigcup_{x \in S} [f_1, r_1^*] \times [f_2, r_2^*] \times \dots \times [f_m, r_m^*]) \quad (20)$$

where  $\text{VOL}(\bullet)$  denotes the Lebesgue measure.  $\mathbf{r}^* = (r_1^*, r_2^*, \dots, r_m^*)$  is a reference point in the objective space that is dominated by all solutions in a PF approximation  $S$ . HV metric measures the size of the objective space dominated by the solutions in  $S$  and bounded by  $\mathbf{r}^*$ . The larger the HV value, the closer  $S$  is to the entire PF.

$$OO = \frac{T_{\text{Total}} - T_{\text{PFP}}}{T_{\text{PFP}}} \quad (21)$$

where  $OO$  stands for the optimizer overhead.  $T_{\text{Total}}$  denotes the total CPU time taken and  $T_{\text{PFP}}$  denotes the time taken for pure function evaluations. The lower  $OO$  metric corresponds to the higher efficiency of the algorithm.

#### 4.2 Fitness evaluation

Fuzzy comprehensive evaluation is one of the effective decision-making methods for objectives affected by various factors, which adopts a fuzzy membership function to describe the fitness factor of an objective function [11, 32]. For the objective function minimization problem, the fuzzy membership function can be expressed as

$$\eta_i(j) = (f_{i\max} - f_i(j)) / (f_{i\max} - f_{i\min}) \quad (22)$$

where  $f_i(j)$  represents the objective function,  $i$  denotes the number of the objective function, and  $j$  denotes the  $j$ -th solution at the PF.  $f_{i\max}$  and  $f_{i\min}$  are the maximum and minimum value of the objective function.

According to Ref. [11], the synthetical membership value can be written as

$$\eta_{\text{syn}} = (\eta_1 + \eta_2) / \max(\eta_1 + \eta_2) \quad (23)$$

The larger synthetical membership value indicates the better fitness of the Pareto solution. The highest synthetical membership value is 1, which can be considered as the most satisfactory solution for the decision-maker.

### 5 Numerical example

The purpose of conducting simulation is to verify the search capability and convergence speed of the proposed INSGA-II as well as validity and competency of the

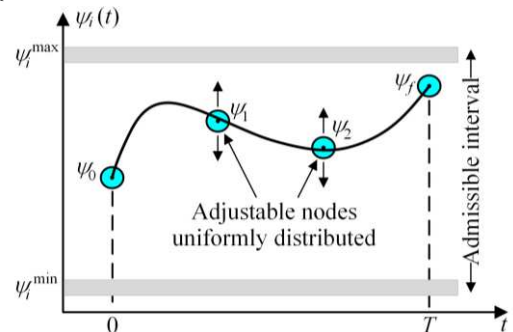
composite polynomial approach for creating trajectory. In this section, taking a serial-parallel hybrid manipulator as instance [33], we start by analyzing the performance of the proposed INSGA-II, MO-INSGA-II [34], success history-based adaptive multi-objective differential evolution with whale optimization (SHAMODE-WO) [35], IMOPSO [36], many-objective evolutionary algorithm based on decomposition with random and adaptive weights (MOEA/D-URAW) [37] and IMODE [38] for the trajectory optimization of a point-point motion. In a second phase, we compared the composite polynomial with the quintic B-splines approach to evaluate its effectiveness.

#### 5.1 Comparison with MO-NSGA-II, SHAMODE-WO, IMOPSO, MOEA/D-URAW and IMODE

Given position and posture of the hand at initial and final instance, including one starting point and eight ending points, the OAJ trajectories are parameterized here by composite polynomial functions with four nodes uniformly distributed along time scale (Fig. 4). There are sixteen unknown parameters, where two intermediate adjustable nodes contain fourteen, ending point only includes one by applying inverse kinematics transformation and the travelling time is one of them. These parameters are optimized by multi-objective algorithm until non-dominated solutions satisfying constraints have been reached. In Fig. 4, the boundary value of the designed parameter  $\psi$  can be obtained by considering the equality and inequality constraints in Eq. (2) and the workspace limitation of the manipulator.

The initial position and posture of the end-effector is  $[x_0 \ y_0 \ z_0 \ \alpha_0 \ \beta_0 \ \gamma_0] = [0 \ 0 \ -0.83 \ \pi/2 \ 0 \ -\pi]$ , and the final positions and postures of the hand (FPH) are set as follow.

$$\begin{cases} \text{FPH}_1 = [-0.02, 0.02, -0.77, 5\pi/12, -\pi/12, -8\pi/9] \\ \text{FPH}_2 = [-0.03, 0.03, -0.75, 5\pi/13, -\pi/13, -8\pi/10] \\ \text{FPH}_3 = [-0.04, 0.04, -0.73, 5\pi/14, -\pi/14, -8\pi/11] \\ \text{FPH}_4 = [-0.05, 0.05, -0.71, 5\pi/15, -\pi/15, -8\pi/12] \\ \text{FPH}_5 = [-0.06, 0.06, -0.69, 5\pi/16, -\pi/16, -8\pi/13] \\ \text{FPH}_6 = [-0.07, 0.07, -0.67, 5\pi/17, -\pi/17, -8\pi/14] \\ \text{FPH}_7 = [-0.08, 0.08, -0.65, 5\pi/18, -\pi/18, -8\pi/15] \\ \text{FPH}_8 = [-0.09, 0.09, -0.63, 5\pi/19, -\pi/19, -8\pi/16] \end{cases} \quad (24)$$



**Figure 4** Composite polynomial curve of joint output angle temporal evolution

To validate the superiority of the proposed INSGA-II, we

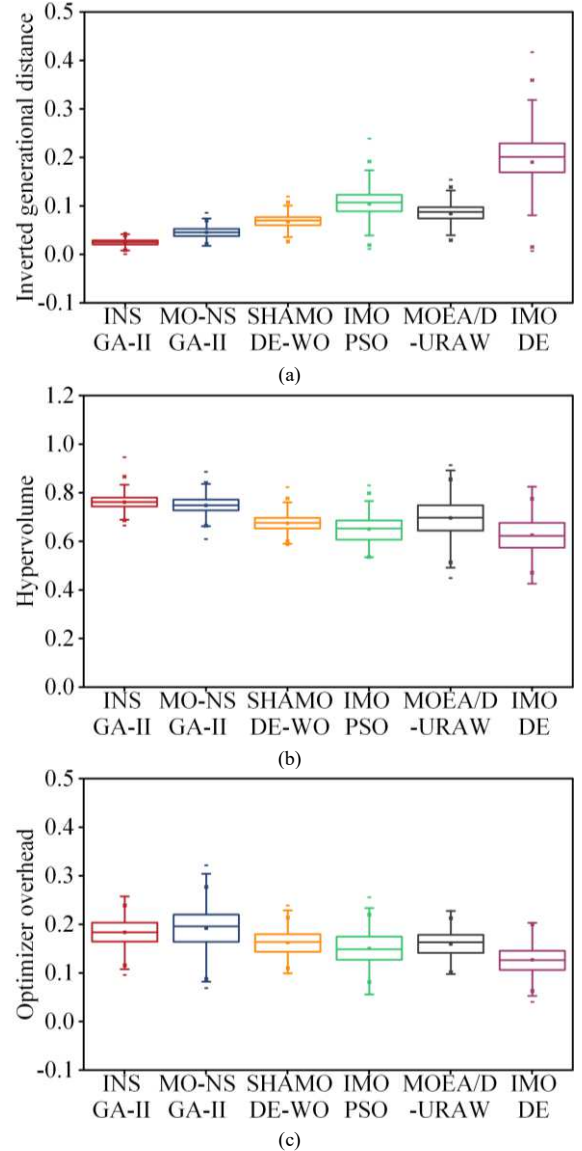


compare its performances with some state-of-the-art representatives from different categories of multi-objective algorithms. All non-dominated solutions of the trajectory optimizations, from the starting point to the eight ending points, offered by INSGA-II over 100 runs are compared to that of MO-NSGA-II, SHAMODE-WO, IMOPSO, MOEA/D-URAW and IMODE in terms of IGD, HV and OO, and then these experiment results are gathered for statistical analysis. It is noted that all objective functions are normalized by adopting the min-max standardization method to have a same range, which can avoid the function with largest range would dominate selection.

The initialization parameters for NSGA-II are as follows: the population size = 100, the generation number = 80. Mutation probability =  $1/16$ , which is selected as  $1/n$  (where  $n$  represents the number of variables) proposed by Deb [25]. For constraint-optimization problems, the distribution indexes for real coded crossover and mutation operators are 20 and 100, respectively. The values of the parameter that have been used in SHAMODE-WO technique are as follow: the population size = 100, the generation number = 80, the historical memory of scaling factor = 0.5, the historical memory of crossover ratio = 0.5, the memory index = 1 and the memory size = 5. In overall tested experiments, IMOPSO was run using the parameters as follows: the population size = 100, the generation number = 80, the jump improved operation mechanism number = 100, the disturbance rate range =  $[0.1, 0.3]$ . For MOEA/D-URAW, the population size = 100, the generation number = 80, the historical memory of scaling factor = 0.5, the historical memory of crossover ratio = 0.5, the memory index = 1 and the memory size = 5. For IMODE, the population size = 100, the maximum number of iterations = 80, the crossover probability = 0.1, the scaling factor = 0.5, the size of initial Pareto front approximation = 100, the number of points desired by the decision maker = 100 and the selection parameter = 0.1.

The mean and standard deviation values (SD) of all the instances, as shown in Table 1 (The best results for each index are marked in bold). Demonstrated in mean and SD of the IGD evaluation results, INSGA-II finds better solutions, which has superior values in all test problems when compared to related works while its IGD remains approximate to zero. In most of the test problems, the INSGA-II performs better than other related methods in the HV evaluation results. The two evaluation results indicate that the convergence and diversity of the non-dominated solutions obtained by INSGA-II performs better than that of other related methods. However, the experiment results of the INSGA-II are not satisfactory in the OO evaluation. As for the OO evaluation results, the calculation efficiency of all the test instances addressed by IMODE is the best.

In order to intuitively reflect the performance of each algorithm, the corresponding boxplots (Fig. 5) is drawn by synthesizing the results in Table 1. The IGD and HV evaluation results clearly show that the INSGA-II can perform exceptionally in solving the problem of manipulator's trajectory planning, which demonstrates its convergence and diversity are better than other algorithms. MO-NSGA-II takes second place in the convergence and diversity performances, but the computational efficiency of the INSGA-II and MO-NSGA-II are undesirable. Additionally, although the convergence and diversity performance of IMODE technique is not good, it gives minimum OO thereby it is the better one for a multicriterion to obtain a best optimal solution trade-off very quickly.



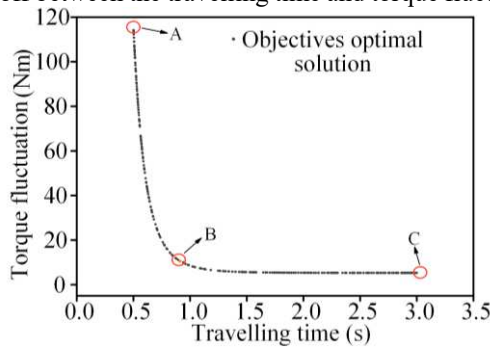
**Figure 5** Performances of INSGA-II, MO-NSGA-II, SHAMODE-WO, IMOPSO, MOEA/D-URAW and IMODE in optimizing the problem of HRRR trajectory planning. (a) IGD. (b) HV. (c) OO

**Table 1** Mean and standard deviation values of GD, SSM and OO between INSGA-II, MO-NSGA-II, SHAMODE-WO, IMOPSO, MOEA/D-URAW and IMODE over 100 runs on eight given different final points (unit: e-2)

Final point	Performance metric	INSGA-II		MO-NSGA-II		SHAMODE-WO		IMOPSO		MOEA/D-URAW		IMODE	
		Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
FPH <sub>1</sub>	IGD	<b>2.15</b>	<b>0.698</b>	4.02	1.07	6.53	1.62	10.4	3.04	8.25	2.16	20.1	8.27
	HV	<b>76.4</b>	3.23	73.3	<b>3.19</b>	67.1	<b>3.59</b>	64.9	5.20	61.7	6.37	60.1	7.62
	OO	17.0	2.93	20.9	4.61	16.1	<b>2.18</b>	16.0	2.84	15.9	2.48	<b>10.9</b>	3.22
FPH <sub>2</sub>	IGD	<b>2.24</b>	<b>0.636</b>	4.17	0.997	6.79	1.82	11.8	4.53	8.31	2.20	19.8	7.92
	HV	75.2	<b>3.05</b>	<b>76.7</b>	3.29	67.3	3.50	64.9	5.19	63.9	5.97	60.7	7.02
	OO	17.4	2.58	20.4	4.37	15.4	<b>2.12</b>	11.9	2.83	15.9	2.48	<b>11.4</b>	3.01
FPH <sub>3</sub>	IGD	<b>2.34</b>	<b>0.587</b>	4.32	0.955	6.77	1.78	11.3	4.01	8.34	2.23	19.5	7.58
	HV	<b>78.1</b>	<b>2.92</b>	76.1	3.28	67.3	3.45	64.9	5.19	66.1	5.63	61.4	6.57
	OO	17.8	2.34	19.9	4.13	15.7	<b>2.17</b>	12.9	2.53	15.9	2.50	<b>11.9</b>	2.83
FPH <sub>4</sub>	IGD	<b>2.43</b>	<b>0.554</b>	4.47	0.946	6.76	1.73	10.9	3.51	8.38	2.26	19.2	7.24
	HV	<b>76.9</b>	<b>2.82</b>	75.4	3.32	67.5	3.45	64.9	5.18	68.3	5.38	62.1	6.30
	OO	18.2	<b>2.23</b>	19.4	3.90	15.9	2.25	13.9	2.36	15.9	2.53	<b>12.4</b>	2.68
FPH <sub>5</sub>	IGD	2.52	0.540	4.62	0.971	6.74	1.69	10.4	3.04	8.42	2.29	18.9	6.90
	HV	<b>75.8</b>	<b>2.77</b>	74.8	3.39	67.3	3.51	64.9	5.11	70.6	5.21	62.8	6.23
	OO	18.5	<b>2.28</b>	18.9	3.69	16.3	2.35	14.9	2.35	15.9	2.57	<b>12.9</b>	2.57
FPH <sub>6</sub>	IGD	<b>2.62</b>	<b>0.546</b>	4.76	1.03	6.73	1.65	9.93	2.60	8.45	2.33	18.6	6.57
	HV	<b>74.6</b>	<b>2.77</b>	74.2	3.49	67.3	3.63	64.9	5.13	72.8	5.14	63.5	6.39
	OO	18.9	2.48	18.5	3.49	16.5	<b>2.47</b>	15.9	2.49	15.9	2.60	<b>13.5</b>	2.51
FPH <sub>7</sub>	IGD	<b>2.71</b>	<b>0.573</b>	4.91	1.11	6.72	1.61	9.46	2.22	8.49	2.36	18.26	6.24
	HV	73.4	<b>2.82</b>	<b>73.5</b>	3.63	67.3	3.79	64.9	5.15	75.0	5.18	64.1	6.73
	OO	19.3	2.78	17.9	3.31	16.8	2.61	16.9	2.78	16.0	2.65	<b>13.9</b>	<b>2.48</b>
FPH <sub>8</sub>	IGD	<b>2.80</b>	<b>0.62</b>	5.06	1.21	6.70	1.58	8.99	1.94	8.53	2.39	17.9	5.91
	HV	<b>77.3</b>	<b>2.91</b>	72.9	3.79	67.3	3.98	64.8	5.14	77.2	5.32	64.8	7.25
	OO	19.7	3.18	17.5	3.15	17.1	2.77	17.9	3.15	16.1	2.69	<b>14.5</b>	<b>2.51</b>

## 5.2 Comparison of trajectory planning methods

Taking the point-to-point mission with FPH1 as an example, to gain the better designed parameters of trajectory optimization, the non-dominated solutions obtained by the four algorithms are all taken as candidates for decision-makers. As shown in Fig. 6, the travelling time ranges from 0.513 s to 2.96 s while the torque fluctuation ranges from 5.29 N•m to 114.17 N•m. Solution A requires the shortest travelling time but the maximum torque fluctuation while Solution C has the least torque fluctuation but the maximum travelling time. Other solutions are the trade-off between the travelling time and torque fluctuation.

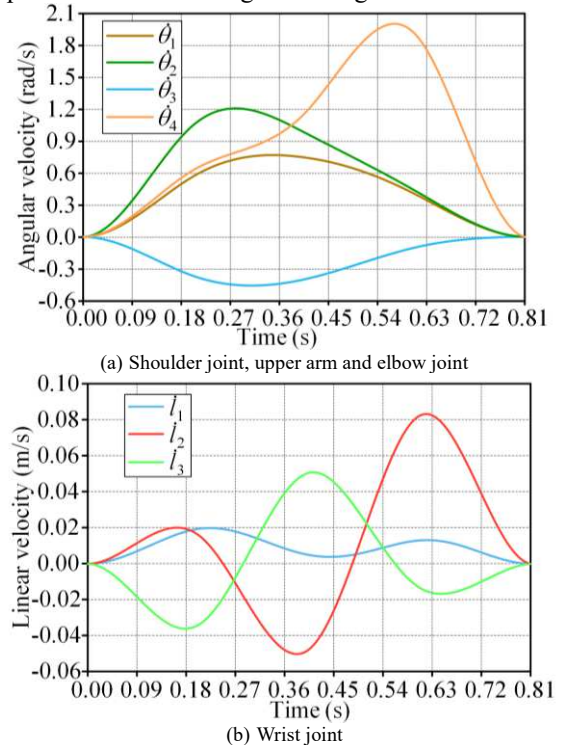


**Figure 6** Approximate true Pareto front for trajectory planning

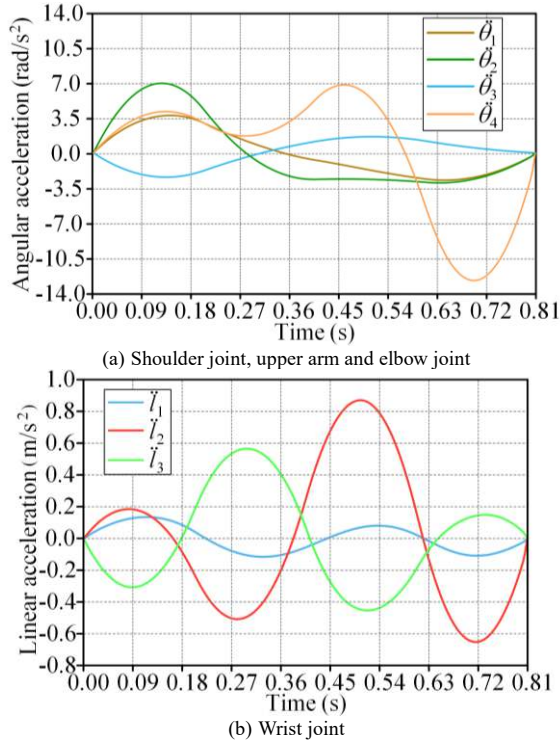
Solution B (The orange circle dot in Fig. 6) is obtained by substituting all non-dominated solutions obtained in the Section 5.2 into the Eq. (23), which corresponds to a synthetical membership value equal to one. The designed parameters matched with Solution B are used to verify the validity and competency of the composite polynomial in comparison with the velocity and acceleration of the quintic B-spline approach [21]. The result values of the designed parameter are: the normalized time  $\lambda = 0.809$ , the

intermediate points  $\psi_{a1} = [-0.0925, -0.0366, 0.194, 0.132, 0.233, -0.0836, -0.172]$  rad and  $\psi_{a2} = [-0.201, -0.0905, 0.351, 0.289, -0.257, -0.0866, 0.234]$  rad.

In order to be able to compare the results yielded by the two trajectory planning methods, the travelling time is consistent with the optimized results. In addition, the velocity and acceleration of the starting and ending points are both set as zero. The profiles of velocity and acceleration for each actuator created using quintic B-splines are shown in Fig. 7 and Fig. 8.

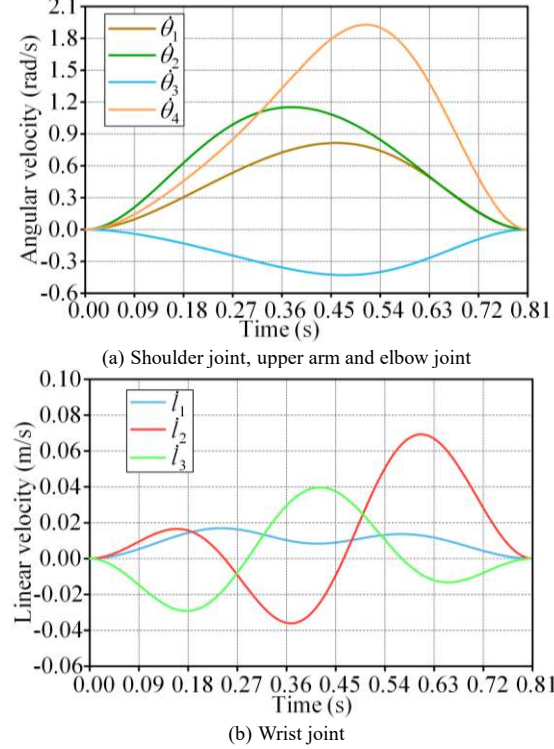


**Figure 7** The velocity profile of each joint using quintic B-splines

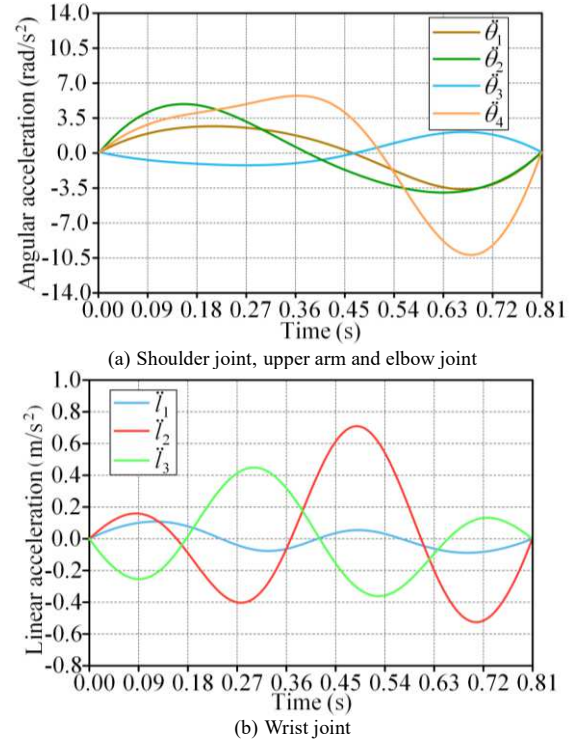


**Figure 8** The acceleration profile of each joint using quintic B-splines

Likewise, the trajectory of each OAJ created by using the composite polynomials can be obtained by substituting the optimum results into Eq. (8), and then the velocity and acceleration of each actuator can be solved. The corresponding velocity and acceleration profiles are shown in Fig. 9 and Fig. 10.



**Figure 9** The velocity profile of each joint using composite polynomials



**Figure 10** The acceleration profile of each joint using composite polynomials

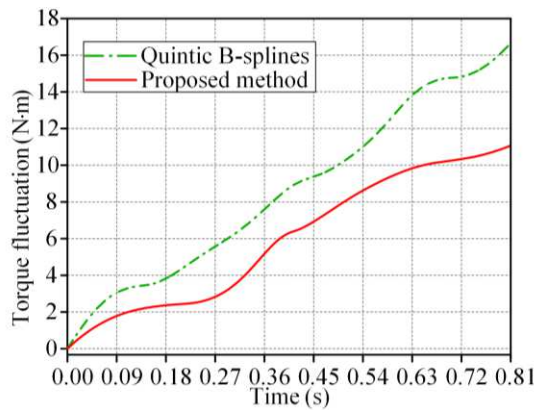
The maximum kinematic values of the profiles in Figs. 10-13 are listed in Table 2, and it can be noticed that the results yielded by the approach described in this article are better than those provided by the approach [21] with respect to the maximum values of velocity and acceleration. It is well known that the lower maximum velocity provides an advantage because lower velocity extends the life of the actuator. Meanwhile, lower acceleration profiles decreases the noise in the mechanism and increases the mechanical life by reducing wear.

**Table 2** Maximum kinematic values resulting from the trajectory construction approaches

Approach	Joint	1	2	3	4	5	6	7
Quintic B-splines								
$V_{\max}$		0.768	0.458	1.206	2.001	0.0201	0.0839	0.0513
$A_{\max}$		3.757	1.627	6.976	6.808	0.136	0.870	0.565
Composite polynomials								
$V_{\max}$		0.815	0.429	1.152	1.929	0.0169	0.0693	0.0397
$A_{\max}$		2.602	2.029	4.829	5.659	0.108	0.710	0.449

To further demonstrate the superiorities of the proposed method, the torque fluctuations are calculated by the quintic B-spline and composite polynomial approach. The reduction of the torque fluctuation is 33.47%, as shown in Fig. 11. Consequently, the manipulator can work with higher stability via the proposed method.





**Figure 11** Comparison chart of torque fluctuation

## 6 Conclusion

A new methodology for optimal trajectory planning has been described in this article. The methodology adopts the composite polynomials to construct the trajectory of each OAJ and optimizes the trajectory with INSGA-II technique. The objective functions take into account both the travelling time and the torque fluctuation along the whole trajectory.

(1) A new composite polynomial is created by combining quintic polynomials with Bezier curves based on cubic Bernstein polynomials. The Bezier curve part improves the convergence to the starting and ending points and the djustability of the trajectory, while the polynomial part provides a smooth transition in the vicinity of the endpoints.

(2) INSGA-II adopts three improved genetic operators: RGS can increase the differences between the paired individuals and the diversity of the paired genes; DBX can expand the search space and improve the probability of individuals with high adaptability; APCM can accelerate the convergence to the optimal solution by the adaptive mutation operator.

(3) Given eight different ending points in trajectory mission, the convergence, diversity and efficiency of INSGA-II, MO-NSGA-II, SHAMODE-WO, IMOPSO, MOEA/D-URAW and IMODE are calculated based on IGD, HV and OO. The simulation results demonstrated that well-converged and well-diversified non-dominated solutions can be obtained by INSGA-II, but the efficiency is lower than that of IMODE.

(4) Using the synthetical fuzzy membership function to obtain a trade-off for decision-users, the trajectory of the OAJ constructed by composite polynomials compared in the velocity and acceleration with quintic B-splines. The former velocity and acceleration are lower, which increases the mechanical life by reducing wear. Moreover, the reduction of the torque fluctuation is 33.47%, thereby ensuring higher motion stability of the manipulator.

Future work will aim to reduce the time complexity of the INSGA-II to improve the calculation efficiency so that the optimization method can be used in the real-time trajectory

planning for the manipulator.

## Authors' contributions

The author' contributions are as follows: Z-SW conceived the basic idea, and carried out research, analysis and writing of the manuscript. Y-BL provided theoretical guidance. KS assisted with formula analyses. W-TZ was in charge of drawing Figures. BC and KC revised the final manuscript. All authors read and approved the final manuscript.

## Author Details

<sup>1</sup> College of Mechanical Engineering, Zhejiang University of Technology, Hangzhou 310023, China

<sup>2</sup> Key Laboratory of E & M, Ministry of Education & Zhejiang Province, Zhejiang University of Technology, Hangzhou 310023, China.

## Authors' Information

Yanbiao Li, born in 1978, is currently a professor and a PhD candidate supervisor at *Key Laboratory of E & M, Ministry of Education & Zhejiang Province, Zhejiang University of Technology, China*. He received his PhD degree from *Yanshan University, China*, in 2008. His research interests include parallel mechanism, robotics. E-mail: lybrory@zjut.edu.cn

Zesheng Wang, born in 1994, is studying his PhD degree in *Zhejiang University of Technology, China*. His main research interests include parallel mechanism, hybrid mechanism, and robotics.

## Competing interests

The authors declare no competing financial interests.

## Acknowledgements

## Funding

The authors gratefully acknowledge the support of the Zhejiang Provincial Natural Science Foundation for Distinguished Young Scientists [grant number LR18E050003]; the National Natural Science Foundation of China (NSFC) [grant number 51975523 and 51905481]; the Students in Zhejiang Province Science and technology Innovation Plan (Xinmiao Talents Program) [grant number 2020R403054]; and the China Postdoctoral Science Foundation [grant number 2020M671784].

## References

- [1] A Gasparetto, P Boscariol, A Lanzutti, et al. Path planning and trajectory planning algorithms: A general overview. *Motion and operation planning of robotic systems*. Springer, Cham, 2015: 3-27.
- [2] H Wang, H Wang, J Huang, et al. Smooth point-to-point trajectory planning for industrial robots with kinematical constraints based on high-order polynomial curve. *Mechanism and Machine Theory*, 2019, 139: 284-293.
- [3] H Liu, X Lai, W Wu. Time-optimal and jerk-continuous trajectory planning for robot manipulators with kinematic constraints. *Robotics and Computer-Integrated Manufacturing*, 2013: 29 (2): 309-317.
- [4] X Shi, H Fang, L Guo. Multi-objective optimal trajectory planning of manipulators based on quintic NURBS. *2016 IEEE International*

- Conference on Mechatronics and Automation*. IEEE, 2016: 759-765.
- [5] Ü Dinçer, M Çevik. Improved trajectory planning of an industrial parallel mechanism by a composite polynomial consisting of Bezier curves and cubic polynomials. *Mechanism and Machine Theory*, 2019, 132: 248-263.
  - [6] A Gasparetto, V Zanutto. A new method for smooth trajectory planning of robot manipulators. *Mechanism and Machine Theory*, 2007, 42 (4): 455-471.
  - [7] T G Thuruthel, E Falotico, F Renda, et al. Model-based reinforcement learning for closed-loop dynamic control of soft robotic manipulators. *IEEE Transactions on Robotics*, 2018, 35 (1): 124-134.
  - [8] B Chen, D R. Gao, Y B Li, et al. Investigation of the droplet characteristics and size distribution during the collaborative atomization process of a twin-fluid nozzle. *The International Journal of Advanced Manufacturing Technology*, 2020, 107 (3-4): 1625-1639.
  - [9] L F Tian, C Curtis. An effective robot trajectory planning method using a genetic algorithm. *Mechatronics*, 2004, 14 (5): 455-470.
  - [10] H I Lin. A fast and unified method to find a minimum-jerk robot joint trajectory using particle swarm optimization. *Journal of Intelligent & Robotic Systems*, 2014, 75 (3-4): 379-392.
  - [11] J S Huang, P F Hu, K Y Zeng, et al. Optimal time-jerk trajectory planning for industrial robots. *Mechanism and Machine Theory*, 2018, 121: 530-544.
  - [12] P Huang, G Liu, J Yuan, et al. Multi-objective optimal trajectory planning of space robot using particle swarm optimization. *International Symposium on Neural Networks*. Springer, Berlin, Heidelberg, 2008: 171-179.
  - [13] M da Graça Marcos, J T Machado, T P Azevedo-Perdicoulis. A multi-objective approach for the motion planning of redundant manipulators. *Applied Soft Computing*, 2012, 12 (2): 589-599.
  - [14] R Saravanan, S Ramabalan, C Balamurugan, et al. Evolutionary trajectory planning for an industrial robot. *International Journal of Automation and computing*, 2010, 7 (2): 190-198.
  - [15] D Chen, S Li, J Wang, et al. A multi-objective trajectory planning method based on the improved immune clonal selection algorithm. *Robotics and Computer-Integrated Manufacturing*, 2019, 59: 431-442.
  - [16] S F Saramago, V S Junior. Optimal trajectory planning of robot manipulators in the presence of moving obstacles. *Mechanism and Machine Theory*, 2000, 35 (8): 1079-1094.
  - [17] M Benzaoui, H Chekireb, M Tadjine, et al. Trajectory tracking with obstacle avoidance of redundant manipulator based on fuzzy inference systems. *Neurocomputing*, 2016, 196: 23-30.
  - [18] J Garrido, W Yu, X O Li. Robot trajectory generation using modified hidden Markov model and Lloyd's algorithm in joint space. *Engineering Applications of Artificial Intelligence*, 2016, 53: 32-40.
  - [19] A Reiter, A Müller, H Gattringer. On higher order inverse kinematics methods in time-optimal trajectory planning for kinematically redundant manipulators. *IEEE Transactions on Industrial Informatics*, 2018, 14 (4): 1681-1690.
  - [20] Y Fang, J Hu, W Liu, et al. Smooth and time-optimal S-curve trajectory planning for automated robots and machines. *Mechanism and Machine Theory*, 2019, 137: 127-153.
  - [21] Y B Li, L Wang, B Chen, et al. Optimization of dynamic load distribution of a serial-parallel hybrid humanoid arm. *Mechanism and Machine Theory*, 2020, 149, 103792.
  - [22] Y B Li, Z S Wang, P Sun, et al. Dynamic Load Distribution Optimization for a 4-DOF Redundant and Series-parallel Hybrid Humanoid Arm. *Journal of Mechanical Engineering*, 2020, 56(09):45-54. (in Chinese)
  - [23] Z Wang, Z Wang, W Liu, et al. A study on workspace, boundary workspace analysis and workpiece positioning for parallel machine tools. *Mechanism and Machine Theory*, 2001, 36 (5): 605-622.
  - [24] N Srinivasan, K Deb. Multi-objective function optimisation using non-dominated sorting genetic algorithm. *Evolutionary Computation*, 1994, 2(3): 221-248.
  - [25] K Deb, A Pratap, S Agarwal, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, 2002, 6 (2): 182-197.
  - [26] K Deb, H Jain. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. *IEEE transactions on evolutionary computation*, 2013, 18 (4): 577-601.
  - [27] Y Song, F Wang, X Chen. An improved genetic algorithm for numerical function optimization. *Applied Intelligence*, 2019, 49 (5): 1880-1902.
  - [28] K Zhang, Z W Xu, S L Xie, et al. Evolution Strategy-Based Many-Objective Evolutionary Algorithm Through Vector Equilibrium. *IEEE Transactions on Cybernetics*, 2020.
  - [29] X Cai, Y Xiao, M Li, et al. A grid-based inverted generational distance for multi/many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 2020.
  - [30] L H Wu, Y N Wang, X F Yuan, et al. Environmental/economic power dispatch problem using multi-objective differential evolution algorithm. *Electric Power Systems Research*, 2010, 80 (9): 1171-1181.
  - [31] M Asafuddoula, T Ray, R Sarker. A decomposition-based evolutionary algorithm for many objective optimization. *IEEE Transactions on Evolutionary Computation*, 2015, 19 (3): 445-460.
  - [32] L H Wu, Y N Wang, X F Yuan, et al. Environmental/economic power dispatch problem using multi-objective differential evolution algorithm. *Electric Power Systems Research*, 2010, 80 (9): 1171-1181.
  - [33] Z S Wang, Y B Li, Y Q Luo, et al. Dynamic analysis of a 7-DOF redundant and hybrid mechanical arm. *Journal of Zhejiang University*, 2020, 54 (8): 1505-1515. (In Chinese)
  - [34] K Deb, H Jain. Handling many-objective problems using an improved NSGA-II procedure. *2012 IEEE Congress on Evolutionary Computation*. IEEE, 2012: 1-8.
  - [35] N Panagant, S Bureerat, K Tai. A novel self-adaptive hybrid multi-objective meta-heuristic for reliability design of trusses with simultaneous topology, shape and sizing optimisation design variables. *Structural and Multidisciplinary Optimization*, 2019, 60 (5): 1937-1955.
  - [36] S J Tsai, T Y Sun, C C Liu, et al. An improved multi-objective particle swarm optimizer for multi-objective problems. *Expert Systems with Applications*, 2010, 37(8): 5872-5886.
  - [37] L R Farias, A F Araújo. Many-Objective Evolutionary Algorithm Based On Decomposition With Random And Adaptive Weights. *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*. IEEE, 2019: 3746-3751.
  - [38] W Gong, Z Cai. An improved multiobjective differential evolution based on Pareto-adaptive  $\epsilon$ -dominance and orthogonal design. *European Journal of Operational Research*, 2009, 198 (2): 576-601.

# Figures

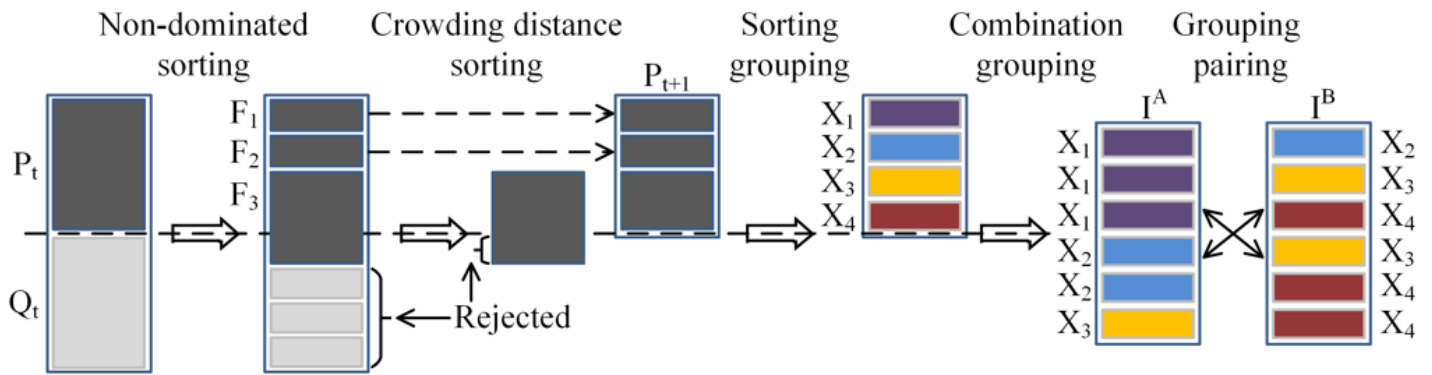


Figure 1

Schematic of the RGS operation

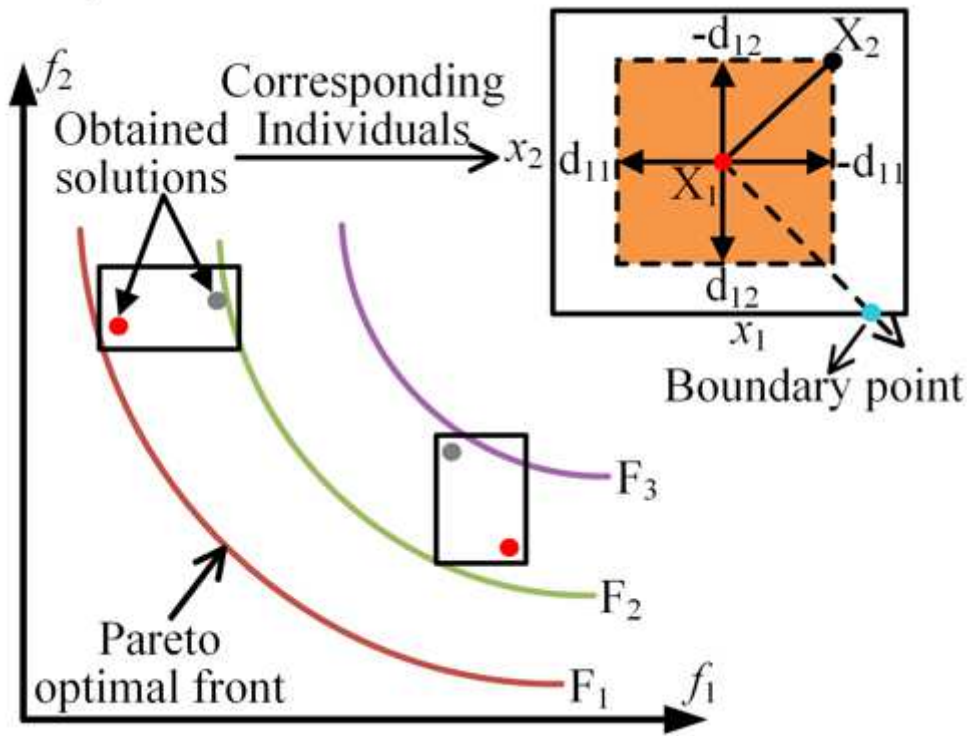
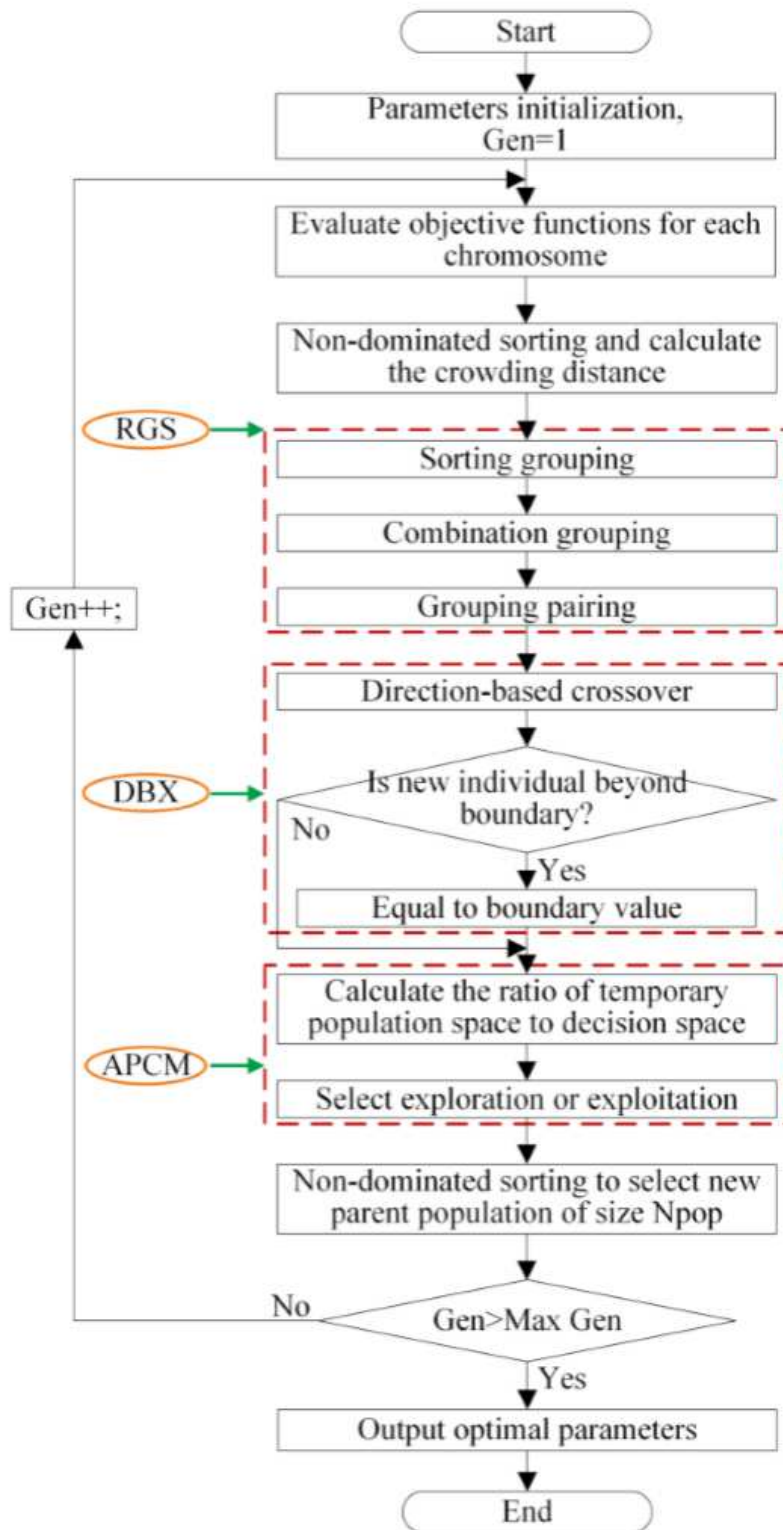


Figure 2

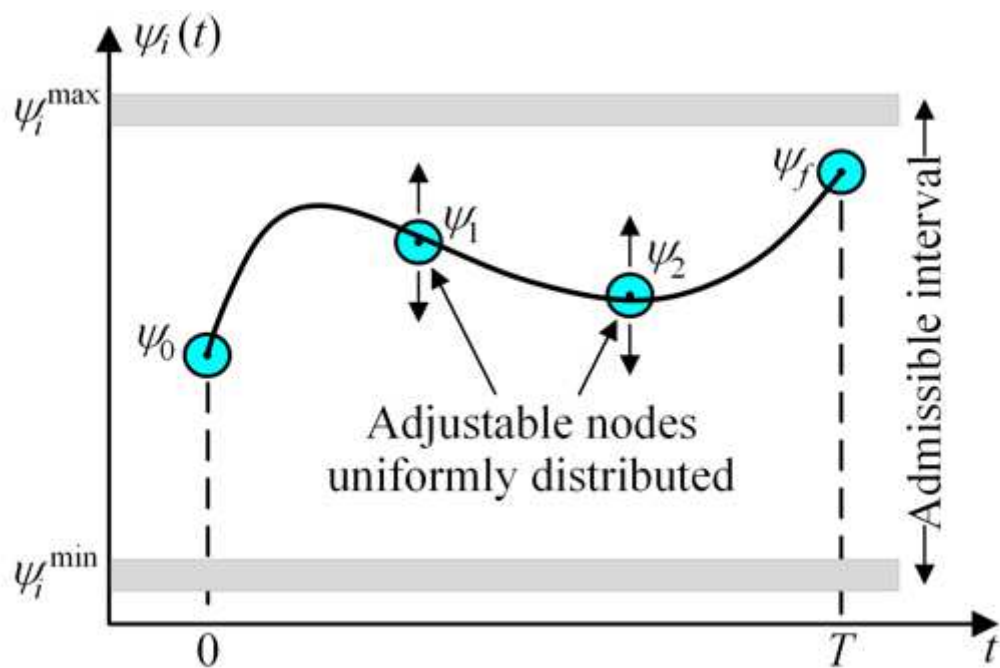
Schematic of the DBX operation





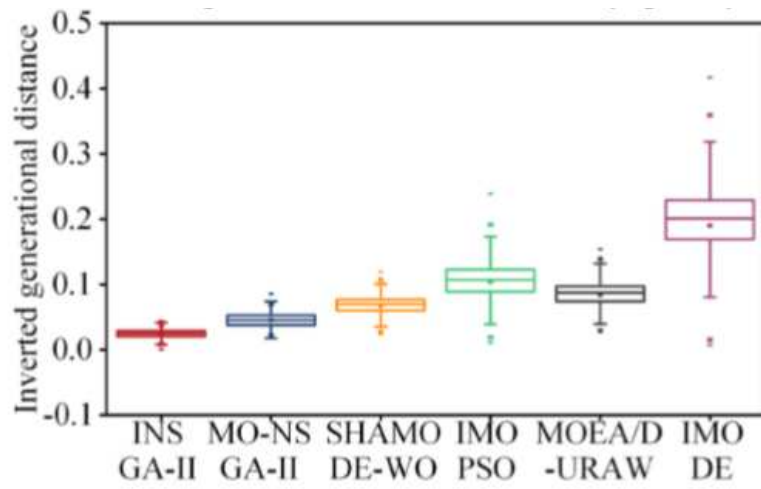
**Figure 3**

Flowchart of INSGA-II algorithm

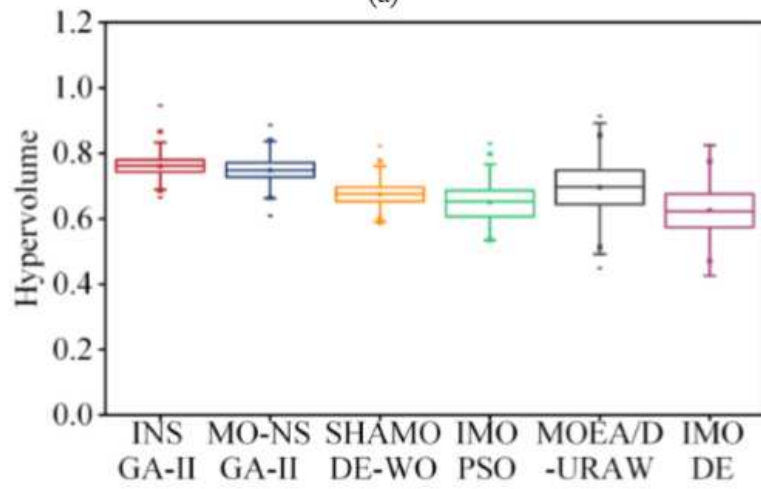


**Figure 4**

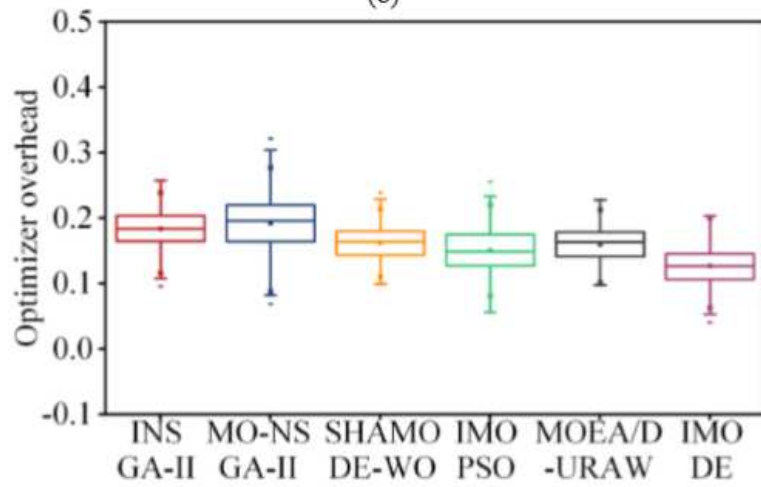
Composite polynomial curve of joint output angle temporal evolution



(a)



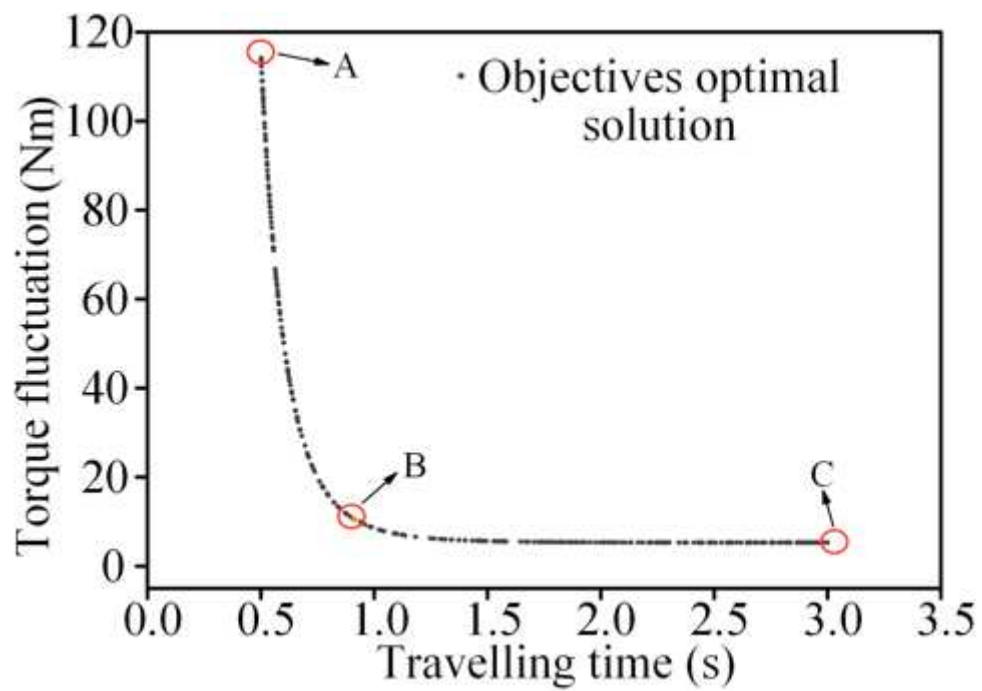
(b)



(c)

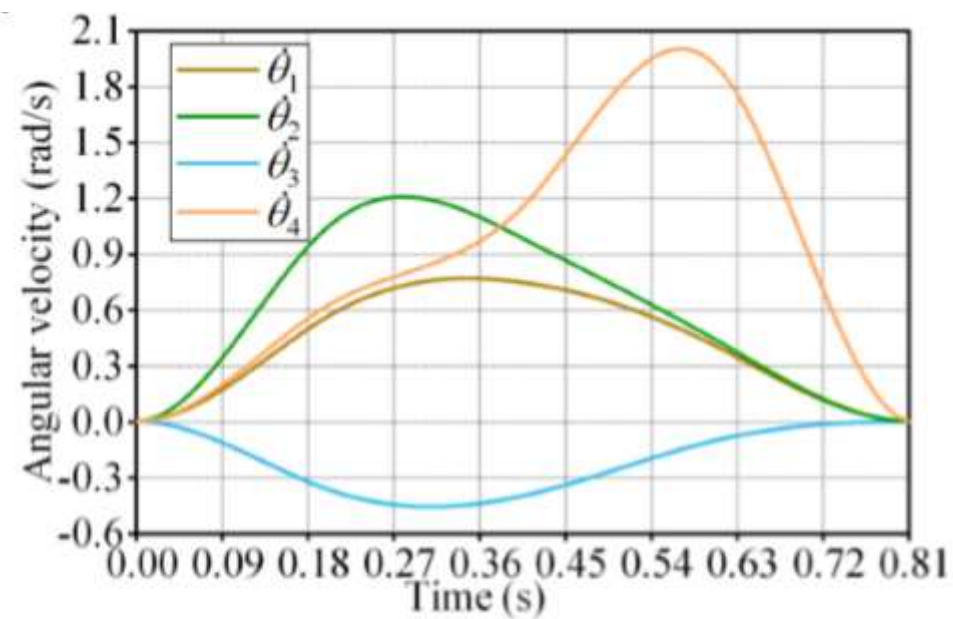
**Figure 5**

Performances of INSGA-II, MO-NSGA-II, SHAMODE-WO, IMOPSO, MOEA/D-URAW and IMODE in optimizing the problem of HRRR trajectory planning. (a) IGD. (b) HV. (c) OO

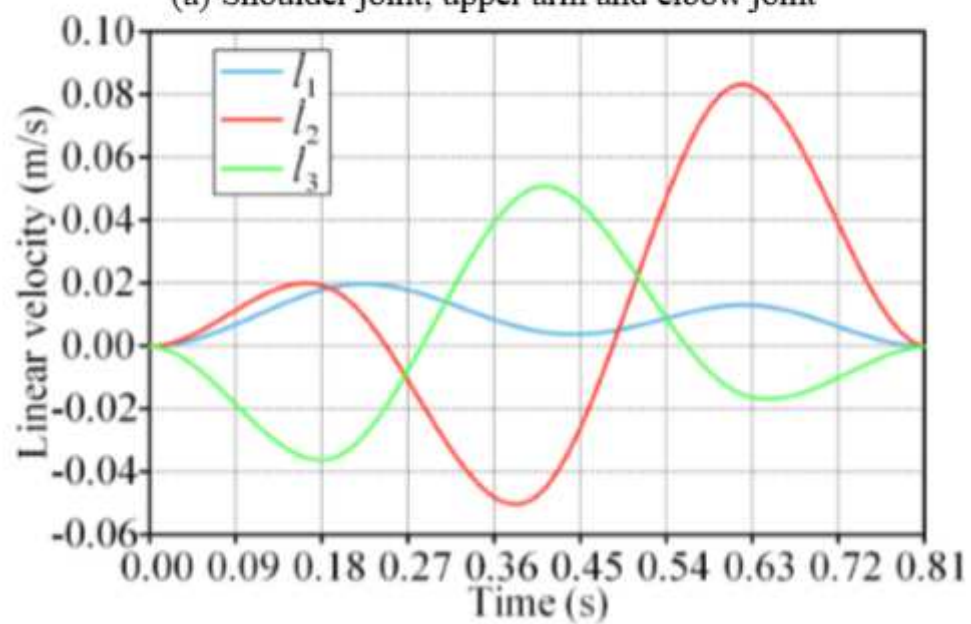


**Figure 6**

Approximate true Pareto front for trajectory planning



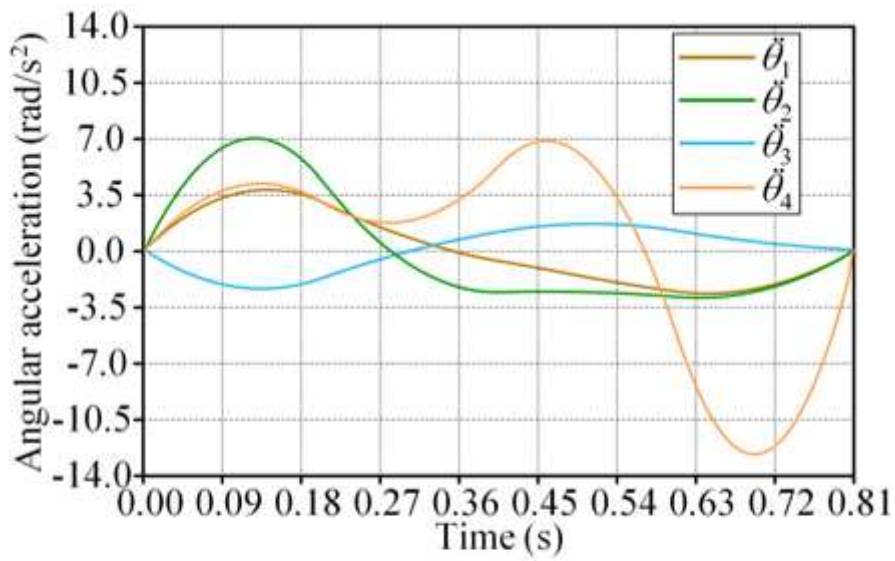
(a) Shoulder joint, upper arm and elbow joint



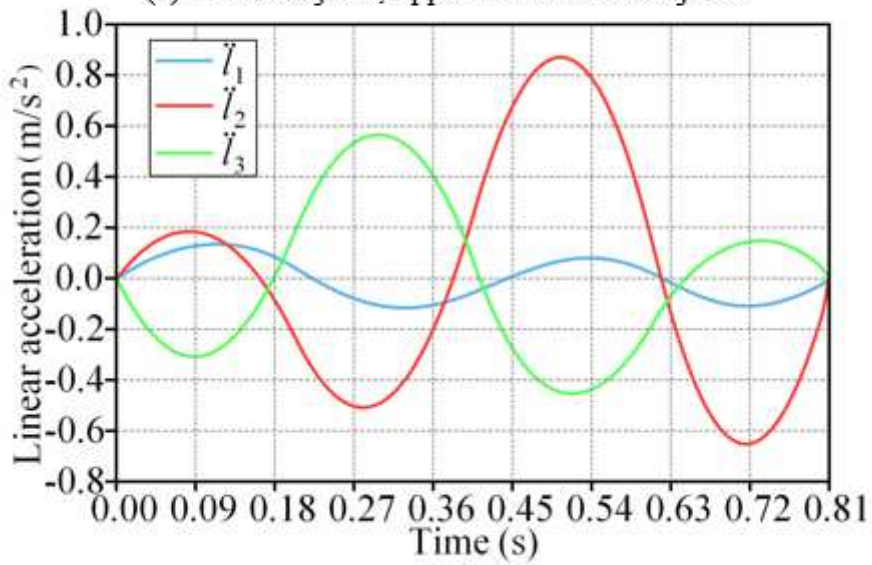
(b) Wrist joint

Figure 7

The velocity profile of each joint using quintic B-splines



(a) Shoulder joint, upper arm and elbow joint

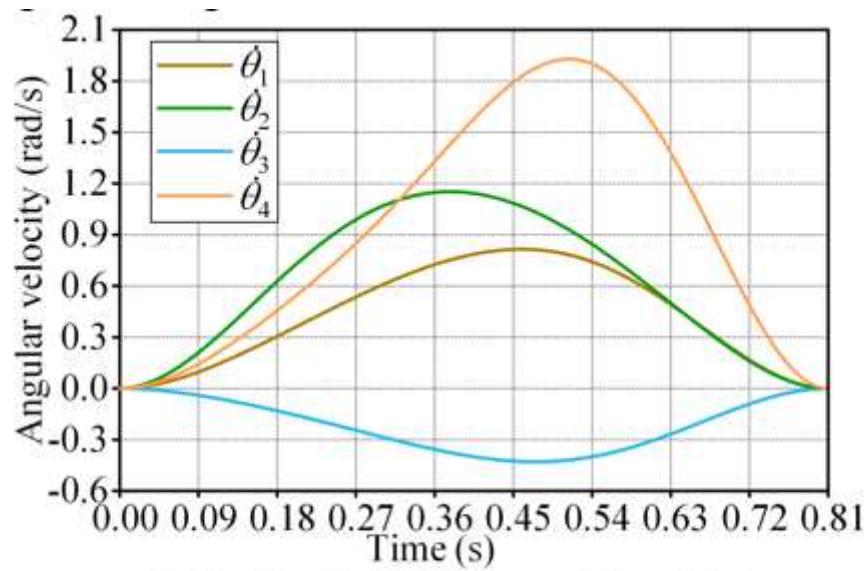


(b) Wrist joint

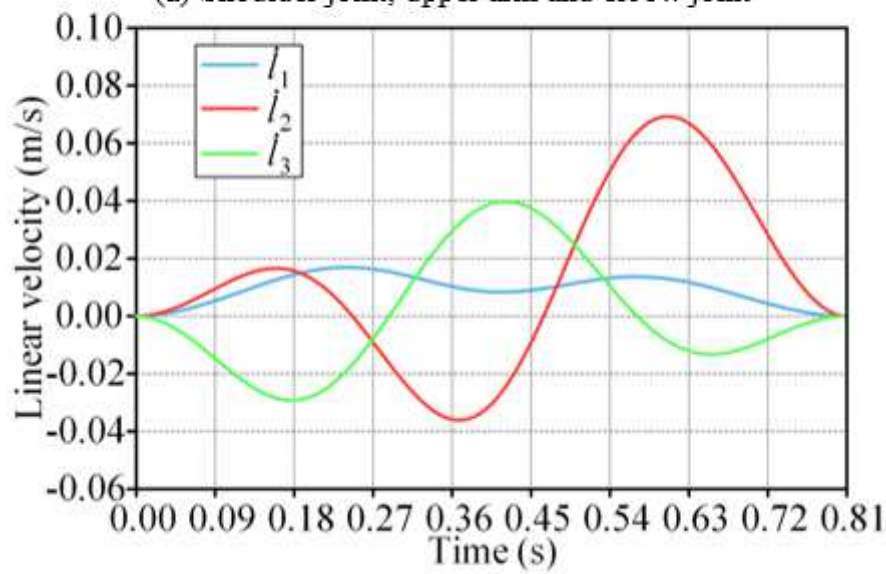
Figure 8

The acceleration profile of each joint using quintic B-splines





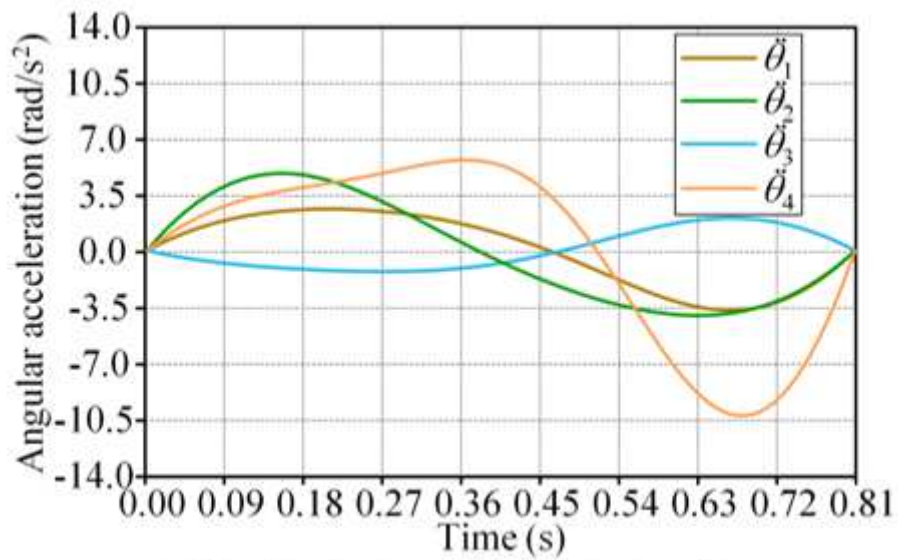
(a) Shoulder joint, upper arm and elbow joint



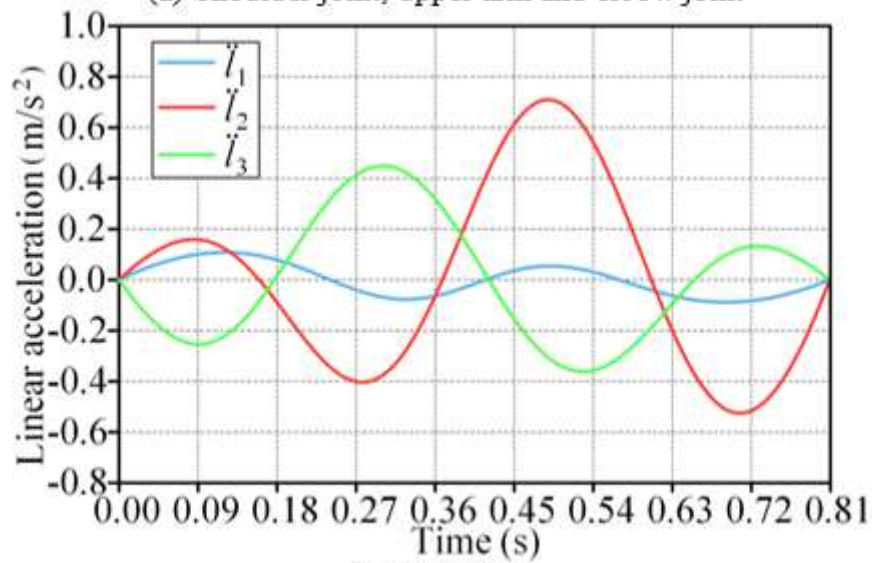
(b) Wrist joint

**Figure 9**

The velocity profile of each joint using composite polynomials



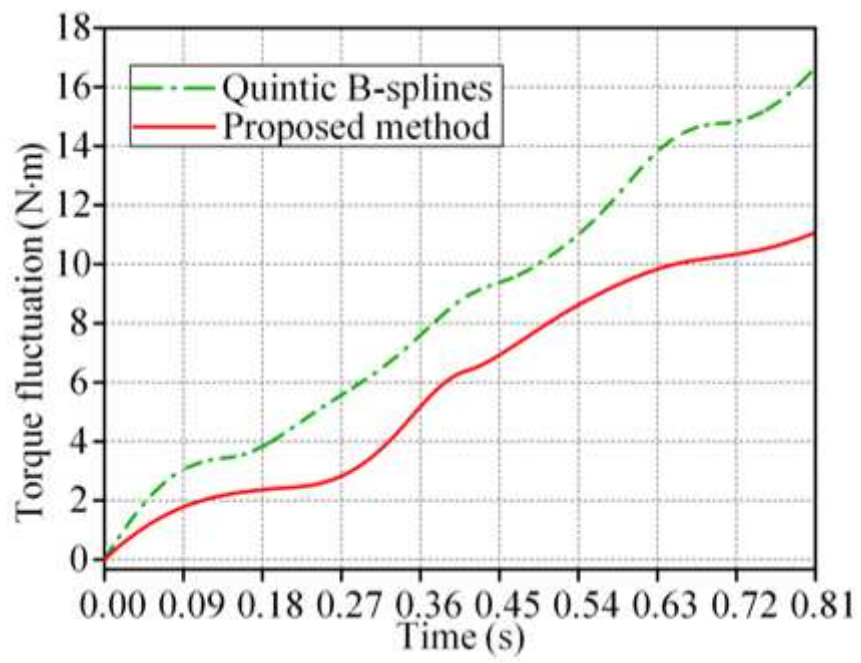
(a) Shoulder joint, upper arm and elbow joint



(b) Wrist joint

Figure 10

The acceleration profile of each joint using composite polynomials



**Figure 11**

Comparison chart of torque fluctuation