

## Research Article

# A Manipulator Pose Planning Algorithm Based on Matrix Information Geometry

Xiaomin Duan <sup>1</sup>, Anqi Mu <sup>1</sup>, Hao Guo <sup>1</sup> and Xinyu Zhao <sup>2,3</sup>

<sup>1</sup>School of Science, Dalian Jiaotong University, Dalian, China

<sup>2</sup>School of Material Science and Engineering, Dalian Jiaotong University, Dalian, China

<sup>3</sup>Dalian Ruidi Sound Light Technology Co. Ltd., Dalian, China

Correspondence should be addressed to Xinyu Zhao; [xyz@djtu.edu.cn](mailto:xyz@djtu.edu.cn)

Received 27 July 2023; Revised 27 November 2023; Accepted 28 December 2023; Published 16 January 2024

Academic Editor: Afaque Manzoor Soomro

Copyright © 2024 Xiaomin Duan et al. 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.

In an automatic ultrasonic testing system constituted by an ultrasonic probe and a six-axis manipulator, the manipulator needs to run from a static state to the target velocity. To prevent equipment damage caused by sudden acceleration or deceleration, it is necessary to plan the position and pose of the end effector of the manipulator at each detected point. In this manuscript, an algorithm for planning the position and pose of the manipulator is proposed based on the information geometry structure of special orthogonal groups. As the linear operation of the orthogonal matrix corresponding to the manipulator pose is not closed, the manipulator pose at each detected point was calculated using the straightness of the Lie algebra of the special orthogonal group. The matrix information geometry algorithm enabled not only the manipulator to accelerate and decelerate uniformly along the detection trajectory, but also the angular acceleration of the end effector to accelerate uniformly at first, then keep a uniform velocity, and finally decelerate uniformly. The platform motion experiments with the Turin TKB070S six-axis manipulator are carried out to verify the effectiveness of the matrix information geometry algorithm for planning the pose of the manipulator.

## 1. Introduction

Ultrasonic testing is an important nondestructive testing approach frequently applied to detect the internal defects of composite materials with simple structures [1–4]. However, the fast-growing composite manufacturing technology brings about the problem of complex shapes in many workpieces to be detected, such as curved shapes, variable thickness, and intricate rotating structures. Aircraft engine blades must have extremely high self-reliability as they serve in high-temperature and high-pressure environments. There are many methods for manufacturing blades, and we discuss diffusion welding of the aviation blades. Its manufacturing process first involves diffusion welding of flat alloy materials, followed by high-temperature torsional plastic forming, and finally, curved surface machining to obtain geometric contours consistent with the design drawings. Blades belong to the thin-walled parts, and after torsional deformation, the unloading of the forming load will cause a rebound phenomenon, which leads to the error of the formed blade surface.

Usually, after the blade is twisted and deformed, and before the curved surface is machined, it is still necessary to perform ultrasonic testing on the blade to ensure the quality of the diffusion welding interface. To achieve blade surface scanning, a line laser profilometer is first used in conjunction with a robotic multi-axis system to obtain blade point clouds, and then point cloud data are used to fit and obtain the scanning trajectory. Therefore, it is crucial to achieve the automatic detection of workpieces with the complex structures.

Figure 1 shows the automatic monitoring system for civil aviation engine blades. According to the requirements of blade structure and water immersion ultrasonic C-scan, Zhao et al. [5] designed a multi-axis detection system. The blade was fixed on the workpiece rotation axis, and the six-axis manipulator stood upright on the side of the blade. The manipulator base could be translated parallel to the blade rotation axis. During ultrasonic scanning inspection, a six-axis manipulator held the ultrasonic probe and incident vertically along the normal direction of the blade surface. The manipulator and the blade rotation axis achieved a seven-

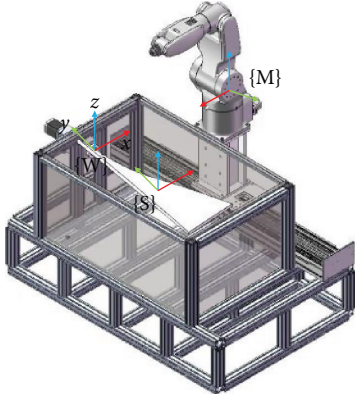


FIGURE 1: Automatic monitoring system for civil aviation engine blades.

axis linkage profile scanning, and then the manipulator base stepped along the track and reversed the profile motion to achieve the next line of scanning. By analogy, it completed the overall scanning of this blade. A manipulator having six degrees of freedom can reach any position and make any pose within its working range. The position and pose of a six-axis manipulator equipped with an ultrasonic probe on its motion trajectory are simulated on an offline simulation platform. This method greatly improves the curved surface detection efficiency and accuracy [6–11].

In this manuscript, we mainly propose how to plan the space trajectory of the manipulator and present a concrete and feasible algorithm. The trajectory in space is a description of the expected motion of the manipulator, and the trajectory in this paper specifically refers to the position and pose of the flange at the manipulator end or the tool clamped on the flange.

The positions and poses at detected points should be subjected to interpolation treatment in trajectory planning. The purpose of interpolation is to facilitate the smooth transition of the manipulator from the initial position and pose to the desired position and pose. And it ensures that each axis of the manipulator can arrive at the specified position and angle within each interpolation period. Besides, interpolation prevents fast acceleration or deceleration of the manipulator during the detection process, helping the manipulator to accomplish the detection task smoothly and efficiently.

In the moving process of the manipulator, the position and pose of the end effector are described as the  $3 \times 1$  vector and  $3 \times 3$  orthogonal matrix in the three-dimensional (3D) real vector space, respectively [12]. According to the linear relationship of the 3D real vector space, the positions at all detected points are planned at first, followed by the pose planning at each detected point based on the proportional relationship of position variations at the detected points. The linear operation of the orthogonal matrix corresponding to the pose is unclosed, so the Euler angle or quaternion transformation method is usually adopted for interpolation [13]. However, the Euler angle method has multiple solutions, and the quaternion method may also encounter illegal situations

caused by the rounding errors. To resolve the above problems, a matrix information geometry method [14–16] is used to plan the pose of the manipulator at each detected point in this paper. The pose of the manipulator in motion corresponds to an element in the special orthogonal group [17]. Based on the linear property of the Lie algebra of the special orthogonal group, an algorithm is proposed to plan the pose of the manipulator at each detected point. The algorithm facilitates the steady movement of the probe at the manipulator end according to the planned position and pose. A matrix Lie group and its Lie algebra are transformed through the exponential map and the logarithmic map. In general, the information geometry algorithm does not need to discuss multiple solutions [18], and the resulting pose matrices are also legal. And, according to the physical meaning of the Lie algebra of special orthogonal groups, the angular acceleration of the end effector is a uniform acceleration and deceleration motion.

The matrix information geometry algorithm for the pose planning of the manipulator can address the issue of unclosed linear operation of the orthogonal matrices. In fact, due to the nonlinear relationship between orthogonal matrices of two different poses, it is difficult to determine the transition pose of these two poses by ratio. In this manuscript, the pose planning algorithm is constructed based on the geometric structure of the special orthogonal group, and the pose at the detected point is planned using the straightness of the Lie algebra. The proposed algorithm promotes the stable movement of the manipulator along the planned trajectory.

The geometric structure of the special orthogonal group will be introduced first in the following text. Then, the algorithm for planning the pose of the manipulator will be proposed based on the linear closure of the Lie algebra on the special orthogonal group. Finally, the algorithm will be verified through manipulator motion experiments.

## 2. Geometric Structure of the Special Orthogonal Group

The manipulator's position and pose in the space can be represented by  $4 \times 4$  matrices. All the orthogonal matrices of the same order with the determinant of 1 form a special orthogonal group, which is a critical matrix compact Lie group. Next, a brief introduction is given to the concepts of special orthogonal groups and their Lie algebras, which are the theoretical basis for designing the pose-planning algorithm [18–21].

**2.1. Matrix Lie Group.** The Lie group has both group and manifold structures. Addition and inverse operations in the group structure are analytical maps in the manifold structure. If a group  $G$  satisfies

- (1)  $G$  is a smooth manifold,
- (2)  $G \times G \rightarrow G$  is closed,
- (3)  $h \rightarrow h^{-1}$  is smooth,  $\forall h \in G$ ,

$G$  becomes a Lie group.

The tangent space at the identity element of the Lie group  $G$  is its Lie algebra  $\mathfrak{g}$ , which is a linear space. The exponential map from the Lie algebra  $\mathfrak{g}$  to the Lie group  $G$  is denoted as  $\exp$ . Although, the exponential map is neither injective nor surjective, it is a diffeomorphism from the neighborhood of the identity element  $I$  of  $G$  to the neighborhood of the identity element  $0$  of  $\mathfrak{g}$ . The logarithmic map is indicated by  $\log$ , which is the local inverse map of the exponential map.

Lie groups in which all elements are matrices, are described as the matrix Lie groups. Exponential and log maps enable the transformation between the matrix Lie group and Lie algebra, and play an essential role in the Lie group theories. The exponential map from the Lie algebra to the matrix Lie group  $G$  is represented as follows [22, 23]:

$$\exp(V) = \sum_{m=0}^{\infty} \frac{V^m}{m!}, \quad (1)$$

where  $V$  is an element in the Lie algebra  $\mathfrak{g}$ . The log map from the matrix Lie group  $G$  to its Lie algebra  $\mathfrak{g}$  is defined as follows:

$$\log(X) = \sum_{m=1}^{\infty} (-1)^{m+1} \frac{(X-I)^m}{m}, \quad (2)$$

where  $X$  belongs to the neighborhood of the identity element  $I$  of  $G$ .

**2.2. Special Orthogonal Group.** The orthogonal group is expressed as follows:

$$O(n) = \{R \in \mathfrak{R}^{n \times n} | RR^T = I\}, \quad (3)$$

where  $\mathfrak{R}^{n \times n}$  denotes all the elements in the real matrix. Its Lie algebra represents all the elements in the antisymmetric matrix as follows:

$$\mathfrak{o}(n) = \{U | U \in \mathfrak{R}^{n \times n}, U^T = -U\}. \quad (4)$$

The special orthogonal group is as follows:

$$SO(n) = \{R \in O(n) | \det(R) = 1\}. \quad (5)$$

Its Lie algebra is  $\mathfrak{so}(n)$ , and

$$\mathfrak{so}(n) = \mathfrak{o}(n). \quad (6)$$

The pose variation of the rotating manipulator end effector is described by a  $3 \times 3$  rotation matrix  $R$ , which is an element in the special orthogonal group  $SO(3)$ . A series of motion poses of the manipulator corresponds to the several elements in  $SO(3)$ .

The only expression for the antisymmetric matrix  $U$  in  $\mathfrak{so}(3)$  is determined as follows:

$$U = \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix}, \quad (7)$$

where  $\omega = (\omega_x, \omega_y, \omega_z)^T \in \mathfrak{R}^3$  and  $\omega$  is the angular velocity of the manipulator end effector [24]. In particular, the basis of the Lie algebra  $\mathfrak{so}(3)$  of  $SO(3)$  is as follows:

$$\begin{aligned} U_1 &= \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}, U_2 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix}, \\ U_3 &= \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \end{aligned} \quad (8)$$

which satisfies

$$[U_1, U_2] = U_3, [U_2, U_3] = U_1, [U_3, U_1] = U_2, \quad (9)$$

where  $[\cdot, \cdot]$  is the Lie bracket. The Lie bracket of  $U_1, U_2 \in \mathfrak{so}(n)$  is defined as follows:

$$[U_1, U_2] = U_1 U_2 - U_2 U_1. \quad (10)$$

For any arbitrary  $U \in \mathfrak{so}(3)$ , the exponential map has a close form [24]:

$$\exp(U) = \begin{cases} I, & \theta = 0 \\ I + \frac{\sin\theta}{\theta} U + \frac{1 - \cos\theta}{\theta^2} U^2, & \theta \in (0, 2\pi) \end{cases}, \quad (11)$$

with  $\theta = \sqrt{\frac{1}{2} \text{tr}(U^T U)}$ . The corresponding log map  $SO(3) \rightarrow \mathfrak{so}(3)$  is as follows:

$$\log(R) = \begin{cases} 0, & \theta = 0 \\ \frac{\theta}{2\sin\theta} (R - R^T), & |\theta| \in (0, \pi) \end{cases}, \quad (12)$$

with  $R \in SO(3)$  and  $\text{tr}(R) = 2\cos\theta + 1$ .

As a connected compact Lie group, the special orthogonal group  $SO(n)$  is a complete manifold, and any two points are connected by a geodesic curve. The geodesic curve on  $SO(n)$  is represented as follows:

$$\gamma(t) = R \exp(tU), 0 \leq t \leq 1, \quad (13)$$

where  $\gamma(t)$  begins with  $R$  and takes  $RU$  as its direction. The geodesic curve connecting the two points  $R_1, R_2$  is expressed by:

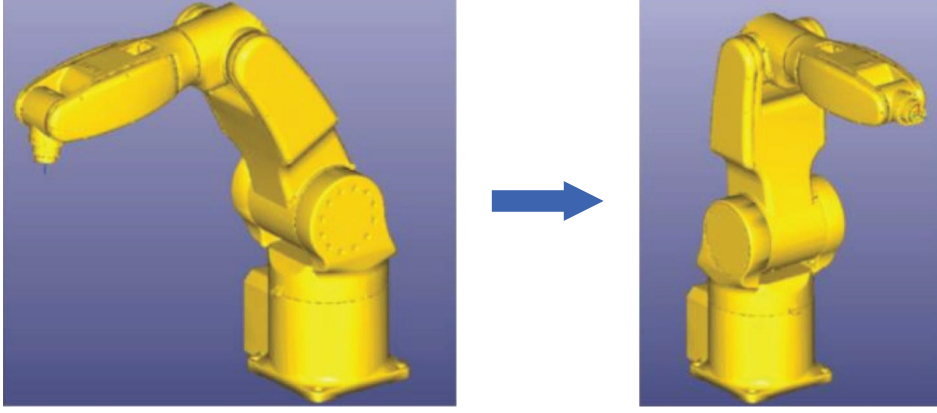


FIGURE 2: Manipulator moving from the initial pose to the target pose.

$$\gamma(t) = R_1(R_1^T R_2)^t = R_1 \exp(t \log(R_1^T R_2)), 0 \leq t \leq 1. \quad (14)$$

The geodesic distance between the two orthogonal matrices  $R_1, R_2$  is calculated by:

$$d(R_1, R_2) = \|\log(R_1^T R_2)\|. \quad (15)$$

### 3. Algorithm for Planning the Manipulator Trajectory

**3.1. Generation of Aerial Paths.** The basic description of trajectory planning is to move a manipulator from the initial position and pose  $\{T_s\}$  to the desired position and pose  $\{T_e\}$ . As shown in Figure 2, the tool coordinate system of the manipulator moves from the current value to the final target value. This process includes changes in the position and the pose of the tool. The detected points on the motion trajectory of the manipulator should be planned according to the motion period  $T$  of the manipulator and the external axis. Owing to the introduction of the automatic control system, the manipulator velocity at the starting stationary state is zero. Therefore, the detected points should be planned to make sure that the manipulator can arrive at the next detected point within a motion period. Violent acceleration or deceleration give rise to the parts' depreciation and the sudden vibration of the manipulator. Therefore, the manipulator movement should conform to the acceleration or deceleration movement law and the movement path should be smooth.

The velocity at the manipulator end is defined as a continuous function of time, which makes the displacement a function of time with a continuous first derivative. Figure 3 illustrates the relationship between the time and the velocity of the tool at the manipulator end on the whole planned path. From 0 to  $t_0$ , the manipulator moves with a constant acceleration  $a$ . When the maximum velocity  $V_{\max}$  of the manipulator is reached, it continues to move at a constant velocity. From  $t_1$  to  $t_2$ , the manipulator moves at a constant deceleration  $-a$ , where  $t_2 - t_1 = t_0$ ,  $V_{\max} = at_0$ . Using the movement velocity function of the manipulator:

$$v(t) = \begin{cases} at, & 0 \leq t < t_0 \\ V_{\max}, & t_0 \leq t < t_1 \\ -a(t - t_2), & t_1 \leq t < t_2 \end{cases}, \quad (16)$$

the displacement function of the manipulator is obtained by:

$$s(t) = \begin{cases} \frac{1}{2}at^2 & 0 \leq t < t_0 \\ \frac{1}{2}at_0^2 + V_{\max}(t - t_0) & t_0 \leq t < t_1 \\ \frac{1}{2}at_0^2 + V_{\max}(t_1 - t_0) + \frac{1}{2}a(t_1 - t_2)^2 - \frac{1}{2}a(t - t_2)^2, & t_1 \leq t < t_2 \end{cases}. \quad (17)$$

The above trajectory planned ensures the steady acceleration and deceleration of the manipulator end and prevents its rapid acceleration and deceleration in the specified motion period.

**3.2. Algorithm for Planning the Position and Pose of the Manipulator End.** First, we introduce how to convert a motion trajectory segment of the manipulator into a matrix in  $SO(3)$ .

The detected point is described as the coordinate system of the tool position and pose relative to the operating platform. The position and pose at each detected point are expressed as  $(x, y, z, \alpha, \beta, \gamma)$ . Here,  $x, y, z$  is the position of the manipulator end in the coordinate system of the operating platform, and  $\alpha, \beta, \gamma$  are the angles by which the coordinate system at the detected point turns relative to the coordinate system of the operating platform.

By using the  $X - Y - Z$  Euler angle coordinate system method, the rotation matrix  $A$  is expressed as follows:

$$\begin{pmatrix} c\beta c\gamma & -c\beta s\gamma & s\beta \\ s\alpha s\beta c\gamma + c\alpha\gamma & -s\alpha s\beta s\gamma + c\alpha\gamma & -s\alpha c\beta \\ -c\alpha s\beta c\gamma + s\alpha\gamma & c\alpha s\beta s\gamma + s\alpha\gamma & c\alpha c\beta \end{pmatrix}, \quad (18)$$

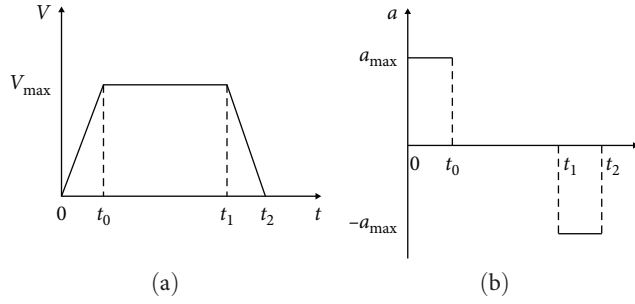


FIGURE 3: Velocity and acceleration of the manipulator end: (a) velocity and (b) acceleration.

where  $s\alpha$ ,  $s\beta$ ,  $s\gamma$  are the abbreviations of  $\alpha$ ,  $\beta$ ,  $\gamma$  sine values and  $c\alpha$ ,  $c\beta$ ,  $c\gamma$  are abbreviations of  $\alpha$ ,  $\beta$ ,  $\gamma$  cosine values.

In addition to equidistant points in the space, the motion velocity, acceleration  $a$  and motion period  $T$  of the manipulator end effector should also be planned in the trajectory planning process. The motion velocity should be smaller than the maximum rated velocity  $V_{\max}$  of the manipulator end, and the maximal motion velocity is generally fixed at 80% of the maximum rated velocity. In our experiments, the velocity is 100 mm/s, the acceleration is 100 mm/s<sup>2</sup>, and the motion interpolation period is 1 ms. If the motion velocity exceeds the maximum rated velocity of the manipulator end, the torque overload may occur to the manipulator in its moving process, resulting in abnormal sounds, increased wear, and even damage to the manipulator. The acceleration should also be lower than the maximum rated acceleration of the manipulator. The motion period should be set according to the German Beckhoff motion controller scan period of the integrated system.

The motion period of the manipulator is set to be the same as the scan period  $T$  (1 ms) of the automatic control system. Therefore, the distance between equidistant points should not be greater than the product of the maximum velocity  $V_{\max}$  and the period  $T$ . When the distance exceeds the product, the manipulator is unable to arrive at the next detected point from the current position within one period  $T$  even if it moves at the maximum velocity. In ultrasonic testing, the ultrasonic probe transmits and receives ultrasonic signals according to the position. Thus, a manipulator position deviation on the space trajectory will cause errors in the imaging results of ultrasonic testing. Due to the symmetry between the accelerated motion and decelerated motion of the manipulator, the distance and the time of the uniformly decelerated motion can be acquired in the same way. In ultrasonic testing, trajectory points refer to the position and pose of the probe held by the manipulator. In the experiments, the uniform acceleration, uniform velocity, and uniform deceleration motion trajectory are planned based on the probe detection speed of 100 mm/s, acceleration of 100 mm/s<sup>2</sup>, and motion interpolation motion period of 1 ms. According to the principle of trapezoidal acceleration and deceleration, the polynomial scanning trajectory is interpolated. The typical cubic curve interpolation data points are shown in Figure 4.

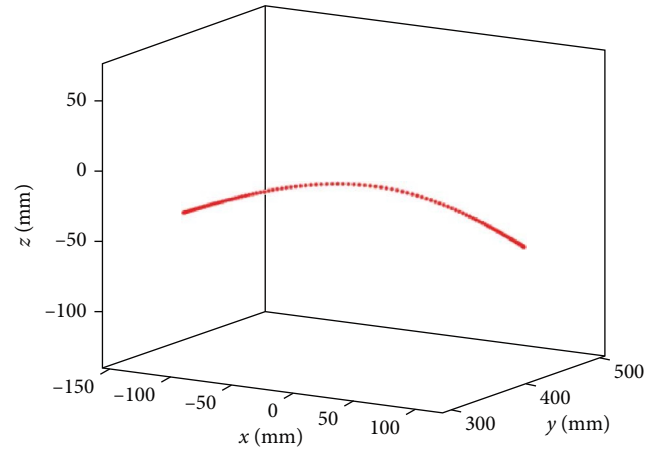


FIGURE 4: Interpolate data points.

The algorithm for planning the position and the pose at each detected point is developed as follows.

**3.2.1. Calculation of the Positions at Detected Points.** The coordinates of scattered points on the workpiece collected by the laser are fitted to acquire the cross-sectional curve of the workpiece surface. Let the number of the detected points on the cross-sectional curve be  $N + 1$ , and the detected point is represented as  $P_0, P_1, \dots, P_N$ . Their corresponding position coordinates and pose matrices are denoted by  $X_0, X_1, \dots, X_N$  and  $A_0, A_1, \dots, A_N$ , respectively. Assuming that the coordinates  $X_0$  and pose matrix  $A_0$  at the initial point  $P_0$  and the coordinates  $X_N$  and pose matrix  $A_N$  at the target point  $P_N$  are known, the acceleration  $a$  is adjusted so that the duration of accelerated, uniform, and decelerated motion is an integral multiple of the period  $T$ .

Due to the small distance between adjacent detected points, the arc length between two points approximates a straight-line segment.

If the coordinates  $X_k$  of the detected point  $P_k$  are known, the coordinates  $X_{k+1}$  of the detected point  $P_{k+1}$  can be obtained by using the cross-sectional curve equation and derivative function of the workpiece to be detected. The arc length between detected points  $P_{k-1}$  and  $P_k$  is denoted by  $s_k$ . As the manipulator end tool moves at a constant acceleration, uniformly and at a constant deceleration on the entire trajectory planned, the arc length  $s_k$  is symmetrical.



In fact, according to the displacement function (Equation (17)), we have

$$s_k = s(kT) - s((k-1)T) = \begin{cases} \frac{1}{2} aT^2(2k-1) & 0 \leq k < N_0 \\ V_{\max} T & N_0 \leq k < N_1 \\ \frac{1}{2} aT^2(2(N+1-k)-1) & N_1 \leq k < N \end{cases} \quad (19)$$

with  $N_0 = \frac{t_0}{T}$ ,  $N_1 = \frac{t_1}{T}$ ,  $N = \frac{t_2}{T}$ . The arc length  $s_k$  satisfies as follows:

$$s_k = s_{N+1-k}. \quad (20)$$

**3.2.2. Calculation of the Poses at Detected Points.** Given the pose matrices of the end effector of the manipulator at the starting and target points, using logarithmic mapping, their Lie algebras are obtained. By utilizing the proportional relationship of arc lengths between detected points, calculate the Lie algebra of all detected points. Finally, using exponential mapping, the pose matrices of all detected points can be obtained.

First, using the ratio relationship between the arc lengths of detected points, let

$$l_k = \frac{s_1 + s_2 + L + s_k}{s_1 + s_2 + L + s_N}, \quad k = 1, \dots, N. \quad (21)$$

Based on the linear property of the Lie algebra on the special orthogonal group, the Lie algebra at each detected point is described as follows:

$$U_k = U_0 + l_k(U_N - U_0), \quad k = 1, \dots, N-1, \quad (22)$$

where

$$U_0 = \log(A_0), \quad U_N = \log(A_N). \quad (23)$$

According to the exponential map from the Lie algebra to the Lie group, the pose matrix at each detected point is further gotten as follows:

$$A_k = \exp(U_k), \quad k = 1, \dots, N-1. \quad (24)$$

Thus, the pose matrix at each detected point between the start and target points is acquired (Figure 5).

Figure 6 shows the flowchart of Algorithm 1.

### 3.3. Algorithm Analysis

**3.3.1. The Rotation Equivalent Axes of the Poses at All Detected Points Are Located on the Same Plane.** The manipulator can make any pose by rotating a certain angle around an appropriate axis [25]. Specifically, the coordinate system  $\{B\}$  first coincides with a known coordinate system  $\{A\}$ . Then

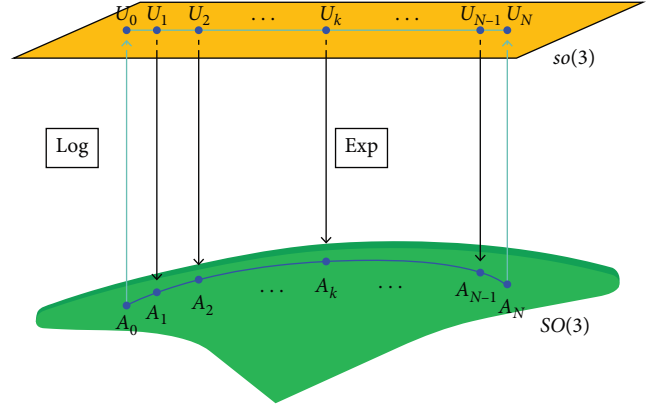


FIGURE 5: Algorithm for the pose matrix at each detected point.

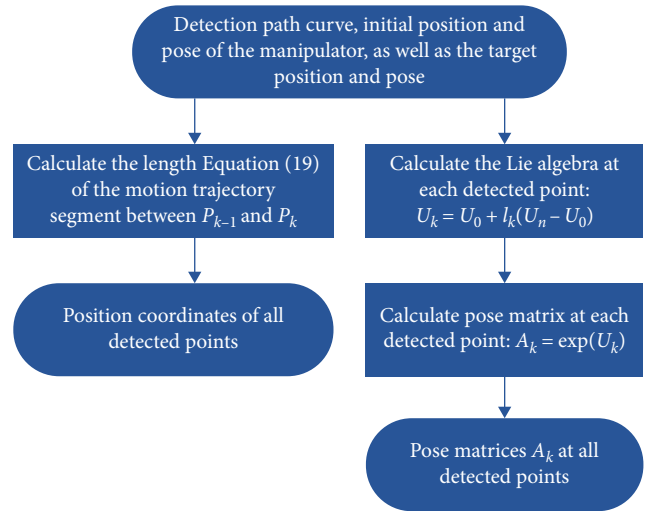


FIGURE 6: Flowchart of Algorithm 1.

**Input:** detection path curve, initial and target positions, and poses of the manipulator  $(x, y, z, \alpha, \beta, \gamma)_i$ .

**Output:** position coordinates and pose matrices at all detected points.

(1) According to the length Equation (19) of the motion trajectory segment between  $P_{k-1}$  and  $P_k$  as well as the curve equation of the detection trajectory, the position coordinates  $X_k$  ( $k = 1, \dots, N$ ) of each detected point  $P_k$  are calculated.

(2) Let the poses of the manipulator at the initial and target positions be  $A_0$  and  $A_N$ , respectively, and the Lie algebra at each detected point is expressed as

$$U_k = U_0 + l_k(U_N - U_0), \quad k = 1, \dots, N-1.$$

Then the pose matrix at each detected point can be calculated by

$$A_k = \exp(U_k), \quad k = 1, \dots, N-1.$$

Algorithm 1: Algorithm for the position and the pose matrix at each detected point.

based on the right-hand rule, the coordinate system rotates  $\theta$  angle around a unit vector  $\hat{K}$ .  $\hat{K}$  is called the equivalent axis of a finite rotation. Let the pose matrix be  $R$ , and  $\hat{K}$  is the unitized angular velocity:

$$\widehat{K} = \frac{\omega}{\theta} = \frac{1}{\theta} \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix}, \quad (25)$$

with

$$\theta = \arccos\left(\frac{\text{tr}(R) - 1}{2}\right). \quad (26)$$

The pose at each detected point in Algorithm 1 is acquired by calculating the ratio of the Lie algebra of the initial pose to that of the target pose. From the one-to-one mapping relationship between the Lie algebra (Equation (7)) and the equivalent axis  $\widehat{K}$ , it can be seen that the rotation equivalent axes of the poses at all detected points obtained by Algorithm 1 are located on the same plane.

**3.3.2. The Manipulator End Effector Moves at a Constant Angular Acceleration and Deceleration.** According to Equation (22), the angular velocity  $\omega_k$  of the manipulator end effector at the detected point  $P_k$  satisfies

$$\omega_k = \omega_0 + l_k(\omega_N - \omega_0), \quad k = 1, \dots, N - 1. \quad (27)$$

The angular acceleration  $a_k$  at the detected point  $P_k$  is approximated by the reverse difference of the angular velocity  $\omega_k$  as follows:

$$a_k = \frac{\omega_k - \omega_{k-1}}{T}. \quad (28)$$

Bringing Equation (22) into the above equation, we have

$$a_k = \begin{cases} \frac{2k-1}{2N}(\omega_N - \omega_0)a & 0 \leq k < N_0 \\ \frac{N_0}{N}(\omega_N - \omega_0)a & N_1 \leq k < N_2 \\ \frac{2(N+1-k)-1}{2N}(\omega_N - \omega_0)a, & N_2 \leq k < N. \end{cases} \quad (29)$$

Obviously, Equation (29) satisfies symmetry as follows:

$$a_k = a_{N+1-k}. \quad (30)$$

Hence, the manipulator end effector moves at a constant acceleration, uniformly and at a constant deceleration in the detection process.

#### 4. Numerical Simulations of Pose Planning

In the simulations, a Turin TKB070S manipulator is applied to read the motion trajectory, and a Gocator-2350D laser is used to retrieve the coordinates of discrete points on a cross-section of the aircraft engine blade, where the step of the scattered points was 0.5 mm.

First, the spatial position of each detected point is calculated by Algorithm 1.

The scattered points are fitted with a quadratic curve according to the curve shape. The motion period  $T$  of the manipulator adopts 1 ms and the acceleration  $a$  is set to 100 m/s<sup>2</sup>. The coordinates of the start and target points obtained by the laser are  $X_0$  and  $X_N$  ( $N = 11049$ ), respectively. The arc between two adjacent points approximates to a straight-line segment. The coordinates  $X_k$  of all detected points of the manipulator can be calculated by Equation (19).

Second, the pose matrices of the manipulator at all detected points are computed. In the experiment, the pose matrices at the start point  $P_0$  and target point  $P_N$  are as follows:

$$A_0 = \begin{pmatrix} -0.9407 & 0.2361 & 0.2434 \\ 0.2434 & 0.9699 & 0 \\ -0.2361 & 0.0593 & -0.9699 \end{pmatrix}, \quad (31)$$

and

$$A_N = \begin{pmatrix} 0.7822 & 0.4127 & 0.4667 \\ 0.4667 & 0.8844 & 0 \\ -0.4127 & 0.2178 & -0.8844 \end{pmatrix}, \quad (32)$$

respectively. Through log mapping, the Lie algebras of the pose matrices  $A_0$  and  $A_N$  are obtained as follows:

$$U_0 = \begin{pmatrix} 0 & -0.0439 & 2.8753 \\ 0.0439 & 0 & -0.3553 \\ -2.8753 & 0.3553 & 0 \end{pmatrix}, \quad (33)$$

and

$$U_N = \begin{pmatrix} 0 & -0.1587 & 2.5877 \\ 0.1587 & 0 & -0.6408 \\ -2.5877 & 0.6408 & 0 \end{pmatrix}, \quad (34)$$

respectively. The angular velocities of the pose matrices  $A_0$  and  $A_N$  are as follows:

$$\omega_1 = \begin{pmatrix} 0.3553 \\ 2.8753 \\ 0.0439 \end{pmatrix}, \omega_N = \begin{pmatrix} 0.6408 \\ 2.5877 \\ 0.1587 \end{pmatrix}, \quad (35)$$

respectively. The equivalent axes of the pose matrices  $A_0$  and  $A_N$  are as follows:

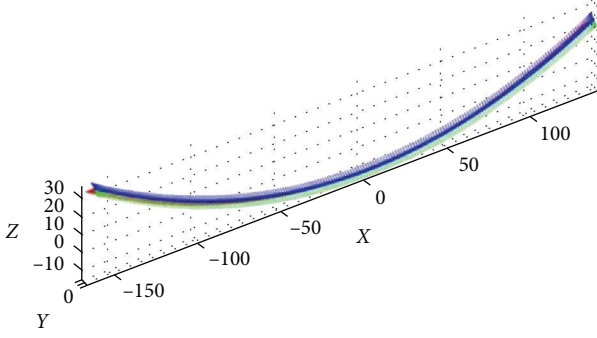


FIGURE 7: Pose at each detected point.

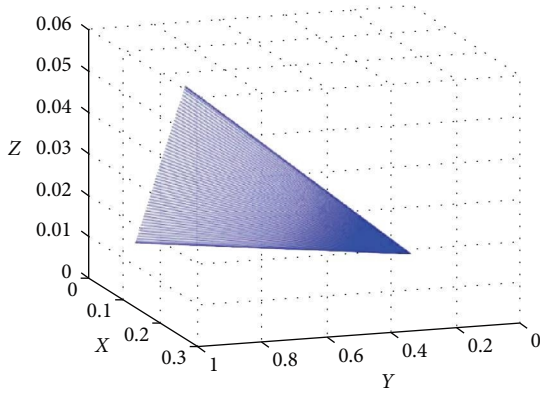


FIGURE 8: Rotation equivalent axis corresponding to each pose.

$$\hat{K}_1 = \begin{pmatrix} 0.1226 \\ 0.9923 \\ 0.0151 \end{pmatrix}, \hat{K}_N = \begin{pmatrix} 0.2399 \\ 0.9689 \\ 0.0594 \end{pmatrix}, \quad (36)$$

respectively.

To acquire the pose matrix  $A_k$  at each detected point on the curve, the Lie algebra of each point is solved by Equation (23) based on its linear property, and then the Lie algebra is exponentially mapped to the Lie group by Equation (24) (Figure 7). According to the one-to-one mapping relationship between the Lie algebra Equation (7) and the equivalent axis  $\hat{K}$  Equation (25), the rotation equivalent axes of pose matrices are obtained, which are located on the same plane (Figure 8).

Third, the desired angle of each axis of the manipulator effector is calculated by solving the inverse kinematics. By arranging the effector angles at all detected points in order of arrival time, the functional relationship of the axis variation of the manipulator with time is obtained. Figure 9 shows the change in the axis angle when the tandem-type six-axis manipulator walks along a trajectory segment. The changes in the manipulator axis and external axis with time can be observed directly.

However, it is impossible to determine whether the axis velocity changes suddenly or the angular velocity of the rotation axis exceeds the motor limit merely according to the displacement of an axis in a time period. Figure 10 presents

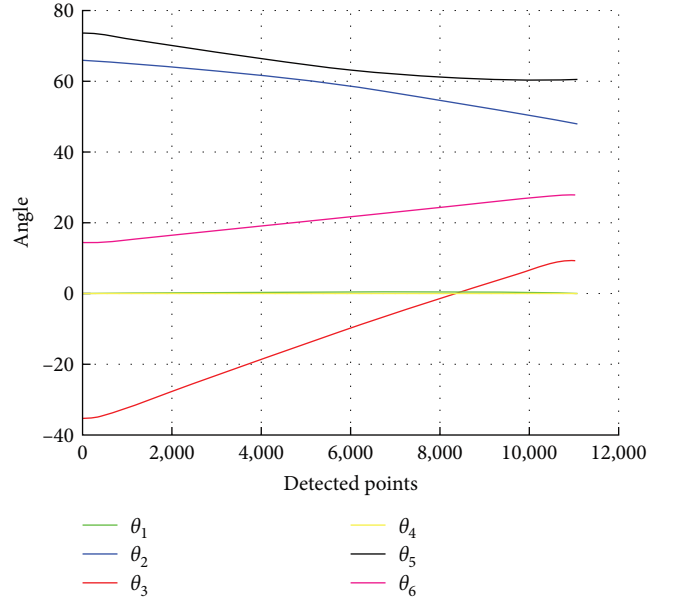


FIGURE 9: Angular displacement of each joint of the six-axis manipulator.

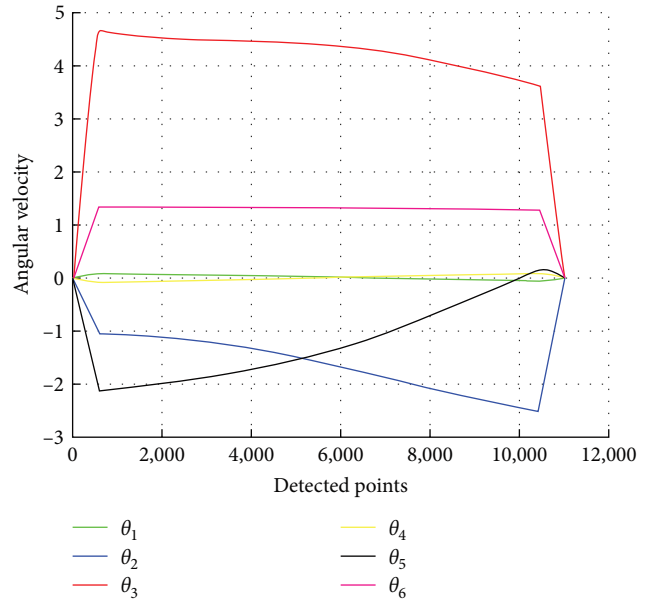


FIGURE 10: Angular velocity of each joint of the six-axis manipulator.

the functional relationship between the time and the axis angular velocity of the tandem-type six-axis manipulator. It can be seen that the function is continuous, which confirms that the axis variation is also continuous and no sudden velocity change occurs when the manipulator moves along this trajectory segment.

It can be seen that the function is continuous, which confirms that when the manipulator moves along this trajectory segment, the changes in the axis are also continuous and there is no sudden velocity change.

The calculation results of Algorithm 1 are compared with those of the classical Euler angle method and the quaternion



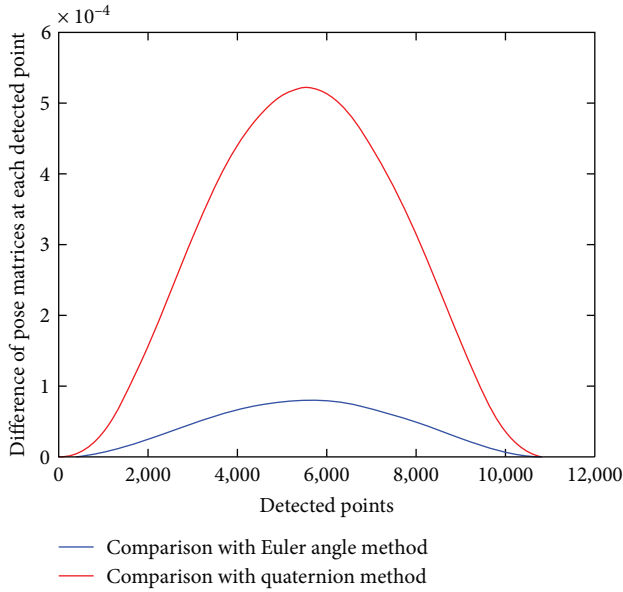


FIGURE 11: Comparison with classical algorithms.

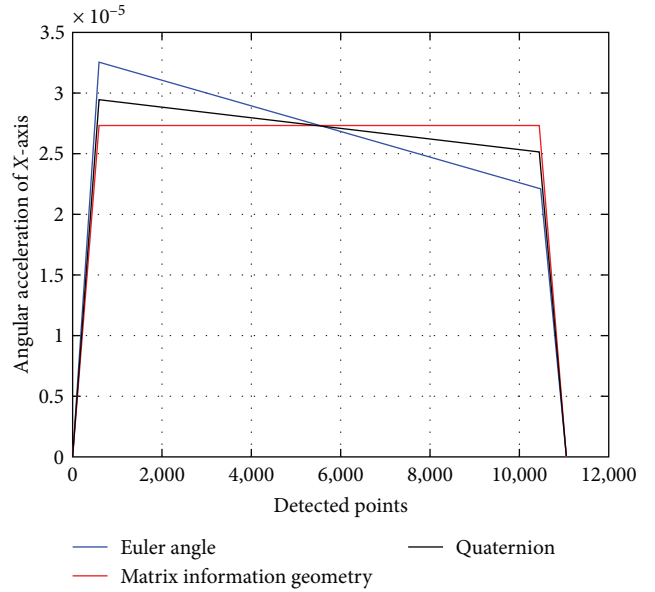


FIGURE 13: Angular acceleration of the end effector (X-axis).

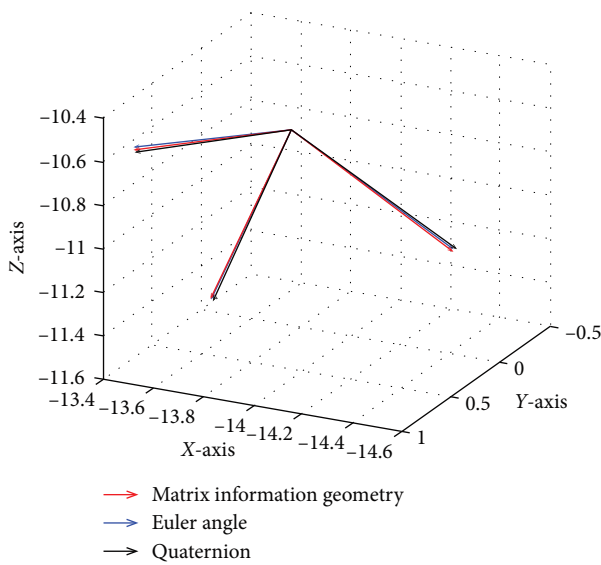


FIGURE 12: Poses obtained by three algorithms at the 5,500th detected point.

method. For this set of data, the geodesic distance between  $A_k$  and the pose matrices gotten by the Euler angle method or the quaternion method is calculated by Equation (15). As shown in Figure 11, the maximum error of the geodesic distance at each detected point does not exceed 0.025. It indicates that Algorithm 1 is consistent with the two classical methods. At the 5,500th detected point, Algorithm 1 and the two classical methods show the largest difference. Figure 12 compares the poses obtained by Algorithm 1, the Euler angle method, and the quaternion method at the 5,500th detected point. It can be seen that the difference in pose matrices obtained by the three algorithms is very small.

Finally, the angular accelerations of the manipulator end effector obtained by three algorithms are compared.

Figures 13–15 show the angular accelerations of the manipulator end effector relative to the X, Y, and Z axes in the detection process. It shows that the end effector moves at a constant acceleration, uniformly, and at a constant deceleration. This motion pattern ensures stable and smooth rotation of the manipulator end effector and prevents rapid acceleration.

To sum up, the numerical experiments verify the effectiveness of the trajectory point planning algorithm based on the matrix information geometry. In addition, this algorithm also has the following advantages: all the rotation equivalent axes of the pose matrices are located on the same plane. During the detection process, the end actuator moves at a constant angular acceleration and a constant angular deceleration, which helps the manipulator move more stably.

## 5. Experiment of Trajectory Planning

In numerical simulations, discrete point coordinates of a cross-section of an aircraft engine blade are retrieved using a Gocator-2350D laser. After curve fitting, the coordinates of 11,050 detected points are obtained through interpolation. For the given poses Equations (31) and (32) of the starting and ending points, the poses at each detected point are obtained using Algorithm 1, the Euler angle method, and the quaternion method, respectively.

In this section, based on the inverse kinematics solutions of three sets of pose matrices, the Turing TKB070S six-axis manipulator is operated separately. First, the equivalent axis of the attitude matrix obtained by Algorithm 1 is verified to be located in the same plane. Second, for the three algorithms, the actual data and numerical simulation data of the end effector angular acceleration are compared.

Specifically, the pose matrices obtained from Algorithm 1 are transmitted to the controller. After the manipulator executes according to instructions, the timer captures the 170 real-time pose matrices corresponding to the end effectors of

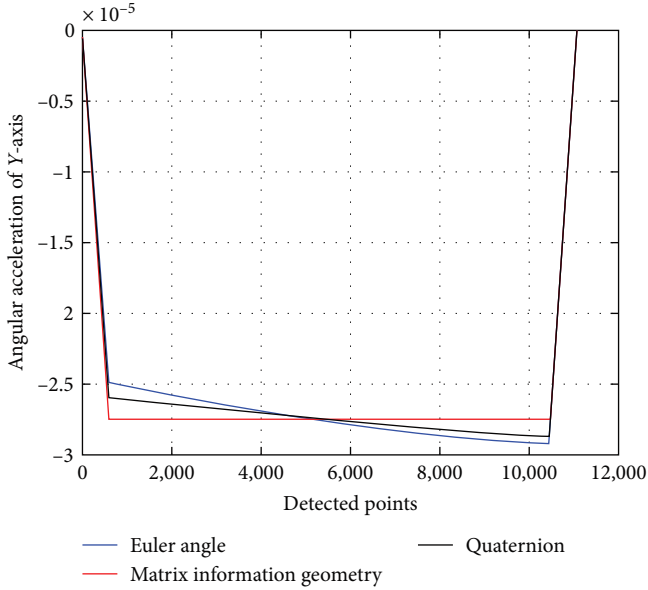


FIGURE 14: Angular acceleration of the end effector (Y-axis).

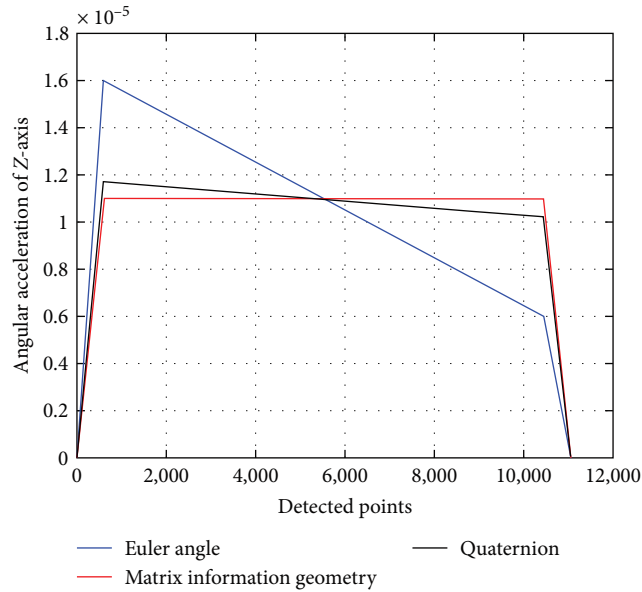


FIGURE 15: Angular acceleration of the end effector (Z-axis).

the manipulator. Then, 170 pose matrices are obtained by forward solutions and their equivalent axes are calculated from Equation (25). Figure 16 shows that the rotation equivalent axes of the 170 pose matrices of the manipulator end effector are located in the same plane, which is consistent with Figure 8.

Then, compare the experimental data and numerical simulations of the angular acceleration of the end effector of the manipulator obtained from the three algorithms. Figure 17(a) shows that the data of Algorithm 1 is distributed

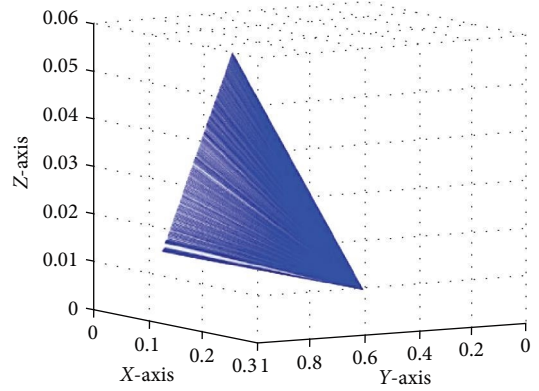


FIGURE 16: Rotation equivalent axis corresponding to each pose obtained from the experiment.

near the trapezoidal curve. Figures 17(b) and 17(c) illustrate that the data from the Euler angle algorithm and the quaternion algorithm are also consistent with the numerical simulation curves. Therefore, the three algorithms are feasible in the motion experiments of the manipulator and consistent with numerical simulations.

In summary, the experimental results are consistent with the numerical simulation results, indicating the controllability of the detected point planning algorithm based on matrix information geometry. During the test, the angular acceleration of the end effector is a uniform acceleration and deceleration movement, which ensures a more stable movement of the manipulator.

## 6. Conclusion

- (1) A manipulator detection trajectory planning algorithm is proposed based on the geometric structure of the special orthogonal group. The linear operation of the orthogonal matrix corresponding to the manipulator pose at each moment is unclosed. Thus, based on the mapping relationship between the special orthogonal group and its Lie algebra and the linear property of the Lie algebra, the manipulator pose at each detected point is acquired.
- (2) The trajectory planning algorithm based on the matrix information geometry achieves the same results as the typical Euler angle and quaternion transformation methods. The matrix information geometry algorithm does not have the problem of multiple solutions in the calculation process. Meanwhile, the rotation-equivalent axes of the pose matrices are located in the same plane. Moreover, the manipulator end effector moves with constant angular acceleration and deceleration in the detection process, enabling the effector to move stably along the given trajectory.
- (3) The experiment on the Turin TKB070S manipulator demonstrates that the manipulator can operate

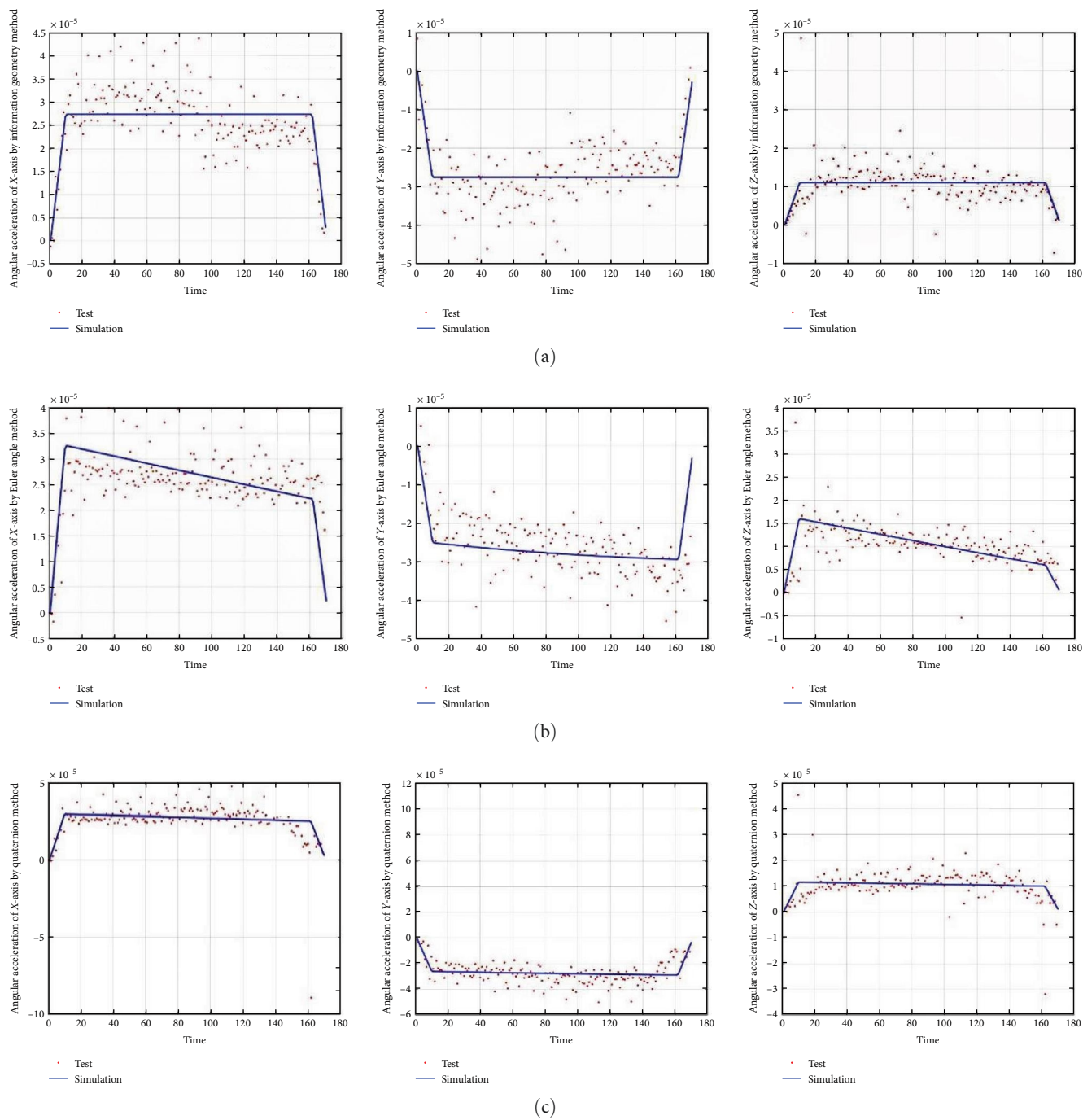


FIGURE 17: Comparison of test data and simulation of the angular acceleration of end effector: (a) matrix information geometry method, (b) Euler angle method, and (c) quaternion method.

smoothly and steadily according to the detected point position and pose planned.

### Data Availability

The simulations and experiment data used to support the findings of this study are available from the corresponding author upon request.

### Conflicts of Interest

The authors declare that there is no conflicts of interests regarding the publication of this article.

### Acknowledgments

The present research is supported by the National Natural Science Foundation of China (No. 61401058) and the Basic

Scientific Research Projects of Liaoning Provincial Education Department (No. JYTMS20230010).

## References

- [1] S. Cantero-Chinchilla, P. D. Wilcox, and A. J. Croxford, "Deep learning in automated ultrasonic NDE—developments, axioms and opportunities," *NDT & E International*, vol. 131, Article ID 102703, 2022.
- [2] A. N. Sinclair and R. Malkin, "Sensors for ultrasonic nondestructive testing (NDT) in harsh environments," *Sensors*, vol. 20, no. 2, Article ID 456, 2020.
- [3] H.-C. Wu, N. Gupta, and P. Mylavarapu, "Blind multiridge detection for automatic nondestructive testing using ultrasonic signals," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 53, no. 10, pp. 1902–1911, 2006.
- [4] C. Miki, N. Kazuhiro, S. Hiromi, and T. Minoru, "Performance evaluation test of automatic ultrasonic testing systems for welded joints of steel bridges," *Journal of Testing and Evaluation*, vol. 33, no. 4, pp. 246–257, 2005.
- [5] X. Zhao, P. Li, H. Guo, X. Duan, and B. Zhang, "Motion planning of curved surface part for automatic ultrasonic testing," *Journal of Mechanical Engineering*, vol. 58, no. 24, pp. 41–48, 2022.
- [6] B. Yuan, G. Shui, and Y. Wang, "Advances in research of nonlinear ultrasonic wave mixing detection technology in non-destructive evaluation," *Journal of Mechanical Engineering*, vol. 55, no. 16, pp. 33–46, 2019.
- [7] C. G. Xu, H. B. Wang, and D. G. Xiao, "Ultrasonic transmission detection techniques for large complex curved surface of composite material," *Fiber Composites*, vol. 30, no. 3, pp. 33–38, 2013.
- [8] C. G. Xu, H. M. Zhang, and C. Z. Guo, "Nondestructive test technology using robot manipulator," *Electro Mechanical Engineering*, vol. 33, no. 2, pp. 1–12, 2017.
- [9] X. B. Li, Z. N. Xu, H. W. Hu, and X. Zhou, "Probe's auto-alignment in curved components ultrasonic survey," *China Mechanical Engineering*, vol. 19, no. 11, Article ID 1269, 2008.
- [10] S. Zhu, S. Liu, X. Wang, and H. Wang, "Time-optimal and jerk-continuous trajectory planning algorithm for manipulators," *Journal of Mechanical Engineering*, vol. 46, no. 3, pp. 47–52, 2010.
- [11] X. Zhao, X. Wen, and X. Duan, "Recognition method for testing manipulator trajectory change-point," *Journal of Mechanical Engineering*, vol. 56, no. 18, pp. 15–21, 2020.
- [12] Y. Yang, X. Zhao, Q. Sun, J. Zhang, and B. Li, "Trajectory optimization of manipulator for minimum working time based on multi-body dynamic characters," *Journal of Mechanical Engineering*, vol. 50, no. 7, pp. 8–14, 2014.
- [13] J. Mei, J. Zang, Z. Qiao, S. Liu, and T. Song, "Trajectory planning of 3-DOF delta parallel manipulator," *Journal of Mechanical Engineering*, vol. 52, no. 19, pp. 9–17, 2016.
- [14] F. Nielsen and R. Bhatia, *Matrix Information Geometry*, Springer, Berlin, Germany, 2013.
- [15] X. Hua, Y. Ono, L. Peng, Y. Cheng, and H. Wang, "Target detection within nonhomogeneous clutter via total Bregman divergence-based matrix information geometry detectors," *IEEE Transactions on Signal Processing*, vol. 69, pp. 4326–4340, 2021.
- [16] S. Feng, X. Hua, and X. Zhu, "Matrix information geometry for spectral-based SPD matrix signal detection with dimensionality reduction," *Entropy*, vol. 22, no. 9, Article ID 914, 2020.
- [17] W. Chen and S. Chena, "Application of the non-local log-Euclidean mean to radar target detection in nonhomogeneous sea clutter," *IEEE Access*, vol. 99, pp. 36043–36054, 2019.
- [18] M. L. Kevin and C. P. Frank, *Modern Robotics: Mechanics, Planning, and Control*, Cambridge University Press, Cambridge, 2017.
- [19] Z. Bai, Y. Shen, and Y. Shui, *Riemannian Geometry Preliminary*, Higher Education Press, Beijing, 2004.
- [20] A. Cont, S. Dubnov, and G. Assayag, "On the information geometry of audio streams with applications to similarity computing," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 837–846, 2011.
- [21] M. L. Curtis, *Matrix Groups*, Universitext, Springer, New York, 2 edition, 1984.
- [22] S. Aminikhanghahi and D. J. Cook, "A survey of methods for time series change point detection," *Knowledge and Information Systems*, vol. 51, no. 2, pp. 339–367, 2017.
- [23] A. Kirillov, *An Introduction to Lie Groups and Lie Algebras*, Cambridge University Press, 2010.
- [24] H. F. Sun, Z. N. Zhang, L. Y. Peng, and X. M. Duan, *An Elementary Introduction to Information Geometry*, Science Press, Beijing, 2016.
- [25] J. J. Craig, Y. Chao, and W. Wei, *Introduction to Robotics*, Mechanical Industry Press, Beijing, 2006.