International Core Journal of Engineering Volume 11 Issue 12, 2025
ISSN: 2414-1895 DOI: 10.6919/ICJE.202512_11(12).0023

Application of Improved B-Spline in Trajectory Planning for Cleaning Robots

Jie Li, Liying Deng*

School of Chemical Equipment, Shenyang University of Technology, Liaoyang 111000, China

Abstract

During the metallurgical production process, the inner walls of steel ingot molds are prone to the accumulation of oxides, residues, and other attachments, which severely affect the service life of the molds and the surface quality of the ingots. To address the insufficient smoothness in the trajectory planning of existing cleaning robots, an improved B-spline interpolation method is proposed and applied to the trajectory planning of a gantry-type cleaning robot. Based on the traditional B-spline, the proposed method introduces auxiliary control points to achieve smoother interpolation in terms of velocity and acceleration, thereby reducing mechanical impact and ensuring uniform cleaning coverage. A simulation of a plum blossom–shaped cross-section was conducted on the MATLAB platform, and the results were compared with those obtained from linear interpolation and conventional B-spline methods. The results demonstrate that the improved B-spline effectively reduces trajectory error and acceleration peaks, resulting in smoother robot motion and meeting the requirements of layered cleaning processes.

Keywords

Steel Ingot Mold; Cleaning Manipulator; Improved B-spline Interpolation; Trajectory Planning.

1. Introduction

In the modern steel production industrial system, steel ingot molds serve as key carriers for ingot forming, and their inner wall cleanliness is directly related to ingot quality and production efficiency. Therefore, mold cleaning is regarded as a crucial link in the metallurgical production process. Most current cleaning methods rely on manual labor, with problems such as low efficiency and unstable cleaning performance. However, existing automated cleaning solutions, especially trajectory planning technologies, still encounter severe challenges when coping with complex curved surfaces of mold inner walls. Although traditional trajectory planning methods are simple in calculation, they lack sufficient smoothness and are prone to sudden changes in velocity and acceleration, which reduces cleaning effects, increases mechanical vibration, and causes damage to mold surfaces. Therefore, how to ensure the cleaning brush accurately tracks preset path points while realizing smooth and stable transition of motion among various axes has become an urgent core technical problem to be solved.

In the field of robot trajectory planning, extensive research has been conducted by numerous scholars, and the methods can be mainly classified into two categories: joint space planning and Cartesian space planning[1].Cheng Huan[2]proposed a 3-5-3 piecewise polynomial interpolation method to plan the trajectory of the robot's working arm on the section to be sprayed, ensuring that the working arm maintains continuous acceleration during the spraying process. Although polynomial methods can guarantee that the trajectory passes through preset points strictly, sudden changes in curvature and acceleration are prone to occur under complex curves, leading to unsmooth motion of the

ISSN: 2414-1895 DOI: 10.6919/ICJE.202512_11(12).0023

robot.Wang Yuming[3]combined the adaptive S-curve with B-spline curves and utilized an adaptive genetic algorithm to optimize complex trajectories, by which the vibration problem of Pecvd equipment during transportation was effectively improved.Kong Qingbo[4]proposed a method combining quintic polynomial interpolation with B-spline interpolation, by which the trajectory smoothness was effectively improved and sudden changes were eliminated, though the computational load was increased.Yang Fu[5]combined B-spline curves with flight constraints to optimize UAV trajectories, by which the smoothness and dynamic feasibility were improved. Zhu Yongguo[6]proposed a trajectory planning method for grinding robots based on multi-objective collaborative optimization and quintic B-spline interpolation algorithm, through which the motion stability was significantly enhanced. Hu Shengsun[7]addressed the problem of large deviation between the theoretical trajectory and the actual trajectory in the trajectory teaching of J-shaped groove welding robots, and designed a B-spline approximation algorithm for the equidistant line of the intersecting line trajectory. A seventh-order B-spline curve was constructed in Reference[8],and the kinematic constraints of the robot were transformed into constraints on the control vertices of the B-spline, realizing multi-objective trajectory optimization. Lu Zhiguo[9]combined B-splines with the Whale Optimization Algorithm, by which the operation efficiency and smoothness of joint space trajectories were improved.These studies indicate that B-spline and its improved methods have significant advantages in robot trajectory planning.

Nevertheless, existing studies have mostly focused on traditional serial industrial robots for typical scenarios like welding and grinding, with limited research on truss manipulators' trajectory planning in complex curved surface cleaning. Notably, in cleaning tasks, achieving precise trajectory fitting to mold inner walls while ensuring velocity/acceleration continuity and smoothness requires further study. To address this, an improved method is proposed herein by adding auxiliary points beside control points, enabling the trajectory to strictly pass through points with good smoothness and locality. Based on this, trajectory planning and simulation for truss cleaning manipulators under plum-blossom-shaped mold sections are conducted. This method effectively enhances geometric accuracy and motion smoothness, offering a feasible technical path for automated steel ingot mold cleaning.

2. Mathematical Model and Cleaning Process Planning of the Manipulator for Inner Wall Cleaning of Steel Ingot Molds

2.1 Mathematical Model of the Cleaning Manipulator

Fig. 1 Physical Diagram of the Cleaning Manipulator

The inner wall cleaning task of steel ingot molds requires the manipulator to possess a large working space and high trajectory smoothness, so as to ensure that the wire brush contacts the inner wall uniformly without missing any area or causing excessive grinding. The cleaning manipulator studied

ISSN: 2414-1895 DOI: 10.6919/ICJE.202512_11(12).0023

in this paper adopts a gantry structure, which is equipped with 3 linear degrees of freedom (X, Y, Z) and 1 rotational degree of freedom (for the end wire brush). The maximum strokes are 2000 mm in the X-direction, 1800 mm in the Y-direction, and 2500 mm in the Z-direction, which can cover the entire space of typical steel ingot molds.

The structural model of the cleaning robot body is shown in Fig. 1. The x-axis, y-axis, z-axis, and end effector are respectively driven by 4 servo motors to perform movements. Pulses are sent by the PLC motion control module to the servo drivers, thereby achieving precise control of the manipulator.

A coordinate system for the cleaning manipulator is established, as shown in Fig. 2. The improved D-H parameters of the robot are presented in Table 1.

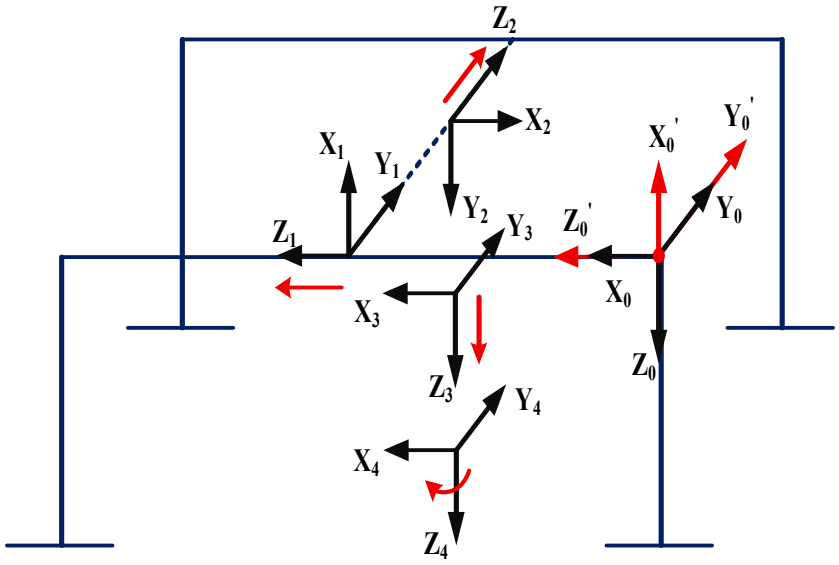

Fig. 2 Equivalent MD-H Parameter Model of the Cleaning Manipulator

Table 1. Parameters of the Robot's MD-H Model

Link	α_{i-1} (°)	a_{i-1} (mm)	d_i (mm)	θ_i (°)	Joint Variable	Variable Range
1	0	0	d_1+300	0	d_1	0~2000mm
2	-90	0	d_2+200	90	d_2	0~+1800mm
3	-90	0	d_3+400	180	d_3	0~+2500mm
4	0	0	0	0	θ_4	-180°~+180°

Based on the above parameters, a manipulator model was established using MATLAB Robotics Toolbox, as shown in Fig. 3. This model can not only intuitively display the structural characteristics of the XYZ linear axes and the rotating end effector of the cleaning manipulator but also provide a visual verification platform for subsequent trajectory planning. Through this kinematic model, the desired path of the mold inner wall can be mapped to the joint variable space, thereby serving as the input for the improved B-spline interpolation method to realize the optimized generation of the end trajectory.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202512_11(12).0023

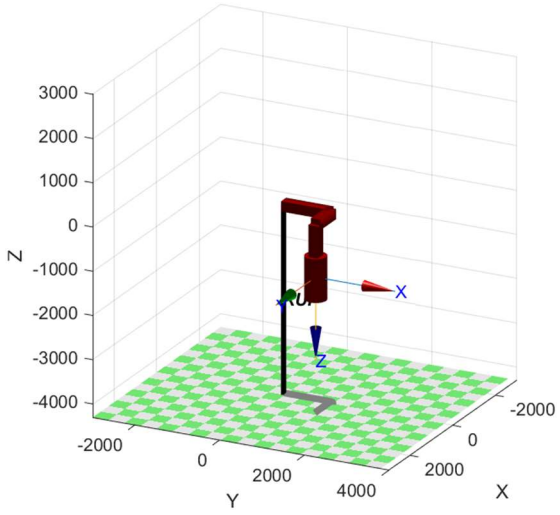

Fig. 3 Model of the Cleaning Manipulator

2.2 Cleaning Process Planning

As a key load-bearing structure in the steel making process, the inner wall of the steel ingot mold is often adhered with substantial residues due to erosion by high-temperature molten metal. If not cleaned promptly, it will impair the quality of steel ingots and even shorten the service life of the mold. Particularly for plum-blossom-shaped steel ingot molds, their cross-sections exhibit a symmetrical arc contour with six or eight petals, featuring distinct concave geometric characteristics. Such a structure imposes special requirements on the path planning of the cleaning manipulator: it is necessary to achieve full coverage of each concave petal area, while ensuring smooth movement without excessive abrupt changes in acceleration.

To address the cleaning challenges of the plum-blossom-shaped inner cavity, a truss structure-based cleaning manipulator is adopted in this study, with full-range cleaning of the mold inner wall achieved through the coordinated movement of three axes (X, Y, Z). The specific cleaning path strategy is as follows:

The plum-blossom-shaped steel ingot mold can be regarded as a cylindrical structure formed by stacking a series of horizontal cross-sections along the Z-axis. The steel ingot mold is placed perpendicular to the ground. Firstly, it is processed in layers along the height direction, i.e., divided into several horizontal plum-blossom cross-sections at fixed Z-axis heights, as shown in Fig. 4. Each cross-section is treated as an independent cleaning unit. After the cleaning brush head is positioned at a specific height, the manipulator completes the cleaning of the inner wall contour of this layer within the XOY plane, and then steps down along the Z-axis to the next layer, where the cleaning task is executed sequentially.

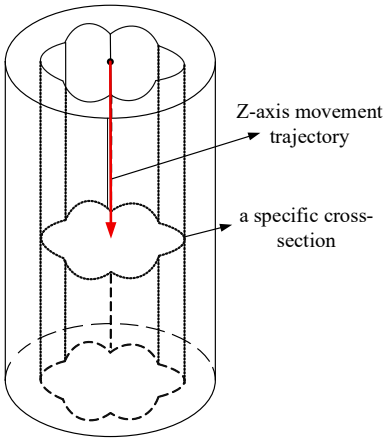

Fig. 4 Division of Cleaning Cross-Sections

ISSN: 2414-1895 DOI: 10.6919/ICJE.202512_11(12).0023

Within each horizontal cross-section, as shown in Fig. 5, a polar coordinate equation is employed to mathematically describe the contour based on the geometric characteristics of the plum-blossom shape, and a continuous, smooth 2D cleaning path is generated. The distribution density of path points is optimized according to the coverage range of the cleaning wire brush, ensuring full-range cleaning without repetition. The cleaning process starts from the top of the mold; after the manipulator completes the cleaning task for each horizontal cross-section in sequence, it moves downward gradually until the entire inner cavity of the mold is covered. This "layered-circulating" 3D cleaning strategy not only adapts to the special structure of the plum-blossom-shaped mold but also ensures the systematicness and integrity of the cleaning operation.

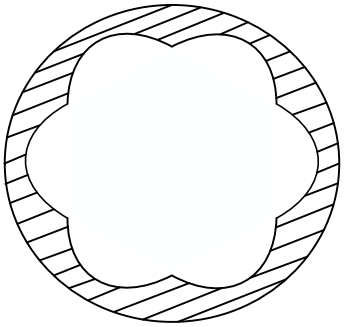

Fig. 5 Cross-Section Shape at a Specific Height

3. Application of Improved B-Spline Curves

3.1 Principle of B-Spline Curve Interpolation

In traditional linear interpolation methods, the trajectory is often formed by splicing multiple straight line segments, which tends to suffer from issues such as discontinuous velocity and unsmooth corners[10].For the steel ingot mold cleaning manipulator, its end wire brush needs to move continuously along the complex curved surface of the mold. If there are abrupt changes in the trajectory, it will not only cause vibration and impact of the manipulator, resulting in unstable cleaning effects, but also may lead to secondary wear on the mold surface. Therefore, it is particularly important to adopt an interpolation method capable of generating continuous and smooth trajectories.

The mathematical form of a B-spline curve can be expressed as the weighted sum of control points and basis functions:

$$C(t)=\sum_{i=0}^{n} N_{i,k}(t)P_i \quad t\in[t_k,t_{n+1}] \tag{1}$$

In this expression, $N_{i,k}(t)$ is defined as the B-spline basis function of degree (k−1), while P_i is taken as the control point, which indicates the coordinates of the i-th waypoint in the Cartesian space.

Although higher-order B-splines offer greater continuity, they incur higher computational cost and increased sensitivity to control points, often leading to undesired oscillations.In engineering applications, the cubic B-spline achieves a good balance between computational efficiency and smoothness, ensuring continuity of position, velocity, and acceleration. It is therefore adopted as the basic interpolation method for trajectory generation, upon which an improved strategy is introduced to further enhance accuracy and smoothness.Based on comprehensive considerations, the position–time sequence points are set as control points. A cubic uniform B-spline is defined by four control points, and its basis functions under equally spaced knots are given as [11]:

ISSN: 2414-1895 DOI: 10.6919/ICJE.202512_11(12).0023

$$\begin{cases} N_0(t)=\dfrac{(1-t)^3}{6} \\ N_1(t)=\dfrac{(3t^3-6t^2+3t+1)}{6} \\ N_2(t)=\dfrac{(-3t^3+3t^2+3t+1)}{6} \\ N_3(t)=\dfrac{t^3}{6} \end{cases} \tag{2}$$

In this expression,t is the local variable, $0 \le t \le 1$.

3.2 Improved B-spline Interpolation Method

Although B-splines offer advantages such as smoothness and local adjustability in robotic trajectory generation, certain limitations remain in practical applications. For the ingot mold cleaning manipulator, conventional B-splines often fail to ensure that the curve strictly passes through predefined points, resulting in incomplete cleaning in local areas. Moreover, abrupt velocity changes near target positions can affect the contact stability between the brush and mold surface, thereby reducing cleaning efficiency. To address these issues, an improved B-spline interpolation method is introduced, in which auxiliary control points are added to enhance both path smoothness and accuracy.

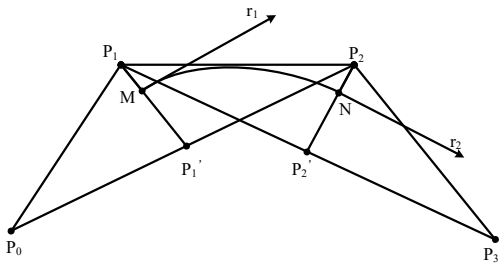

Fig. 6 Cubic B-spline curve

As shown in Fig. 6 , when generating a curve using the cubic uniform B-spline interpolation method, each segment is determined by four control points. Let P_0, P_1, P_2,and P_3 be the control points. The starting point of the segment lies on the midpoint M of the line $P_1P_1^{'}$ formed by the triangle $P_0P_1P_2$, where $P_1M = 1/3P_1P_1^{'}$.Similarly, the endpoint lies on the midpoint N of the line $P_2P_2^{'}$ of the triangle $P_1P_2P_3$, satisfying $P_2N = 1/3P_2P_2^{'}$.The cubic B-spline curve constructed from MNMNMN does not strictly pass through the control points P_0, P_1, P_2,and P_3; instead, it is influenced by their relative positions, with local shape adjustment achieved by varying individual control points while the overall curve continuity remains unchanged. Each curve segment connects smoothly at its endpoints, ensuring global smoothness. By linking these segments successively, a continuous cubic B-spline curve is obtained. Therefore, the cubic B-spline trajectory is determined by the control points but does not pass through them exactly; rather, it maintains a certain spacing, which varies slightly with the adjustment of control point positions.

As the control point P_1 gradually approaches $P_1^{'}$, point M also moves closer to $P_1^{'}$. Similarly, when P_2 approaches $P_2^{'}$, the corresponding point N shifts accordingly. With these variations, the start and end points of each curve segment progressively approach the bases of the triangles $\triangle P_0P_1P_2$ and triangle $\triangle P_1P_2P_3$. When points P_1 and $P_1^{'}$, as well as P_2 and $P_2^{'}$, coincide, the start and end points of the curve coincide with these control points.It can thus be observed that when three equally spaced

ISSN: 2414-1895 DOI: 10.6919/ICJE.202512_11(12).0023

control points lie on a straight line, the B-spline curve segment passes directly through the middle control point. This property reflects the inherent geometric characteristics of cubic B-splines. To achieve precise interpolation, the control points must lie on the sides of the corresponding triangles, with a spacing ratio of 1:3 between the edge and its midpoint. When the chord length tends to zero, meaning the three vertices are collinear, the curve passes through the corresponding point. This indicates that if the curve is expected to pass precisely through all control points, auxiliary points must be added on both sides of each key position to locally modify the curve shape and maintain overall smoothness.

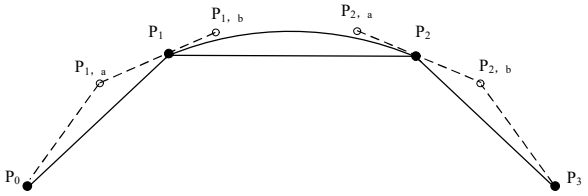

Fig. 7 Improved B-spline curve

In the improved B-spline interpolation method, two auxiliary points are added on both sides of each control point to ensure that the curve passes through the original reference points.As shown in Figure 7, two auxiliary points $P_{1,a}$ and $P_{1,b}$ are introduced on both sides of control point P_1, making points P_1, $P_{1,\ a}$, and $P_{1,\ b}$ collinear and parallel to P_0P_2. The distances satisfy $P_{1,\mathrm{a}}P_1 = P_1P_{1,\mathrm{b}} = \mathrm{h}P_0P_2$, among them, the proportional coefficient is $h = 1/6$. Similarly, for control point P_2, two auxiliary points $P_{2,a}$ and $P_{2,b}$ are added on both sides, following the same geometric relationships.

Based on the geometric characteristics of B-spline curves, if two auxiliary control points are additionally introduced at P_0 and P_3, the curve can be made to pass through the corresponding endpoints. Consequently, by taking P_0, $P_{1,\mathrm{a}}$, P_1, $P_{1,b}$, $P_{2,a}$, P_2, $P_{2,\mathrm{b}}$, and P_3 as control points, three cubic uniform B-spline segments can be constructed, ensuring that the generated trajectory precisely passes through points P_1 and P_2. Furthermore, to guarantee that the entire trajectory passes through all specified points, two auxiliary points are added on both sides of each original control point. For n given control points, this process ultimately yields $3\mathrm{n}-3$ curve segments.

The cubic B-spline curve generated by this method passes sequentially through all the original control points, and the calculation formula for the newly added auxiliary points is as follows:

$$\begin{aligned} P_{\mathrm{ia}} &= P_{\mathrm{i}} - \mathrm{h}\left(P_{\mathrm{i+1}} - P_{\mathrm{i-1}}\right) \\ P_{\mathrm{ib}} &= P_{\mathrm{i}} + \mathrm{h}\left(P_{\mathrm{i+1}} - P_{\mathrm{i-1}}\right) \end{aligned} \tag{3}$$

From Eqs. (2) and (3), the equation of the improved cubic B-spline curve can be derived as follows:

$$r_{3i+1}(t) = \frac{1}{6}\begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix}\begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix}\begin{bmatrix} P_{i,\mathrm{a}} \\ P_i \\ P_{i,\mathrm{b}} \\ P_{i+1,\mathrm{a}} \end{bmatrix} \tag{4}$$

ISSN: 2414-1895 DOI: 10.6919/ICJE.202512_11(12).0023

$$r_{3i+2}(t)=\frac{1}{6}\begin{bmatrix}t^3 & t^2 & t & 1\end{bmatrix}\begin{bmatrix}-1 & 3 & -3 & 1\\ 3 & -6 & 3 & 0\\ -3 & 0 & 3 & 0\\ 1 & 4 & 1 & 0\end{bmatrix}\begin{bmatrix}P_i\\ P_{i,\mathrm{b}}\\ P_{i+1,\mathrm{a}}\\ P_{i+1}\end{bmatrix} \tag{5}$$

$$r_{3i+3}(t)=\frac{1}{6}\begin{bmatrix}t^3 & t^2 & t & 1\end{bmatrix}\begin{bmatrix}-1 & 3 & -3 & 1\\ 3 & -6 & 3 & 0\\ -3 & 0 & 3 & 0\\ 1 & 4 & 1 & 0\end{bmatrix}\begin{bmatrix}P_{i,\mathrm{b}}\\ P_{i+1,\mathrm{a}}\\ P_{i+1}\\ P_{i+1,\mathrm{b}}\end{bmatrix} \tag{6}$$

In the formula, $i=0,1,\cdots,n-1$

All auxiliary control points can be derived from the original control points.Therefore,by substituting Eq. (3) into Eqs. (4), (5), and (6), the equation of the improved cubic B-spline curve represented by $P_0, P_1, P_2, \ldots P_{\mathrm{n}}$ can be obtained as follows:

$$r_{3i+1}(t)=\frac{1}{6}\begin{bmatrix}t^3 & t^2 & t & 1\end{bmatrix}\begin{bmatrix}1/3 & -5/6 & 2/3 & -1/6\\ 0 & 0 & 0 & 0\\ -1 & 0 & 1 & 0\\ 0 & 6 & 0 & 0\end{bmatrix}\begin{bmatrix}P_i\\ P_{i+1}\\ P_{i+2}\\ P_{i+3}\end{bmatrix} \tag{7}$$

$$r_{3i+2}(t)=\frac{1}{6}\begin{bmatrix}t^3 & t^2 & t & 1\end{bmatrix}\begin{bmatrix}-1/2 & 3/2 & -3/2 & 1/2\\ 1 & -5/2 & 2 & -1/2\\ 0 & -5/2 & 3 & -1/2\\ -2/3 & 31/6 & 5/3 & -1/6\end{bmatrix}\begin{bmatrix}P_i\\ P_{i+1}\\ P_{i+2}\\ P_{i+3}\end{bmatrix} \tag{8}$$

$$r_{3i+3}(t)=\frac{1}{6}\begin{bmatrix}t^3 & t^2 & t & 1\end{bmatrix}\begin{bmatrix}1/6 & -2/3 & 5/6 & -1/3\\ -1/2 & 2 & -5/2 & 1\\ 1/2 & -3 & 5/2 & 0\\ -1/6 & 1/6 & 31/6 & -2/3\end{bmatrix}\begin{bmatrix}P_i\\ P_{i+1}\\ P_{i+2}\\ P_{i+3}\end{bmatrix} \tag{9}$$

Note:In Eqs. (7)~(9), the parameter range does not include 0 or $n-1$. This is because the two additional curve segments are formed using auxiliary points that are not computed directly by the original formula but are determined according to the condition in Eq. (2).

From the derivation of Eqs. (7)~(9), a globally continuous and smooth B-spline curve can be obtained. The improved curve-fitting method is still established within the framework of traditional B-spline basis functions. Therefore, the generated curve not only preserves local control, continuity, and convex-hull properties, but also accurately passes through all control points, achieving higher fitting precision.

4. Analysis of Constraint Conditions

4.1 Analysis of Velocity and Acceleration of the Improved B-splin Trajectory in Cartesian Space

The three-axis gantry manipulator is composed of three translational degrees of freedom, each independently driven. The spatial trajectory of the end-effector is expressed in the Cartesian coordinate system as a position–time sequence (X, T), where X =(x, y, z) denotes the Cartesian

ISSN: 2414-1895 DOI: 10.6919/ICJE.202512_11(12).0023

position of the manipulator end. To distinguish it from the local parameter of the B-spline curve, the time variable is denoted by T. Based on this definition, the velocity and acceleration expressions of the improved B-spline curve with respect to position (coordinates) and time in Cartesian space are derived in the following.

In Cartesian space, the trajectory function in each dimension can be expressed as:

$$x(T),\ y(T),\ z(T)\ ,T=T(t),t\in[0,1] \tag{10}$$

In the formula, t is the local parameter of the interpolation segment, the spatial position components are given by the spline functions, and the time variable $T=T(t)$ represents the actual physical time corresponding to that curve segment.

Except for the terminal B-spline segments, each intermediate segment is determined by four consecutive points $P_i(x_i, y_i, z_i, T_i)$, $P_{i+1}(x_{i+1}, y_{i+1}, z_{i+1}, T_{i+1})$, $P_{i+2}(x_{i+2}, y_{i+2}, z_{i+2}, T_{i+2})$ and $P_{i+3}(x_{i+3}, y_{i+3}, z_{i+3}, T_{i+3})$. By differentiating the B-spline basis functions, the first-order derivatives of the Cartesian coordinate components (e.g., x, with y and z treated similarly) with respect to the local curve parameter ttt, and with respect to time T, can be obtained.

For the x-coordinate component, it is given that

$$\dot{x}_{i+1}=x_{i+2}\frac{2t^2+1}{6}+x_i\frac{t^2-1}{6}-x_{i+1}\frac{5t^2}{12}-x_{i+3}\frac{t^2}{12} \tag{11}$$

Similarly, the first-order derivatives of the y-coordinate and z-coordinate components, denoted as $\dot{y}_{i+1}$ and $\dot{z}_{i+1}$, can be obtained in the same manner.

For the time variable T, the first-order derivative with respect to t is given by

$$\dot{T}_{i+1}=T_{i+2}\frac{2t^2+1}{6}+T_i\frac{t^2-1}{6}-T_{i+1}\frac{5t^2}{12}-T_{i+3}\frac{t^2}{12} \tag{12}$$

The linear velocity v of the manipulator end in Cartesian space has its x-direction component vx derived by the chain rule, given by

$$v_x=\frac{dx}{dT}=\frac{\dot{x}}{\dot{T}} \tag{13}$$

By combining the above expressions for $\dot{x}_{i+1}$ and $\dot{T}_{i+1}$, it follows that

$$v_x=\frac{2x_{i+2}\left(2t^2+1\right)+2x_i\left(t^2-1\right)-5t^2x_{i+1}-t^2x_{i+3}}{2T_{i+2}\left(2t^2+1\right)+2T_i\left(t^2-1\right)-5t^2T_{i+1}-t^2T_{i+3}} \tag{14}$$

Similarly, the velocity components in the y- and z-directions, v_y and v_z, can be obtained. The Cartesian linear velocity vector is thus expressed as $v=(v_x,v_y,v_z)$.

The analytical derivation of linear acceleration, involving the second-order differentiation with respect to T, is computationally complex ; therefore, a numerical approximation is adopted. The

ISSN: 2414-1895 DOI: 10.6919/ICJE.202512_11(12).0023

linear acceleration aaa is defined as the ratio of the incremental change in linear velocity to the corresponding time increment, that is,

$$a=\frac{\Delta v}{\Delta T} \tag{15}$$

In the formula, Δv is the change in the end-effector linear velocity between two consecutive instants, and ΔT is the corresponding actual time increment.

The linear accelerations in the remaining directions can be obtained in the same manner. Based on the above derivations, the linear velocity and linear acceleration curves of the manipulator end in the x-, y-, and z-directions of the Cartesian space are obtained separately. These curves enable a direct assessment of whether the end-effector motion exhibits smooth velocity variations and whether any abrupt changes in acceleration, such as impact-like spikes, are present.

4.2 Multi-Constraint Control Analysis of the Cleaning Trajectory

In practical operation, the cleaning manipulator must satisfy multiple kinematic constraints to ensure motion safety, stability, and cleaning quality. For the end-effector trajectory in Cartesian space, the position, velocity, and acceleration constraints are defined as:

$$\begin{cases} x_{\min} \le x(T) \le x_{\max} \\ |v_x(T)| \le v_{x,\max} \\ |a_x(T)| \le a_{x,\max} \end{cases} \tag{16}$$

The constraints in the y- and z-directions are defined in the same manner.These constraints ensure that the B-spline–generated trajectory remains within the manipulator's workspace and satisfies the dynamic capabilities of each axis, preventing excessive velocity or acceleration that may cause vibration or trajectory deviation.

5. Analysis of Simulation Results

To verify the effectiveness of the proposed improved B-spline interpolation algorithm in trajectory planning for the ingot-mold cleaning manipulator, a petal-shaped trajectory was generated and simulated using Matlab/Robotics Toolbox. The trajectory shape was first planned in Cartesian space and then mapped to joint space through inverse kinematics, where interpolation was performed using the improved B-spline algorithm.

The petal-shaped trajectory to be generated is defined by:

$$r = R_0 + A\cos(6\theta) \tag{17}$$

In the formula,R0=500 mm, A=50 mm, $\theta \in [0, 2\pi]$.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202512_11(12).0023

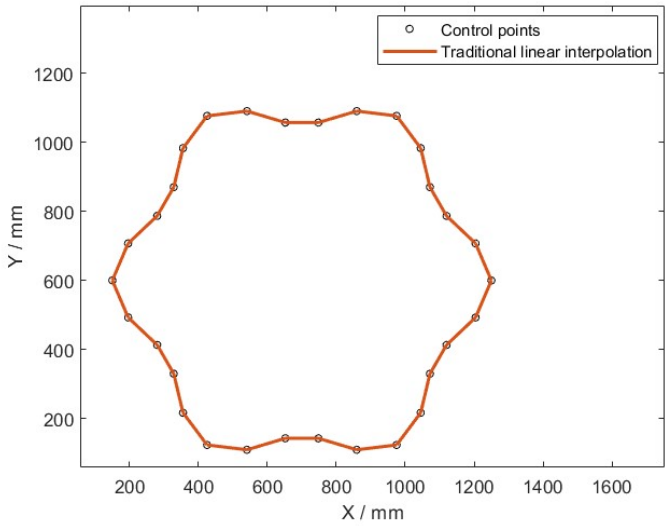

Fig. 8 Joint linear interpolation path in 3D space

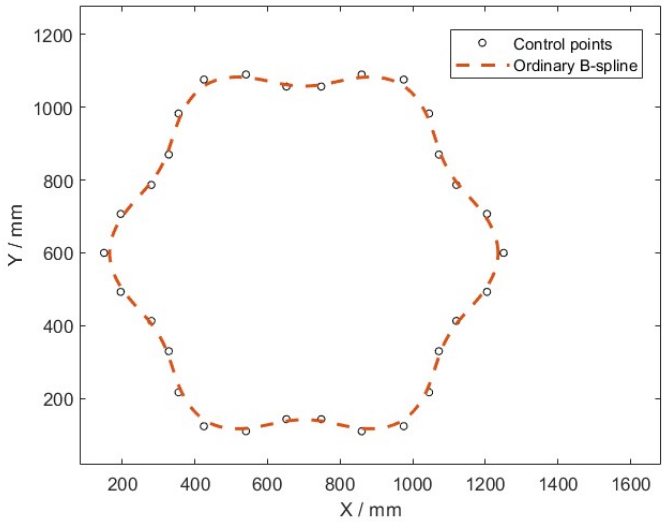

Fig. 9 Traditional B-spline interpolation path in 3D space

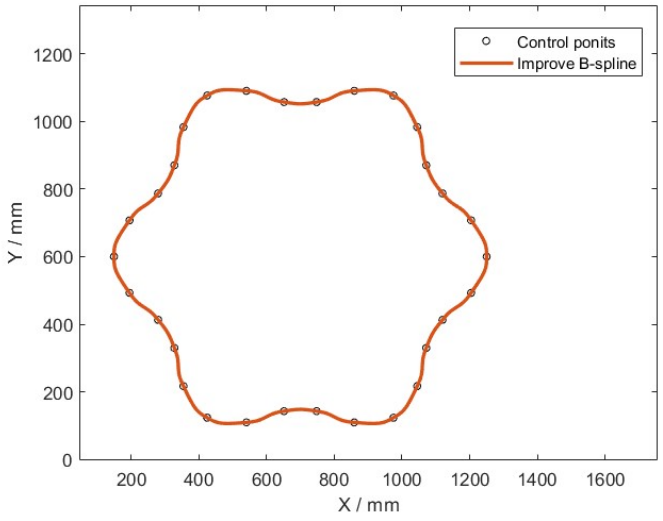

Fig. 10 Improved B-spline interpolation path in 3D space

ISSN: 2414-1895 DOI: 10.6919/ICJE.202512_11(12).0023

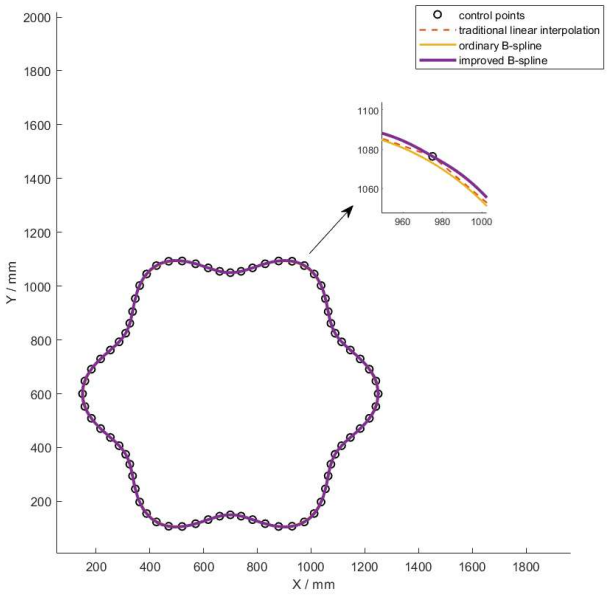

Fig. 11 Spatial trajectories obtained by the three methods

In Fig. 8, 9, and 10 the trajectories obtained by conventional linear interpolation, standard spline interpolation, and the improved B-spline interpolation for the petal-shaped cross-section are compared. It can be observed that linear interpolation produces broken-line segments between adjacent key points, leading to poor continuity and pronounced corners. Standard spline interpolation generates smoother curves but fails to pass exactly through all specified points, which may result in uncleaned regions in practical operations. With the improved B-spline interpolation, the trajectory passes precisely through the designated points and exhibits a smoother and more rounded profile. Path fluctuation is reduced, and the smoothness is markedly improved, ensuring complete coverage of the cleaning path.In Fig. 11 ,a Matlab-based program was developed to evaluate the fitting error, and the errors generated by the different curve-fitting methods are presented in Table 2.

Table 2. Comparison of different methods fitting error

Joint Interpolation Method	Spatial Path Error/mm
Linear Interpolation	0.992mm
Traditional B-spline	1.727mm
Improved B-spline	0.329mm

To further examine the influence of different trajectory planning methods on the dynamic performance of the manipulator, Fig. 12 and Fig. 13 present the displacement, velocity, and acceleration responses of Joint 1 (X-axis) and Joint 2 (Y-axis), respectively.

The results show that the velocity curve obtained by linear interpolation exhibits clear discontinuities, and sharp spikes appear in the acceleration profile, making mechanical impact likely. Although the conventional B-spline generates an overall smooth trajectory, its approximating nature prevents the curve from passing exactly through the specified points, resulting in limited geometric accuracy. In contrast, the improved B-spline maintains trajectory continuity while ensuring exact point interpolation. The velocity varies more uniformly, acceleration fluctuations are significantly reduced, and the resulting curve is smoother. Overall, the improved B-spline provides superior geometric accuracy and motion stability compared with the other methods, demonstrating evident advantages in enhancing the manipulator's operational stability and dynamic response.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202512_11(12).0023

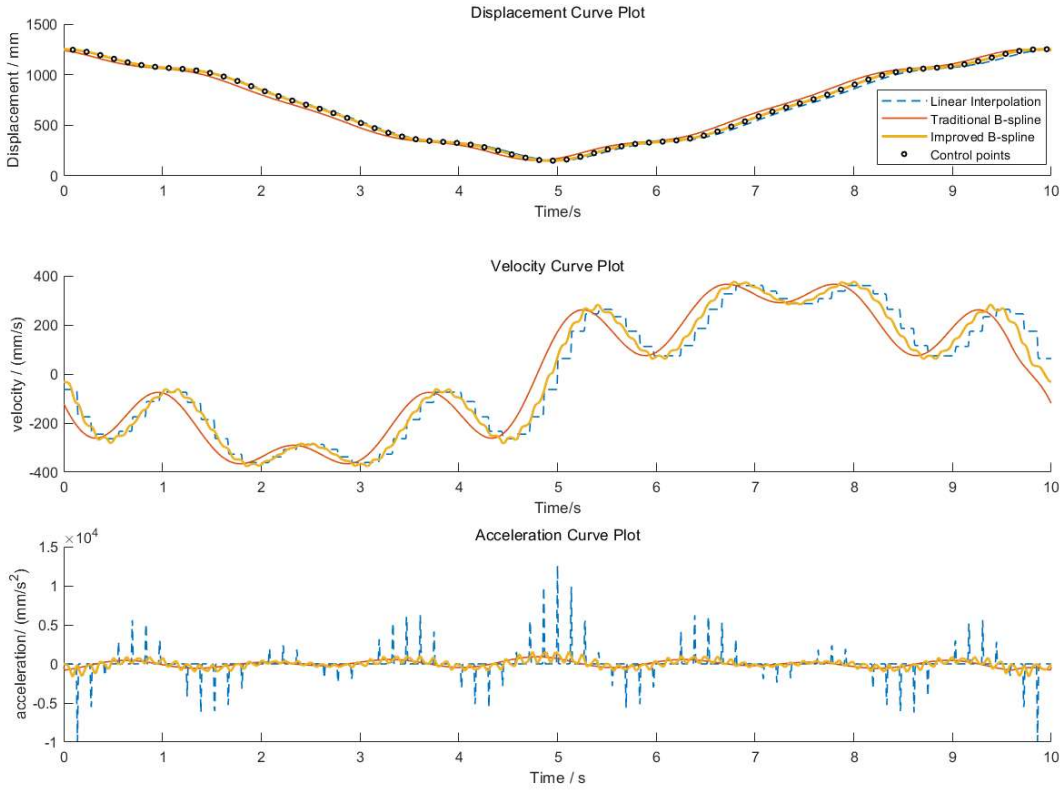

Fig. 12 X-axis displacement, velocity, and acceleration curves

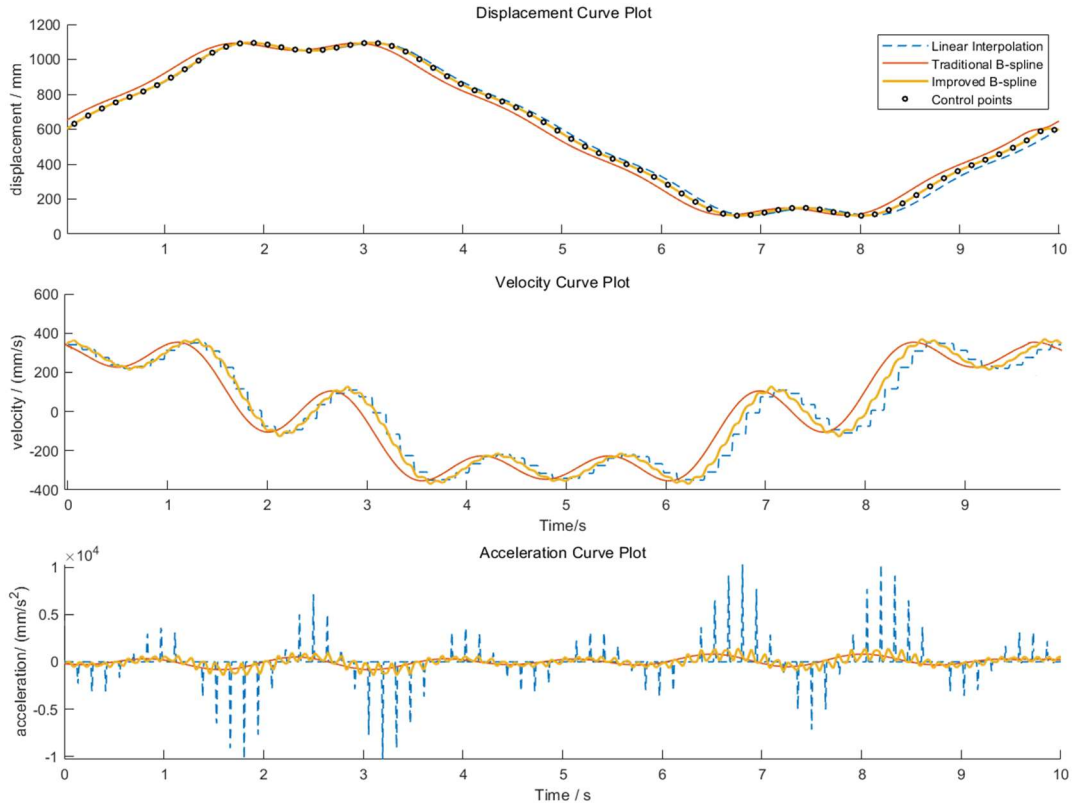

Fig. 13 Y-axis displacement, velocity, and acceleration curves

ISSN: 2414-1895 DOI: 10.6919/ICJE.202512_11(12).0023

6. Conclusion

In this study, the improved B-spline interpolation method was applied to joint-space trajectory planning for the ingot-mold cleaning manipulator. Simulation results show that the method generates smoother and more continuous cleaning paths while ensuring that the trajectory passes exactly through all designated key points, significantly outperforming conventional linear interpolation and standard B-spline interpolation. The enhanced smoothness reduces abrupt variations in joint velocity and acceleration, improves the stability of manipulator motion, and ensures that the cleaning brush covers the entire inner mold surface without omission. These results indicate that the improved B-spline interpolation method provides a more reliable means of ensuring cleaning quality and offers effective technical support for efficient and thorough mold cleaning.

References

[1] Cao Jinqi, Han Xuesong. Review of Research Methods for Industrial Robot Trajectory Planning[J].Information and Control, 2024,53(04):471-486+498.

[2] Cheng Huan, Deng Liying. Trajectory planning and tracking control of a seven degree of freedom shotcrete robot in coal mine roadway[J].Journal of Mine Automation,2024,50(01): 115-121.

[3] Wangyu Ming, Yangbao Hai, Yaoyu Xuan, et al.Segmented Trajectory Planning of Graphite Boat Transfer Mechanism Based on s-Curve and B-Spline Curve,2025,(01):84-87.

[4] Kongqing Bo,Kongqing Bo, Yuan Liang, Jiang Wei.Research of an Improved Trajectory Planning Method for Industrial Robot[J].Journal of Mechanical Transmission, 2019, 43(02): 30-36.

[5] Fu Y ,Wang J ,Chen J .Trajectory optimization algorithm based on B-spline curves[J].Journal of Physics: Conference Series,2025,3062(1):012011-012011.

[6] Zhu Y ,Wang X ,YafeiWang , et al.Robot trajectory planning for gear chamfer grinding based on multi-objective collaborative optimization and quintic B-spline interpolation algorithm[J].The International Journal of Advanced Manufacturing Technology,2025,138(9):1-18.

[7] Husheng Sun, Tuoyu Kun, Shenjun Qi, et al. Robot Trajectory Optimization Algorithm Based onSpatial Offset B-Spline[J].Journal of Tianjin University(Science and Technology), 2015, 48(8): 723-727.

[8] A. Abe, "Minimum energy trajectory planning method for robot manipulator mounted on flexible base," 2013 9th Asian Control Conference (ASCC), Istanbul, Turkey, 2013, pp. 1-7, doi: 10.1109/ASCC.2013.6606088.

[9] Luzhi Guo,Wang Xiao.Mechanical Arm Trajectory Planning Based on B-Splineand Whale Optimization Algorithm[J].Journal of Northeastern University(Natural Science), 2024, 45(05): 683-689.

[10] Dong Li, Yang Dong, Lujian Sen. Trajectory Planning Methods for Industrial Robots: A Review[J].Control Engineering of China, 2022, 29(12): 2365-2374.

[11] Shibu Hai, Sunhui Hui .B-spline Curve Algorithm Research Based on New S Velocity Planning[J]. Machine Tool & Hydraulics, 2016, 44(15): 72- 79.