Research on Engineering Structures & Materials
www.jresm.org
Research Group

Research Article

Continuous toolpath and feedrate smoothing control in five-axis CNC based on Dual-Projection tool axis optimization

Yangyang Li [*, a], Guo Chen [b]

Engineering Training Center, Shandong Huayu University of Technology, Dezhou 253034, China

Article Info

Article History:

Received 21 July 2026

Accepted 29 Sep 2026

Keywords:

Five-axis CNC machining;
Tool axis optimization;
Dual projection;
Trajectory smoothing;
Feed rate planning;
Machining efficiency;
Numerical simulation;
Kinematic constraints

Abstract

In five-axis CNC machining, abrupt tool axis direction shifts and over-limit rotary axis motions trigger vibration, accuracy degradation and low processing efficiency. To smooth tool axis motion and realize feed rate synchronization, this paper proposes an innovative greedy search-based tool axis optimization algorithm, which integrates rotary axis motion constraints with an asymmetric seven-segment S-curve acceleration and deceleration scheme to construct a complete five-axis machining trajectory optimization strategy. Numerical simulation results show that the proposed method reduces the average tool orientation vector variation rate to 0.018, with a maximum reduction of 85.48% compared with conventional G01 linear interpolation. The 90th percentile of the angle between adjacent tool orientation vectors is only 2.01°, the median feed rate reaches 29.20 mm/s, and rotary axis angular velocity complies with machine tool constraints throughout machining; total processing time falls to 81.34 s, boosting machining efficiency by 35.24%. Even in high-curvature regions, the average feed rate remains 26.89 mm/s, directional angle error is controlled within 0.53°, and no angular velocity limit violation occurs. This method effectively eliminates trajectory mutations and feed rate fluctuations in five-axis machining through geometric and kinematic optimal smoothing, delivering an effective technical solution for machining complex surface parts.

© 2026 MIM Research Group. All rights reserved.

1. Introduction

With the accelerated implementation of Industry 4.0 and Industry 5.0, five-axis Computer Numerical Control (CNC) machining technology has become a critical method for machining core components in aerospace, high-end equipment manufacturing, and other fields, owing to its advantages in achieving efficient and high-precision machining of complex surfaces. The machining quality and efficiency directly determine equipment performance and production benefits [1-2]. In actual machining processes, abrupt changes in tool axis orientation and frequent fluctuations in feed rate can easily trigger issues such as rotary axis motion overrun and machine tool vibration, which in turn lead to decreased machining accuracy, deteriorated surface quality, and limited machining efficiency [3]. The conventional planning approach for the tool axis usually conducts the vector modification in the workpiece coordinate frame without adequate consideration of the restrictions associated with the rotary axis motions and varying curvature of the trajectories, thereby hindering the attainment of a compromise between the smoothness of the tool axis, stability of motions, and machining efficiency. In light of the increasingly stringent needs for high precision and machining efficiency for complex surface parts, the current trajectory planning approaches fall short in meeting the needs for cooperative machining [4]. Therefore, conducting research on tool axis trajectory optimization and feed rate smoothing control, and constructing an integrated machining trajectory planning method that balances geometric accuracy and motion

*Corresponding author: YangyangLii@outlook.com
[a]orcid.org/0009-0007-6021-9994, [b]orcid.org/0009-0000-8646-4022
DOI: http://dx.doi.org/10.17515/resm2026-1953vk0721rs

constraints, has significant practical importance for promoting the engineering application of five-axis CNC machining technology in high-end manufacturing fields [5].

Recent studies have further demonstrated the close relationship between machining dynamics, vibration stability, and machined surface quality. Li et al. reported that chatter deteriorates the surface integrity and fatigue performance of Ti-6Al-4V components [6]. Karimi and Altintas showed that considering tool flexibility and blade dynamics can reduce chatter and surface-finish errors in five-axis ball-end milling [7]. Swan et al. demonstrated that combined low- and high-frequency vibration modes contribute to surface location error and dimensional deviations [8]. Hajdu et al. further showed that tool runout alters milling stability and may generate additional stable or unstable regions [9]. Binder et al. experimentally confirmed strong correlations among process vibrations, cutting forces, and surface topography [10]. These findings indicate that tool-axis orientation and feed rate planning should be evaluated not only in terms of geometric continuity, but also with respect to machine-tool dynamics, vibration stability, and final surface quality.

In order to achieve higher trajectory continuity and stability in five-axis CNC milling operations, much research has been done regarding the optimization of tool axis vectors, trajectory smoothing techniques, and velocity planning. Among these, the projection method, as a dimension reduction strategy, has been widely applied in interference detection and tool axis planning. For example, Liu and Xing proposed a tool axis vector optimization method based on automatic smoothing of C-axis rotation angle [11]. Zhao et al. proposed a multi-factor vector optimization method for face milling cutters [12]. Nevertheless, in case of the abovementioned single-projection approaches, global smoothing of the axis trajectory cannot be accomplished. To solve the above problem, the idea of dual dimensional projection has been developed, wherein the tool positions and orientations are processed separately. Xu et al. proposed Dual Non-Uniform Rational B-Spline (Dual-NURBS), adopting a master-slave curve strategy to achieve synchronized control of tool position and orientation [13]. Xiao et al. introduced a Dual-NURBS interpolator using interval partitioning under drive axis kinematic constraints [14]. Shi et al. proposed a G^3 continuous path smoothing method for hybrid robots [15]. The single-projection and dual-projection methods discussed above are compared in Table 1.

Table 1. Comparison of existing tool-axis optimization methods

Method type	Main assumption	Main constraints	Main feature and limitation
Rotary-angle optimization [11]	Smoothness improved by adjusting rotary angles	C-axis angle variation and machine limits	C-axis smoothing only, no dual-projection
Tool-axis-vector optimization [12]	Orientation selected by feasibility and motion	Collision, force, angular displacement and velocity	Multi-factor but no dual-projection framework
Dual-NURBS interpolation [13]	Position and orientation represented by two NURBS curves	Chord error, curvature, velocity fluctuation, synchronization	Dual-trajectory interpolation, not dual-projection
Online dual-NURBS interpolation [14]	Motion coordinated through master-slave curves	Axis velocity, acceleration, jerk, feed rate	Real-time interpolation with axial constraints
Local B-spline smoothing [15]	Local high-order smoothing improves continuity	Position, orientation and geometric errors, jerk	G^3 continuous machine-coordinate smoothing for hybrid robot

As shown in Table 1, single-projection methods focus on local optimization, while dual-curve methods separate position and orientation but lack explicit modeling of rotary axis angular velocity, acceleration, and jerk constraints.

In trajectory smoothing and velocity planning, related research has further focused on collaborative optimization of kinematic constraints and machining efficiency. For example, Yan et al. proposed a G^3 continuous dual-tangent corner arc transition method, which effectively limited the jerk of rotary axes [16]. Wang et al. proposed an adaptive feed rate generation technique for NURBS paths, integrating the S-curve approach and bidirectional scanning, producing feed rate

profiles with enhanced processing time by 46.73% and 22.33% [17]. Guevel et al. proposed a geometric error compensation method that corrects position detection in real time, achieving a 98% correction rate for virtual volume errors [18]. Additionally, Jacso et al. introduced artificial neural networks into feed rate planning, training models with pre-cutting experimental data to achieve constant tool load control, reducing machining time by 50% without increasing peak cutting force [19]. Another approach was proposed by Li et al. where differential vector optimization was used for generation of smooth axis trajectories, constructed using Jacobian matrices, through optimization targets, and validated on A-C type five-axis machine tools [20].

To address the above limitations, a five-axis CNC trajectory continuity and feed rate smoothing control method based on dual-projection tool axis optimization is proposed. It is worth noting that commercial CNC systems (e.g., Siemens Sinumerik COMPCURV/COMPCAD and Fanuc Smooth Local Interpolation) also provide trajectory smoothing functions, but their core is to compress and fit micro-segments at the NC program level to achieve inter-block velocity and acceleration continuity. In contrast, the proposed method reduces the tool axis orientation to the A-C rotary axis space for global greedy search through dual-projection mapping, suppressing tool axis mutations from the source, and explicitly models rotary axis angular velocity, angular acceleration, and angular jerk constraints, forming a complementary relationship with commercial CNC smoothing functions. The main contributions of this study are as follows:

- Theoretical mapping innovation: A bidirectional closed-loop mathematical mapping theory between the workpiece coordinate system and the machine A-C rotary axis space is established, and a complete dimensionality-reduction analytical model for high-dimensional tool orientation is constructed. The high-DOF tool orientation optimization problem is analytically mapped to a 2D A-C plane, and the tool path with minimum motion cost is obtained via greedy search, theoretically guaranteeing first-order tangential continuity of tool orientation.
- Differential constraint modeling innovation: The chain differential equations coupling rotary axis motion and feed rate are derived, and a mathematical model of asymmetric seven-segment S-curve feed rate planning with hard constraints on rotary axis angular velocity, angular acceleration, and angular jerk is constructed. The rotary axis kinematic constraints are explicitly and analytically embedded into feed rate optimization, breaking through the theoretical limitation of conventional feed rate planning those neglects rotary axis motion limits.
- Collaborative optimization framework innovation: A two-phase coupled collaborative optimization framework of geometric smoothing and motion smoothing is proposed. The synchronous parameterized coupling mechanism between tool tip trajectory and tool orientation is established, unifying the optimization variables of geometric smoothing and motion smoothing, and theoretically suppressing rotary axis angular velocity overrun in high-curvature regions.

2. Coordinated Control Method for Tool Axis Smoothing and Feed rate Smoothing

2.1 Trajectory Continuity Optimization Method Based on Dual-Projection Tool Axis

In five-axis CNC milling, sudden alterations in the orientation of the tool axis often result in inadequate trajectory continuity and contribute to vibrations in the machine tool, thus lowering its surface finish quality [21]. Conventional approaches in designing the tool axis orientation typically focus on altering the tool axis vector in the workpiece coordinate system in three dimensions. To address this issue, this study proposes a Greedy-based Dual-Projection Tool Orientation Optimization (GDPTOO). The structure of this method is shown in Figure 1.

As shown in Figure 1, GDPTOO maps high-dimensional tool axis vectors onto a low-dimensional A-C plane through the first projection. Projection domain planning yields the optimal angle path, and linear interpolation constructs a complete angle sequence. The second projection maps the optimized angles back to the workpiece coordinate system to output smoothed tool axis vectors. Before dimensionality-reduction optimization, tool axis vectors are converted into angular

information to facilitate smoothing. To guarantee that the mapping conforms to the physical structure of the machine tool, a dedicated forward kinematics model of the machine tool is established to build the correspondence between tool axis vectors and rotary axis angles, rather than adopting isolated spherical coordinate transformation. For the A-C dual rotary table machine tool, its kinematic chain follows: workpiece → C-axis → A-axis → cutting tool. Let $R_A(\theta_A)$ and $R_C(\theta_C)$ denote the rotation matrices of the A-axis (rotating about the X-axis) and C-axis (rotating about the Z-axis), respectively. The rotational transformation from the workpiece coordinate system to the machine tool coordinate system can be expressed as Equation (1).

$$R_{MW}(\theta_A, \theta_C) = R_A(\theta_A) \cdot R_C(\theta_C) \tag{1}$$

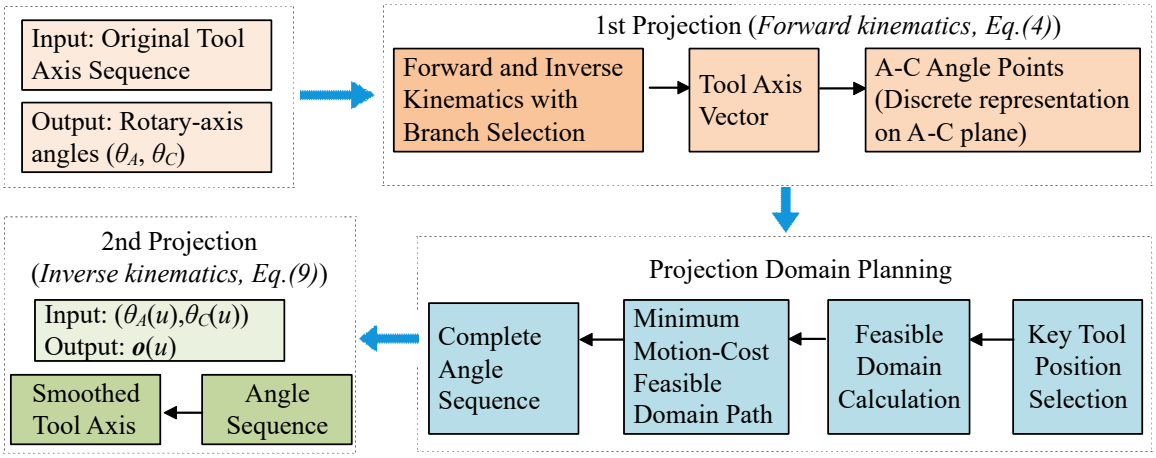


Fig. 1. Structure of GDPTOO method

In Equation (1), R_{MW} is the rotation matrix from the workpiece coordinate system to the machine tool coordinate system; θ_A is the swing angle of the A-axis; θ_C is the rotation angle of the C-axis. The specific forms of $R_A(\theta_A)$ and $R_C(\theta_C)$ are shown in Equation (2).

$$R_A(\theta_A) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_A & \sin\theta_A \\ 0 & -\sin\theta_A & \cos\theta_A \end{bmatrix}, \quad R_C(\theta_C) = \begin{bmatrix} \cos\theta_C & \sin\theta_C & 0 \\ -\sin\theta_C & \cos\theta_C & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

The complete machine tool kinematics adopts a 4×4 homogeneous transformation matrix. In addition to the above 3×3 rotational submatrix, it contains translational terms corresponding to the A-C pivot distance and worktable offset. The tool axis vector is a pure directional vector, and the translational component has no effect under homogeneous representation. Therefore, only the rotational submatrix is used for tool axis orientation mapping, and the pivot translation parameters are introduced in the solution of cutter-tip position. The expression of the tool axis vector in the workpiece coordinate system is shown in Equation (3).

$$\boldsymbol{o}_i^W = R_{MW}(\theta_{A,i}, \theta_{C,i}) \cdot \boldsymbol{e}_z \tag{3}$$

In Equation (3), $\boldsymbol{o}_i^W$ denotes the unit tool axis vector at the i-th cutter location; $\boldsymbol{e}_z$ is the unit vector along the Z-axis of the machine tool coordinate system. Expanding Equation (3) yields the forward kinematic relationship between the tool axis vector and rotary axis angles, as shown in Equation (4).

$$\begin{cases} o_{x,i} = \sin\theta_{A,i}\cos\theta_{C,i} \\ o_{y,i} = \sin\theta_{A,i}\sin\theta_{C,i} \\ o_{z,i} = \cos\theta_{A,i} \end{cases} \tag{4}$$

In Equation (4), $o_{x,i}$, $o_{y,i}$ and $o_{z,i}$ are the components of the tool axis vector along the X, Y and Z axes in the workpiece coordinate system, respectively. The inverse kinematic relationship for solving rotary axis angles from tool axis vectors can be obtained by inverting Equation (4). Since the A-axis

of the A-C dual rotary table machine tool can tilt in both positive and negative directions, solving merely with $\arccos(o_{z,i})$ will lose the sign information of negative angles. Therefore, a positive-negative branch selection strategy based on the arctan2 function is adopted. Let $r_i = \sqrt{o^2{}_{x,i} + o^2{}_{y,i}}$ denote the projection length of the tool axis vector on the XY-plane, then two sets of candidate solutions can be obtained as shown in Equation (5).

$$\begin{cases} \theta_{A,i}^{(+)} = atan2(r_i, o_{z,i}) \\ \theta_{C,i}^{(+)} = atan2(o_{y,i}, o_{x,i}) \end{cases}, \quad \begin{cases} \theta_{A,i}^{(-)} = -atan2(r_i, o_{z,i}) \\ \theta_{C,i}^{(-)} = atan2(-o_{y,i}, -o_{x,i}) \end{cases} \tag{5}$$

In Equation (5), superscripts (+) and (-) represent the positive and negative angle branches of the A-axis, respectively. The branch with physical feasibility is selected by minimizing the weighted angular variation relative to the previous cutter location, as shown in Equation (6).

$$(\theta_{A,i}, \theta_{C,i}) = argmin_{b\in\{+,-\}} [w_A|\theta_{A,i}^{(b)} - \theta_{A,i-1}| + w_C d_{\text{circ}}(\theta_{C,i}^{(b)}, \theta_{C,i-1})] \tag{6}$$

In Equation (6), w_A and w_C are weighting coefficients; $d_{\text{circ}}(\cdot,\cdot)$ is the shortest circular angular distance of the C-axis; $\theta_{A,i-1}$ and $\theta_{C,i-1}$ are the determined rotary axis angles of the previous cutter location. After selecting the optimal branch via the cost-minimization criterion, the solved angles are clipped within the mechanical limits of machine rotary axes to discard solutions exceeding the physical travel range. If both candidate branches of a cutter location exceed the travel limits, the corresponding orientation is unreachable for the machine tool, and feasible orientations can be generated by interpolation of adjacent cutter locations. The complete handling of this extreme case will be reserved for future research. After the first projection, the set of tool axis vectors $\{O_i\}$ is converted into a discrete angle point set $\{(\theta_{A,i}, \theta_{C,i})\}$ on the A-C plane, transforming the spherical orientation optimization problem into a 2D planar path optimization problem.

To construct a smooth tool axis trajectory within the 2D angle space, this work further adopts an optimization strategy based on key tool poses and greedy local optimization. First, the rotary-axis motion cost function between adjacent key tool poses is defined, as expressed in Equation (7).

$$\text{Cost}(\Theta_k^{(p)}, \Theta_{k+1}^{(q)}) = |\theta_{A,k}^{(p)} - \theta_{A,k+1}^{(q)}| + d_{\text{circ}}(\theta_{C,k}^{(p)}, \theta_{C,k+1}^{(q)}) \tag{7}$$

In Equation (7), $Cost(\cdot)$represents the cost function of rotary axis motion. $\Theta_k^{(p)}$ and $\Theta_{k+1}^{(q)}$ represent the k -th and $k+1$-th candidate angle points in the feasible domains of the p -th and q -th key cutter locations. This function quantifies the intensity of angular changes. Based on this, a greedy local optimization algorithm performs optimal search of the angle sequence. Starting from the feasible domain of the initial point, the method selects the optimal angle position Θ_{k+1} from the feasible domain Θ_{k+1}^* for each key cutter location in sequence. The selection criterion is the minimum motion cost relative to the previous key point, as shown in Equation (8).

$$\Theta_{k+1}^* = arg \min_{\Theta\in\Theta_{k+1}} Cost(\Theta_k^*, \Theta) \tag{8}$$

In Equation (8), $\Theta = (\theta_A, \theta_C)$, Θ_k^* and Θ_{k+1}^* represent the selected optimal angle points of the k -th and $k+1$ -th key cutter locations. $argmin(\cdot)$ represents the optimal solution operator that minimizes the cost function. It should be noted that the greedy algorithm is merely a local optimization strategy, which only optimizes based on the current state at each step and cannot guarantee a globally optimal solution. When the A-axis approaches the kinematic singularity at 0°, the feasible domain tends to shrink or fracture, and the algorithm faces the risk of local locking. Three strategies are adopted to improve search robustness: first, densify sampling of key tool poses to avoid sparse feasible domains; second, introduce the previous pose angle in branch selection to impose continuity constraints; third, automatically refine local poses and restart searching if no feasible solution is found. Iterative greedy search yields the key angle sequence with minimum

motion cost in the A–C plane, supporting the generation of smooth tool axes [22-23]. The single-step optimization and iterative workflow of the algorithm are illustrated in Figure 2.

In Figure 2, each key cutter location corresponds to a feasible domain with multiple candidate points in the A–C plane. The greedy algorithm starts from the optimal angle point p_k of the previous key cutter location. It traverses all candidate points of the current cutter location and calculates their motion cost. It then selects the point with the minimum cost as the optimal solution p_{k+1} of the current cutter location, thus completing a single-step optimization. After optimization in the angle space, the method maps the optimized angles back to the workpiece coordinate system through the second projection. The inverse transformation is expressed in Equation (9).

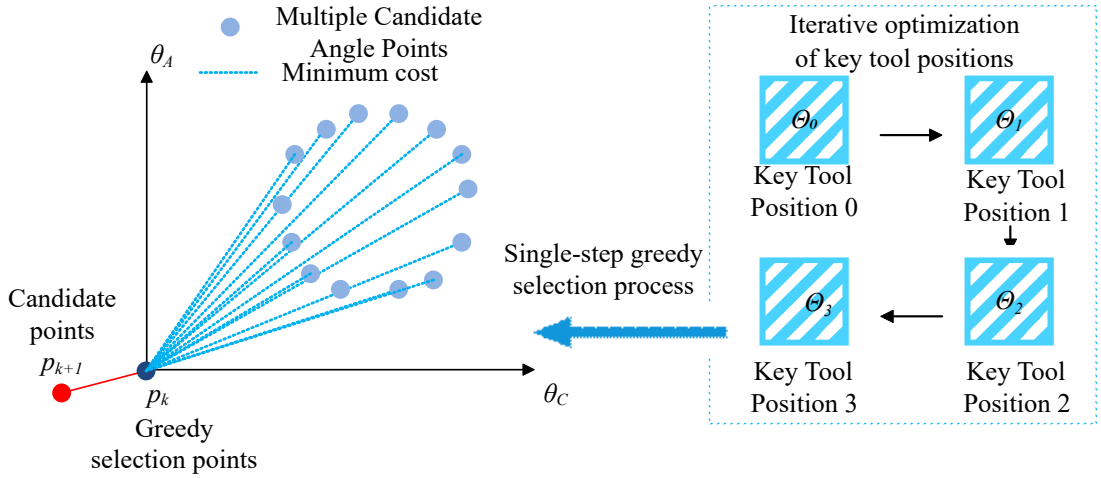


Fig. 2. Greedy optimization process in A–C plane

$$O(u) = [\sin\theta_A(u)\cos\theta_C(u),\ \sin\theta_A(u)\sin\theta_C(u),\ \cos\theta_A(u)]^T \tag{9}$$

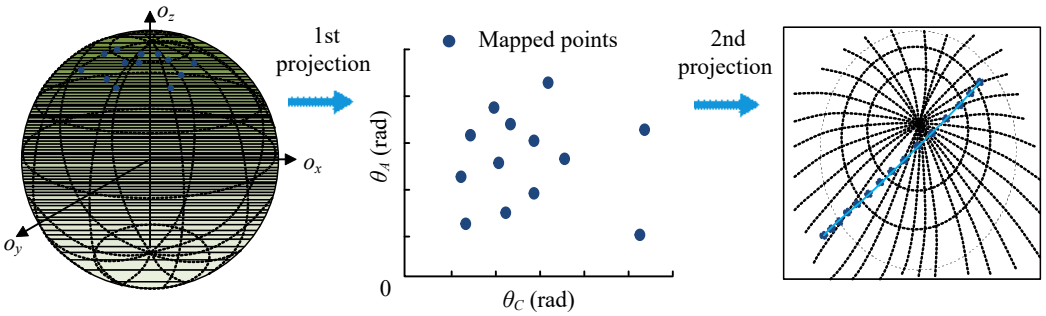
In Equation (9), $\theta_A(u)$ and $\theta_C(u)$ represent the interpolated A-axis and C-axis angles at trajectory parameter u, respectively. To guarantee continuous and smooth tool axis trajectory, cubic B-spline interpolation is performed between key angle points to achieve C^2 continuity of the angle sequence, eliminating jumps in angular velocity and impulses of angular acceleration. The angle interpolation formula is given in Equation (10).

$$\Theta(u) = \sum_{j=0}^{m} N_{j,3}(u)\Phi_j \tag{10}$$

In Equation (10), $\Theta(u) = (\theta_A(u), \theta_C(u))$. Φ_j denotes the angle control vertex; $N_{j,3}(u)$ is the cubic B-spline basis function; m+1 is the number of control vertices. Φ_j are determined by inverse calculation from key angle points via B-spline interpolation. According to the properties of cubic B-splines, Φ_j possesses C^2 continuity, which means the position, first and second derivatives of the angle sequence are continuous at nodes, corresponding to non-abrupt angular velocity and angular acceleration. Substituting the interpolated angles into Equation (9) yields the globally smooth tool axis vector sequence $\{O'_i\}$. Benefiting from the C^2 continuity of the angle sequence and the continuous mapping property of forward and inverse kinematics, the final tool axis trajectory satisfies continuity in position, tangent and curvature. The dimensionality-reduction projection process of tool axis vectors is illustrated in Figure 3.

The three-dimensional unit sphere in Figure 3(a) represents the distribution of the original tool axis vectors in the workpiece coordinate system. Through forward kinematics, the method projects the three-dimensional tool axis vectors onto the A–C plane in Figure 3(b) to complete dimensionality reduction. The optimization of greed and trajectory smoothing in the angle space are followed by an inverse kinematic mapping of the angle vector to a continuous tool axis trajectory on the 3D sphere, as illustrated in Figure 3(c). This two-stage projection process achieves dimension reduction and ensures comprehensive reconstruction of the tool axis direction. The tool

axis angle sequence $\Theta(u)$ obtained after dual-projection optimization needs to coordinate with the tool tip position trajectory to form complete five-axis machining instructions. Therefore, cubic B-spline interpolation is applied to the original discrete tool-tip coordinates to construct the parametric tool-tip trajectory $P(u)$. The proposed GDPTOO method implements dimensionality reduction optimization and pose reconstruction of the tool axis via bidirectional projection. It converts the complex spherical smoothing problem into a shortest-path optimization task in the low-dimensional angle space, and simultaneously improves tool axis continuity and machine motion stability within a unified framework.



(a) Tool axis on unit sphere 3D workpiece space (b) Mapped to A-C plane 2D rotation space (c) Optimized angle trajectory

Fig. 3. Dual-projection process of tool axis vector

2.2 Feed rate Smoothing Method Based on Rotary Axis Constraints

GDPTOO generates a tangentially continuous tool axis trajectory at the geometric level. However, there remains a disparity between the two elements of geometric continuity and motion smoothness. With mismatching feed rate and trajectory, the kinematic parameters may still violate their limits, causing motion shock and vibrations in the machine. The movement of the tool axis in five-axis machining is directly transformed into the rotary axis movement. Its dynamic constraints have frequently been identified as the most significant constraint to efficient machining operations. Consequently, accurate feed rate control is necessary for the proposed approach. Based on the GDPTOO framework, this study puts forward a rotary-axis-constrained asymmetric S-curve feed rate smoothing method, named GDPTOO-S. Its overall process is shown in Figure 4.

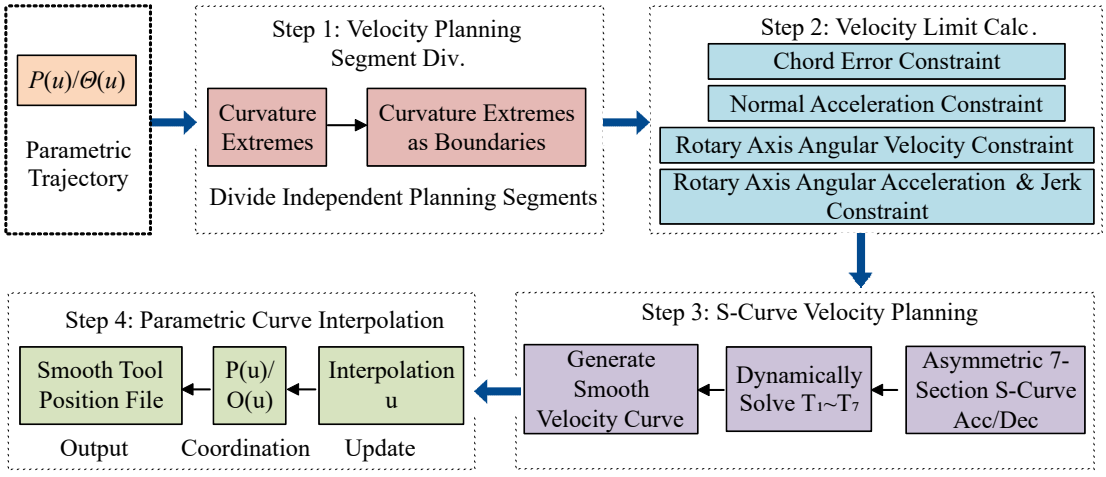


Fig. 4. Workflow of GDPTOO-S method

As shown in Figure 4, the GDPTOO-S method includes four core steps. First, it divides the trajectory into independent planning units using curvature extrema as boundaries. It computes the length of each unit by arc length integration. Second, it calculates the maximum allowable feed rate for each unit under three types of constraints. Third, it uses an asymmetric seven-segment S-curve acceleration and deceleration model to perform look-ahead planning and generate a smooth velocity profile [24]. Finally, it interpolates curves using the Taylor series expansion of the second order. The technique utilizes parameter synchronization for aligning the tip position and

orientation of the tool, and generates a cutter location file. The division of planning units is based on curvature extrema. At these points, curvature reaches its maximum and determines the upper limit of feed rate. Therefore, the feed rate at curvature extrema must satisfy three types of constraints. The first is the chord error constraint. It requires that the maximum deviation between the chord and the curve within one interpolation cycle stays within the allowable range. The constraint is shown in Equation (11).

$$v \le \frac{\sqrt{8\varepsilon_{chmax}/k_{max}}}{T_s} \tag{11}$$

In Equation (11), ε_{chmax} represents the maximum allowable chord error. T_s represents the interpolation period of the CNC system. v represents the feed rate, and k_{max} represents the curvature at the curvature extremum. The second constraint is the dynamic constraint of translational axes, that is, the normal acceleration constraint. The centripetal acceleration caused by trajectory curvature must not exceed the maximum allowable value of the machine translational axes, to avoid overload vibration. The constraint is shown in Equation (12).

$$v \le \sqrt{\frac{a_{nmax}}{k_{max}}} \tag{12}$$

In Equation (12), $a_{n\max}$ represents the maximum allowable normal acceleration of the machine. The third constraint is the key constraint specific to five-axis machining, which is the angular velocity constraint of rotary axes. The continuous tool axis orientation output by GDPTOO determines the swing angles of the A and C axes. The feed rate and rotary-axis angular velocity are strongly coupled through trajectory geometry. This work adopts the seven-segment S-curve acceleration/deceleration model. Within acceleration and deceleration intervals, feed rate v, tangential acceleration a and tangential jerk j are not constant. Full chain differentiation is required to establish the coupling among rotary-axis angular velocity, angular acceleration, angular jerk and feed parameters. $r \in \{A, C\}$ denote the A/C rotary axes, and $\theta_r(s)$ the mapping function of rotation angle with respect to arc length s. The three-layer kinematic relation is expressed as Equation (13).

$$\begin{cases} \dot{\theta_r} = \theta'_r(s)v \\ \dot{\theta_r} = \theta''_r(s)v^2 + \theta'_r(s)a \\ \dot{\theta_r} = \theta'''_r(s)v^3 + 3\theta''_r(s)va + \theta'_r(s)j \end{cases} \tag{13}$$

In Equation (13), $\dot{\theta_r}$, $\dot{\theta_r}$, $\dot{\theta_r}$ represent angular velocity, angular acceleration and angular jerk of rotary axes; $\theta'_r(s)$, $\theta''_r(s)$, $\theta'''_r(s)$ are the first-, second- and third-order derivatives of rotation angle with respect to arc length. To guarantee smooth rotary-axis motion without violation, $\dot{\theta_r}$, $\dot{\theta_r}$, $\dot{\theta_r}$ must satisfy their hardware limits in Equation (14).

$$\begin{cases} |\dot{\theta_r}| \le \dot{\theta}_{r,max} \\ |\dot{\theta_r}| \le \dot{\theta}_{r,max} \\ |\dot{\theta_r}| \le \dot{\theta}_{r,max}\{ \end{cases} \tag{14}$$

In Equation (14), $\dot{\theta}_{r,max}$, $\dot{\theta}_{r,max}$, $\dot{\theta}_{r,max}$ are the maximum allowable angular velocity, angular acceleration and angular jerk of rotary axes. Substituting Equation (13) into Equation (14) and eliminating angular derivatives yields the complete expression of full dynamic constraints for rotary axes, as Equation (15).

$$\begin{cases} |\theta'_r(s)v| \le \dot{\theta}_{r,max} \\ \theta''_r v^2 + \theta'_r(s)a| \le \ddot{\theta}_{r,max} \\ \theta'''_r v^3 + 3\theta''_r(s)va + \theta'_r(s)j| \le \dddot{\theta}_{r,max} \end{cases} \tag{15}$$

From the constraint characteristics in Equation (15), the angular velocity constraint can explicitly solve the upper feed rate bound, while angular acceleration and angular jerk constraints couple v, a and j simultaneously, which need real-time verification during iterative velocity planning. The mechanisms of these three constraints on feed rate are illustrated in Figure 5.

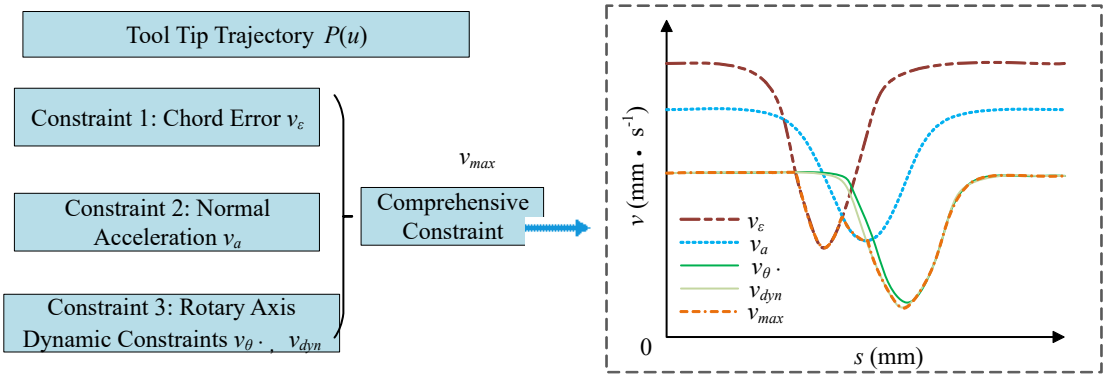


Fig. 5. Schematic of feed rate constraints under multiple conditions

As shown in Figure 5, chord error and normal acceleration constraints are mainly governed by trajectory curvature and significantly reduce allowable feed rate at curvature extremes. Rotary-axis angular velocity, angular acceleration and angular jerk constraints depend on the variation rate of tool orientation and become the globally dominant constraint where the tool axis swings violently. Combining all constraints and the machine maximum feed rate F, the maximum allowable feed rate v_{max}) within each segment is determined by the strictest constraint, as Equation (16).

$$v\ min\{v_\varepsilon, v_a, v_{\dot{\theta}}, v_{\mathrm{dyn}}, F\}_{max} \tag{16}$$

In Equation (16), v_ε and v_a denote the feed rate upper bounds limited by chord error and normal acceleration, respectively; $v_{\dot{\theta}}$ is the feed rate upper bound from rotary-axis angular velocity constraint; v_{dyn} represents the maximum feasible feed rate satisfying the coupled angular acceleration and angular jerk constraints in Equation (15). After determining feed rate limits, an asymmetric seven-segment S-curve acceleration/deceleration model is adopted for look-ahead velocity planning. The parameter calculation flow of this model is shown in Figure 6.

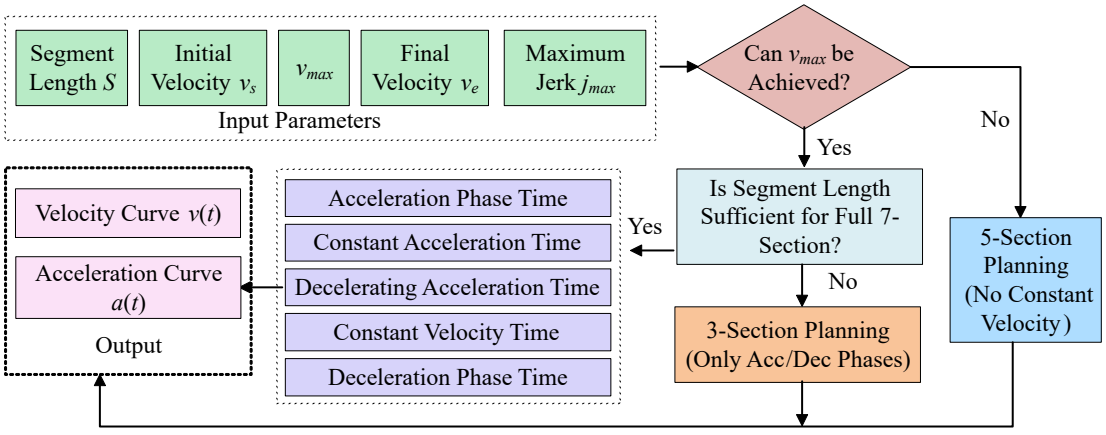


Fig. 6. Parameter computation process of S-curve planning

As illustrated in Figure 6, the model ensures continuous acceleration by limiting jerk. Its asymmetric property allows different dynamic parameters for acceleration and deceleration phases, better adapting to non-uniform tool-axis variation. The planning targets time optimality, and dynamically solves phase durations according to segment length, start/end velocity and feed rate upper bound. Within each interpolation period, current v, a and j are substituted into Equation (15) to verify all rotary-axis dynamic indices. If any parameter exceeds the limit, the feed

rate upper bound is reduced and the planning is recalculated. Short trajectories automatically degrade to five-section or three-section planning while maintaining acceleration continuity. The complete calculation formulas for segment transition times of the asymmetric seven-segment S-curve are provided in Appendix A. After velocity planning, interpolation converts velocity commands into discrete cutter location points [25]. For $P(u)$, this study uses a second-order Taylor expansion to update the curve parameter and achieve high-precision interpolation, as shown in Equation (17).

$$u_{i+1} = u_i + \frac{v_i T_s}{|P'(u_i)|} + \frac{a_i T_s^2}{2|P'(u_i)|} - \frac{v_i^2 T_s^2 P'(u_i) \cdot P''(u_i)}{2|P'(u_i)|^4} \tag{17}$$

In Equation (17), u_i and u_{i+1} represent the current and next interpolation parameters. $P'(u_i)$ and $P''(u_i)$ represent the first and second derivatives of the tool tip trajectory. While updating the tool tip position, the method uses a parameter synchronization mechanism. It substitutes the same parameter u_{i+1} into the tool axis vector function O(u). This ensures coordinated motion of position and orientation, and finally outputs the GDPTOO-S smoothed cutter location file. The truncation error of this second-order Taylor expansion converges at the order of $\mathcal{O}(T_s^3)$. The complete error expression consists of a third-order dominant term and higher-order residual terms of order four and above, as shown in Equation (18).

$$\varepsilon_{\text{Taylor}} = \frac{1}{6}\frac{d^3u}{dt^3}T_s^3 + \mathcal{O}(T_s^4) \tag{18}$$

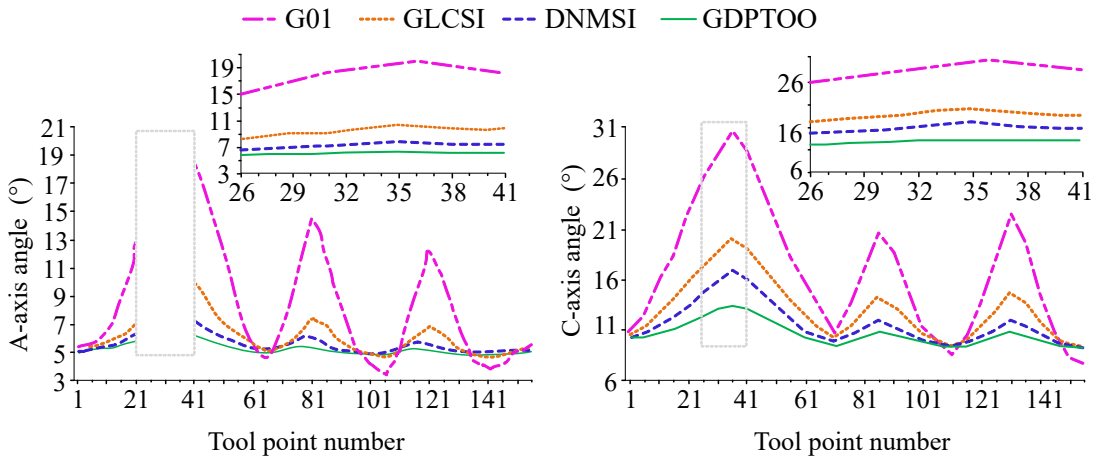
In Equation (18), $\varepsilon_{\text{Taylor}}$ denotes the truncation error of Taylor expansion. In high-curvature trajectory regions, $\frac{d^3u}{dt^3}$ increases with the growth of curvature, which raises $\varepsilon_{\text{Taylor}}$. The interpolation cycle adopted in this work is T_s=0.001. The magnitude of the dominant truncation error term is far smaller than the geometric characteristic scale of the trajectory and can be neglected in practical machining. Therefore, the second-order Taylor interpolation achieves favorable convergence and numerical stability in high-curvature regions.

3. Numerical Simulation Verification

3.1 Validation of Trajectory Continuity of GDPTOO

To verify the effectiveness of GDPTOO in tool axis trajectory smoothing and machining stability, a single blade of an aero-engine integral impeller was selected as the test object. The blade surface is complex, and the tool axis vector varies sharply at the root and tip regions, which is prone to causing machine tool vibration. A five-axis machining cutter location file containing 156 cutter location points was generated by UG CAM using the iso-scallop height method. The cutter location file adopts the GOTO statement format, where each statement contains the tool tip coordinates (X, Y, Z) and the tool axis vector (I, J, K). A MATLAB script reads the file line by line, extracts the six parameters from each GOTO statement, and constructs the tool tip sequence and tool axis vector sequence as the input of GDPTOO. The B-spline interpolation in GDPTOO is applied to the angle sequence and tool tip trajectory rather than the original CAM output.

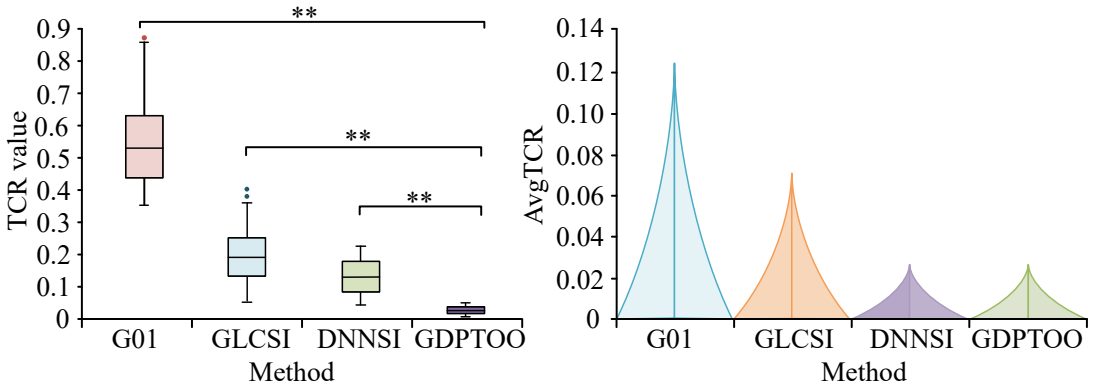
The experimental hardware used an Intel Core i7-12700K processor with 32GB memory, and the software was built on MATLAB R2022a. The interpolation period was set to 0.001 s, the chord height error limit was 0.01 mm, the maximum normal acceleration was 2000 mm/s^2, and the maximum angular velocities of the A/C axes were 30°/s and 60°/s, respectively. In order to verify the role of GDPTOO algorithm on smooth operation of rotary axes, this study chose the fluctuating behavior of swing angle of A-axis and rotation angle of C-axis as primary evaluation criteria. Meanwhile, GDPTOO was compared with traditional G01 linear interpolation, Global and Local Composite Smoothing Interpolation (GLCSI), and Dual-NURBS Master-Slave Interpolation (DNMSI). Among these, GLCSI uses a dual-domain critical criterion to achieve real-time smoothing interpolation, and DNMSI adopts a master-slave curve strategy to achieve synchronized control of position and orientation. The results are shown in Figure 7.



(a) A-axis swing angle variation curve (b) C-axis rotation angle variation curve

Fig. 7. Comparison of rotary axis angle variations

As shown in Figure 7(a), for the A-axis swing angle, G01 reaches a peak of 20.12° at tool-point 36. The peak values of GLCSI and DNMSI drop to 10.45° and 7.89°, respectively, while GDPTOO yields the lowest peak of 6.34° with the mildest variation. As shown in Figure 7(b), G01 exhibits severe fluctuation and peaks at 30.50° for the C-axis rotation angle at tool-point 36. The peak values of GLCSI and DNMSI are 19.80° and 16.80°, and GDPTOO achieves a peak of only 13.20°, restricting the fluctuation within 9.00°~13.20°. The inset magnified views further reveal obvious slope corners at tool-point junctions for G01, whereas the GDPTOO curve is free of slope discontinuity. These results demonstrate that the GDPTOO method can suppress both angle amplitude and segment-joint-induced angle mutation for A and C axes, and realize smooth rotary-axis motion. To quantitatively evaluate the effectiveness of different methods in suppressing tool axis vector abrupt changes, this study used Tool Axis Vector Change Rate (TCR) as an evaluation indicator. When TCR ≤ 0.05, tool axis motion can be considered smooth. The comparison of TCR distribution and average TCR among the four methods is shown in Figure 8.



(a) TCR distribution for different methods (b) Average TCR across different methods

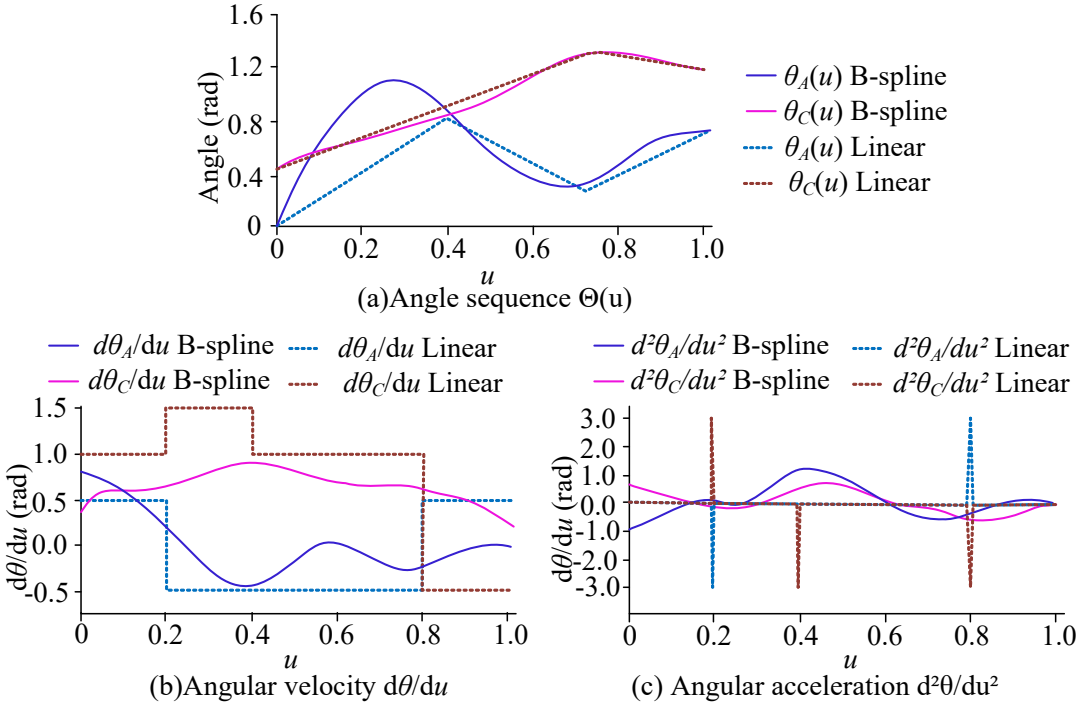
Fig. 8. Comparison of tool axis vector change rates

*Note: Wilcoxon signed-rank test was used in Figure (a), ** indicates $p<0.01$.*

As shown in Figure 8(a), the G01 method had the widest TCR distribution, with a median of 0.53 and a maximum value of 0.87, with all samples exceeding 0.05. The TCR distributions of GLCSI and DNMSI narrowed successively, but some samples still exceeded the threshold. The GDPTOO method had the most compact TCR distribution, with a maximum value of only 0.04, and all samples met the smoothness requirement below 0.05. As shown in Figure 8(b), the average TCR of G01 was 0.124, while the average TCR of the GDPTOO method was 0.018, representing reductions of 85.48%, 74.65%, and 35.71% compared to G01, GLCSI, and DNMSI, respectively. This suggests that GDPTOO is capable of effectively suppressing the sudden changes in tool axis orientation via the dual projection approach and greedy searching process, as all TCRs of the trajectory satisfy the

requirement for smooth machining. For verifying the improvement of cubic B-spline interpolation on angle sequence continuity in the GDPTOO method, the angle sequence and its first and second derivatives under piecewise linear interpolation and cubic B-spline interpolation are compared, as shown in Figure 9.

As shown in Figure 9(a), linear interpolation ensures only position continuity at knots, whereas cubic B-spline interpolation achieves smooth transitions. As shown in Figure 9(b), the first derivative under linear interpolation exhibits step discontinuities at knots, while that under cubic B-spline interpolation remains continuous. As shown in Figure 9(c), the second derivative under linear interpolation produces sharp impulse spikes at knots, whereas that under cubic B-spline interpolation is smooth without spikes. These results indicate that cubic B-spline interpolation achieves C^2 continuity of the angle sequence, eliminating the second-derivative discontinuity caused by linear interpolation.



(a)Angle sequence Θ(u)

(b)Angular velocity dθ/du

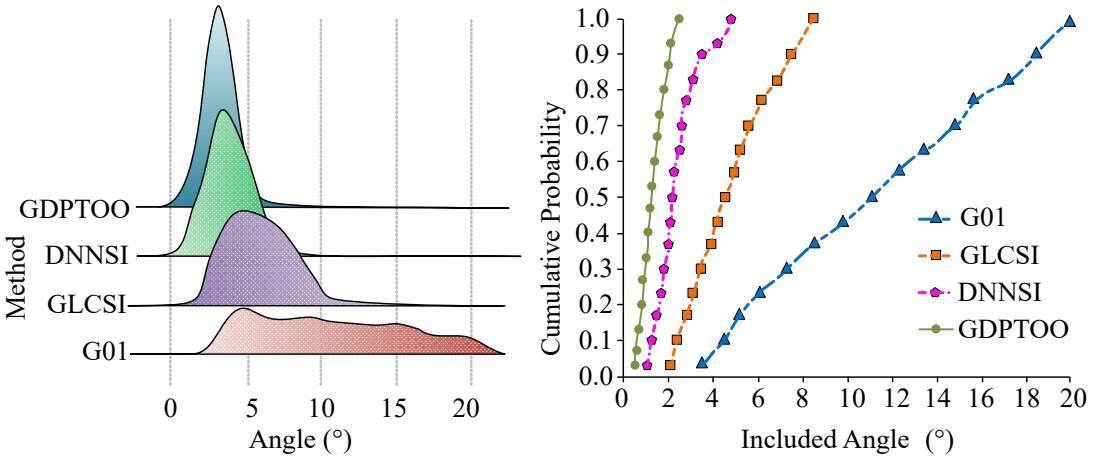
(c) Angular acceleration d²θ/du²

Fig. 9. C^2 continuity verification: linear vs cubic B-spline interpolation

In order to quantify the effectiveness of each method on the suppression of the variation amplitude of the tool axis orientations at two adjacent cutter locations, this study utilized the spatial angle between two adjacent tool axis vectors as the assessment criterion. The results are shown in Figure 10. In Figure 10(a), the G01 method had the widest-angle distribution range, with the longest tail extending to 20.16°, indicating frequent tool axis abrupt changes. The distribution ranges of GLCSI and DNMSI methods gradually narrowed. The GDPTOO method had the most compact distribution, concentrated in the low-angle range of 0°–5°, with no obvious long tail. As shown in Figure 10(b), the curve obtained using the GDPTOO approach was on the farthest left side and its 90th percentile was 2.01°, whereas the 90th percentiles of DNMSI, GLCSI, and G01 were 3.73°, 7.84°, and 18.42°, respectively. This indicates that GDPTOO can suppress abrupt changes in tool axis orientation, providing high-quality geometric trajectories for stable machining. To analyze the influence of key tool position selection and feasible domain search step on algorithm performance, a sensitivity analysis is conducted under different cutter location point densities and search steps, as shown in Table 2.

As shown in Table 2, with an increasing CL point density, the average TCR decreases slightly and the machining time shortens slightly, while the computation time increases significantly. When the CL point number increases from 156 to 1000, the average TCR drops from 0.018 to 0.014 (22.2% reduction), while the computation time rises from 0.42 ms to 2.03 ms, exceeding the 1 ms real-time threshold. With decreasing search step, the average TCR decreases and the computation time

increases. Overall, a CL point number of 156–300 and a search step of 0.2° achieve a good balance between optimality and computational efficiency.



(a) Adjacent tool axis angle distribution (b) Cumulative distribution of adjacent tool axis angle

Fig. 10. Distribution of spatial angles between adjacent tool axis vectors

Table 2. Sensitivity analysis of algorithm parameters

Variable	Parameter value	Average TCR	Average CPU time per interpolation frame (ms)	Machining execution time (s)
CL point density	156	0.018	0.42	81.34
	300	0.016	0.68	80.12
	500	0.015	1.15	79.56
	1000	0.014	2.03	79.18
A-C feasible domain search step (°)	0.5	0.021	0.18	/
	0.2	0.018	0.42	/
	0.1	0.016	0.85	/
	0.05	0.015	1.62	/

3.2 Validation of Feed rate Smoothing Control of GDPTOO-S

In order to assess the effect of GDPTOO-S on feed rate kinematic properties and machining productivity, this study adopted the optimal tool axis path derived from GDPTOO, utilizing G01, GLCSI, and DNMSI as control strategies, and validated its efficiency in smoothing feed rate and satisfying rotary axis constraints within the same experimental conditions as discussed previously. The experiment strictly adhered to five-axis machine kinematic constraints, and kinematic data was gathered at 10% machining times. The feed rate distribution and A/C-axis angular velocity variations for each method are shown in Figure 11.

Figure 11(a) shows that the A-axis angular velocity peaks of G01 and GLCSI reach 45.23°/s and 35.63°/s, both exceeding the threshold. The DNMSI peak is 28.48°/s, while the GDPTOO-S peak is only 24.50°/s. Figure 11(b) shows that the C-axis angular velocity peak of G01 reaches 82.56°/s, while those of GLCSI and DNMSI decrease to 58.45°/s and 42.35°/s, respectively. The GDPTOO-S peak is only 30.52°/s, satisfying the constraint throughout. The zoomed-in subplots indicate that G01 and GLCSI exhibit slope corners at peak transitions, whereas the GDPTOO-S curve transitions smoothly without slope discontinuities. As shown in Figure 11(c), the median feed rate of G01 is only 18.45 mm/s, while those of GLCSI and DNMSI are 24.63 mm/s and 28.50 mm/s, respectively. The GDPTOO-S method has the most concentrated distribution with a median of 29.20 mm/s. These results demonstrate that GDPTOO-S effectively improves feed rate stability and strictly controls the A/C-axis angular velocities within machine tool limits. To evaluate the control effect of GDPTOO-S on tool axis orientation accuracy and machining efficiency, the direction angle error distributions and machining execution times of four methods are compared, as shown in Figure 12.

As shown in Figure 12(a), the G01 method has a median direction angle error of 1.87° with the widest distribution. The medians of GLCSI and DNMSI are 1.12° and 0.68°, respectively. The GDPTOO-S method has a median of 0.35° with the narrowest distribution and the best control accuracy, showing significant differences from all other methods. This indicates that GDPTOO-S can effectively control tool axis direction deviation through rotary axis angular velocity constraints. Figure 12(b) shows that G01 has the longest machining time of 125.63 s. The machining times of GLCSI and DNMSI are improved by 13.66% and 26.19% compared with G01, respectively. GDPTOO-S achieves the shortest machining time of 81.34 s, which is 35.24% shorter than G01 and 12.26% shorter than DNMSI, demonstrating significant machining efficiency advantage. To evaluate the real-time control feasibility, the CPU computation time per interpolation frame in MATLAB R2022a is summarized in Table 3.

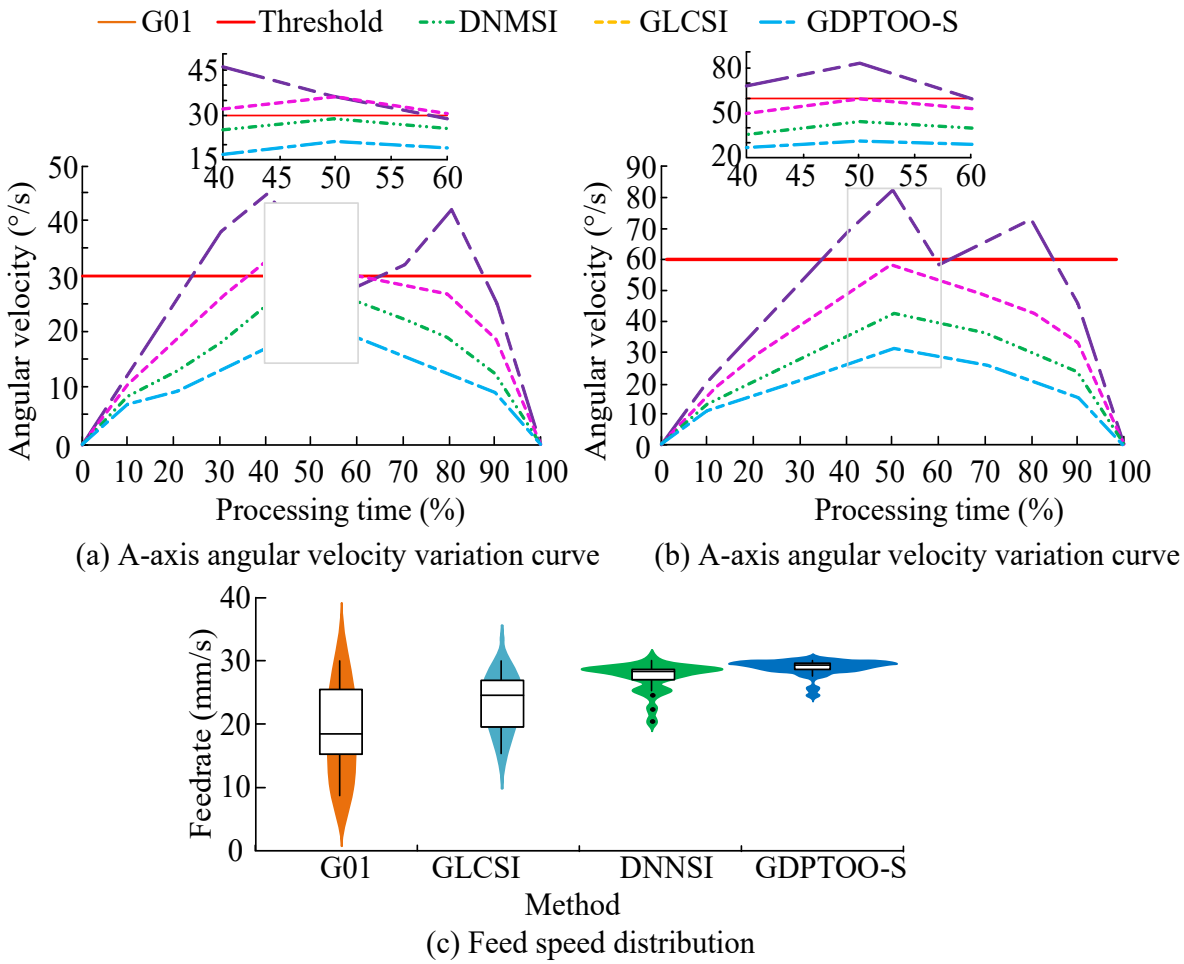


Fig. 11. Feed rate and rotary axis angular velocity distributions

Note: The feed rate distribution in (a) aggregates data of the full machining process; angular velocity curves in (b) and (c) plot against normalized processing time. All subplots share the identical machining case.

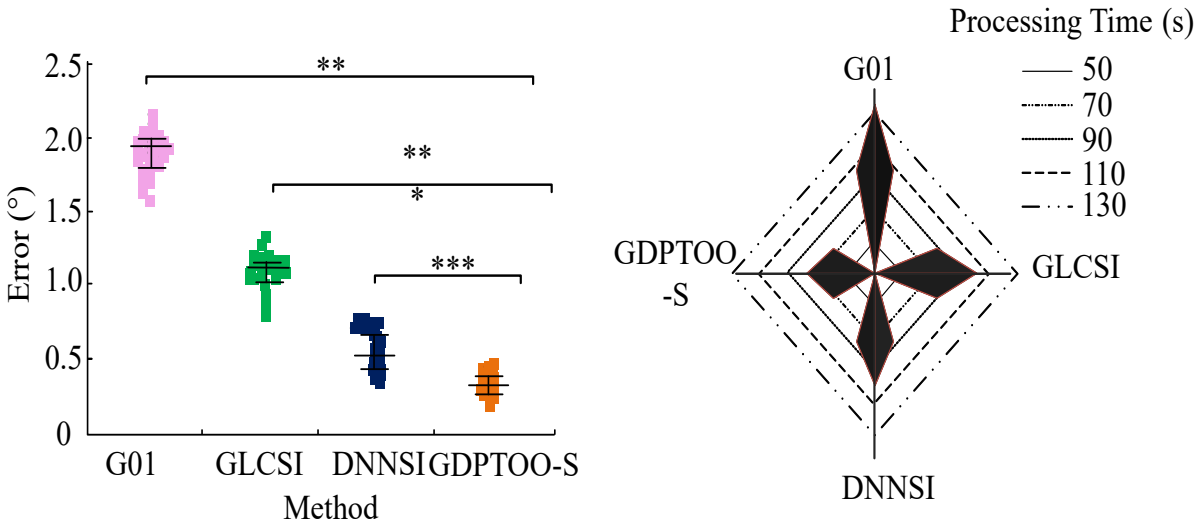
Table 3. CPU computation time comparison

Method	Average CPU time per interpolation frame (ms)	Real-time feasibility (1 ms)
G01	0.005	Yes
GLCSI	0.18	Yes
DNMSI	0.26	Yes
GDPTOO-S	0.42	Yes

Note: Computation environment: MATLAB R2022a, Intel Core i7-12700K CPU. The time refers to the average solving time per tool axis interpolation frame. 1 ms is the standard CNC interpolation cycle limit.

As shown in Table 3, G01 has the shortest computation time of only 0.005 ms. The computation times of GLCSI and DNMSI are 0.18 ms and 0.26 ms, respectively. GDPTOO-S has the longest computation time of 0.42 ms, which is still less than the 1 ms interpolation cycle, satisfying the real-

time control requirement. It should be noted that MATLAB is an interpreted language, and the computation time can be further reduced after porting to C++. Future work will port GDPTOO-S to an open-architecture CNC system to verify its real-time performance in an actual CNC controller.



(a) Azimuth angle error distribution (b) Machining Processing Time Comparison

Fig. 12. Comparison of Azimuth Angle Error and Machining Efficiency

*Note: Wilcoxon signed-rank test, ** indicates p<0.01, *** indicates p<0.001.*

To further analyze the performance differences of each method in different curvature regions, this study divided the trajectory into low, medium, and high curvature intervals according to curvature magnitude k, and compared the average feed rate, A-axis angular velocity overrun count, and mean direction angle error of each method in each interval. The results are shown in Table 4.

Table 4. Performance comparison in different curvature regions

Curvature Interval (mm^{-1})	Metric	G01	GLCSI	DNMSI	GDPTOO-S
k<0.05	Feed rate (mm/s)	22.53	26.37	28.19	29.54
	A-axis violations	0	0	0	0
	Orientation error (°)	1.23	0.67	0.44	0.23
0.05≤k<0.15	Feed rate (mm/s)	15.87	22.54	26.81	28.67
	A-axis violations	8	3	0	0
	Orientation error (°)	2.39	1.19	0.74	0.39
k≥0.15	Feed rate (mm/s)	8.24	16.83	22.57	26.89
	A-axis violations	15	7	2	0
	Orientation error (°)	3.87	1.89	1.07	0.53

Note: $k = \frac{|P'(u) \times P''(u)|}{|P'(u)|^3}$ *; the low curvature region (k<0.05 mm⁻¹) corresponds to the flat region at the blade middle, the medium curvature region (0.05≤k<0.15 mm⁻¹) corresponds to the blade transition region, and the high curvature region (k≥0.15 mm⁻¹) corresponds to the curvature mutation regions at the blade root and tip.*

As shown in Table 4, the differences among methods in the low-curvature region were relatively small. GDPTOO-S achieved an average feed rate of 29.54 mm/s, a direction angle error of 1.23°, and zero angular velocity overruns. In the medium-curvature region, G01 had 8 overruns, while GDPTOO-S maintained zero overruns with an average feed rate of 28.67 mm/s, significantly higher than other methods. The performance gap further widened in the high-curvature region, where G01 achieved a feed rate of only 8.24 mm/s with 15 overruns. In contrast, GDPTOO-S was able to achieve a speed of 26.89 mm/s, an error of 0.53°, and no overruns at all, proving that it is capable of balancing efficiency and stability even in curved paths.

4. Conclusion and Future Work

To ensure continuous smoothing of tool axis paths and stable feed rate control in five-axis CNC milling operations, this study developed the GDPTOO dual projection tool axis optimization algorithm using greedy search technique, incorporated rotation axis constraints into asymmetric seven-segment S-curve speed planning, and formulated the GDPTOO-S feed rate smoothing control approach. Experimental results showed that GDPTOO reduced the average tool axis vector change rate to 0.018, representing reductions of 35.71%–85.48% compared to the comparison methods. The 90th percentile of adjacent tool axis vector angles was only 2.01°, with tool axis smoothness and continuity significantly superior to the comparison methods. Additionally, GDPTOO-S achieved a feed rate median of 29.20 mm/s, with A/C-axis angular velocity peaks of 24.50°/s and 30.52°/s, respectively, satisfying machine tool constraints throughout. The machining time was only 81.34 s, representing a 35.24% reduction compared to traditional methods. Under high-curvature complex trajectories, the proposed method maintained an average feed rate of 26.89 mm/s, controlled the direction angle error within 0.53°, and achieved zero angular velocity overruns. Based on these results, it is evident that GDPTOO-S possesses superb comprehensive performance capabilities in smoothness, motion constraint satisfaction, machining accuracy, and machining efficiency, successfully avoiding sharp fluctuations in the tool axis motion and overshooting in the rotary axis motion.

However, the validation results in this study are based on MATLAB R2022a offline numerical simulations without physical cutting experiments. Although the kinematic constraint parameters are set according to typical five-axis machine tool specifications, factors such as machine tool dynamics, tool wear, thermal deformation, and cutting forces in actual machining may affect the machining quality. In addition, near the kinematic singularity where the A-axis approaches 0°, the feasible domain may become disconnected or sharply shrink, and the greedy search is prone to local locking. In the asymmetric seven-segment S-curve planning, the angular acceleration and angular jerk constraints involve multiple coupled variables, requiring iterative verification during planning, which increases computational complexity. Future work will address these limitations from three aspects: first, conducting physical cutting experiments on Al7075 or titanium alloy workpieces on a five-axis machining center to validate the engineering effectiveness by collecting vibration signals, surface roughness, and contour accuracy data; second, porting the GDPTOO-S method to an open-architecture CNC system with C++ implementation and hardware acceleration to verify real-time performance within a 1 ms interpolation cycle; third, providing a complete convergence proof of the GDPTOO method near kinematic singularities and exploring a hybrid strategy combining greedy search with dynamic programming to balance computational efficiency and global optimality.

Funding

The research is supported by: School Grants platform for Scientific Research of Shandong Huayu University of Technology in 2025: CNC tool manufacturing process research and development center PT2025KJX001.

Appendix A: Calculation of Time Intervals for Asymmetric Seven-Segment S-Curve Acceleration/Deceleration

The formulas below apply to the full seven-segment condition where the maximum acceleration a_{acc} on the acceleration side and a_{dec} on the deceleration side can be achieved. Let S denote the length of the planning segment, v_s the start feed rate, v_e the end feed rate, and F the maximum feed rate. The maximum acceleration and maximum jerk of the acceleration side are a_{acc} and j_{acc}, respectively. The maximum acceleration and maximum jerk of the deceleration side are a_{dec} and j_{dec}, respectively. The duration of each phase ($\Delta T_1 - \Delta T_7$) is calculated as Equation (A1).

In Equation (A1), S_{acc} and S_{dec} represent the total displacement of the acceleration phase and deceleration phase, respectively. If $\Delta T_2 \leq 0$ or $\Delta T_6 \leq 0$, the trajectory length is insufficient to reach the preset maximum acceleration, the uniform-acceleration or uniform-deceleration phase disappears, and the velocity profile automatically degrades to a five-segment or three-segment form. The cumulative segment transition time points (T_1–T_7, time instants counted from t=0, corresponding to the transition moments in Figure 6 of Section 2.2) are obtained by summing the durations of each phase, as given in Equation (A2).

$$\begin{cases} \Delta T_1 = \Delta T_3 = \dfrac{a_{\text{acc}}}{j_{\text{acc}}} \\ \Delta T_2 = \dfrac{F - v_s}{a_{\text{acc}}} - \dfrac{a_{\text{acc}}}{j_{\text{acc}}} \\ \Delta T_4 = \dfrac{S - S_{\text{acc}} - S_{\text{dec}}}{F} \\ \Delta T_5 = \Delta T_7 = \dfrac{a_{\text{dec}}}{j_{\text{dec}}} \\ \Delta T_6 = \dfrac{F - v_e}{a_{\text{dec}}} - \dfrac{a_{\text{dec}}}{j_{\text{dec}}} \end{cases} \tag{A1}$$

In Equation (A2), ΔT_1–ΔT_7 denote the duration of each motion phase; T_1–T_7 denote cumulative transition instants starting from t=0.

$$\begin{cases} T_1 = \Delta T_1 \\ T_2 = T_1 + \Delta T_2 \\ T_3 = T_2 + \Delta T_3 \\ T_4 = T_3 + \Delta T_4 \\ T_5 = T_4 + \Delta T_5 \\ T_6 = T_5 + \Delta T_6 \\ T_7 = T_6 + \Delta T_7 \end{cases} \tag{A2}$$

Reference

[1] Waqas M, Naseem A. Artificial intelligence in sustainable industrial transformation: A comparative study of industry 4.0 and industry 5.0. FinTech and Sustainable Innovation. 2025;1:A2. https://doi.org/10.47852/bonviewFSI52025321

[2] Osei S, Wei W, Yu J, Ding Q. A new effective decoupling method to identify the tracking errors of the motion axes of the five-axis machine tools. Journal of Intelligent Manufacturing. 2024;35(7):3377-3392. https://doi.org/10.1007/s10845-023-02220-2

[3] Mgherony A, Mikó B. Controlling the spindle speed when milling free-form surfaces using ball-end milling cutter. Acta Polytechnica Hungarica. 2023;20(6):135-149. https://doi.org/10.12700/APH.20.6.2023.6.8

[4] Karimi B, Altintas Y. Chatter avoidance by spindle speed and orientation planning in five-axis ball-end milling of thin-walled blades. Journal of Manufacturing Science and Engineering. 2024;146(2):021006. https://doi.org/10.1115/1.4063654

[5] Petráček P, Falta J, Stejskal M, Šimůnek A, Kupka P, Sulitka M. Chatter-free milling strategy of a slender Blisk blade via stock distribution optimization and continuous spindle speed change. The International Journal of Advanced Manufacturing Technology. 2023;124(3):1273-1295. https://doi.org/10.1007/s00170-022-10264-6

[6] Li M, Zhao W, Li L, He N, Stepan G. Influence of milling stability on machined surface integrity and fatigue performance of Ti-6Al-4V titanium alloy. Engineering Failure Analysis. 2025;168:109103. https://doi.org/10.1016/j.engfailanal.2024.109103

[7] Karimi B, Altintas Y. Optimal stock removal to reduce chatter and deflection errors for five-axis ball-end milling of thin-walled blades. CIRP Annals - Manufacturing Technology. 2024;73(1):293-296. https://doi.org/10.1016/j.cirp.2024.03.003

[8] Swan R, Penney J, Corson G, Nazario J, Schmitz T. Surface location error in robotic milling: Effect of combined low frequency and high frequency vibration modes. CIRP Journal of Manufacturing Science and Technology. 2024;49:203-215. https://doi.org/10.1016/j.cirpj.2024.01.011

[9] Hajdu D, Franco O, Sanz-Calle M, Totis G, Munoa J, Stépán G, Dombóvári Z. Stable tongues induced by milling tool runout. International Journal of Machine Tools and Manufacture. 2025;206:104258. https://doi.org/10.1016/j.ijmachtools.2025.104258

[10] Binder M, Henkel S, Bliedtner J, Fritzsche M, Biegler E. Investigations on process vibrations and forces and their effects on resulting optical component qualities in a CNC grinding process. The International Journal of Advanced Manufacturing Technology. 2025;136:4847-4863. https://doi.org/10.1007/s00170-025-15123-8

[11] Liu H, Xing F. Tool axis vector optimization based on automatic smoothing rotary angles of five-axis machining for laser additive manufacturing. The International Journal of Advanced Manufacturing Technology. 2024;134(7):3481-3493. https://doi.org/10.1007/s00170-024-14286-0

[12] Zhao P, Liu Z, Li Z, Cao Z. Research on tool axis vector optimization when face milling complex surfaces. The International Journal of Advanced Manufacturing Technology. 2023;128(11):5081-5099. https://doi.org/10.1007/s00170-023-12031-7
[13] Xu Y, Zaman MHM, Zhou F. Optimized dual NURBS curve interpolation for high-accuracy five-axis CNC path planning. Scientific Reports. 2025;15(1):25409. https://doi.org/10.1038/s41598-025-10937-8
[14] Xiao J, Zhang H, Li B. A dual NURBS curve five-axis interpolator with interval division under axial kinematic constraints. IEEE/ASME Transactions on Mechatronics. 2023;29(2):1170-1180. https://doi.org/10.1109/TMECH.2023.3295369
[15] Shi ZK, Zhang WJ, Ding Y. A local toolpath smoothing method for a five-axis hybrid machining robot. Science China Technological Sciences. 2023;66(3):721-742. https://doi.org/10.1007/s11431-022-2315-7
[16] Yan G, Zhang D, Xu J, Sun Y. A C3 continuous double circumscribed corner rounding method for five-axis linear tool path with improved kinematics performance. Journal of Computational Design and Engineering. 2023;10(4):1490-1506. https://doi.org/10.1093/jcde/qwad066
[17] Wang L, Zhang J, Li W, Wang Y, Zhou Y. Research on feedrate scheduling method for NURBS toolpath interpolation in high-speed CNC machining. The International Journal of Advanced Manufacturing Technology. 2025;138(7):3293-3313. https://doi.org/10.1007/s00170-025-15756-9
[18] Guevel F, Viprey F, Euzenat C, Fromentin G, Masciantonio U. Development and performance evaluation of real-time geometric error compensation through position feedback modification in 5-axis machining. The International Journal of Advanced Manufacturing Technology. 2025;137(11-12):5565-5584. https://doi.org/10.1007/s00170-025-15476-0
[19] Jacso A, Szalay T, Sikarwar BS, Phanden RK, Singh RK, Ramkumar J. Investigation of conventional and ANN-based feed rate scheduling methods in trochoidal milling with cutting force and acceleration constraints. The International Journal of Advanced Manufacturing Technology. 2023;127(1):487-506. https://doi.org/10.1007/s00170-023-11506-x
[20] Li J, Lu L, Wang S, Dai S, Sun L, Wang Z. Axis path planning of five-axis surface machining by optimizing differential vector of the axis movement considering tool posture limits. The International Journal of Advanced Manufacturing Technology. 2024;132(9):4841-4855. https://doi.org/10.1007/s00170-024-13603-x
[21] Liu Y, Sun Y. Optimization of disc tool parameters and tool path of five-axis milling of twist channels. The International Journal of Advanced Manufacturing Technology. 2024;133(9):4415-4435. https://doi.org/10.1007/s00170-024-13782-7
[22] Schuetz MJA, Brubaker JK, Katzgraber HG. Reply to: Inability of a graph neural network heuristic to outperform greedy algorithms in solving combinatorial optimization problems. Nature Machine Intelligence. 2023;5(1):26-28. https://doi.org/10.1038/s42256-022-00588-z
[23] Aslan S. Back-and-Forth (BaF): a new greedy algorithm for geometric path planning of unmanned aerial vehicles. Computing. 2024;106(8):2811-2849. https://doi.org/10.1007/s00607-024-01309-7
[24] Halinga MS, Nyobuya HJ, Uchiyama N. Generation and experimental verification of time and energy optimal coverage motion for industrial machines using a modified S-curve trajectory. The International Journal of Advanced Manufacturing Technology. 2023;125(7):3593-3605. https://doi.org/10.1007/s00170-023-10912-5
[25] Shen M, Li N, Liu S. Module design of S-curve acceleration and deceleration control based on FPGA for numerical-control systems. Journal of Computers. 2026;37(1):157-171. https://doi.org/10.63367/19911599202602370101O