Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Error-Envelope-Optimal G2 Dual Bezier Corner Transitions for High-Speed Five-Axis CNC Toolpaths

JIANGRUI DING

dingjiangrui@buaa.edu.cn

Beihang University

Xin Jiang

https://orcid.org/0000-0003-4762-3101

Yi Kuang

Jin Zhang

Hexiong Li

Yifei Hu

Research Article

Keywords: Five-axis machining, Corner smoothing, Curve parameter synchronization, Real-time interpolation

Posted Date: September 7th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10365053/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Error-Envelope-Optimal G^2 Dual Bézier Corner Transitions for High-Speed Five-Axis CNC Toolpaths

Jiangrui Ding[a], Xin Jiang[a,*], Yi Kuang[b], Jin Zhang[b], Hexiong Li[c,e,**], Yifei Hu[d]

[a]*Beijing Key Laboratory of Artificial Intelligence Innovation and Application in the Machine Tool Industry, Institute of Artificial Intelligence, Beihang University, Beijing, China*
[b]*School of Mathematical Sciences, Beihang University, Beijing, China*
[c]*School of Mathematics and Computational Science, Xiangtan University, Hunan, China*
[d]*School of Advanced Engineering, Great Bay University, Dongguan, China*
[e]*Zhengzhou Aerotropolis Institute of Artificial Intelligence, Zhengzhou, China*

Abstract

Toolpaths in practical five-axis CNC machining are typically programmed as sequences of G01 linear blocks. At block junctions, curvature discontinuities and abrupt curvature variation reduce achievable feedrates and may excite vibration in high-speed machining. This paper proposes a local corner-smoothing method that replaces each junction with parameter-synchronized dual quartic Bézier transition segments, providing G^2 continuity with continuously varying tool position and orientation. An overall error envelope is defined with respect to the programmed polyline specified by G01 block endpoints, and Bézier control points are obtained by solving a constrained optimization that minimizes differential smoothness energy to suppress curvature peaks and promote a more uniform curvature distribution within tolerance. To enhance robustness near kinematic singularities, a lightweight rotary-axis stabilization strategy based on a proxy azimuth-rate signal with soft-lock blending is incorporated to stabilize rotary commands without explicit Jacobian evaluation, while maintaining computational feasibility for online interpolation. Junctions are further classified into two types according to neighboring G01 block characteristics and treated accordingly. Simulations and five-axis machining experiments on impeller and butterfly parts demonstrate improved curvature profiles, reduced feedrate fluctuation, and higher interpolation efficiency compared with conventional corner-transition schemes.

Keywords: Five-axis machining, Corner smoothing, Curve parameter synchronization, Real-time interpolation

1. Introduction

High-precision and complex components, such as blades, impellers, and propellers, commonly found in aerospace, automotive, and mold manufacturing, are typically produced using five-axis computer numerical control (CNC) machine tools[1]. In modern CNC machining, toolpath optimization plays a critical role in balancing high precision with operational efficiency[2].

The overall performance of machining (speed, surface quality, and tool wear) is strongly influenced by toolpath continuity [3, 4, 5]. CAM toolpaths are usually issued as G01 segments, and poor corner handling can cause vibration and speed loss when tangential or curvature continuity is not maintained[6, 7]. Curvature is another key factor for speed and stability. Reducing corner curvature and its local variation improves interpolation efficiency and machining time[8, 9]. Excessive curvature increases stress, vibration, and wear, and can degrade surface finish, so low-curvature continuous transitions are essential for performance.

Five-axis toolpaths include tool tip position (TTP) and tool axis orientation (TAO). Existing smoothing methods are commonly grouped as global vs. local, and workpiece coordinate system (WCS) vs. machine coordinate system (MCS) strategies.

Global approaches smooth the full path, often via NURBS or related splines[10, 11, 12]; they offer high smoothness but complicate error control due to nonlinear arc-length parameterization[13]. Local approaches add short transition curves at corners, enabling direct error control by enforcing continuity at junctions; methods target both geometric (G) and parametric (C) continuity. G continuity ensures geometric smoothness, while C^k continuity enforces smooth time derivatives for stable motion. Achieving C^k continuity requires consistent velocity, acceleration, and jerk within each sampling interval. Representative C^2/C^3 methods include synchronized WCS splines[14], cubic B-spline approximations[15], semi-analytical corner smoothing[16], efficient C^2 interpolation[17], constrained micro-splines[18], and analytical C^3 schemes without iteration[19, 20]. Later work uses quintic B-splines or asymmetric splines to decouple or coordinate position–orientation smoothing[9, 3, 21].

In this context, G continuity is widely used because it is less restrictive than C^k continuity and offers more flexibility in control-point selection. Yong and Cheng [22] optimized a

*Corresponding author
**Corresponding author
Email addresses: jiangxin@buaa.edu.cn (Xin Jiang), hexiongli9609@xtu.edu.cn (Hexiong Li)

geometric Hermite cubic G^1 curve, and Jaklič and Žagar [23] simplified boundary conditions for smoother transitions. Lu [24] extended this to planar quintic G^2 curves. Xavier et al.[4] proposed a G^2 cubic B-spline corner rounding model with contour/orientation tolerance control in WCS. Zhao et al.[6] used curvature-continuous B-splines with five control points for real-time blending and curvature control. Bi et al.[25] developed an analytical real-time G^2 dual-Bézier method in MCS. Shi et al.[26] used quintic PH curves for analytic G^2 rounding. Huang et al.[27] replaced linear corners with a cubic B-spline to raise feedrates, and Jin et al.[28] proposed a G^2 Bézier-based feed interpolator. Sun and Altintas [29] introduced a G^3 method that optimizes feedrate along transition splines under error and kinematic limits.

Production efficiency is often assigned the highest priority, and most existing studies aim to optimize various performance indicators by minimizing machining time while maintaining trajectory continuity. In this context, the selection of machining toolpaths can be likened to the determination of an optimal racing line. A commonality between a racing line and the machining error band is that both the driver and the machine tool must operate strictly within their designated spatial boundaries. Inspired by racing cornering strategies, minimizing toolpath curvature and its derivatives greatly improves feedrate smoothness[30], thereby reducing unnecessary acceleration–deceleration cycles and enhancing machining efficiency and quality. In racing, when a vehicle enters a spin, the sustained lateral G-force reaches a critical threshold [31] :

$$\text{Lateral G value}(m/s^2) = \frac{\text{velocity}(m/s)^2}{\text{Radius of curvature}(m)}$$

The turning radius governs allowable speed: larger radii enable higher velocities, while excessive corner curvature in CNC machining triggers frequent acceleration–deceleration cycles and degrades efficiency and quality. Prior work therefore emphasizes curvature reduction under error-band constraints, spanning inscribed/inside strategies, circumscribed/outside strategies, and symmetric/centered blending. Representative inside or conservative approaches include inscribed cubic B-splines for 3-axis paths (Zhao et al.[6]) and inscribed quintic Bézier rounding (Sencer [32]). Circumscribed methods include double cubic B-splines for five-axis G^2 smoothing (Xu et al.[33]) and quintic B-spline smoothing with synchronized position–orientation for five-axis robots (Zhang et al.[9]). After curvature is controlled, overlap can be eliminated, as shown by type-based elimination (Yan et al.[34]) and LP-based elimination (Hu [21]). These categories clarify the remaining gap: most methods reduce peak curvature, but explicit control of curvature consistency within the error band, together with robust singularity handling and synchronized position–orientation smoothing, is still limited.

The racing apex, defined as the intersection point between the vehicle's trajectory and the innermost corner point, serves as a critical reference for achieving optimal cornering. The optimal racing strategy follows an outside-inside-outside trajectory, where drivers begin the turn from the outer lane, approach the apex near the inner lane, and exit the turn by returning to the outer lane[35, 36]. This trajectory pattern minimizes the turning radius, allowing the vehicle to maintain higher speeds while optimizing stability. The route diagram illustrating this strategy is shown in Figure 1.

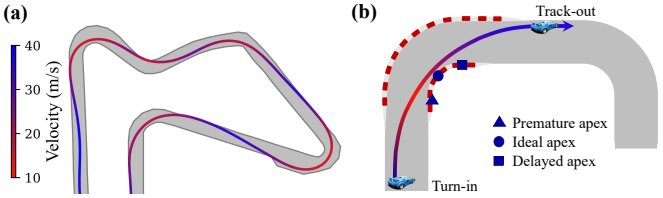


Figure 1: Optimal racing route. (a) Global trajectory with velocity distribution. (b) Local cornering strategy showing different apex positions.

This paper applies optimal racing trajectory design principles to toolpath planning. The transition curves at toolpath corners are optimized by adopting an outside-inside trajectory pattern within the error tolerance zone, minimizing curvature. This strategy enhances production efficiency while strictly adhering to specified error tolerance constraints. Additionally, the proposed methodology integrates impact energy constraints into the geometric design of transition curves, aiming to reduce curvature variations and generate low-curvature curves that maintain G^2 continuity. This WCS-based approach guarantees stable and uniform tool motion during high-speed machining, effectively mitigating the time inefficiencies and vibration effects typically caused by deceleration at angular transitions. The overall process is illustrated in the accompanying schematic diagram.

Table 1 summarizes the structured comparison of inside/outside blending strategies and highlights the innovations of the proposed method.

The remainder of this work is structured as follows: Section 2 introduces a smoothing method for the tool tip position toolpath that minimizes differential smoothness energy under error-band constraints. Section 3 presents a hyperbolic smoothing method for the tool-axis orientation toolpath, synchronized with the TTP parameterization. In Section 4, simulation and experimental results are reported. Section 5 concludes the paper.

2. Generation of G^2 Continuous Tool Tip Position Splines

Five-axis toolpath planning introduces two additional rotational axes compared to three-axis planning. The cutter-location (CL) data generated by CAM consists of the tool tip position $\mathbf{P}_n = [P_{x,n}, P_{y,n}, P_{z,n}]^T$ and tool orientation $\mathbf{O}_n = [O_{i,n}, O_{j,n}, O_{k,n}]^T$, where $n = 0, 1, ..., N$ represents discrete points along the toolpath, relative to the workpiece.

In this section, we propose a quartic Bézier method for smoothing the local corners of linear tool tip position segments, subject to user-defined error tolerance and differential smoothness energy limits. The control points of the Bézier curve are calculated with precision to fully leverage the allowable error tolerance, analogous to the cornering strategies used in high-performance racing vehicles. Concurrently, differential smoothness energy is applied to regulate the curvature of the transition curve, optimizing the overall feed rate and improving machining efficiency.

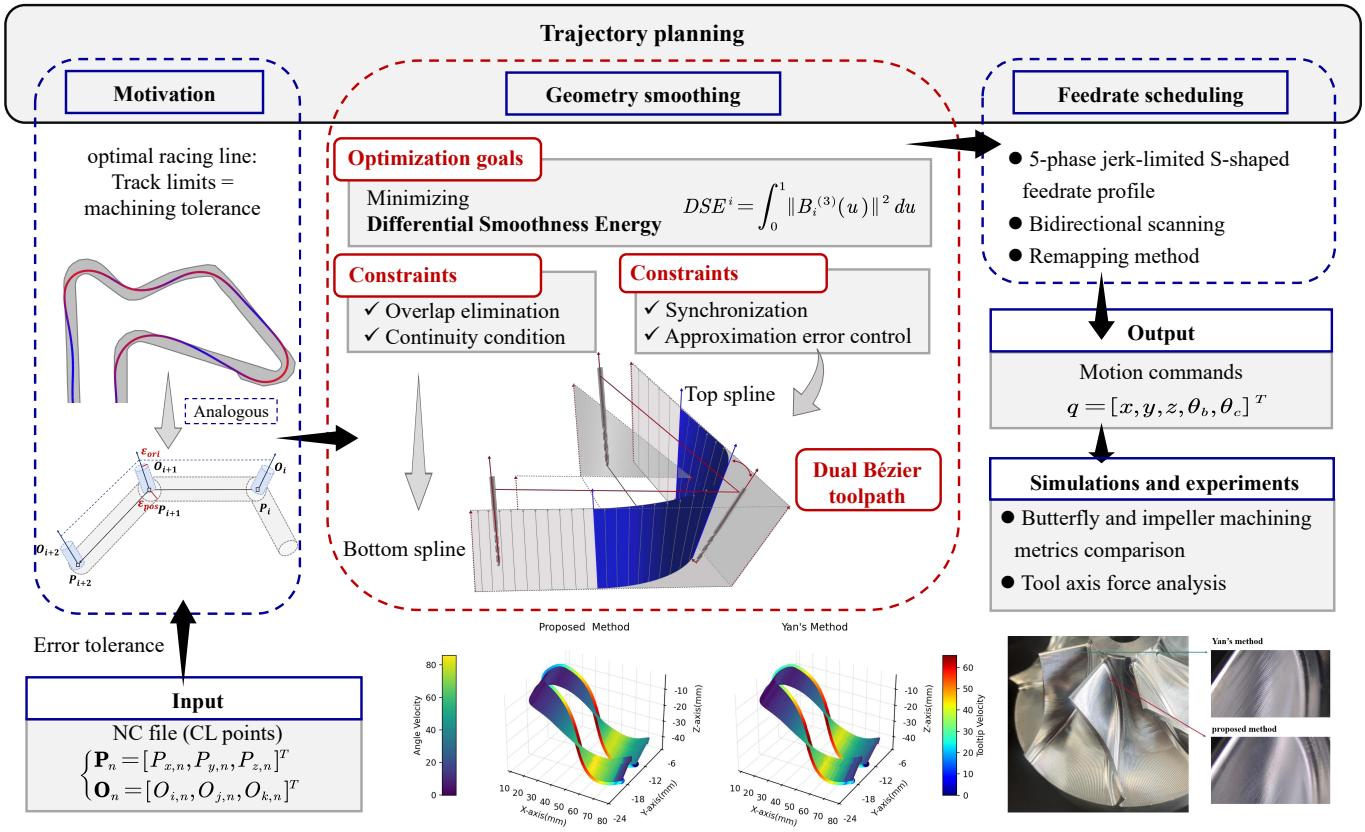


Figure 2: Flowchart of the research methodology.

2.1. Designing Corner Transition Curves

2.1.1. Parametric Position Spline Construction

The Bézier curve with degree k inserted between two liner segments is given by Farin [37] as

$$\mathbf{B}_i(u) = \sum_{j=0}^{k} \binom{k}{j} (1-u)^{k-j} u^j \mathbf{Q}_j^i, \tag{1}$$

where $0 \le u \le 1, \mathbf{Q}_j^i$ are the control points.

To ensure G^2 continuity of the toolpath at the corner, the selected control points $\{\mathbf{Q}_0, \mathbf{Q}_1, \mathbf{Q}_2\}$ and $\{\mathbf{Q}_{k-2}, \mathbf{Q}_{k-1}, \mathbf{Q}_k\}$ must be collinear respectively, necessitating a minimum of five control points ($k \ge 4$) for the entire curve. To optimize computational efficiency in machine tool processing, the lower bound of k is adopted, i.e. $k = 4$. Consequently, a quartic Bézier curve with five control points is selected as the optimal transition curve. The curve is constructed with symmetry to precisely regulate its maximum deviation, which occurs at the midpoint, ensuring it remains within the acceptable range. As illustrated inFigure 3, $\|\mathbf{Q}_1^i\mathbf{Q}_2^i\| = \|\mathbf{Q}_2^i\mathbf{Q}_3^i\| = a_i, \|\mathbf{Q}_0^i\mathbf{Q}_1^i\| = \|\mathbf{Q}_3^i\mathbf{Q}_4^i\| = a_i b_i$.

In CNC machining, accuracy and time are the primary concerns, typically reflected by contour error [38] and feedrate. Classical local methods [19] use the Hausdorff distance between the transition curve and the local line segment joining adjacent interpolation points to quantify contour error, which typically reduces to the shortest point-to-curve distance. Maximizing

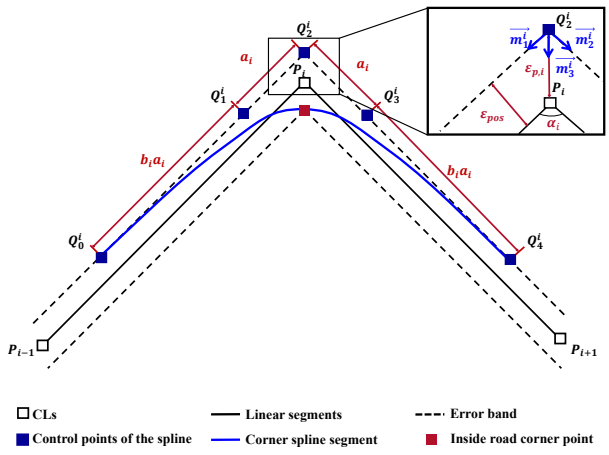


Figure 3: Toolpath position smoothing using quartic Bézier curve.

this distance within tolerance yields the smoothest admissible curve and higher speed, but the requirement that end segments align with the connecting line constrains the transition and limits tool-tip smoothness.

In practice, contour error is defined as half the width of the narrowest band enclosing the machined contour around the target path [38]. We adopt this notion and use an error-tolerance pipeline of width ε_{pos} along the line connecting tool-tip points to shape the curve, providing more intuitive accuracy control

Table 1: Comparison of transition strategies and implementation features for corner smoothing in CNC machining.

Papers	Scope	Transition type	Overlap	Singularity	Error envelope	Solver	Real-time
H. Zhao, IJMTM, 2013 [6]	3-axis	Inscribed cubic B-spline	0.5L recursive	–	2D plane	Opt(LP)	✓
B. Sencer, IJAMT, 2015 [32]	3-axis	Inscribed quintic Bézier	–	–	2D plane	Analytic	–
F. Xu, IJAMT, 2018 [33]	5-axis	Circumscribed cubic B-spline	0.5L adjustment	–	2D plane	Analytic	–
G. Yan, JMP, 2023 [34]	5-axis	Asymmetrical quintic B-spline	OE(type)	–	2D plane	Analytic	–
Y. Hu, CIRP, 2024 [21]	5-axis	Asymmetrical quintic B-spline	OE(LP)	–	2D plane	Opt(LP)	–
S. Zhang, JMP, 2024 [9]	5-DoF robot	Circumscribed quintic B-spline	self-intersection avoidance	–	3D tube	Analytic	–
This paper	5-axis	Circumscribed quartic Bézier	self-intersection avoidance	Soft-lock mitigation	3D tube	Opt(KKT)	✓

Note: OE(type): type-based overlap elimination; OE(LP): LP-based overlap elimination; Opt(LP): LP optimization; Opt(KKT): KKT/Lagrange-based optimization.

consistent with industrial standards.

Each transition curve, denoted as $\mathbf{B}_i(u)$, is constructed based on three adjacent CL points $\mathbf{P}_{i-1}, \mathbf{P}_i, \mathbf{P}_{i+1}$. The corner angle formed by these interpolation points is $\angle\mathbf{P}_{i-1}, \mathbf{P}_i, \mathbf{P}_{i+1} = \alpha_i$. The distance from point $\mathbf{P}_i$ to the outer corner point of the error band (highlighted by the red square in Fig.3) is defined with respect to the position error tolerance ε_{pos}, and is expressed as $\varepsilon_{p,i} = \varepsilon_{pos} \sin^{-1}\left(\frac{\alpha_i}{2}\right)$.

To minimize the overall curvature of the transition curve while ensuring that it remains within the defined error band, the convex hull property of the Bézier curve requires that all the five control points $\mathbf{Q}_0^i, \mathbf{Q}_1^i, \mathbf{Q}_2^i, \mathbf{Q}_3^i, \mathbf{Q}_4^i$ lie on the boundary of the error band. Moreover, the midpoint of the curve $\mathbf{B}(u)|_{u=\frac{1}{2}}$ should coincide with the inflection point within the error band. Let the unit direction vectors of $\overrightarrow{\mathbf{P}_i\mathbf{P}_{i-1}}, \overrightarrow{\mathbf{P}_i\mathbf{P}_{i+1}}, \overrightarrow{\mathbf{Q}_2^i\mathbf{P}_i}$ be denoted as $\overrightarrow{m_1^i}, \overrightarrow{m_2^i}, \overrightarrow{m_3^i}$ respectively. Denote the length of $\overrightarrow{\mathbf{Q}_1^i\mathbf{Q}_2^i}$ as a_i. Define the ratio coefficient between $\overrightarrow{\mathbf{Q}_0^i\mathbf{Q}_1^i}$ and $\overrightarrow{\mathbf{Q}_1^i\mathbf{Q}_2^i}$ as b_i:

$$b_i = \frac{\left\|\mathbf{Q}_0^i\mathbf{Q}_1^i\right\|}{\left\|\mathbf{Q}_1^i\mathbf{Q}_2^i\right\|}. \tag{2}$$

Then for the midpoint a of the curve, we have:

$$\begin{aligned}\mathbf{B}_i(u)|_{u=\frac{1}{2}} &= \frac{1}{16}\mathbf{Q}_0^i + \frac{1}{4}\mathbf{Q}_1^i + \frac{3}{8}\mathbf{Q}_2^i + \frac{1}{4}\mathbf{Q}_3^i + \frac{1}{16}\mathbf{Q}_4^i \\ &= \mathbf{P}_i + \left[\frac{(b_i+5)a_i}{8}\cos\left(\frac{\alpha_i}{2}\right) - \varepsilon_{p,i}\right]\cdot\overrightarrow{m_3^i} \\ &= \mathbf{P}_i + \varepsilon_{p,i}\cdot\overrightarrow{m_3^i}.\end{aligned} \tag{3}$$

From this, the relationship between a_i and b_i can be derived as.

$$a_i = \frac{16\varepsilon_{p,i}}{(b_i+5)\cos\left(\frac{\alpha_i}{2}\right)}. \tag{4}$$

2.1.2. Minimizing Differential Smoothness Energy

The smoothness of a curve can be quantitatively characterized by the curvature variation energy (CVE) [39], defined as:

$$\mathrm{CVE} = \int_0^1 [\kappa'(u)]^2 \mathrm{d}u, \tag{5}$$

where $\kappa(u)$ denotes the curvature, computed as $\kappa(u) = \frac{\|B'(u)\times B''(u)\|}{\|B'(u)\|^3}$.

For the quartic Bézier transition curve $\mathbf{B}(u)$, the curvature $\kappa(u)$ is a nonlinear function, and its derivaive $\kappa'(u)$ results in a high-degree rational expression. Consequently, evaluating CVE becomes computationally intensive and unsuitable for real-time machining applications.

To address this issue, we adopt the differential smoothness energy (DSE) as a practical approximation of CVE [40], expressed as

$$DSE = \int_0^1 \left\| B^{(3)}(u)\right\|^2 du. \tag{6}$$

It should be noted that the energy function defined in (6) is sensitive to parameterization. Large variations in the magnitude of the curve's derivatives may reduce the accuracy of this approximation. However, when applied to curves parameterized by arc length, the DSE formulation provides a highly accurate approximation to CVE. Therefore, it offers a computationally efficient and reliable criterion for curvature smoothness, making it particularly suitable for the proposed toolpath smoothing framework.

Theorem 1. *Consider the quartic Bézier curve* $\mathbf{B}(u)$ *defined by the five control points*$\{\mathbf{Q}_0, \mathbf{Q}_1, \mathbf{Q}_2, \mathbf{Q}_3, \mathbf{Q}_4\}$. *Then, the differential smoothness energy (DSE) of* $\mathbf{B}(u)$ *satisfied the following relation:*

$$DSE = 48\|J_0\|^2 + 144\|J_1\|^2, \tag{7}$$

where J_0, J_1 *are given by*

$$\begin{aligned}J_0 &= \mathbf{Q}_0 - 4\mathbf{Q}_1 + 6\mathbf{Q}_2 - 4\mathbf{Q}_3 + \mathbf{Q}_4, \\ J_1 &= -\mathbf{Q}_0 + 2\mathbf{Q}_1 - 2\mathbf{Q}_3 + \mathbf{Q}_4.\end{aligned} \tag{8}$$

Theorem 1 established the algebraic relationship between the differential smoothness energy of a quartic Bézier curve and its control points. Substituting the parameterized expression of the transition curve into Eq.(8), we obtain

$$\begin{cases} J_0^i = a_i\,(b_i - 3)\left(\overrightarrow{m_1^i} + \overrightarrow{m_2^i}\right), \\ J_1^i = a_i(1 - b_i)\left(\overrightarrow{m_1^i} - \overrightarrow{m_2^i}\right). \end{cases} \tag{9}$$

By further substitution and simplification, the explicit relationship between $DS\,E^i$ of the transition curve $\mathbf{B}_i(u)$ and the proportional coefficient b_i can be expressed as:

$$DS\,E^i = 49152 \cdot \varepsilon_{p,i}^2 \left[\left(\frac{b_i - 3}{b_i + 5}\right)^2 + 3\tan^2\left(\frac{\alpha_i}{2}\right)\left(\frac{1 - b_i}{b_i + 5}\right)^2\right]. \tag{10}$$

To minimize the curve's jerk energy, the corresponding b_i is obtained as:

$$b_i = \frac{9\tan^2\left(\frac{\alpha_i}{2}\right) + 12}{9\tan^2\left(\frac{\alpha_i}{2}\right) + 4}. \tag{11}$$

Accordingly, the expressions for a_i and $a_i b_i$ can be derived as:

$$\begin{cases} a_i = \frac{8\varepsilon_{p,i}\left(9\tan^2\left(\frac{\alpha_i}{2}\right)+4\right)}{\left(24\tan^2\left(\frac{\alpha_i}{2}\right)+16\right)\cos\left(\frac{\alpha_i}{2}\right)}, \\ a_i b_i = \frac{3\varepsilon_{p,i}\left(\cos^2\left(\frac{\alpha_i}{2}\right)+3\right)}{\cos\left(\frac{\alpha_i}{2}\right)\left(3-\cos^2\left(\frac{\alpha_i}{2}\right)\right)}. \end{cases} \tag{12}$$

The above derivation established how the transition curve passes through the corner point while minimizing the differential smoothness energy. However,it does not ye account for the distance between adjacent interpolation points. To avoid overlap between neighboring transition curves, the five control points of each transition curve (segment) are computed based on the angle α_i, the unit vectors $\overrightarrow{m_1^i}, \overrightarrow{m_2^i}, \overrightarrow{m_3^i}$, and the specified error tolerance $\varepsilon_{p,i}$. The resulting expressions are given by:

$$\begin{cases} \mathbf{Q}_2^i = \mathbf{P}_i - \varepsilon_{p,i} \cdot \overrightarrow{m_3^i}, \\ \mathbf{Q}_1^i = \mathbf{Q}_2^i + a_i \cdot \overrightarrow{m_1^i}, \\ \mathbf{Q}_3^i = \mathbf{Q}_2^i + a_i \cdot \overrightarrow{m_2^i}, \\ \mathbf{Q}_0^i = \mathbf{Q}_1^i + a_i b_i \cdot \overrightarrow{m_1^i}, \\ \mathbf{Q}_4^i = \mathbf{Q}_3^i + a_i b_i \cdot \overrightarrow{m_2^i}, \end{cases} \tag{13}$$

where

$$a_i = \begin{cases} \dfrac{8\varepsilon_{p,i}\left(9\tan^2\left(\frac{\alpha_i}{2}\right)+4\right)}{\left(24\tan^2\left(\frac{\alpha_i}{2}\right)+16\right)\cos\left(\frac{\alpha_i}{2}\right)}, & \Delta_{i,1} \le L_i,\ \Delta_{i,2} \le L_i, \\ \dfrac{16\varepsilon_{p,i} - L_i\cos\left(\frac{\alpha_i}{2}\right)}{4\cos\left(\frac{\alpha_i}{2}\right)}, & \Delta_{i,1} \le L_i,\ \Delta_{i,2} > L_i, \\ \dfrac{L_i\ (5\ \cos\alpha_i - 13)}{2\ (\cos\alpha_i - 17)}, & \text{else}, \end{cases}$$

$$a_i b_i = \begin{cases} \dfrac{3\varepsilon_{p,i}\left(\cos^2\left(\frac{\alpha_i}{2}\right)+3\right)}{\cos\left(\frac{\alpha_i}{2}\right)\left(3-\cos^2\left(\frac{\alpha_i}{2}\right)\right)}, & \Delta_{i,1} \le L_i,\ \Delta_{i,2} \le L_i, \\ \dfrac{5L_i\cos\left(\frac{\alpha_i}{2}\right) - 16\varepsilon_{p,i}}{4\cos\left(\frac{\alpha_i}{2}\right)}, & \Delta_{i,1} \le L_i,\ \Delta_{i,2} > L_i, \\ -\dfrac{L_i\ (3\ \cos\alpha_i + 21)}{2\ (\cos\alpha_i - 17)}, & \text{else}. \end{cases} \tag{14}$$

such that $\Delta_{i,1} = \frac{2\varepsilon_{p,i}}{\cos\left(\frac{\alpha_i}{2}\right)}, \Delta_{i,2} = \frac{-2\varepsilon_{p,i}\cos^2\left(\frac{\alpha_i}{2}\right)+9\varepsilon_{p,i}+3}{3\cos\left(\frac{\alpha_i}{2}\right)-\cos^3\left(\frac{\alpha_i}{2}\right)}, L_i = \max\left\{\frac{\|\mathbf{P}_{i-1}\mathbf{P}_i\|}{4}, \frac{\|\mathbf{P}_i\mathbf{P}_{i+1}\|}{4}\right\}$. This threshold-driven, piecewise assignment guarantees a minimum spacing between successive transition segments and thus avoids excessively long or overlapping control polygons—even near angular configurations where denominators would otherwise approach zero.

2.2. Construction of Connecting Curve

The shape of each connecting segment curve is governed by four adjacent CL points. Specifically, the points $\mathbf{P}_{i-1}, \mathbf{P}_i, \mathbf{P}_{i+1}, \mathbf{P}_{i+2}$ jointly define the tool tip transition curve $\mathbf{T}_{i+1}(u)$ for the $(i+1)$-th segment, connecting $\mathbf{P}_i$ and $\mathbf{P}_{i+1}$. As illustrated in Figure 4, the corner connection segments can be classified into two categories based on whether $\mathbf{Q}_4^{i+1}, \mathbf{Q}_0^{i+2}$ lie on the same side of the error band boundary.

This paper employs a normal vector projection method to evaluate the relative positions of control points for adjacent transition curves. Let $\overrightarrow{\boldsymbol{D_i}} = \mathbf{P}_{i+2} - \mathbf{P}_{i+1}, \overrightarrow{\boldsymbol{D}_1^i} = \mathbf{Q}_4^{i+1} - \mathbf{P}_{i+1}, \overrightarrow{\boldsymbol{D}_2^i} = \mathbf{Q}_0^{i+2} - \mathbf{P}_{i+1}$. A stable reference direction is established via the normal vector: $\overrightarrow{\boldsymbol{N_i}} = \overrightarrow{\boldsymbol{D_i}} \times (\overrightarrow{\boldsymbol{D}_1^i} + \overrightarrow{\boldsymbol{D}_2^i})$,which enables robust assessment of the distribution of control points across diverse five-axis toolpath scenarios. By defining $d_1^i = \overrightarrow{\boldsymbol{N_i}} \cdot \overrightarrow{\boldsymbol{D}_1^i}, \quad d_2^i = \overrightarrow{\boldsymbol{N_i}} \cdot \overrightarrow{\boldsymbol{D}_2^i}$, the positional relatinship is determined: if $d_1^i \cdot d_2^i < 0$, $\mathbf{Q}_4^{i+1}, \mathbf{Q}_0^{i+2}$ lie on opposite sides of the error tube; otherwise, they are on the same side.

If on the same side, the two corner transition curves are connected by a line segment. When $\mathbf{Q}_4^{i+1}$ and $\mathbf{Q}_0^{i+2}$ are on the opposite side, the connecting curve $\mathbf{T}_{i+1}(u)$ implements a "lane change" function, ensuring that the tool transitions smoothly (G^2 continuous) from the one side of the error band to the other. The control points of this connecting curve are symmetrically distributed about the corner control point while satisfying predefined length constraints within the transition region. This guarantees G^2 continuity between adjacent splines. Based on these conditions, the specific analytic expression of the six con-

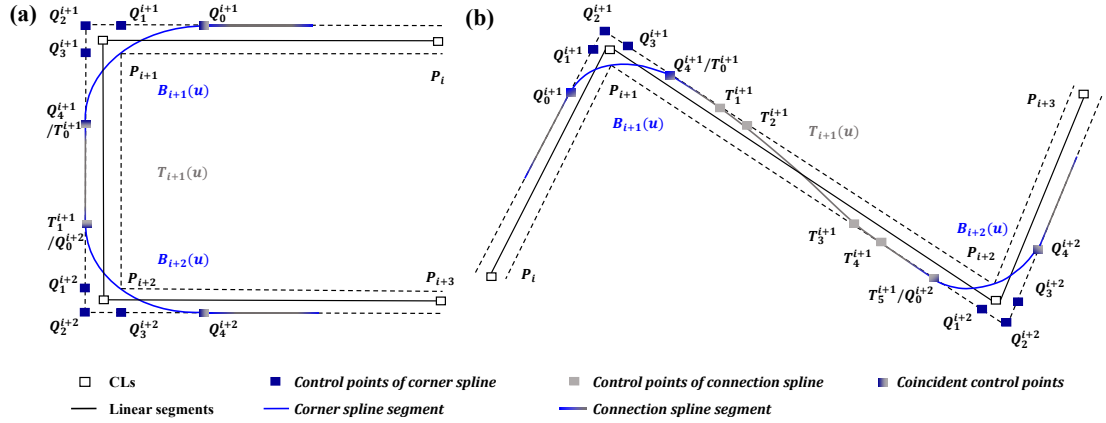


Figure 4: Transition curve types (C and N)

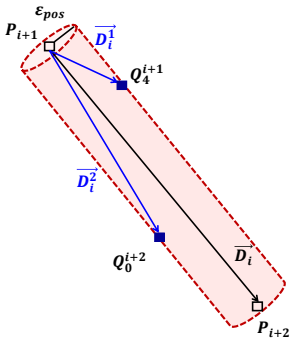


Figure 5: Illustration of normal vector projection–based judgment

trol points of $\mathbf{T}_{i+1}(u)$ are defined as (15):

$$\begin{cases} \mathbf{T}_0^{i+1} = \mathbf{Q}_4^{i+1}, \\ \mathbf{T}_1^{i+1} = \mathbf{T}_0^{i+1} + a_{i+1}b_{i+1} \cdot \overrightarrow{m_2^{i+1}}, \\ \mathbf{T}_2^{i+1} = \mathbf{T}_1^{i+1} + a_{i+1} \cdot \overrightarrow{m_2^{i+1}}, \\ \mathbf{T}_5^{i+1} = \mathbf{Q}_0^{i+2}, \\ \mathbf{T}_4^{i+1} = \mathbf{T}_5^{i+1} + a_{i+2}b_{i+2} \cdot \overrightarrow{m_1^{i+2}}, \\ \mathbf{T}_3^{i+1} = \mathbf{T}_4^{i+1} + a_{i+2} \cdot \overrightarrow{m_1^{i+2}}. \end{cases} \tag{15}$$

2.3. Smoothness Constraints for the Head and Tail of the Path

Unlike the intermediate transition curves, all control points are positioned on the boundary of the error band. The first segment curve $\mathbf{T}_0(u)$ should start at $\mathbf{P}_0$, while the last segment curve $\mathbf{T}_N(u)$ should terminate at $\mathbf{P}_N$. Therefore, the construction of the curves $\mathbf{T}_0(u)$ and $\mathbf{T}_N(u)$ requires independent consideration.

As Figure 6 demonstrates, $\mathbf{P}_0$ coincides with $\mathbf{T}_0^0$ of the first curve $\mathbf{T}_0(u)$, while $\mathbf{T}_4^0$ coincides with $\mathbf{Q}_0^1$ of the transition curve $\mathbf{B}_1(u)$. Furthermore, the intermediate control points $\mathbf{T}_3^0, \mathbf{T}_2^0$ should align with respect to $\mathbf{Q}_0^1, \mathbf{Q}_1^1, \mathbf{Q}_2^1$ respectively. The position of the remaining control point $\mathbf{T}_1^0$ (highlighted in red square in Figure 6)is determined by applying the minimal differential smoothness energy constraint.

The explicit expressions for the control points of the starting and ending curves are provided in (16) and (17), respectively,

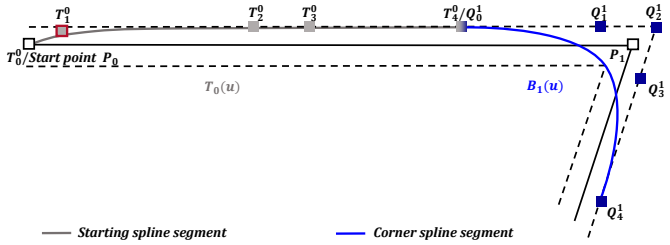


Figure 6: Control point setting of the head of CLs.

with the derivation detailed in the Appendix A.

$$\begin{cases} \mathbf{T}_0^0 = \mathbf{P}_0, \\ \mathbf{T}_4^0 = \mathbf{Q}_0^1, \\ \mathbf{T}_3^0 = \mathbf{T}_4^0 + a_1 b_1 \cdot \overrightarrow{m_1^1}, \\ \mathbf{T}_2^0 = \mathbf{T}_3^0 + a_1 \cdot \overrightarrow{m_1^1}, \\ \mathbf{T}_1^0 = \frac{5}{14}\mathbf{T}_0^0 + \frac{6}{7}\mathbf{T}_2^0 - \frac{1}{7}\mathbf{T}_3^0 + \frac{1}{14}\mathbf{T}_4^0. \end{cases} \tag{16}$$

$$\begin{cases} \mathbf{T}_4^N = \mathbf{P}_N, \\ \mathbf{T}_0^N = \mathbf{Q}_4^1, \\ \mathbf{T}_1^N = \mathbf{T}_1^N + a_N b_N \cdot \overrightarrow{m_3^N}, \\ \mathbf{T}_2^N = \mathbf{T}_1^N + a_N \cdot \overrightarrow{m_3^N}, \\ \mathbf{T}_3^N = -\frac{1}{14}\mathbf{T}_0^N - \frac{1}{7}\mathbf{T}_1^N + \frac{6}{7}\mathbf{T}_2^N + \frac{5}{14}\mathbf{T}_4^N. \end{cases} \tag{17}$$

3. Generation of G^2 Continuous Tool Orientation Curves

In this section, the five-axis toolpath is mathematically represented using dual quartic Bézier curves. To improve computational efficiency and reduce precision loss, the tool orientation is smoothed within the WCS. The bottom curve $\mathbf{B}(u)$ has been defined in Section 2, while the top curve is denoted as $\mathbf{\Theta}(u)$. They jointly define the instantaneous machining state. The difference between the two curves at the same parameters u represents the tool orientation $V(u)$. Tool axis interpolation is performed by finding geodesics on the unit sphere, ensuring the shortest transition in tool axis orientation while maintaining topological connectivity, preventing path isolation, singularities, and redundancies.

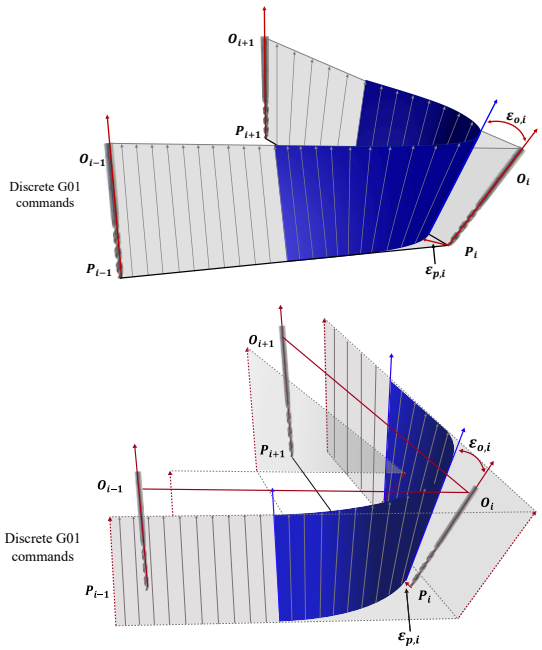


Figure 7: Illustration of five-axis corner smoothing. (Top) Transition of the corner angle is performed inside the connecting line of CL points, constrained by the position and orientation error bands. (Bottom) The proposed method achieves smooth toolpath generation within the prescribed error bands, ensuring continuity of both tool tip position and tool-axis orientation.

3.1. Parametric Orientation Spline Construction

The inserted top quartic Bézier curves can be defined with the following norm:

$$\mathbf{\Theta}_i(u) = \sum_{j=0}^{4} \binom{4}{j} (1-u)^{4-j} u^j \mathbf{\Theta}_j^i, \tag{18}$$

where $0 \le u \le 1$, $\mathbf{\Theta}_j^i$ are the control points. Analogous to the tool tip position control points settings, the control points of the upper curve also fall within the error band, ensure the G^2 continuity of the joint simultaneously, see Figure 7.

During toolpath interpolation, the bottom curve is parameterized by arc length to obtain the parameter u_{bottom} for each interpolation interval. To ensure synchronization between the two curves, the corresponding upper curve parameter u_{top} also needs to be determined. The two paths are synchronized using simple linear parameters along the straight line, i.e. $u_{top} = u_{bottom} = u$.

Since the tool length is constant, the orientation vector must maintain unit magnitude $\|V(u)\| = 1$ and lie on the surface of the unit sphere. The nominal tool orientation vector is

$$V(u) = \frac{\mathbf{L}_{\mathbf{\Theta}}(u) - \mathbf{L}_{\mathbf{B}}(u)}{\| \mathbf{L}_{\mathbf{\Theta}}(u) - \mathbf{L}_{\mathbf{B}}(u) \|} = \frac{\mathbf{\Theta}(u) - \mathbf{B}(u)}{\| \mathbf{\Theta}(u) - \mathbf{B}(u) \|}. \tag{19}$$

However, if the linear parameters are applied on the curve, it will cause significant discontinuity in direction changes, resulting in a reduced feed rate at the curve's entrance and exit. Although G^2 continuity is achieved, a degree of freedom remains in the determination of the tool direction. The next section details the method for setting the upper curve control points to ensure both synchronization and smooth orientation transitions.

3.2. Optimal Control Points Locations for Asymmetric Curves

Unlike the bottom curve, the top curve might exhibit asymmetry, as it must satisfy specific proportional constraints to preserve continuity at the curve junctions.

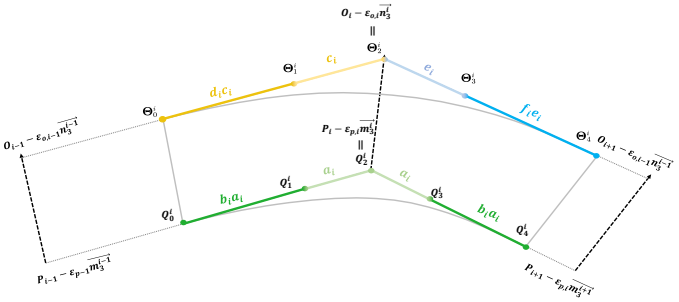


Figure 8: Control polygon lengths of transition Bézier curves.

Let the unit direction vectors $\overrightarrow{n_1^i}, \overrightarrow{n_2^i}, \overrightarrow{n_3^i}$ correspond to $\overrightarrow{\mathbf{O}_i\mathbf{O}_{i-1}}, \overrightarrow{\mathbf{O}_i\mathbf{O}_{i+1}}, \overrightarrow{\mathbf{\Theta}_2^i\mathbf{O}_i}$ respectively. The control point $\mathbf{\Theta}_2^i$ is defined by $\mathbf{\Theta}_2^i = \mathbf{O}_i - \varepsilon_{o,i} \cdot \overrightarrow{n_3^i}$. Here $\varepsilon_{o,i}$ constrains the maximal tool orientation error. Specifically, the angle between the given tool orientation vector $\overrightarrow{\mathbf{P}_i\mathbf{O}_i}$ and the tool's actual orientation $V(0.5)$, must not exceed the given tool orientation error tolerance ε_{ori}. Define the polygon edge lengths as follows: $\|\overrightarrow{\mathbf{\Theta}_1^i\mathbf{\Theta}_2^i}\|$ and $\|\overrightarrow{\mathbf{\Theta}_3^i\mathbf{\Theta}_2^i}\|$ are defined as c_i and e_i, and $\|\overrightarrow{\mathbf{\Theta}_0^i\mathbf{\Theta}_1^i}\| = d_i c_i$, $\|\overrightarrow{\mathbf{\Theta}_3^i\mathbf{\Theta}_4^i}\| = f_i e_i$. The control polygon structure of the two curves is illustrated in Figure 8. The condition for continuous direction change at the junction between a linear segment and the transition curve can be expressed as:

$$\begin{cases} \dfrac{\left\| \mathbf{\Theta}_0^i - \left(\mathbf{O}_{i-1} - \varepsilon_{o,i} \overrightarrow{n_3^{i-1}} \right) \right\|}{\left\| \mathbf{Q}_0^i - \left(\mathbf{P}_{i-1} - \varepsilon_{p,i} \overrightarrow{e_3^{i-1}} \right) \right\|} = \dfrac{c_i d_i}{a_i b_i}, \\ \dfrac{\left\| \mathbf{\Theta}_4^i - \left(\mathbf{O}_{i+1} - \varepsilon_{o,i} \overrightarrow{n_3^{i+1}} \right) \right\|}{\left\| \mathbf{Q}_4^i - \left(\mathbf{P}_{i+1} - \varepsilon_{p,i} \overrightarrow{e_3^{i+1}} \right) \right\|} = \dfrac{e_i f_i}{a_i b_i}. \end{cases} \tag{20}$$

The proof of (20) is given in Appendix B. According to (14), the coefficient a_i, b_i depend on $\alpha_i, \varepsilon_{p,i}$ and α_i respectively, i.e. $a_i(\alpha_i, \varepsilon_{p,i}), b_i(\alpha_i)$. (20) shows that $c_i d_i$ and $e_i f_i$ are functions of $a_i b_i$, which can deduce $d_i(c_i, \alpha_i, \varepsilon_{p,i}), f_i(e_i, \alpha_i, \varepsilon_{p,i})$. Substitute the relationship between control points into (8)(9) , the components of $DS\,E^i$ are

$$\begin{cases} J_0^i = c_i(d_i - 3)\overrightarrow{n_1^i} + e_i(f_i - 3)\overrightarrow{n_2^i}, \\ J_1^i = c_i(1 - d_i)\overrightarrow{n_1^i} - e_i(1 - f_i)\overrightarrow{n_2^i}. \end{cases} \tag{21}$$

Finally, the representation of $DS\,E^i$ of the asymmetric curve, which can be simplified to a function of c_i, e_i.

3.3. Smoothing Error Constraints for Orientation

The values of c_i and e_i must be selected to guarantee the upper transition curve within prescribed error tolerance. Because the blending curve can be asymmetrical, the maximum error in

the smoothing spline does not necessarily occur at its midpoint $\mathbf{\Theta}(0.5)$. To address this issue, we propose an alternative method by constructing two auxiliary symmetrical curves $\mathbf{\Theta}'(u)$ with control points located on the same side, as shown in Figure 9.

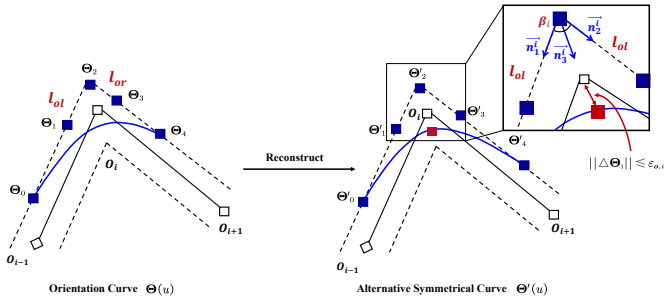


Figure 9: Alternative curve to calculate l_o.

The unit vector $\overrightarrow{n_3^i}$ is defined along the angle bisector of the line segment connecting the control points of $\mathbf{\Theta}'(u)$. Although the tool orientation smoothing error cannot be directly computed, its upper bound can be estimated using (22):

$$\arccos\left\langle \frac{\mathbf{\Theta}'(0.5)}{\|\mathbf{\Theta}'(0.5)\|}, \mathbf{O}_i \right\rangle \leq \varepsilon_{o,i}. \tag{22}$$

By taking the equality, which corresponds to the worst-case condition when the curve is exactly tangent to the error band, we can derive the control polygon length limits:

$$\begin{cases} \varepsilon_1 = \frac{(d_i+5)c_i}{8}\cos\left(\frac{\beta_i}{2}\right) - 2\varepsilon_{o,i}, \\ \varepsilon_2 = \frac{(f_i+5)e_i}{8}\cos\left(\frac{\beta_i}{2}\right) - 2\varepsilon_{o,i}. \end{cases} \tag{23}$$

Accordingly, for the upper curve to remain within the permissible orientation error tolerance, the control polygon lengths must satisfy the following constraint:

$$c_i, e_i \leq l^i_{max} = \frac{16\varepsilon_{o,i}}{(k^i_{max}+5)\cos\left(\frac{\beta_i}{2}\right)}, \tag{24}$$

where k^i_{max} is the corresponding coefficient associated with l^i_{max} (d_i or f_i) , β_i is the corner angle.

3.4. Solving the Optimization Problem for Control Points

Sections 3.2 and 3.3 establish the requirements that the tool orientation vector must vary continuously while remaining within the prescribed error tolerance. These requirements translate into scale and length constraints on the control polygon. By exploiting these relationships, the initial four unknown variables governing the placement of control points in an asymmetric curve are reduced to two.

To resolve the remaining unknowns, we employ a Lagrange multiplier optimization framework. The optimization objective is to minimize the differential smoothness energy, subject to the error constraints derived earlier. The Lagrangian function is defined as:

$$L = E_{jerk} + \lambda_1\varepsilon_1 + \lambda_2\varepsilon_2, \tag{25}$$

with the Karush–Kuhn–Tucker (KKT) conditions expressed as:

$$\text{s.t.} \begin{cases} \frac{\partial L}{\partial c}, \frac{\partial L}{\partial e}, \frac{\partial L}{\partial \lambda_1}, \frac{\partial L}{\partial \lambda_2} = 0, \\ \varepsilon_1, \varepsilon_2 \leqslant 0, \\ \lambda_1, \lambda_2 \geqslant 0, \\ \varepsilon_1\lambda_1, \varepsilon_2\lambda_2 = 0. \end{cases} \tag{26}$$

Solving this constrained optimization problem yields the values of c_i and e_i. The corresponding parameters d_i and f_i can then be obtained using (20). At this stage, the positions of all control points on the upper curve are fully determined, completing the construction of the dual-curve five-axis smoothing model.

4. Simulation and Experimental Results

In this section, we compare the performance of the proposed tool path smoothing method with four widely used local smoothing techniques, both at the three-axis and five-axis levels. The transition methods selected for comparison are summarized in Table 1, which outlines key characteristics and performance metrics of each approach.

The proposed five-axis path smoothing algorithm was implemented in Python 2023 and executed on a PC equipped with an Intel i7-13700H 2.40 GHz CPU and 2 GB of SDRAM to generate interpolation points. The algorithm's performance was experimentally validated on a Litz LU-400 vertical machining center, which is equipped with an SPIKE® sensor tool holder for real-time cutting force detection during machining operations. The spindle is subjected to simultaneous tension (axial load), torque (rotational load), and bending moments (rotational load). Data is wirelessly transmitted to a receiver, enabling visual diagnostics and facilitating process optimization. Figure 10 shows the motion modes for each axis of the five-axis CNC system. The machine's linear axes include the X-axis, which supports the Y-axis and the rotary table, and the Z-axis, which carries the spindle. The rotational axes consist of the B-axis (tilt axis) and the C-axis (rotary axis), with the C-axis mounted on the B-axis. The workpiece is fixed on the rotary table.

In 5-axis milling, the CAM system generates CL points from the Computer-Aided Design (CAD) model of the part. The primary function of a 5-axis CNC program is to convert these CL data points into machine command data (G-code).

The configuration of rotary commands such as AB, BC, or AC, varies with the structural design of the machine tool. In this study, a B-C rotary type machine is employed. Through forward kinematics transformation, the five-axis motion commands $q = [x, y, z, \theta_b, \theta_c]^T$ in the machine coordinate system can be converted into the tool tip position $P = [P_x, P_y, P_z]^T$ and tool orientation $O = [O_i, O_j, O_k]^T$ in the workpiece coordinate system. DH notations provide a theoretical foundation for the precise mapping of machine commands to workpiece

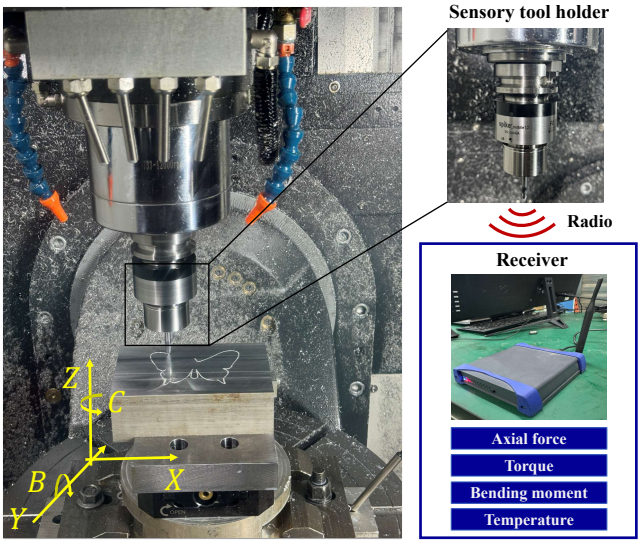


Figure 10: Five-axis experimental test bed.

coordinates in five-axis machining[41].

$$\begin{cases} P_x = C_{\theta_C} x + S_{\theta_C} C_{\theta_B} y + S_{\theta_C} S_{\theta_B} z + L_x, \\ P_y = -S_{\theta_C} x + C_{\theta_C} C_{\theta_B} y + C_{\theta_C} S_{\theta_B} z + L_y, \\ P_z = -S_{\theta_B} y + C_{\theta_B} z + L_z, \\ O_i = S_{\theta_B} S_{\theta_C}, \\ O_j = S_{\theta_B} C_{\theta_C}, \\ O_k = C_{\theta_B}, \end{cases} \tag{27}$$

where $\left(L_x, L_y, L_z\right)$ represent the pivot point offset of the rotation axes B and C relative to the workpiece coordinate system, which remain constant vectors. $S_{(\cdot)}, C_{(\cdot)}$ represent $\sin(\cdot), \cos(\cdot)$ respectively.

The inverse kinematics, which transforms tool tip positions and tool orientations to five-axis motion commands, can be listed as:

$$\begin{cases} x = C_{\theta_C}(P_x - L_x) - S_{\theta_C}(P_y - L_y), \\ y = S_{\theta_C} C_{\theta_B}(P_x - L_x) + C_{\theta_C} C_{\theta_B}(P_y - L_y) - S_{\theta_B}(P_z - L_z), \\ z = S_{\theta_C} S_{\theta_B}(P_x - L_x) + C_{\theta_C} S_{\theta_B}(P_y - L_y) + C_{\theta_B}(P_z - L_z), \\ \theta_B = \arctan 2(\sqrt{{O_i}^2 + {O_j}^2}, O_k), \\ \theta_C = \arctan 2(O_i, O_j). \end{cases} \tag{28}$$

In Eq. (28), the rotary command $\theta_C = \arctan 2(O_i, O_j)$ becomes ill-conditioned when the tool axis approaches the machine Z direction, i.e., the XY-projection magnitude $\rho = \sqrt{O_i^2 + O_j^2}$ tends to zero. To suppress abrupt C-axis motion near this singular region without Jacobian evaluation, an engineering-oriented hybrid strategy consisting of joint soft-locking, azimuth-rate-based velocity prediction, and weighted blending is adopted.

Let $\psi_k = \arctan 2(O_{i,k}, O_{j,k})$ be the azimuth of the XY projection at the k-th interpolation step, T_s the interpolation period, and wrap$(\cdot)$ map an angle to $(-\pi, \pi]$:

$$\rho_k = \sqrt{O_{i,k}^2 + O_{j,k}^2}, \tag{29}$$

$$\dot{\psi}_k = \frac{\text{wrap}(\psi_k - \psi_{k-1})}{T_s}. \tag{30}$$

We use $\dot{\psi}_k$ as a physical proxy of the C-axis angular rate and apply a decay factor to realize soft-locking. Specifically, define a blending weight $\lambda_k \in [0, 1]$ based on the projection magnitude:

$$\lambda_k = \text{clamp}\left(\frac{\rho_1 - \rho_k}{\rho_1 - \rho_0}, 0, 1\right), \tag{31}$$

where $0 < \rho_0 < \rho_1$ are user-defined thresholds (lock region / normal region). The predicted C-axis command is

$$\theta_{C,k}^{\text{pred}} = \theta_{C,k-1} + T_s \cdot \text{sat}\left(\dot{\psi}_k, \omega_C^{\max}\right), \tag{32}$$

with sat$(\cdot)$ bounding the rate by $\omega_C^{\max}$. The final command is obtained by soft-lock blending

$$\theta_{C,k} = \lambda_k \theta_{C,k-1} + (1 - \lambda_k)\theta_{C,k}^{\text{pred}}. \tag{33}$$

When $\rho_k \le \rho_0$, $\lambda_k \approx 1$ and θ_C is nearly frozen (soft-lock); when $\rho_k \ge \rho_1$, $\lambda_k \approx 0$ and θ_C follows the predicted update smoothly. This lightweight procedure is suitable for point-wise real-time interpolation loops since it only involves vector projection, angle wrapping, and scalar blending.

The bidirectional scanning speed planning technique[42] is employed to meet both kinematic and geometric constraints. The nominal feedrate is determined by (34), as outlined in [6].

$$v_m = \min\left(\frac{2}{T_s}\sqrt{\frac{1}{\kappa_{\max}^2} - \left(\frac{1}{\kappa_{\max} - \delta}\right)^2}, \sqrt{\frac{A_{\max}}{\kappa_{\max}}}, \sqrt{\frac{J_{\max}}{\kappa_{\max}^2}}, V_{\max}\right), \tag{34}$$

where T_s is the sampling interval, $\kappa_{\max}$ is the curvature of the segmentation point, $V_{\max}, A_{\max}, J_{\max}$ denote the limits of the feedrate, normal acceleration, normal jerk, and δ is the limit of chord error, respectively. Once the tool position movement is determined, the corresponding tool orientation movement is synchronized by aligning its curve parameters with those of the tool position.

The proposed algorithm was validated by smoothing three distinct test toolpaths composed of linear segments, evaluated on both three-axis and five-axis systems. The results demonstrated that the algorithm significantly improves machining efficiency and surface integrity while offering broad applicability across multi-axis machining operations.

4.1. Results of Three-Axis Machining

This section provides a comprehensive comparison of three methods in terms of path representation and kinematic performance, using a rhombic shape as a demonstration and butterfly-shaped workpieces as a machining example. In line with this paper, we also consider the inscribed corner rounding method proposed by Zhao et al.[6] and the circumscribed corner rounding method introduced by Xu et al.[33], both of which aim to

enhance the smoothness of local toolpaths. (35) and (36) provide an explicit formulation for the curvature extreme value of the Zhao's method.

$$\kappa_{\max} = \frac{4\sin\beta}{3d_2\cos^2\beta}, \tag{35}$$

$$d_2 = \frac{2\varepsilon_p}{\cos\left(\frac{\alpha}{2}\right)}, \tag{36}$$

where $\beta = \frac{\pi-\alpha}{2}$. After substitution and simplification, derive the expression relating the maximum curvature to the tolerance error ε_p and the angle α in Zhao's technique:

$$\kappa_{\max,Zhao} = -\frac{4\cos\left(\frac{\alpha}{2}\right)^2}{3\varepsilon_p\left(\cos\left(\alpha\right)-1\right)}. \tag{37}$$

The expression for the extreme value of curvature in our method is given by (38), with a detailed derivation provided in the Appendix C .

$$\kappa_{\max,our} = \frac{\left(9\tan\left(\frac{a}{2}\right)^2+8\right)\left(27\tan\left(\frac{a}{2}\right)^2+16\right)\cos\left(\frac{a}{2}\right)^2}{432\varepsilon_{p,i}\left(1-\cos\left(a\right)\right)\left(\tan\left(\frac{a}{2}\right)^2+\frac{2}{3}\right)^2}. \tag{38}$$

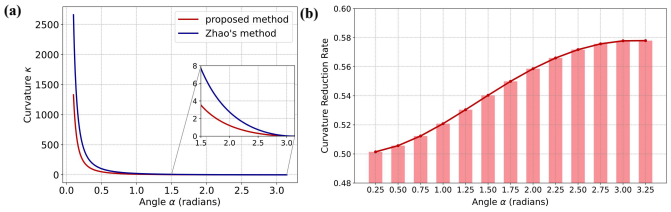


Figure 11: Curvature extreme value comparison.

According to (37)(38), when the interpolation point angle $\alpha > 0$, the condition $\kappa_{\max,our} < \kappa_{\max,Zhao}$ is always valid. Figure 11(a) shows the variation in curvature extreme values for both approaches as a function of the interpolation point angle. Similarly, Figure 11(b) presents the relationship between the curvature decay rate and arc length for both methodologies, with data points sampled at intervals of 15 degs. The results indicate a positive correlation between the interpolation point angle and the rate of curvature reduction. As indicated by (34), the reduction in curvature leads to a higher velocity, resulting in shorter machining time. To further validate the effectiveness of the proposed corner smoothing algorithm under different conditions, various user-defined error tolerances and kinematic limitations are set for the two testing toolpaths, as summarized in Table 2.

In the rhombic demo, the sampling interval is set to 1 ms. Figure 12 illustrates the variation of the curvature comb with respect to the arc length of the path. The curvature comb visualizes both the magnitude and direction of curvature at uniformly distributed points along a curve. As shown in Figure 12 (b), the curvature extreme has been reduced, accompanied by a more gradual rate of change. The maximum curvature of the entire toolpath for the proposed method, Xu's method, and Zhao's method are 12.73, 30.79, and 66.67 $(1/mm)$, respectively.

Simulations	Error tolerance	Kinematic constraints
rhombic	$\varepsilon_{pos} = 0.01$ mm	$V_{\max} = 30$ mm/s $A_{\max} = 300$ mm/s^2 $J_{\max} = 1000$ mm/s^3
butterfly	$\varepsilon_{pos} = 0.1$ mm	$V_{\max} = 10$ mm/s $A_{\max} = 100$ mm/s^2 $J_{\max} = 500$ mm/s^3
impeller	$\varepsilon_{pos} = 0.01$ mm $\varepsilon_{ori} = 0.01°$	$V_{\max} = 100$ mm/s $A_{\max} = 1000$ mm/s^2 $J_{\max} = 10000$ mm/s^3

Table 2: Constraints and parameters used in three-axis simulation.

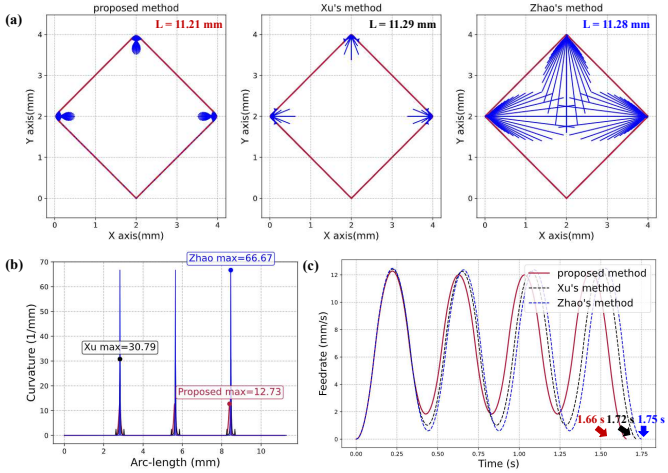


Figure 12: The result of smoothing the rhombic demo. (a) Toolpath with curvature combs: the red curve denotes the smoothed trajectory, while the blue lines indicate the magnitude and orientation of curvature; (b) Variation of curvature magnitude along arc length; (c) the feedrate profiles.

The method is further validated through simulations and experiments on a complex butterfly-shaped 3-axis workpiece, with the G-code containing 100 points. The interpolation period is set to 1 ms. On the PC platform, our algorithm required a total of 0.678 s to compute the interpolation points, including the time for path generation and bidirectional scan velocity planning. Figure 13 shows the corresponding curvature and tangential kinematic profiles comparison. In all settings, the proposed method maintains the defined tolerances and kinematic indicators within acceptable bounds, while achieving lower curvature and more consistent motion performance compared to the other two approaches. It is evident that, in regions with high curvature extremes, our method allows for a higher feedrate. The results in a production efficiency increase of 7.83% and 11.23% compared to the other two methods, respectively. By integrating feedrate information obtained via bidirectional scanning, the final interpolation points are generated at uniform time intervals,

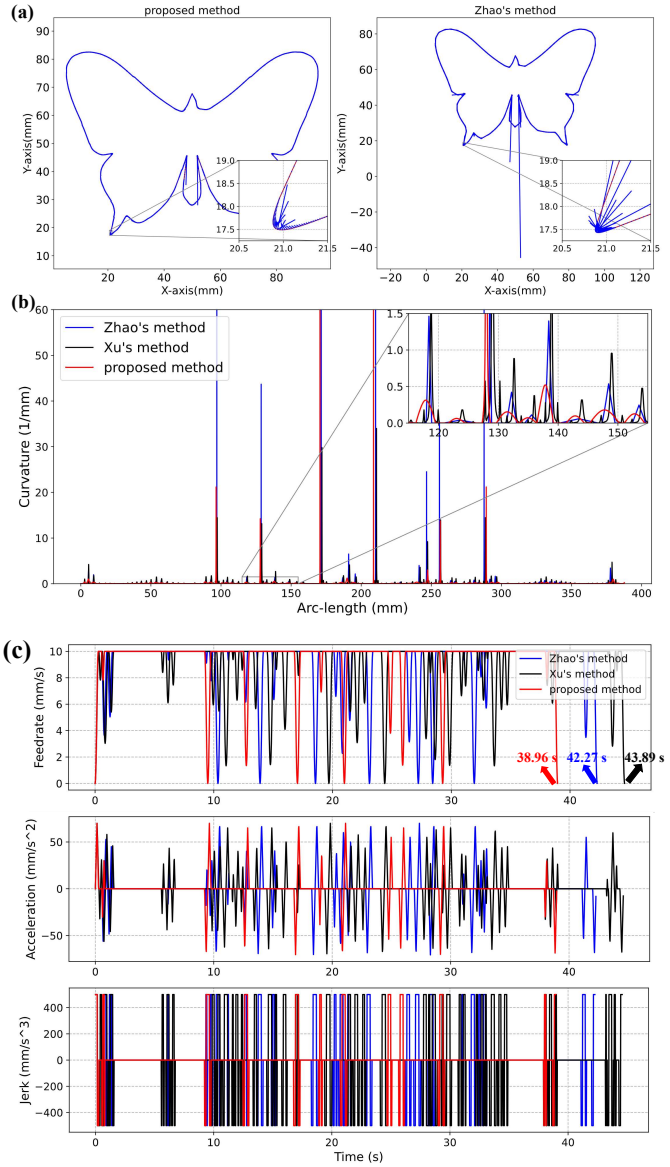


Figure 13: The result of smoothing a butterfly-shaped workpiece. (a) The curvature comb; (b) the magnitude of curvature; (c) the tangential kinematic profiles.

as illustrated in Figure 14. To enhance clarity in observation, a constant offset of 1 was applied to both x- and y-axis. As shown in Part I of Figure 14, it is evident that the proposed interpolation path is smoother. And in part II, the proposed interpolation toolpath appears sparser compared to the other two methods, indicating a higher interpolation speed.

During the machining process, the forces measured by the dynamometer at the tool tip are analyzed to optimize cutting parameters and predict tool wear and machining quality. In this experiment, tip milling is employed. Accordingly, the cutting force parameters monitored by the dynamometer include axial tension and torque along the Z-axis. The measured axial tension refers to the tensile or compressive force exerted on the tool along its axis. Excessive axial tension can lead to tool deformation and vibrations, ultimately degrading surface quality. Torque, on the other hand, represents the cutting resistance-induced moment acting around the tool's rotational axis. High

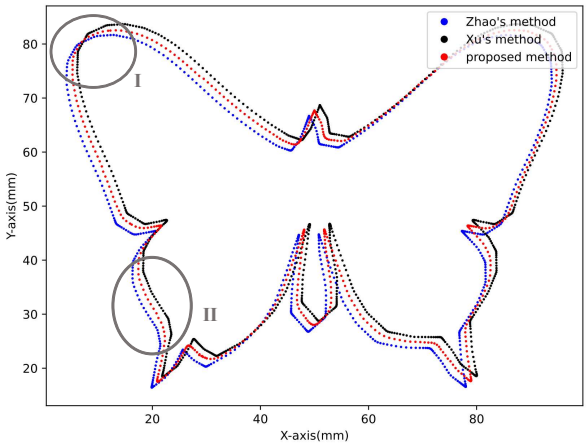


Figure 14: Interpolation results.

or unstable torque may cause spindle overload, affecting the machine's lifespan. Therefore, real-time monitoring of tension and torsion via the dynamometer provides an effective means of evaluating experimental machining parameters. Figure 15 (a) shows that while the toolpaths generated by both methods exhibit substantial similarity, the proposed approach achieves notably smoother corner transitions. The integration of the proposed path-smoothing approach and look-ahead interpolation algorithm leads to a notable improvement in tracking efficiency while simultaneously maintaining high tracking accuracy. A slight time increase was observed in the force measurement experiment due to the inclusion of tool entry and exit times. In terms of force feedback, the proposed method results in lower axial force and more stable tension on the tool. These results validate the effectiveness of the proposed method in improving path smoothness and optimizing motion performance.

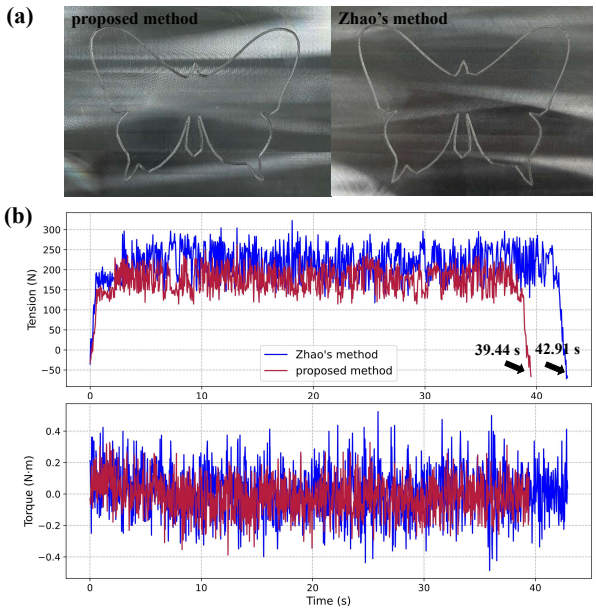


Figure 15: The butterfly-shaped milling results. (a) The machined workpiece toolpath; (b) comparison of tool forces (tension and torque) between the two methods.

4.2. Results of Five-axis Machining

This section presents simulations and five-axis actual machining of a typical turbo impeller part to verify the effectiveness of the proposed algorithm in path smoothing and various kinematic metrics. The turbo impeller, shown in Figure Figure 16, has a diameter of 158 mm, and the a height of 32 mm, with a tool cutting edge length of 35 mm. G01 commands of the toolpath are generated by the UG NX CAM module, containing 212 spatially distributed CLs. The machining error of the finished path is influenced by the acceptable tolerance error during the path generation stage, which should be defined with precision. Considering the actual working conditions, ε_{pos} is set to 0.01 mm and ε_{ori} is set to 0.01 deg. The total time to compute over 4,000 interpolation points is 2.273 s. Figure 16 illustrates the starting points (u = 0) of each segment of the Bézier curves located at the top and bottom.

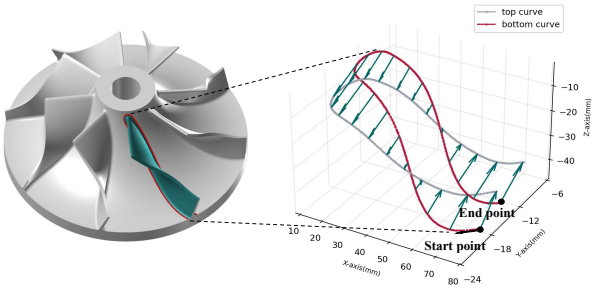


Figure 16: toolpath for flank milling of one blade in an impeller.

The five-axis smoothing algorithms developed by Yan and Hu are both implemented for a comparative analysis. Specific kinematics details are provided to offer a clearer insight into the generated transition curves. The comparison of the velocities, accelerations, and jerks of each axis for the three methods is presented in Figure 17. The tool tip velocity and tool orientation angular velocity of the proposed method and Yan's method are shown in Figure 18.

The execution time of the G01 block is 4.471 s and 4.316 s when smoothed using Yan's and Hu's transition methods, respectively, compared to 4.063 s with the proposed synchronization method. This results in a 9.1% and 5.9% improvement in the average feedrate, respectively. Additionally, the proposed method significantly suppresses feedrate fluctuations, as demonstrated in Figure 17 and Figure 18. The improved acceleration smoothness and controlled jerk contribute to enhanced tracking accuracy and surface quality.

The impeller is manufactured via side-edge milling. In contrast to tip milling, where the primary cutting direction is axial, the bending moment provides a more representative characterization of the load behavior in side milling. Figure 19 presents the experimentally measured bending moment fluctuations and the resulting workpiece surface quality. The proposed method consistently outperforms Yan's method in both metrics.

4.3. Real-time feasibility and computational cost

To assess the deployability in point-wise interpolation loops, we profile the per-cycle computation time under an interpolation

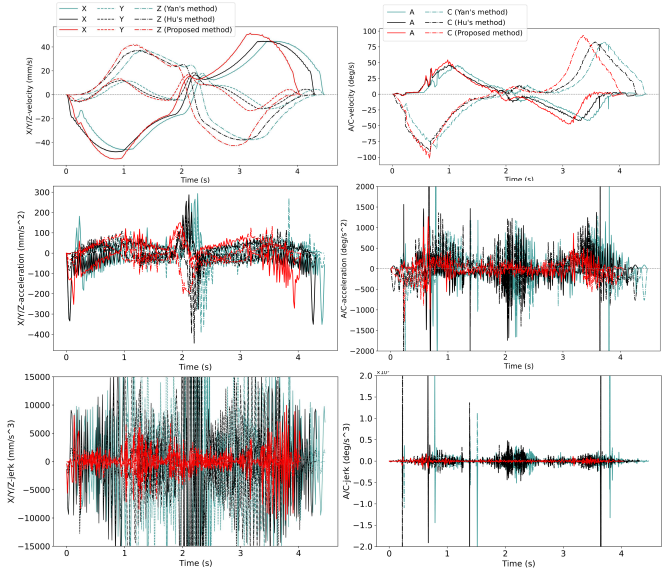


Figure 17: The axial kinematic performance (velocity/acceleration/jerk profile for X/Y/Z/B/C-axis).

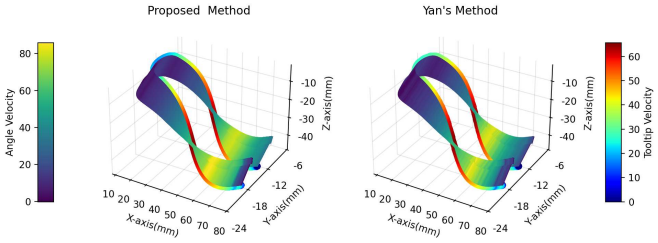


Figure 18: Comparison of tooltip and angular velocities with Yan's approach.

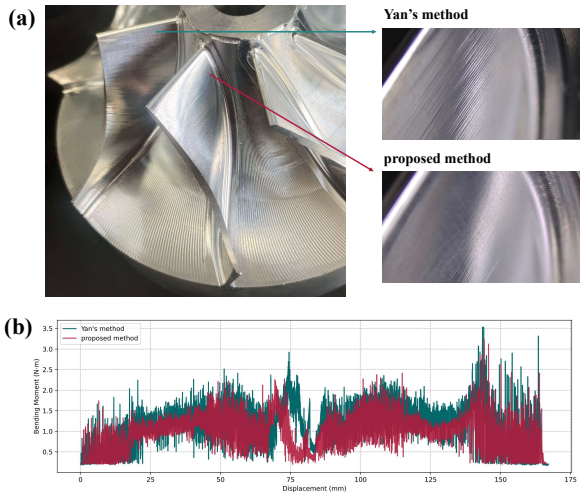


Figure 19: Bending moment response and machining quality comparison of the proposed and Yan's methods.

period of T_s = 1.0 ms. Fig. 20 reports the per-cycle timeline and CPU utilization for both the 3-axis and 5-axis online executions. The measured average computation time is 42 μs (3-axis) and 116 μs (5-axis), corresponding to 4.2% and 11.6% of the cycle budget, respectively, leaving sufficient real-time margin.

We further conduct a sensitivity test by varying the interpolation period T_s and re-evaluating the CPU utilization. As shown in Fig. 20, the current setting (T_s = 1.0 ms) remains well below the cycle-time limit with a conservative safety margin.

5. Conclusions

This paper proposes a five-axis local toolpath smoothing method for G01 corner transitions. Inspired by racing-line principles, parameter-synchronized dual quartic Bézier curves are constructed to achieve G^2 continuity in both tool position and orientation. Control points are determined under an overall error envelope by minimizing the differential smoothness energy, aiming to optimize the global curvature profile and dynamic behavior within tolerance rather than enforcing strict local collinearity at every segment. A lightweight singularity-mitigation strategy based on proxy azimuth-rate soft-lock blending is further integrated to stabilize rotary-axis commands without Jacobian evaluation. Simulations, machining tests, and real-time profiling confirm reduced corner curvature peaks, smoother feedrate behavior, and sufficient computational margin for online execution, indicating strong industrial applicability to high-speed freeform-surface and continuous-contour machining where dynamic stability and efficiency are prioritized.

CRediT authorship contribution statement

Jiangrui Ding: Writing – original draft, Methodology, Software, Formal analysis, Visualization, Data curation. **Xin Jiang**: Supervision, Project administration, Funding acquisition. **Yi Kuang**: Methodology, Software, Formal analysis, Visualization. **Jin Zhang**: Validation, Methodology, Software, Formal analysis **Hexiong Li**: Methodology, Formal analysis. **Yifei Hu**: Methodology, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors acknowledge Xinxiang Vocational and Technical College for providing the experimental setup and the testing place. This research received funding from the National Key Research and Development Program of China [Grant No. 2024YFA1013102], the Beijing Natural Science Foundation [Grant No. Z240002], and the National Natural Science Foundation of China [Grant No.12171023].

References

[1] S. Yuwen, J. Jinjie, X. Jinting, C. Mansen, N. Jinbo, Path, feedrate and trajectory planning for free-form surface machining: A state-of-the-art review, Chinese Journal of Aeronautics 35 (8) (2022) 12–29.
[2] I. Lazoglu, C. Manav, Y. Murtezaoglu, Tool path optimization for free form surface machining, CIRP annals 58 (1) (2009) 101–104.
[3] A. Yuen, K. Zhang, Y. Altintas, Smooth trajectory generation for five-axis machine tools, International Journal of Machine Tools and Manufacture 71 (2013) 11–19.
[4] X. Beudaert, S. Lavernhe, C. Tournier, 5-axis local corner rounding of linear tool path discontinuities, International Journal of Machine Tools and Manufacture 73 (2013) 9–16.
[5] S. Tajima, B. Sencer, Accurate real-time interpolation of 5-axis tool-paths with local corner smoothing, International Journal of Machine Tools and Manufacture 142 (2019) 1–15.
[6] H. Zhao, L. M. Zhu, H. Ding, A real-time look-ahead interpolation methodology with curvature-continuous b-spline transition scheme for cnc machining of short line segments, International Journal of Machine Tools and Manufacture 65 (2013) 88–98.
[7] J. Huang, X. Du, L. M. Zhu, Real-time local smoothing for five-axis linear toolpath considering smoothing error constraints, International Journal of Machine Tools and Manufacture 124 (2018) 67–79.
[8] G. Zhao, Z. Wei, X. Jing, a. et, An analytical local tool path smoothing method with curvature control, The International Journal of Advanced Manufacturing Technology 127 (3) (2023) 1355–1375.
[9] S. Q. Zhang, Z. K. Shi, Y. Ding, Toolpath smoothing with reduced curvature and synchronized motion for hybrid robots, Journal of Manufacturing Processes 109 (2024) 181–197.
[10] X. Du, J. Huang, L. M. Zhu, a. et, An error-bounded b-spline curve approximation scheme using dominant points for cnc interpolation of micro-line toolpath, Robotics and Computer-Integrated Manufacturing 64 (2020) 101930.
[11] D. N. Song, J. W. Ma, Y. G. Zhong, a. et, Global smoothing of short line segment toolpaths by control-point-assigning-based geometric smoothing and fir filtering-based motion smoothing, Mechanical Systems and Signal Processing 160 (2021) 107908.
[12] Y. Hu, X. Jiang, Z. Zheng, A global toolpath smoothing for micro-line toolpath based on edge error estimation and regional optimization, Journal of Manufacturing Processes 141 (2025) 957–975.
[13] G. Zhang, J. Gao, L. Zhang, a. et, Generalised nurbs interpolator with nonlinear feedrate scheduling and interpolation error compensation, International Journal of Machine Tools and Manufacture 183 (2022) 103956.
[14] R. V. Fleisig, A. D. Spence, A constant feed and reduced angular acceleration interpolation algorithm for multi-axis machining, Computer-Aided Design 33 (1) (2001) 1–15.
[15] V. Pateloup, E. Duc, P. Ray, Bspline approximation of circle arc and straight line for pocket machining, Computer-Aided Design 42 (9) (2010) 817–827.
[16] G. Li, H. Liu, S. Liu, a. et, A general c2 continuous toolpath corner smoothing method for a 5-dof hybrid robot, Mechanism and Machine Theory 169 (2022) 104640.
[17] S. Sun, P. Zhao, T. Zhang, a. et, Smoothing interpolation of five-axis tool path with less feedrate fluctuation and higher computation efficiency, Journal of Manufacturing Processes 109 (2024) 669–693.
[18] S. Tulsyan, Y. Altintas, Local toolpath smoothing for five-axis machine tools, International Journal of Machine Tools and Manufacture 96 (2015) 15–26.
[19] J. Yang, A. Yuen, An analytical local corner smoothing algorithm for five-axis cnc machining, International Journal of Machine Tools and Manufacture 123 (2017) 22–35.
[20] X. Wang, B. Liu, X. Mei, a. et, Global smoothing for five-axis linear paths based on an adaptive nurbs interpolation algorithm, The International Journal of Advanced Manufacturing Technology 114 (7) (2021) 2407–2420.
[21] Y. Hu, X. Jiang, G. Huo, a. et, Enhancing five-axis cnc toolpath smoothing: Overlap elimination with asymmetrical b-splines, CIRP Journal of Manufacturing Science and Technology 52 (2024) 36–57.
[22] J. H. Yong, F. F. Cheng, Geometric hermite curves with minimum strain energy, Computer Aided Geometric Design 21 (3) (2004) 281–301.
[23] G. Jaklič, E. Žagar, Planar cubic g1 interpolatory splines with small strain

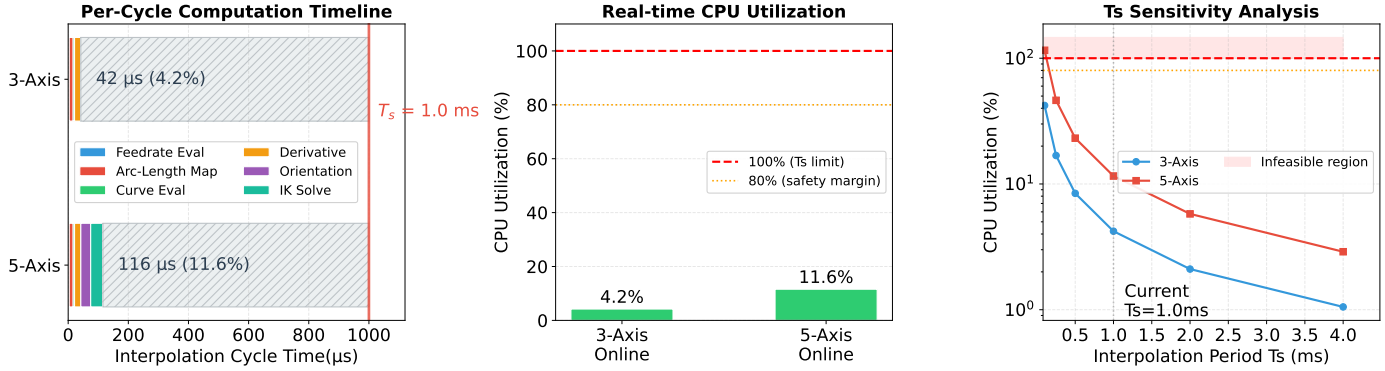


Figure 20: Real-time profiling: per-cycle computation timeline, CPU utilization under T_s = 1.0 ms, and sensitivity to interpolation period T_s.

energy, Journal of computational and applied mathematics 235 (8) (2011) 2758–2765.

[24] L. Lu, Planar quintic g2 hermite interpolation with minimum strain energy, Journal of Computational and Applied Mathematics 274 (2015) 109–117.

[25] Q. Z. Bi, J. Shi, Y. H. Wang, a. et, Analytical curvature-continuous dual-bézier corner transition for five-axis linear tool path, International Journal of Machine Tools and Manufacture 91 (2015) 96–108.

[26] J. Shi, Q. Z. Bi, L. M. Zhu, a. et, Corner rounding of linear five-axis tool path by dual ph curves blending, International Journal of Machine Tools and Manufacture 88 (2015) 223–236.

[27] J. Huang, X. Du, L. M. Zhu, Real-time local smoothing for five-axis linear toolpath considering smoothing error constraints, International Journal of Machine Tools and Manufacture 124 (2018) 67–79.

[28] Y. Jin, S. Zhao, Y. Wang, An optimal feed interpolator based on g 2 continuous bézier curves for high-speed machining of linear tool path, Chinese Journal of Mechanical Engineering 32 (2019) 1–10.

[29] S. Sun, Y. Altintas, A g3 continuous tool path smoothing method for 5-axis cnc machining, CIRP Journal of Manufacturing Science and Technology 32 (2021) 529–549.

[30] A. Heilmeier, A. Wischnewski, L. Hermansdorfer, J. Betz, M. Lienkamp, B. Lohmann, Minimum curvature trajectory planning and control for an autonomous race car, Vehicle System Dynamics (2020).

[31] C. Lopez, Going faster!, Robert Bentley Inc., MA (1997).

[32] B. Sencer, K. Ishizaki, E. Shamoto, A curvature optimal sharp corner smoothing algorithm for high-speed feed motion generation of nc systems along linear tool paths, The International Journal of Advanced Manufacturing Technology 76 (9) (2015) 1977–1992.

[33] F. Xu, Y. Sun, A circumscribed corner rounding method based on double cubic b-splines for a five-axis linear tool path, The International Journal of Advanced Manufacturing Technology 94 (2018) 451–462.

[34] G. Yan, J. Liang, J. Xu, Overlap classification-based and kinematically coordinated corner rounding using double asymmetrical transitions for five-axis short-segmented tool path, Journal of Manufacturing Processes 85 (2023) 1077–1095.

[35] A. Jain, M. Morari, Computing the racing line using bayesian optimization[c]//2020 59th ieee conference on decision and control (cdc), IEEE (2020) 6192–6197.

[36] N. Baumann, E. Ghignone, J. Kühne, a. et, Forzaeth race stack–scaled autonomous head-to-head racing on fully commercial off-the-shelf hardware, Journal of Field Robotics (2024).

[37] G. Farin, Curves and surfaces for computer-aided geometric design: a practical guide, Elsevier, 2014.

[38] B. S. ISO, Test code for machine tools–part 1: Geometric accuracy of machines operating under no-load or quasi-static conditions, ISO230-1 (2012).

[39] G. Farin, Geometric hermite interpolation with circular precision, Computer-Aided Design 40 (4) (2008) 476–479.

[40] D. J. Walton, D. S. Meek, G 2 blends of linear segments with cubics and pythagorean-hodograph quintics, International Journal of Computer Mathematics 86 (9) (2009) 1498–15.

[41] J. Yang, Y. Altintas, Generalized kinematics of five-axis serial machines with non-singular tool path generation, International Journal of Machine Tools and Manufacture 75 (2013) 119–132.

[42] X. Du, J. Huang, L. M. Zhu, A complete s-shape feed rate scheduling approach for nurbs interpolator, Journal of Computational Design and Engineering 2 (4) (2015) 206–217.

Appendix A. Derivation of Theorem 1

The cubic derivative of the quartic Bézier curve $\mathbf{B}(u)$ and the control point are related as follows:

$$\begin{aligned}\mathbf{B}^{(3)}(u) = 24[&(u-1)\mathbf{Q}_0 + (-4u+3)\mathbf{Q}_1 \\ &+ (6u-3)\mathbf{Q}_2 + (-4u+1)\mathbf{Q}_3 + u\mathbf{Q}_4],\end{aligned} \tag{A.1}$$

note $A = \begin{bmatrix} u-1 & -4u+3 & 6u-3 & -4u+1 & u \end{bmatrix}^T$, $Q = \begin{bmatrix} \mathbf{Q}_0 & \mathbf{Q}_1 & \mathbf{Q}_2 & \mathbf{Q}_3 & \mathbf{Q}_4 \end{bmatrix}^T$, then it can be expressed by

$$\|\mathbf{B}^{(3)}(u)\|^2 = 24^2 \left[A^\top Q Q^\top A\right] = 576 \operatorname{tr}\left[Q^\top A A^\top Q\right], \tag{A.2}$$

where

$$AA^\top = \begin{bmatrix} (u-1)^2 & (3-4u)(u-1) & (u-1)(6u-3) & (1-4u)(u-1) & u(u-1) \\ (3-4u)(u-1) & (3-4u)^2 & (3-4u)(6u-3) & (1-4u)(3-4u) & u(3-4u) \\ (u-1)(6u-3) & (3-4u)(6u-3) & (6u-3)^2 & (1-4u)(6u-3) & u(6u-3) \\ (1-4u)(u-1) & (1-4u)(3-4u) & (1-4u)(6u-3) & (1-4u)^2 & u(1-4u) \\ u(u-1) & u(3-4u) & u(6u-3) & u(1-4u) & u^2 \end{bmatrix}. \tag{A.3}$$

Record

$$M = 6\int_0^1 AA^\top \, \mathrm{d}u = \begin{bmatrix} 2 & -5 & 3 & 1 & -1 \\ -5 & 14 & -12 & 2 & 1 \\ 3 & -12 & 18 & -12 & 3 \\ 1 & 2 & -12 & 14 & -5 \\ -1 & 1 & 3 & -5 & 2 \end{bmatrix}, \tag{A.4}$$

then

$$\begin{aligned}\int_0^1 \|\mathbf{B}^{(3)}(u)\|^2 \, \mathrm{d}u &= 96 \operatorname{tr}\left[Q^\top M Q\right] \\ &= 48\left(\|U_1 Q\|^2 + 3\|U_2 Q\|^2\right),\end{aligned} \tag{A.5}$$

where $U_1 = [1, -4, 6, -4, 1]^\top$, $U_2 = [-1, 2, 0, -2, 1]^\top$. let $J_0 = \mathbf{Q}_0 - 4\mathbf{Q}_1 + 6\mathbf{Q}_2 - 4\mathbf{Q}_3 + \mathbf{Q}_4$, $J_1 = -\mathbf{Q}_0 + 2\mathbf{Q}_1 - 2\mathbf{Q}_3 + \mathbf{Q}_4$, the differential smoothness energy of $\mathbf{B}(u)$ can be written as

$$DSE = 48 \parallel J_0 \parallel^2 + 144 \parallel J_1 \parallel^2 . \tag{A.6}$$

Note

$$\begin{aligned} A &= \mathbf{Q}_0 + 6\mathbf{Q}_2 - 4\mathbf{Q}_3 + \mathbf{Q}_4, \\ B &= -\mathbf{Q}_0 - 2\mathbf{Q}_3 + \mathbf{Q}_4, \end{aligned} \tag{A.7}$$

then

$$\min DSE = \min\left(48(A - 4\mathbf{Q}_1)^2 + 144(B + 2\mathbf{Q}_1)^2\right). \tag{A.8}$$

The solution is

$$\mathbf{Q}_1 = \frac{2A - 3B}{14} = \frac{5}{14}\mathbf{Q}_0 + \frac{6}{7}\mathbf{Q}_2 - \frac{1}{7}\mathbf{Q}_3 - \frac{1}{14}\mathbf{Q}_4. \tag{A.9}$$

Similarly, let

$$\begin{aligned} A' &= \mathbf{Q}_0 + 6\mathbf{Q}_2 - 4\mathbf{Q}_1 + \mathbf{Q}_4, \\ B' &= -\mathbf{Q}_0 - 2\mathbf{Q}_1 + \mathbf{Q}_4, \end{aligned} \tag{A.10}$$

have

$$\min DSE = \min\left(48(A\prime - 4\mathbf{Q}_3)^2 + 144(B\prime + 2\mathbf{Q}_3)^2\right). \tag{A.11}$$

The solution is

$$\begin{aligned} \mathbf{Q}_3 &= \frac{2A' - 3B'}{14} \\ &= -\frac{1}{14}\mathbf{Q}_0 - \frac{1}{7}\mathbf{Q}_1 + \frac{6}{7}\mathbf{Q}_2 + \frac{5}{14}\mathbf{Q}_4. \end{aligned} \tag{A.12}$$

Appendix B. Derivation of Eq. (20)

To ensure continuity, the first derivative of the orientation vector must be equal at the junction of the transition section at the entrance and exit. The left-hand derivative at the junction $[\mathbf{Q}_0^i, \mathbf{\Theta}_0^i]^T$ between the straight line and curve is

$$\begin{aligned} \frac{dV(u_L)}{ds} = &\frac{L_{\mathbf{\Theta}}\prime(u_L) - L_{\mathbf{B}}\prime(u_L)}{\|L_{\mathbf{\Theta}}(u_L) - L_{\mathbf{B}}(u_L)\|} \\ &\times \left\|\mathbf{Q}_0^i - (\mathbf{P}_{i-1} - \varepsilon_{p,i-1}\overrightarrow{m_3^{i-1}})\right\|, \quad u_L \in [0,1], \end{aligned} \tag{B.1}$$

where

$$\begin{cases} L_{\mathbf{\Theta},i}\prime(u_L) = \mathbf{\Theta}_0^i - \left(\mathbf{O}_{i-1} - \varepsilon_{o,i-1}\overrightarrow{n_3^{i-1}}\right), \\ L_{\mathbf{B},i}\prime(u_L) = \mathbf{Q}_0^i - \left(\mathbf{P}_{i-1} - \varepsilon_{p,i-1}\overrightarrow{m_3^{i-1}}\right). \end{cases} \tag{B.2}$$

The connection point occurs at the end of the straight line segment, where $u_L = 1$ and $\mathbf{L}_{\mathbf{\Theta_i}}(1) = \mathbf{\Theta}_0^i, \mathbf{L}_{\mathbf{B_i}}(1) = \mathbf{Q}_0^i$. The equation can be rewritten as:

$$\begin{aligned} &\frac{\left(\mathbf{\Theta}_0^i - \left(\mathbf{O}_{i-1} - \varepsilon_{o,i-1}\overrightarrow{n_3^{i-1}}\right)\right) - \left(\mathbf{Q}_0^i - \left(\mathbf{P}_{i-1} - \varepsilon_{p,i-1}\overrightarrow{m_3^{i-1}}\right)\right)}{\left\|\mathbf{\Theta}_0^i - \mathbf{Q}_0^i\right\|} \\ &\quad \Big/\left\|\mathbf{Q}_0^i - \left(\mathbf{P}_{i-1} - \varepsilon_{p,i-1}\overrightarrow{m_3^{i-1}}\right)\right\| \\ &= \frac{\frac{\mathbf{\Theta}_0^i - \left(\mathbf{O}_{i-1} - \varepsilon_{o,i-1}\overrightarrow{n_3^{i-1}}\right)}{\mathbf{Q}_0^i - \left(\mathbf{P}_{i-1} - \varepsilon_{p,i-1}\overrightarrow{m_3^{i-1}}\right)} - 1}{\left\|\mathbf{\Theta}_0^i - \mathbf{Q}_0^i\right\|} \cdot \frac{\mathbf{Q}_0^i - \left(\mathbf{P}_{i-1} - \varepsilon_{p,i-1}\overrightarrow{m_3^{i-1}}\right)}{\left\|\mathbf{Q}_0^i - \left(\mathbf{P}_{i-1} - \varepsilon_{p,i-1}\overrightarrow{m_3^{i-1}}\right)\right\|} \\ &= \frac{\frac{\left\|\mathbf{\Theta}_0^i - \left(\mathbf{O}_{i-1} - \varepsilon_{o,i-1}\overrightarrow{n_3^{i-1}}\right)\right\|\left(-\overrightarrow{n_1^i}\right)}{\left\|\mathbf{Q}_0^i - \left(\mathbf{P}_{i-1} - \varepsilon_{p,i-1}\overrightarrow{m_3^{i-1}}\right)\right\|\left(-\overrightarrow{m_1^i}\right)} - 1}{\left\|\mathbf{\Theta}_0^i - \mathbf{Q}_0^i\right\|} \cdot \left(-\overrightarrow{m_1^i}\right). \end{aligned} \tag{B.3}$$

The right-hand derivative at the junction can be interpreted as:

$$\frac{dV(u_C)}{ds} = \frac{\mathbf{\Theta}_i'(u_C) - \mathbf{B}_i'(u_C)}{\| \mathbf{\Theta}_i(u_C) - \mathbf{B}_i(u_C) \|} \Big/ \| \mathbf{B}_i'(u_C) \|, \quad u_C \in [0,1], \tag{B.4}$$

Considering that the connection is the entrance of the curve, with $u_C = 0$, we substitute this into the equation:

$$\begin{aligned} &\frac{4\left[\left(\mathbf{\Theta}_1^i - \mathbf{\Theta}_0^i\right) - \left(\mathbf{Q}_1^i - \mathbf{Q}_0^i\right)\right]}{\|\mathbf{\Theta}_i(0) - \mathbf{B}_i(0)\|} \Big/ 4\left\|\mathbf{Q}_1^i - \mathbf{Q}_0^i\right\|\} \\ &= \frac{\frac{\mathbf{\Theta}_1^i - \mathbf{\Theta}_0^i}{\mathbf{Q}_1^i - \mathbf{Q}_0^i} - 1}{\|\mathbf{\Theta}_i(0) - \mathbf{B}_i(0)\|} \cdot \frac{\mathbf{Q}_1^i - \mathbf{Q}_0^i}{\left\|\mathbf{Q}_1^i - \mathbf{Q}_0^i\right\|} \\ &= \frac{\frac{c_i d_i\left(-\overrightarrow{n_1^i}\right)}{a_i b_i\left(-\overrightarrow{m_1^i}\right)} - 1}{\|\mathbf{\Theta}_i(0) - \mathbf{B}_i(0)\|} \cdot \left(-\overrightarrow{m_1^i}\right). \end{aligned} \tag{B.5}$$

Let the derivatives from the left and right be equal to ensure the continuity of the directional change. The continuity at the exit junction will be proved in the same manner. Thus, the proof of (20) is complete.

Appendix C. Derivation of curvature explicit expression (38)

From the symmetry property of the transition curve $\mathbf{B}(u)$, it can be determined that the extreme curvature of the proposed curve is achieved at $\mathbf{B}(u)|_{u=\frac{1}{2}}$. In the case of the Bézier curve $\mathbf{B}(u)$, the first-order derivative of the midpoint is calculated using the parameter representation method described in Section 2.1. This is then simplified by substituting into the equations given in (13)(14).

$$\begin{aligned} \mathbf{B}\prime(u)|_{u=\frac{1}{2}} &= -\frac{1}{2}Q_0 - Q_1 + Q_3 + \frac{1}{2}Q_4 \\ &= \left(\frac{1}{2}a + \frac{3}{2}\right) b(\overrightarrow{m_2} - \overrightarrow{m_1}). \end{aligned} \tag{C.1}$$

Similarly, the second-order derivative of the midpoint is

$$\begin{aligned} \mathbf{B}''(u)\big|_{u=\frac{1}{2}} &= 12\left(\frac{1}{4}Q_0 - \frac{1}{2}Q_2 + \frac{1}{4}Q_4\right) \\ &= 3\,(a+1)\,b(\overrightarrow{m_2} + \overrightarrow{m_1}). \end{aligned} \tag{C.2}$$

It can thus be demonstrated that the curvature extreme of the proposed method. The curvature at the midpoint of the curve can be expressed as follows:

$$\begin{aligned} \kappa_{\max,our} &= \kappa\left(\mathbf{B}(u)|_{u=\frac{1}{2}}\right) \\ &= \frac{\left\|\mathbf{B}\prime(u)|_{u=\frac{1}{2}} \times \mathbf{B}''(u)|_{u=\frac{1}{2}}\right\|}{\left\|\mathbf{B}\prime(u)|_{u=\frac{1}{2}}\right\|^3} \\ &= \frac{(9\tan\left(\frac{a}{2}\right)^2 + 8)\left(27\tan\left(\frac{a}{2}\right)^2 + 16\right)\cos\left(\frac{a}{2}\right)^2}{432\varepsilon_{pos}\,(1 - \cos(a))\left(\tan\left(\frac{a}{2}\right)^2 + \frac{2}{3}\right)^2}. \end{aligned} \tag{C.3}$$