

Article

Not peer-reviewed version

Trajectory Smoothing Planning of Delta Parallel Robot Combining Cartesian and Joint Space

[Zhu Dachang](#)^{*}, [He Yonglong](#), Yu Xuezhe, [Li Fangyi](#)^{*}

Posted Date: 9 October 2023

doi: 10.20944/preprints202310.0285.v1

Keywords: Delta parallel robot; kinematic and dynamic modelling; trajectory smoothing planning; combining Cartesian space with joint space



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

Trajectory Smoothing Planning of Delta Parallel Robot Combining Cartesian and Joint Space

Zhu Dachang* , He Yonglong, Yu Xuezhe, Li Fangyi *

School of Mechanical and Electrical Engineering, Guangzhou University, Guangzhou 510006, China; heyonglong@gzhu.edu.cn (H.Y.); 2112307093@gzhu.edu.cn (Y.X.)

* Correspondence: zdc98998@gzhu.edu.cn (Z.D.); fangyi_li@gzhu.edu.cn (L.F.)

Abstract: Delta parallel robot has been widely used in precision parts processing, handling, sorting, and precision assembly, with high efficiency and motion stability being its important performance indexes. Any small line segment corner in trajectory planning causes tangential discontinuous trajectory abrupt change, and the process vibration and shock caused by it seriously affect its high-speed and high-precision performance. In this paper, a trajectory planning method combining Cartesian space and joint space is proposed. Firstly, the vector method and micro-element integration method are used to establish the complete kinematics and dynamics equations of the Delta parallel robot, and the inverse kinematics/dynamics model-solving program is written based on Matlab software. Secondly, the end-effector trajectory of the Delta parallel robot is planned in Cartesian space, and the data points and inverse control points of the end-effector trajectory are obtained by the normalization method. Finally, the data points and control points are mapped to the joint space through the inverse kinematics equation, and the fifth-order B-spline curve is adopted for quadratic trajectory planning, which realizes the high-order continuous smoothing of trajectory planning. The simulation and experiment results show that the trajectory smoothing performance of continuous high-order curvature change can be improved using the proposed method, the tracking error of the end-effector is reduced by 21.12%, and the peak torques change of three joints is reduced by 3.5%, 11.6% and 1.6%, respectively.

Keywords: delta parallel robot; kinematic and dynamic modelling; trajectory smoothing planning; combining Cartesian space with joint space

1. Introduction

Delta parallel robot has been widely used in precision parts processing, handling, sorting, and assembly because of its high speed and high precision [1]. When a straight line or an arc approaches the target curve, the acceleration curve of the trajectory of the end-effector is abrupt due to tangential discontinuity at the connecting corner of the fitted line segment, which makes the process vibration of the Delta parallel robot inevitable in high-speed operation and even causes production safety accidents [2]. Polynomial functions and interpolation methods are often used for robot trajectory planning and optimization by giving constraints on starting and ending points, including displacement, velocity, and acceleration. Boryga [3] presented a smooth trajectory planning method using higher-degree polynomial functions of acceleration to determine accurate dependencies derived for the calculation of polynomial coefficients and motion time in the case of velocity, acceleration, and jerk constraints, thus ensuring the continuity and smoothness of the trajectory. A seventh-order polynomial was applied to the trajectory planning to achieve the position, velocity, acceleration, and jerk of the actuating joints, and the positions/orientations of the parallel mechanism modules were interpolated to get a smooth trajectory without impact [4]. To achieve a smooth and precise trajectory while minimizing jerk, Damasevicius et al. [5] proposed natural cubic splines for finding a Pareto-optimal set of robot arm trajectories. When a robot is required to machine a complex curved workpiece with high precision and speed, the path is generally dispersed into a series of points and transmitted to the robot, which not only reduces precision but also causes damage to the motors and robot. Zhang et al. [6] inserted a quintic

Bezier curve between small adjacent line segments to achieve velocity and acceleration continuity at the connecting points, and a quartic polynomial was adopted to achieve a constant velocity at the velocity limitation. To solve the problem of trajectory continuity, some fitting curves were presented, such as quintic and cubic non-uniform rational NURBS curves [7–9], quintic Pythagorean-Hodograph (PH) curve [10], combining cubic splines with a linear segment with parabolic blends [11], S-curve [12]. However, since the trajectory of the end-effector in Cartesian space needs to be mapped to the joint space through the inverse kinematic equation, it is realized by the joint actuator through the torque control according to the calculated joint variable values. Therefore, the end-effector position, joint variables, and the precise modeling of robot kinematics and dynamics equations impact trajectory smoothing planning and become a very challenging subject in robotics [13,14].

Should surfaces such as sharp corners, holes, or protrusions be considered in trajectory planning [15]? Considering the execution characteristics of joint actuators, most trajectory smoothing planning considers the planning method in joint space. It takes joint velocity, acceleration, and jerk continuity as the trajectory smoothing targets [16,17]. Considering the Gaussian interpolation of boundary conditions, a radial basis function method was proposed by Chettibi [18] to generate point-to-point smooth trajectories of robot arms. A Multi-axis real-time synchronous look-ahead trajectory planning algorithm was proposed by Liang et al. [19] based on dynamically given position and velocity sequences under the constraint of maximum velocity and acceleration of each joint axis. Shrivastava et al. [20] used the cuckoo search algorithm to suppress the linear motion equation to avoid the directional singularity in Cartesian space. Then the Bezier curve was used to depict the shape of the joint trajectory by interpolating linear polynomials with the parabolic blend. In the joint space, the joint movement was divided into acceleration, uniform speed, and deceleration parts; the fourth-order polynomial formed by the root weight was used to plan the acceleration to ensure the continuity of the displacement, speed, acceleration, and jerk of each joint and end-effector of the robot [21].

On the other hand, some literature has studied the trajectory planning of robot end-effectors in Cartesian space [22–26]. However, trajectory smoothing planning in a single space can not guarantee trajectory smoothness in other spaces. Moreover, in the process of high-speed reciprocating motion of a high-speed parallel robot, the influence of its nonlinear dynamic characteristics on the dynamic response and stable operation of the system can not be ignored, especially when the smooth trajectory is mapped from Cartesian space (or joint space) to joint space (or Cartesian space) [27]. Comprehensive kinematics and dynamics characteristics of the 3-DoF parallel manipulator were analyzed under the accepted link Jacobian matrices and proverbial virtual work principle, and the discretized piecewise quintic polynomials were employed to interpolate the sequence of joints' angular position knots, which are transformed from these predefined via-points in Cartesian space [28]. Dupac [29] proposed quintic Hermite piecewise interpolants having functional continuity, and the smoothness of the end-effector path was guaranteed. At the same time, inverse kinematics and inverse dynamics were used to compute the coordinates of the joints. The smoothness of the trajectory of the robot's end-effector is not only smoothly planned in the Cartesian space but also achieved through the kinematic and dynamic changes of the robot joints and the torque generated by the joint actuators [30]. Focus on process shock and vibration caused by abrupt trajectory change, a trajectory smoothing planning combining Cartesian and joint space is proposed in this paper. The main contribution of the current paper are summarized as follows.

(1) The vector method and microelement integration method are used to establish the complete kinematics and dynamics equations of the Delta parallel robot, and the inverse kinematics/dynamics model solving program is written based on Matlab software.

(2) The end-effector trajectory of the Delta parallel robot is planned in Cartesian space, and the data points and inverse control points of the end-effector trajectory are obtained by the normalization method.

The rest of this paper is structured as follows: In section 2, the kinematic and dynamic model of the Delta parallel robot is presented. The trajectory smoothing planning combining Cartesian and joint

space is proposed in Section 3. The simulation results are shown in Section 4. Finally, Conclusions are drawn in Section 5.

2. Kinematic and Dynamic Model of the Delta Parallel Robot

2.1. Kinematic model of the Delta parallel robot

The origin O of the fixed coordinate system $\{O - XYZ\}$ is located in the geometric center of the equilateral triangle $A_1A_2A_3$ which the circumradius is R , the axis of X is along with the direction of $\overrightarrow{OA_1}$, the axis of Y is parallel to the axis of the revolute joint A_1 . The origin o' of the moving coordinate system $\{o' - xyz\}$ is located in the geometric center of the equilateral triangle $C_1C_2C_3$ which the circumradius is r . The length of active links A_iB_i and passive links B_iC_i are the constant value l_a and l_b , respectively. The variable parameters in joint space are $\theta_i, i = 1, 2, 3$, and the output three displacements in the Cartesian space of the moving platform are x, y , and z , respectively. The geometric configuration of the Delta parallel robot is shown in Fig.1.

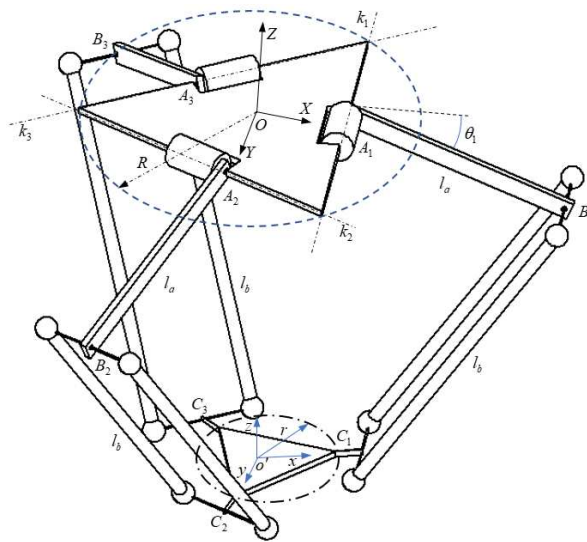


Figure 1. The geometric configuration of Delta parallel robot with three DoFs.

Assumed that the origin of the moving platform o' is $\begin{bmatrix} x & y & z \end{bmatrix}^T$ with respect to the fixed coordinate system. The inverse positional kinematic of the Delta parallel robot can be given by

$$\overrightarrow{B'_iC_i} = \overrightarrow{Oo} - \overrightarrow{OA_i} - \overrightarrow{A_iB'_i} \text{ and } \|\overrightarrow{B'_iC_i}\| = l_b^2 \quad (1)$$

The axis vectors of the revolute joint R_i are k_i ($k_i = \begin{bmatrix} k_{xi} & k_{yi} & k_{zi} \end{bmatrix}^T, i = 1, 2, 3$), respectively. According to the geometric composition characteristics of the Delta parallel robot, k_i can be derived as follows.

$$k_1 = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T, \quad k_2 = \begin{bmatrix} \sqrt{3}/2 & 1/2 & 0 \end{bmatrix}^T, \quad k_3 = \begin{bmatrix} \sqrt{3}/2 & -1/2 & 0 \end{bmatrix}^T \quad (2)$$

The homogeneous transformation matrix of rotation θ about any axis $k = \begin{bmatrix} k_x & k_y & k_z \end{bmatrix}^T$ is satisfied with

$$\text{Rot}(k, \theta) = \begin{bmatrix} k_x k_x (1 - c\theta) + c\theta & k_y k_x (1 - c\theta) - k_z s\theta & k_z k_x (1 - c\theta) + k_y s\theta & 0 \\ k_x k_y (1 - c\theta) + k_z s\theta & k_y k_y (1 - c\theta) + c\theta & k_z k_y (1 - c\theta) - k_x s\theta & 0 \\ k_x k_z (1 - c\theta) - k_y s\theta & k_y k_z (1 - c\theta) + k_x s\theta & k_z k_z (1 - c\theta) + c\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

${}^o P_{B'}$ can be obtained by translation and rotation based on the initial position ${}^o P_B$, yields to

$${}^o P_{B'} = \text{Rot}(k_i, \theta_i) {}^o P_B + \text{Trans}(\overrightarrow{OA_i}) \quad (4)$$

${}^o P_{C'}$ can be obtained according to the motion characteristics of the moving platform, yields to

$${}^o P_{C'} = \text{Trans}(\overrightarrow{O\hat{O}}) {}^o P_C \quad (5)$$

Substitute (4) and (5) into (1), we obtain

$$(x - m \cos \alpha_i)^2 + (y - m \sin \alpha_i)^2 + (z + l_a \sin \theta_i)^2 = l_b^2 \quad (6)$$

where $m = R - r + l_a \cos \theta_i$, $\alpha_i = \frac{2(i-1)\pi}{3}$ are the angle between OA_i and X axis, $i = 1, 2, 3$. Assumed that $\sin(\theta_i) = \frac{2t_i}{1 + t_i^2}$ and $\cos(\theta_i) = \frac{1 - t_i^2}{1 + t_i^2}$, eq.(6) can be calculated as follows

$$\theta_i = 2 \arctan \left(\frac{-U_i \pm \sqrt{U_i^2 - 4K_i V_i}}{2K_i} \right) \quad (7)$$

where

$$K_i = l_b^2 - l_a^2 - x^2 - y^2 - z^2 - (R - r)^2 + 2(x \cos \alpha_i + y \sin \alpha_i)(R - r - l_a) + 2(R - r)l_a$$

$$U_i = -4zl_a$$

$$V_i = l_b^2 - l_a^2 - x^2 - y^2 - z^2 - (R - r)^2 + 2(x \cos \alpha_i + y \sin \alpha_i)(R - r + l_a) - 2(R - r)l_a$$

Assumed that $b_i = B_i C_i$ and derivative of both sides with respect to the time, we obtain

$$A\dot{X} + B\dot{\theta} = 0 \quad (8)$$

where $B = \text{diag}(b_1^T c_1, b_2^T c_2, b_3^T c_3)$ is a diagonal matrix, $A = \begin{bmatrix} b_1^T & b_2^T & b_3^T \end{bmatrix}^T$, and $c_i = \begin{bmatrix} l_a \sin \theta_i \cos \alpha_i & l_a \sin \theta_i \sin \alpha_i & l_a \cos \theta_i \end{bmatrix}^T$, $\dot{X} = \begin{bmatrix} \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^T$ is the velocity vector of the end-effector of Delta parallel robot, $\dot{\theta} = \begin{bmatrix} \dot{\theta}_1 & \dot{\theta}_2 & \dot{\theta}_3 \end{bmatrix}^T$ is the velocity vector of the joints, $0 \in R^{3 \times 1}$ is the zero matrix.

Eq.(8) can be rewritten with

$$\dot{\theta} = -B^{-1}A\dot{X} = J\dot{X} \quad (9)$$

where J is the kinematic Jacobian matrix of the Delta parallel robot.

Derivative (9) with respect to the time, we obtain

$$\ddot{\theta} = J^{-1}(\ddot{X} - \dot{J}\dot{X}) \quad (10)$$

where $\dot{J} = A^{-1}\dot{A}A^{-1}B - A^{-1}\dot{B}$, and \dot{A} , \dot{B} satisfied with

$$\dot{A} = I_a \begin{bmatrix} \cos \alpha_1 \sin \theta_1 & \sin \alpha_1 \sin \theta_1 & \cos \theta_1 \\ \cos \alpha_2 \sin \theta_2 & \sin \alpha_2 \sin \theta_2 & \cos \theta_2 \\ \cos \alpha_3 \sin \theta_3 & \sin \alpha_3 \sin \theta_3 & \cos \theta_3 \end{bmatrix} \dot{\theta}^T + \dot{X}^T \quad (11)$$

and

$$\dot{B} = \text{diag} \left(\dot{b}_1^T c_1 + b_1^T \dot{c}_1, \dot{b}_2^T c_2 + b_2^T \dot{c}_2, \dot{b}_3^T c_3 + b_3^T \dot{c}_3 \right) \quad (12)$$

2.2. Dynamic model of the Delta parallel robot

The dynamic model of the Delta parallel robot with 3-DoFs is give by

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + G(\theta) = \tau \quad (13)$$

where $M(\theta) \in R^{3 \times 3}$ is the inertial mass matrix, $C(\theta, \dot{\theta}) \in R^{3 \times 3}$ is the Coriolis and centripetal force, $\tau \in R^{3 \times 1}$ is the moment, $G(\theta) \in R^{3 \times 1}$ is the gravity matrix. The dynamic equation of the Delta parallel robot has the properties as follows:

Remark 1. The inertial mass matrix $M(\theta)$ is a positive symmetric matrix and is bounded, and satisfy with

$$mI < M(\theta) = M^T(\theta) \leq \bar{m}I \quad (14)$$

where m and \bar{m} are constants more than zero, $I \in R^{3 \times 3}$ is the identity matrix.

The mass of the active link is m_b , the mass of the passive link is m_p , and the mass of the moving platform is m_m . The mass matrix $M(\theta)$ of the Delta parallel robot consists of three parts: active link $M_{m_b}(\theta)$, passive link $M_{m_p}(\theta)$ and moving platform $M_{m_m}(\theta)$.

Considering the motion characteristics of the Delta parallel robot, the mass matrix of the moving platform can be expressed as

$$M_{m_m}(\theta) = m_{pl} J^T J \quad (15)$$

where m_{pl} is the mass of the moving platform and load, J is the Jacobian matrix.

The mass matrix of the active link is given by

$$M_{m_b}(\theta) = \begin{bmatrix} I_{b1} & 0 & 0 \\ 0 & I_{b2} & 0 \\ 0 & 0 & I_{b3} \end{bmatrix} \quad (16)$$

where $I_{b1} = I_{b2} = I_{b3} = I_m + \left(\frac{m_b}{3} + m_c\right) l_a^2$, I_m is the moment of inertia of the actuator, m_c is the mass of the spherical hinge. The linear velocity at any point along the link $B_i C_i$ can be expressed as

$$v_{B_i C_i}(x) = \left(1 - \frac{x}{l_b}\right) v_1 + \frac{x}{l_b} v_2 \quad (17)$$

where v_1 is the velocity at the connection between $B_i C_i$ and $A_i B_i$, v_2 is the velocity at the connection between $B_i C_i$ and the moving platform. the elementary kinematic energy can be given by

$$dT = \frac{1}{2} v_{B_i C_i}^2 dm = \frac{1}{2} v_{B_i C_i}^2 \rho s dx \quad (18)$$

where ρ is the material density of the link, s is the cross-sectional area.

Integrating (17), we obtain

$$T = \int_0^{l_b} dT = \frac{1}{2} \rho s \int_0^{l_b} v_{B_i C_i}^2 dx = \frac{1}{2} \left(\frac{1}{3} m_l (v_1^2 + v_2^2 + v_1 v_2) \right) \quad (19)$$

The Jacobian matrix of the joint B_i is defined as $J_{u,i}$, $M_{m_p}(\theta)$ can be derived as

$$M_{m_p}(\theta) = \sum_{i=1}^3 \frac{1}{3} m_l \left(J^T J + J_{u,i}^T J_{u,i} + J_{u,i}^T J \right) \quad (20)$$

Assumed that the equivalent mass of the moving platform is $m_{eq} = m_{pf} + m_l$, there has

$$\sum_{i=1}^3 \frac{1}{3} m_l \left(J_{u,i}^T J_{u,i} \right) = \frac{1}{3} m_l \begin{bmatrix} l_a^2 & 0 & 0 \\ 0 & l_a^2 & 0 \\ 0 & 0 & l_a^2 \end{bmatrix} \quad (21)$$

The mass matrix of the Delta parallel robot can be expressed completely as

$$M(\theta) = I_{bt} + m_{eq} J^T J \quad (22)$$

$$\text{where } I_{bt} = \begin{bmatrix} I_{eq} & 0 & 0 \\ 0 & I_{eq} & 0 \\ 0 & 0 & I_{eq} \end{bmatrix}, \text{ and } I_{eq} = I_m + \left(\frac{m_b}{3} + \frac{2m_l}{3} + m_c \right) l_a^2.$$

3. Trajectory smoothing planing combining Cartesian and joint space

Trajectory smoothing planning of the Delta parallel robot combining Cartesian and joint space is mainly to solve the problem of process vibration caused by acceleration and jerk change which is caused by discontinuity at the inflection point of the small line segment.

3.1. Trajectory planning in Cartesian space

Assume that the end-effector of the Delta parallel robot passes through the set point $p_i = (x_i, y_i, z_i)$ ($i = 0, 1, \dots, n$), the coordinates of each interpolation point are expressed as

$$x = x_0 + \lambda \Delta x, \quad y = y_0 + \lambda \Delta y, \quad z = z_0 + \lambda \Delta z \quad (23)$$

where λ is a normalized factor, (x_0, y_0, z_0) is the initial point position, Δx , Δy and Δz are increments on the x , y and z , respectively, and satisfied with

$$\Delta x = x_{i+1} - x_i, \quad \Delta y = y_{i+1} - y_i, \quad \Delta z = z_{i+1} - z_i \quad (24)$$

Assume that the velocity of the line in the middle section is v , the acceleration of the parabola segment is a , the movement time and distance of the parabola segment are satisfied with

$$T_p = \frac{v}{a}, \quad L_p = \frac{1}{2} a T_p^2 \quad (25)$$

Considering the (24), the time, displacement and acceleration of the parabola segment after normalization can be expressed as

$$T_{p\lambda i} = T_p / T_i, \quad L_{p\lambda i} = L_p / L_i, \quad a_{p\lambda i} = 2L_{p\lambda i} / T_{p\lambda i}^2 \quad (26)$$

where

$$L_i = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2}$$

$$T_i = 2T_p + (L_i - 2L_p) / v$$

The normalized factor λ is given by

$$\lambda = \begin{cases} \frac{1}{2}a_{\lambda i}t^2 & 0 \leq t \leq T_{p\lambda i} \\ \frac{1}{2}a_{\lambda i}T_{p\lambda i}^2 + a_{\lambda i}T_{p\lambda i}(t - T_{p\lambda i}) & T_{p\lambda i} < t \leq 1 - T_{p\lambda i} \\ \frac{1}{2}a_{\lambda i}T_{p\lambda i}^2 + a_{\lambda i}T_{p\lambda i}(t - T_{p\lambda i}) - \frac{1}{2}a_{\lambda i}(t + T_{p\lambda i} - 1)^2 & 1 - T_{p\lambda i} < t \leq 1 \end{cases} \quad (27)$$

where $t = i/n, i = 0, 1, \dots, n$. $\lambda = 0$ is the starting point of the line segment, and $\lambda = 1$ is the end point of the line segment.

3.2. Trajectory planning in joint space

By using Eqs. (7)-(22), the interpolation points obtained from Cartesian space are inversely mapped to joint space and used as the critical point of trajectory planning in joint space for quadratic interpolation using quintic B-spline curve, which is expressed as

$$\theta_{i,k}(t) = \sum_{l=0}^k P_{i+l} B_{l,k}(t) \quad (28)$$

where $t \in [0, 1]$, $\theta_{i,k}(t)$ is the B-spline curve of k^{th} , $P_{i+l} (l = 0, \dots, k)$ are the control points, $B_{l,k}(t)$ is the B-spline basis function which can be given by

$$B_{l,k}(t) = \frac{1}{k!} \sum_{j=0}^{k-l} (-1)^j C_{k+1}^j (t+k-l-j)^k \quad (29)$$

where $C_{k+1}^j = \frac{(k+1)!}{j!(k+1-j)!}$, $k = 5$.

Substitute (29) into (28), we obtain

$$\theta_{i,5}(t) = \frac{1}{120} \begin{bmatrix} 1 & t & t^2 & t^3 & t^4 & t^5 \end{bmatrix} = A \begin{bmatrix} P_i & P_{i+1} & P_{i+2} & P_{i+3} & P_{i+4} & P_{i+5} \end{bmatrix} \quad (30)$$

where

$$A = \begin{bmatrix} 1 & 26 & 66 & 26 & 1 & 0 \\ -5 & -50 & 0 & 50 & 5 & 0 \\ 10 & 20 & -60 & 20 & 10 & 0 \\ -10 & 20 & 0 & -20 & 10 & 0 \\ 5 & -20 & 30 & -20 & 5 & 0 \\ -1 & 5 & -10 & 10 & -5 & 1 \end{bmatrix}$$

When $t = 0$, the end-point of the B-spline curve of 5^{th} is given by

$$\theta_{i,5}(0) = \frac{1}{120} (P_i + 26P_{i+1} + 66P_{i+2} + 26P_{i+3} + P_{i+4}) \quad (31)$$

Given $m + 1$ data points in joint space $\theta_i (i = 0, 1, \dots, m)$, inverse calculate the control point P_i . Since the number of the control points is $m + 5$, so four initial constraint equations are selected as follows:

$$\begin{cases} \ddot{\theta}_0(0) = 20P_0 + 40P_1 - 120P_2 + 40P_3 + 20P_4 = 0 \\ \dot{\theta}_0(0) = -60P_0 + 120P_1 - 120P_3 + 60P_4 = 0 \\ \ddot{\theta}_m(1) = 20P_m + 40P_{m+1} - 120P_{m+2} + 40P_{m+3} + 20P_{m+4} = 0 \\ \dot{\theta}_m(1) = -60P_m + 120P_{m+1} - 120P_{m+3} + 60P_{m+4} = 0 \end{cases} \quad (32)$$

Simultaneous (30) and (32), we obtain

$$120 \begin{bmatrix} 0 \\ 0 \\ \theta_0 \\ \theta_1 \\ \vdots \\ \theta_m \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -5 & -50 & 0 & 50 & 5 \\ 20 & 40 & -120 & 40 & 20 \\ 1 & 26 & 66 & 26 & 1 \\ & 1 & 26 & 66 & 26 & 1 \\ & & \ddots & \ddots & \ddots & \ddots \\ & & & 1 & 26 & 66 & 26 & 1 \\ & & & 20 & 40 & -120 & 40 & 20 \\ & & & -5 & -50 & 0 & 50 & -5 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ \vdots \\ P_m \\ P_{m+1} \\ P_{m+2} \\ P_{m+3} \\ P_{m+4} \end{bmatrix} \quad (33)$$

The method of inverse control vertex is used to calculate NURBS curve, and the NURBS curve of degree k is defined as

$$p(t) = \frac{\sum_{i=0}^n d_i \omega_i B_{i,k}(t)}{\sum_{i=0}^n \omega_i B_{i,k}(t)} = \frac{TB_{i,k}(t)D_i}{TB_{i,k}(t)W_i}, 0 \leq t \leq 1 \quad (34)$$

where $d_i (i = 0, 1, \dots, n)$ is the control vertex of curve $p(t)$, ω_i is the weight factor whose number is equal to the number of the control vertices.

$$t = \frac{u - u_{i+k}}{u_{i+k+1} - u_{i+k}}$$

$$D_i = \begin{bmatrix} \omega_i d_i & \omega_{i+1} d_{i+1} & \omega_{i+2} d_{i+2} & \omega_{i+3} d_{i+3} \end{bmatrix}^T$$

$$W_i = \begin{bmatrix} \omega_i & \omega_{i+1} & \omega_{i+2} & \omega_{i+3} \end{bmatrix}^T$$

4. Experiments

The experimental platform of the Delta parallel robot based on EtherCAT communication mode is constructed. Motion control card selected from the Googo GEN-009-00 series, EtherCAT's distributed clock device ensures synchronous movement between the various axes, thus avoiding position deviation in time response due to the strong coupling characteristics between the various axes. Based on the EtherCAT master-slave architecture, three slave stations are configured on the experimental platform. The model number is JSDG2S-15A-E, and the connection cables are CAT5E-STP industrial ethernet cable RJ45. The motor model is JSMA-PBC04AAKB, the rated torque of the motor is 3000rpm, the rated power is 400W, the moment of inertia is 0.000073kg-m², and the reduction ratio of the motor reducer is 1:25. The experimental platform of the Delta parallel robot is shown in Fig.2.

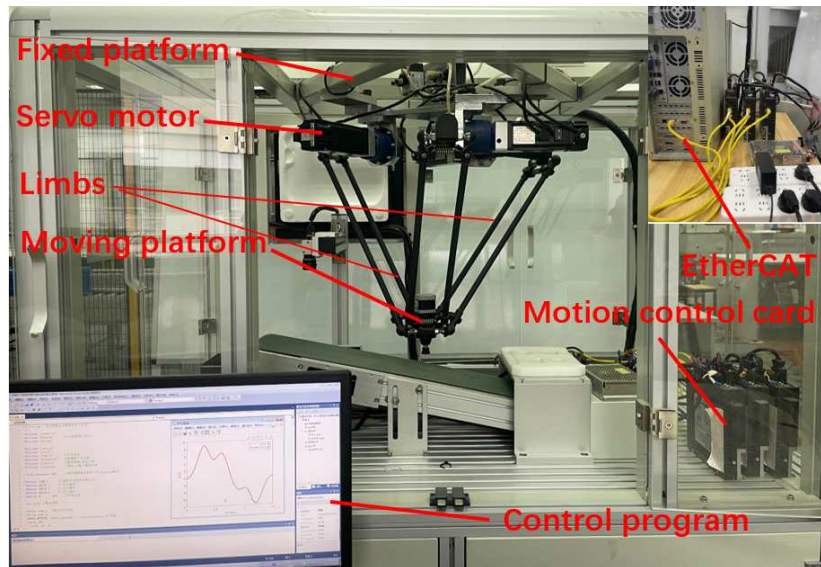


Figure 2. Experimental platform of the Delta parallel robot.

The parameters of the Delta parallel robot are shown in Table 1.

Table 1. Parameters of the Delta parallel robot.

Parameters	Values	Parameters	Values
Length of active link l_a	0.1800 m	Radius of the fixed platform R	0.1350 m
Length of passive link l_b	0.5000 m	Radius of the moving platform r	0.0400 m
Mass of the moving platform m_p	0.2231 kg	Mass of the active link m_b	0.3392 Kg g^2
Mass of the passive link m_l	0.1412 Kg	Moment of inertia of the motor I_m	0.000073 kg · m ²

The test time is set to 10s, and the sampling number is set to 20001. The traditional PID controller is used to verify the single-space (Cartesian space) trajectory smoothing planning, and the proposed trajectory method is compared. PID controller parameters are set to $K_I = \text{diag}[3.05, 4.05, 3.05]^T$, $K_P = \text{diag}[75, 75, 70]^T$ and $K_D = \text{diag}[10, 15, 10]^T$.

The end-effector motion trajectory of the Delta parallel robot is selected as a five-pointed star curve, 11 data points of the curve are given, and the start and end weight factors are selected as 1.0, and the rest are 0.9. The trajectory of the end-effector in Cartesian space is shown in Fig.3.

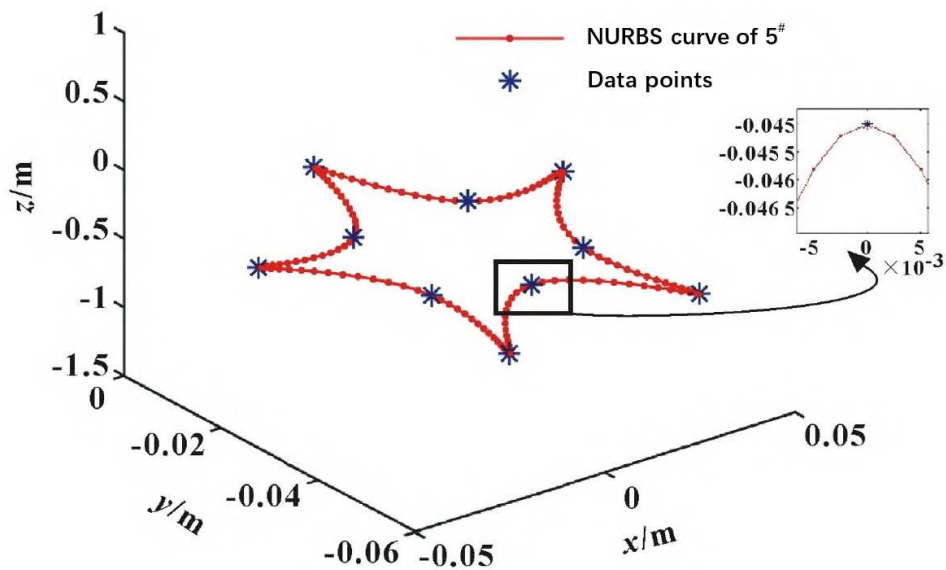


Figure 3. End-effector trajectory of the Delta parallel robot in Cartesian space.

The interpolation points obtained from Cartesian space are mapped to joint space through inverse kinematics, and the fifth uniform B-spline curve is used as input data for the second interpolation and trajectory smoothing planning of the angle, angular velocity, angular acceleration, and angular jerk are obtained in joint space which are shown in Fig.4-Fig.6, respectively.

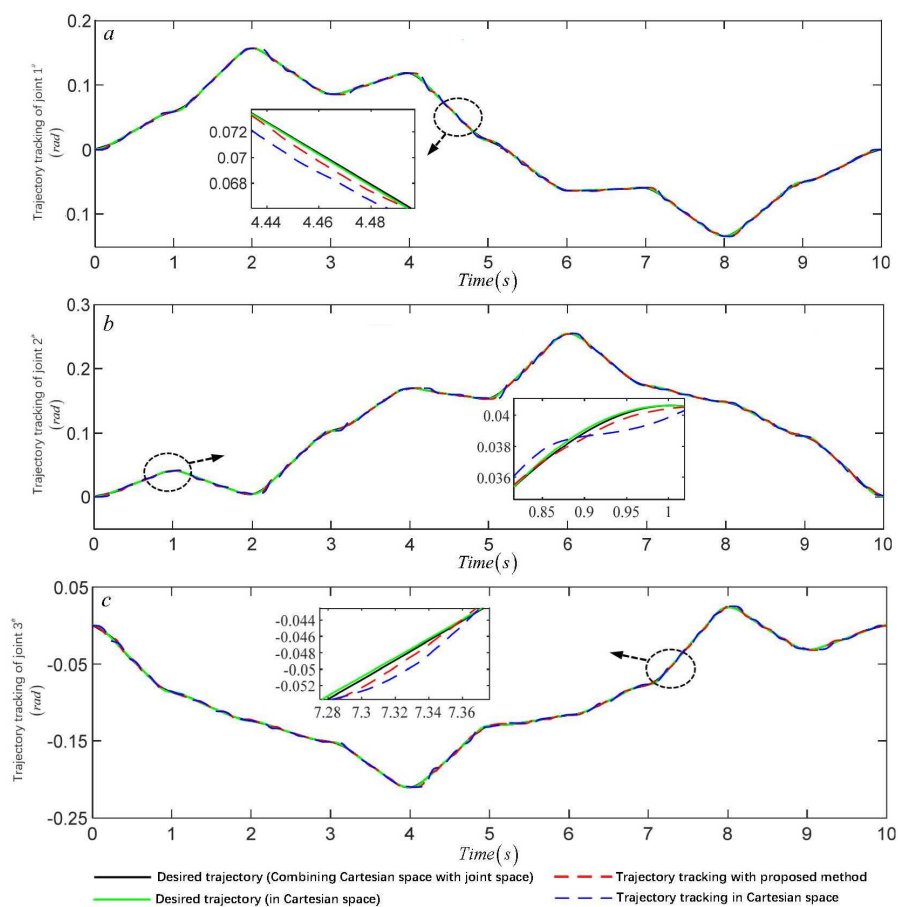


Figure 4. Trajectory tracking of each joint with proposed method and in Cartesian space. (a) joint 1. (b) Joint 2. (c) joint 3.

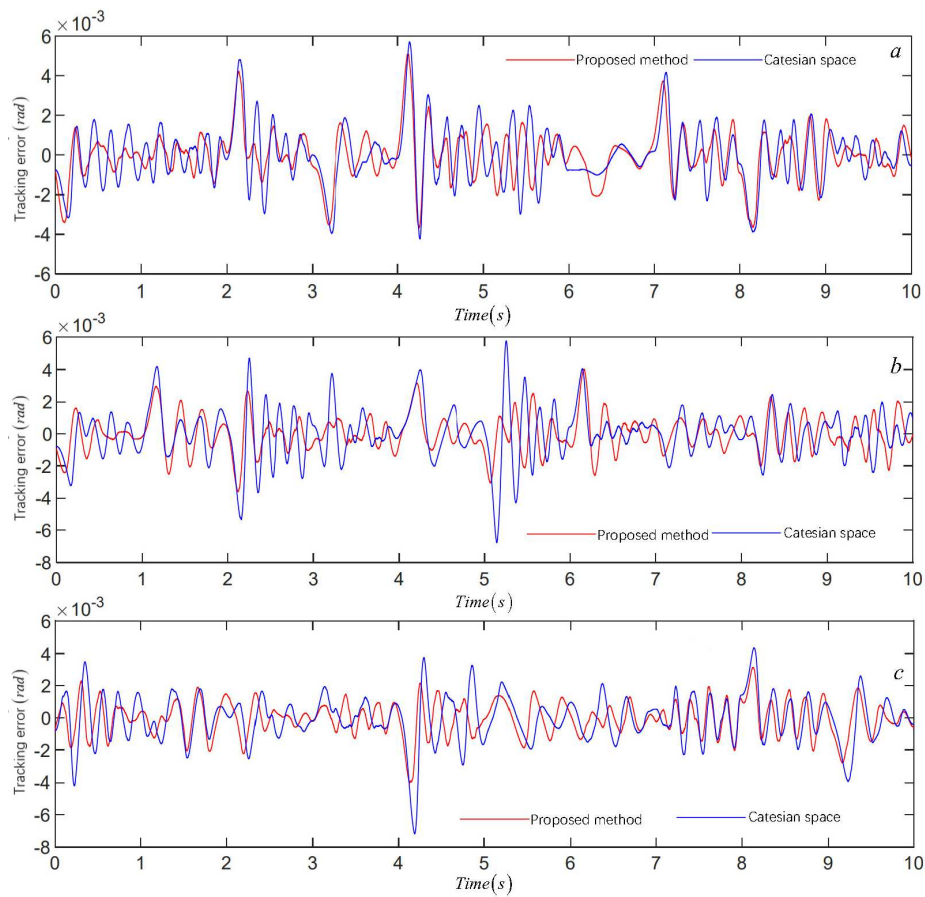


Figure 5. Tracking error of each joint with proposed method and in Cartesian space. Comparison of joint velocity trajectories in joint space.(a) joint 1. (b) Joint 2. (c) joint 3.

Fig.4 and Fig.5 show that compared with single space (Cartesian space) trajectory planning, the trajectory planning algorithm combining joint space can obtain a smoothing trajectory, and the discontinuity of the Jerk curve and trajectory mutation can be effectively improved under the same controller (traditional PID). Among them, the peak tracking error of joint 2 and 3 are 29% and 44% lower than that of single spatial trajectory planning, and the error value of joint 2 is smaller in the 5s transition stage. The errors are worth rapidly converging at the initial stage of the three joints.

The comparison of the peak value of trajectory tracking error is shown in Tables 2.

Table 2. Comparison of the peak value of trajectory tracking error.

Joint	The peak value of trajectory tracking error (rad)	
	Cartesian space	Combining Cartesian and joint space
1 th	0.0057	0.0051
2 th	0.0068	0.0040
3 th	0.0072	0.0040

The control torque of the three joints is shown in Fig.6.

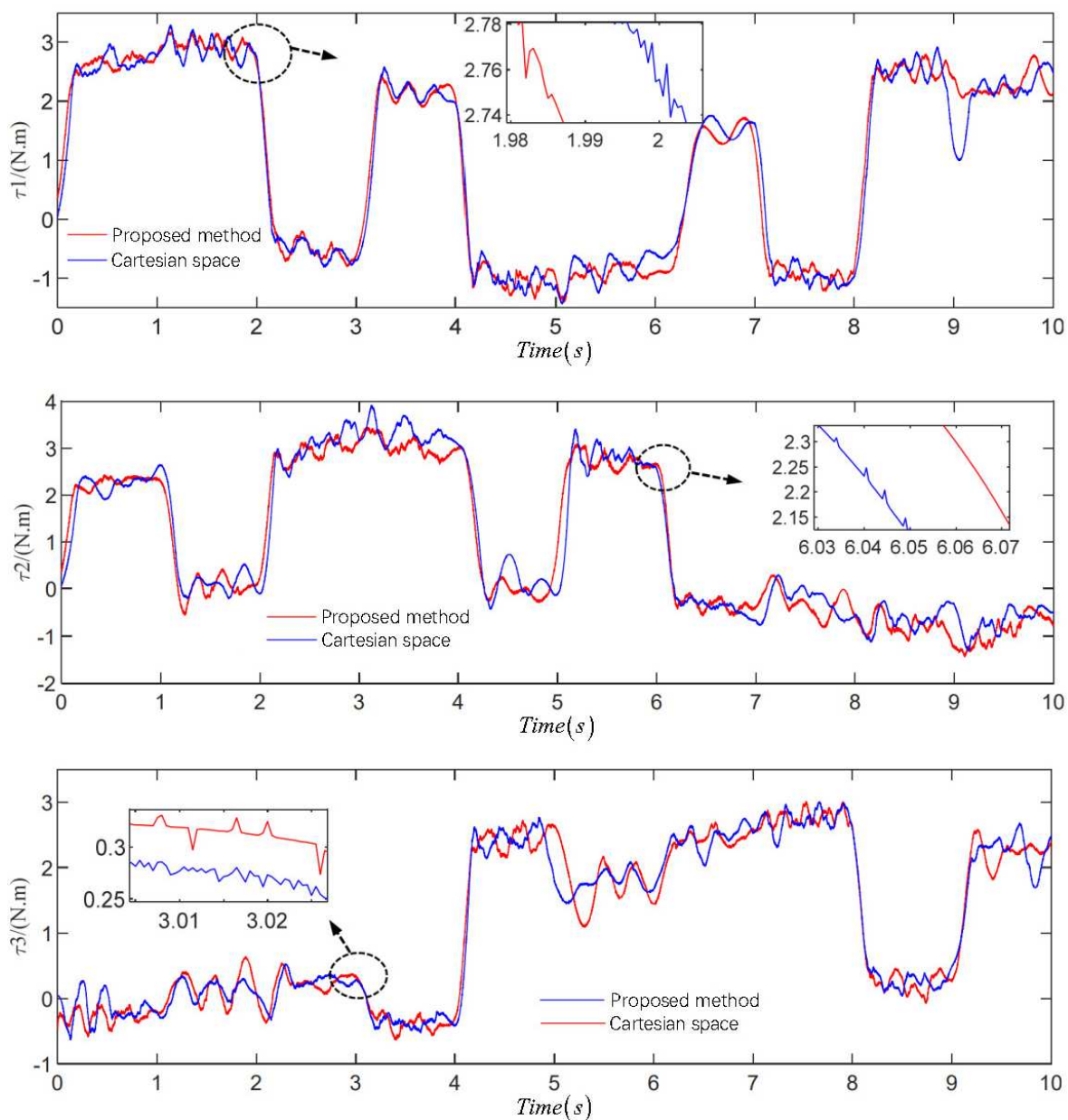


Figure 6. Comparison of joint acceleration trajectories in joint space.

Fig.6 shows the peak torque reduction rate of joint 2 of the trajectory obtained by the proposed trajectory planning algorithm is 11.6%, and the torque change trend is gentle. Although the peak torque reduction rates of joints 1 and 3 are 3.5% and 1.6%, respectively, it can be seen from the torque amplification diagram that the torque vibration and shock phenomena at the transition of adjacent line segments such as 2s, 3s, and 6s are effectively reduced when the acceleration chattering is improved. A comparison of the peak value of control torque is shown in Table 3.

Table 3. Comparison of the peak value of control torque.

Joint	Peak value of control torque τ_{\max} / (N.m)		
	Cartesian space	Combining Cartesian and joint space	reduction rate
1 th	3.2909	3.1743	3.5%
2 th	3.9193	3.4657	11.6%
3 th	3.0056	2.9584	1.6%

5. Conclusions

In order to solve the problem of process shock and vibration caused by tangential discontinuity at the corner of small line segment in the trajectory planning of Delta parallel robot end-effector, a smooth trajectory planning method based on the combination of cubic NURBS curve in Cartesian space and quintic uniform B-spline curve in joint space is proposed in this paper. (1) The trajectory smoothing planning method combined with Cartesian space and joint space effectively solves the trajectory smoothing problem with continuous high order curvature derivatives, and the obtained joint jerk is continuously smooth. (2) The proposed method can make the robot's end-effector move precisely along a smooth trajectory while making the angular velocity, acceleration and jerk of the joint continuously smooth without impact. (3) Smooth trajectory planning is carried out from the two aspects of robot kinematics and dynamics, which reduces the tracking error by 21.12% and the maximum joint torque by 11.65%, effectively reducing the vibration and impact during the movement.

Author Contributions: Conceptualization, Zhu Dachang and Li Fangyi; methodology, He Yonglong; software, Yu Xuezhe; validation, He Yonglong; writing—original draft preparation, Zhu Dachang; writing—review and editing, Li Fangyi; All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Simionescu, I.; Ciupitu, L.; Ionita, L.C. Static balancing with elastic systems of DELTA parallel robots. *Mech Mach Theory* **2015**, *87*, 150–162.
2. Liu, C.; Cao, G.H.; Qu, Y.Y. Safety analysis via forward kinematics of delta parallel robot using machine learning. *Safety Sci* **2019**, *117*, 243–249.
3. Boryga, M. The use of higher-degree polynomials for trajectory planning with jerk, acceleration and velocity constraints. *Int J Comput Appl T* **2020**, *63(4)*, 337–347.
4. Zhao, Y.J.; Yuan, F.F.; Chen, C.W.; Jin, L.; Li, J.Y.; Zhang, X.W.; Lu, X.J. Inverse kinematics and trajectory planning for a hyper-redundant bionic trunklink robot. *Int J Robot Autom* **2020**, *35(3)*, 229–241.
5. Damasevicius, R.; Maskeliunas, R.; Narvydas, G.; Narbutaite, R.; Polap, D.; Wozniak, W. Intelligent automation of dental material analysis using robotic arm with Jerk optimized trajectory. *J Amb Intel Hum Comp* **2020**, *11(12)*, 6223–6234.
6. Zhang, S.; Liu, X.J.; Yan, B.K.; Bi, J.; Han, X.D. A real-time look-ahead trajectory planning methodology for multi small line segments path. *Chin J Mech Eng En* **2023**, *36(1)*, 59.
7. Wu, G.L.; Zhao, W.K.; Zhang, X.P. Optimum time-energy-jerk trajectory planning for serial robotic manipulators by reparameterized quintic NURBS curves. *P I Mech Eng C-J Mec* **2021**, *235(9)*, 4382–4393.
8. Xu, G.P.; Zhang, H.; Meng, Z.; Sun, Y.Z. Automatic interpolation algorithm for NURBS trajectory of shoe spraying based on 7-DOF robot. *Int J Cloth Sci Tech* **2021**, *34(3)*, 434–450.
9. Wu, P.; Wang, Z.Y.; Jing, H.K.; Zhao, P.F. Optimal time-jerk trajectory planning for Delta parallel robot based on improved butterfly optimization algorithm. *Appl Sci -Basel* **2022**, *12(16)*, 8145.
10. Liang, X.; Su, T.T. Quintic Pythagorean-Hodograph curves based trajectory planning for Delta robot with a prescribed geometrical constraint. *Appl Sci -Basel* **2019**, *9(12)*, 4491.
11. Bilal, H.; Yin, B.Q.; Kuma, A.; Ali, M.; Zhang, J.; Yao, J.F. Jerk-bounded trajectory planning for rotary flexible joint manipulator: an experimental approach. *Appl Sci -BaseSoft Comput* **2023**, *27(7)*, 4029–4039.
12. Fang, Y.; Hu, J.; Liu, W.H.; Shao, Q.Q.; Qi, J.; Peng, Y.H. Smooth and time-optimal S-curve trajectory planning for automated robots and machines. *Mech Mach Theory* **2019**, *137*, 127–153.
13. Singh, G.; Banga, V.K. Kinematics and trajectory planning analysis based on hybrid optimization algorithm for an industrial robotic manipulators. *Mech Mach Theory* **2022**, *26(21)*, 11339–11372.
14. Dong, H.; Cong, M.; Chen, H.P. Effective algorithms to find a minimum-time yet high smooth robot joint trajectory. *Int J Robot Autom* **2019**, *34(4)*, 331–343.
15. Brown, S.; Pierson, H.A. Adaptive path planning of novel complex parts for industrial spraying operations. *Prod Manuf Res* **2020**, *8(1)*, 335–368.

16. Wang, F.; Wu, Z.J.; Bao, T.T. Time-jerk optimal trajectory planning of industrial robots based on a hybrid WOA-GA algorithm. *Processes* **2022**, *10*(5), 1014.
17. Nadir, B.; Mohammed, O.; Minh-Tuan, N.; Abderrezak, S. Optimal trajectory generation method to find a smooth robot joint trajectory based on multiquadric radial basis functions. *Int J Adv Manuf Technol* **2022**, *120*(1-2), 297–312.
18. Chettibi, T. Smooth point-to-point trajectory planning for robot manipulators by using radial basis functions. *Robotica* **2019**, *37*(3), 539–559.
19. Liang, Y.Y.; Yao, C.Z.; Wu, W.; Wang, L.; Wang, Q.Y. Design and implementation of multi-axis real-time synchronous look-ahead trajectory planning algorithm. *Int J Adv Manuf Technol* **2022**, *119*(7-8), 4991–5009.
20. Shrivastava, A.; Dalla, V.K. Multi-segment trajectory tracking of the redundant space robot for smooth motion planning based on interpolation of linear polynomials with parabolic blend. *P I Mech Eng C-J Mec* **2022**, *236*(16), 9255–9269.
21. Wang, H.; Wang, H.; Huang, J.H.; Zhao, B.; Quan, L. Smooth point-to-point trajectory planning for industrial robots with kinematical constraints based on high-order polynomial curve. *Mech Mach Theory* **2019**, *139*, 284–293.
22. Wang, G.R.; Xu, F.; Zhou, K.; Pang, Z.H. S-velocity profile of industrial robot based on NURBS curve and slerp interpolation. *Processes* **2022**, *10*(11), 2195.
23. Grassmann, R.M.; Burgner-Kahrs, J. Quaternion-based smooth trajectory generator for via poses in SE(3) considering kinematic limits in Cartesian space. *IEEE Robot Autom Let* **2019**, *4*(4), 4192–4199.
24. Singh, G.; Banga, V.R. Combinations of novel hybrid optimization algorithms-based trajectory planning analysis for an industrial robotic manipulators. *J Field Robot* **2022**, *39*(5), 650–674.
25. Zhao, Y.Q.; Mei, J.P.; Niu, M.T. Vibration error-based trajectory planning of a 5-dof hybrid machine tool. *Robot Cim-Int Manuf* **2021**, *69*, 102095.
26. Wang, Q.; Wang, Z.B.; Shuai, M. Trajectory planning for a 6-DoF manipulator used for orthopaedic surgery. *Int J Intell Robot Appl* **2020**, *4*(1), 82–94.
27. Yang, D.; Xie, F.G.; Liu, X.J. Velocity constraints based approach for online trajectory planning of high-speed parallel robots. *Chin J Mech Eng En* **2022**, *35*(1), 127.
28. Lu, S.; Ding, B.X.; Li, Y.M. Minimum-jerk trajectory planning pertaining to a translational 3-degree-of-freedom parallel manipulator through piecewise quintic polynomials interpolation. *Adv Mech Eng* **2020**, *12*(3), 1687814020913667.
29. Dupac, M. Mathematical modeling and simulation of the inverse kinematic of a redundant robotic manipulator using azimuthal angles and spherical polar piecewise interpolation. *Math Comput Simulat* **2023**, *209*, 282–298.
30. Rout, A.; Deepak, B.B.V.L.; Biswal, B.B.; Mahanta, G.B. Trajectory generation of an industrial robot with constrained kinematic and dynamic variations for improving positional accuracy. *Int J Appl Metaheur* **2021**, *12*(3), 163–179.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.