ARCHIVE OF MECHANICAL ENGINEERING

DOI: 10.24435/ame/2026.157609 2026, Vol. 73, No. 1, pp. 167–194

Xu-Lin CAI [1,2], Wen-Ming ZHANG [1], Wen-An YANG [2], You-Peng YOU [2]

Catmull–Rom spline-based C^3 continuous tool path smoothing method for five-axis CNC milling

Received 9 October 2025, **Revised** 21 January 2026, **Accepted** 27 January 2026, **Published online** 8 March 2026

Keywords: tool path smoothing, Catmull–Rom spline, five-axis NC milling, C^3 continuity, jerk continuity

Tool-path smoothing is essential for ensuring continuous motion at transition corners between linear segments, since kinematic discontinuities degrade both machining efficiency and surface quality. Most existing spline-based methods achieve only G^2 or C^2 continuity and therefore produce discontinuous jerk, which can excite high-order structural resonances. Achieving true C^3 continuity remains challenging, particularly because synchronizing tool-tip position and orientation is complicated by the nonlinear relationship between arc length and spline parameterization. This study presents an analytical Catmull–Rom (CR) spline–based method for C^3-continuous tool-path smoothing in five-axis CNC milling. Transition corners are replaced by adjustable Catmull–Rom (ACR) splines, whose control points and tuning parameters are designed or optimized to constrain the deviation from the original path. The remaining linear segments are also substituted with ACR splines to enforce position–orientation synchronization, with control points that can be chosen analytically to guarantee zero synchronization error. The proposed method is fully analytical and non-iterative. Numerical simulations demonstrate that the generated tool paths satisfy prescribed geometric tolerances, produce smooth and continuous jerk profiles, and achieve exact synchronization between tool-tip position and orientation.

✉ Xu-Lin CAI, e-mail: caixulin@nuaa.edu.cn

[1]Luoyang Institute of Electro-Optical Equipment AVIC, LuoYang, People's Republic of China

[2]College of Mechanical and Electrical Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, People's Republic of China

1. Introduction

With the advancement of numerical control systems and equipment, modern computer numerical control (CNC) machine tools increasingly support spline interpolation to meet the demands for lightweight, integrated, complex, and high-precision manufacturing of aerospace structural components [1]. Nevertheless, discrete linear motion commands (G01) generated by CAM software remain the predominant tool paths in CNC machining [2]. Tangential discontinuities between successive linear segments both lengthen cycle times and degrade surface finish during milling [3]. Therefore, it is essential to smooth these discrete linear commands to attain high-order geometric and parametric continuity before sending them to the numerical control system.

Tool-path smoothing methods in the literature can be broadly classified into global and local approaches according to the spline construction technique. Global methods represent the entire discrete tool path as one continuous spline-examples include NURBS [4], B-splines [5], and polynomial splines [6]. While these approaches readily provide overall smoothness, they often struggle to evaluate and constrain deviation errors between the smoothed and original tool paths. By contrast, local methods improve smoothness by inserting micro-splines at junctions between adjacent linear segments [7, 8]. Local smoothing more directly addresses deviation constraints, but it raises the challenge of ensuring high-order continuity at the interfaces between inserted splines and linear segments.

Numerous local smoothing techniques have been developed for three-, four-, and five-axis machines to achieve different orders of motion continuity [9–12]. In three-axis machining, transition corners are commonly smoothed by inserting curves such as quartic Bézier splines [13], cubic B-splines [14], or Pythagorean-Hodograph (PH) curves [15]; only the tool tip position requires smoothing because tool orientation is usually fixed. Multi-axis machines, however, require smoothing of both tool tip position and tool orientation, and independently smoothing those two aspects introduces additional synchronization challenges compared with three-axis cases [16, 17]. This complexity is especially pronounced in five-axis machining, which has attracted substantial research due to its complex kinematics and drive constraints [16, 18, 19]. Drive and axis constraints (including axis jerk constraints) have been explicitly considered in several studies that combine geometric smoothing with drive-level constraints or feed rate interpolation techniques to ensure feasible axis motions during execution [18, 20]. Reparameterization and feed rate scheduling methods have also been used to generate jerk-minimized or drive-aware multi-axis toolpaths (for example, Hashemian et al. on reparameterization for jerk-minimized flank milling [19], and Beudaert et al. on feed rate interpolation with axis jerk constraints [20]).

Continuity is typically described using geometric continuity (G^n) and parametric continuity (C^n), where Cn is the stricter requirement: C^n continuity implies G^n continuity, but not vice versa. Jin et al. [21] proposed a G^2 local smoothing

method for five-axis tool paths using double G^2 continuous cubic Bézier splines, and Huang et al. [22] introduced a real-time G^2 approach replacing cornered segments with cubic B-splines. To attain higher continuity, Zhao et al. [23] presented an analytical C^2 local smoothing method combining an asymmetric cubic B-spline with symmetric quartic spherical Bézier splines, and Huang et al. [24] proposed a C^2 method using a high-order airthoid spline with peak-constrained jerk. Yang et al. [25] developed an analytical smoothing algorithm for 6-DOF robots using a fifth-degree PH-spline that smooths both position and orientation in the workpiece coordinate system (WCS) and synchronizes them by replacing remaining linear segments with specially constructed PH-splines. Although these G^2/C^2 methods ensure continuous acceleration, they generally result in discontinuous jerk. As Yuen et al. [26] demonstrated, discontinuous jerk can excite high-order resonances in the mechanical system, increasing vibration and tracking error. To address this, later studies have pursued C^3 continuity: Zhang et al. [27] devised an analytical G^3 local smoothing method using two specially designed quartic curves; Tulsyan et al. [28] inserted quintic and septic micro-splines for position and orientation to realize C^3 continuity; Yang et al. [29] proposed a C^3 method based on a specially designed quintic micro-spline; and more recently Hu et al. [30] developed a real-time C^3 local smoothing technique using C^3 PH-splines. Despite their effectiveness, these C^3 approaches typically employ complex synchronization techniques that often require iterative computation of spline control points, which reduces computational efficiency and can produce unsatisfactory synchronization errors.

Beyond pure smoothing and continuity concerns, five-axis tool-path generation must also address inverse-kinematics multiplicity, optimal orientation selection, and collision/clearance issues. Several works study multiple inverse-kinematics solutions and multipoint five-axis tool positioning strategies to exploit redundancy and improve feasibility (e.g., Sharma et al. [31]). Optimal orientation control and collision-aware orientation selection have been addressed to improve surface quality and avoid interference (Farouki and Li [32]; Ezair and Elber [33]), and more recent research has extended collision-free motion planning to specialized cutters such as toroidal tools (Chichell et al. [34]). These studies highlight that smoothing algorithms for five-axis milling need to integrate with orientation selection, kinematic solution choice, and collision avoidance strategies to be practical in industrial settings. Catmull–Rom (CR) splines are an attractive choice for local smoothing because they are interpolating splines with intuitive local control and well-studied geometric properties [35]. Their interpolatory nature makes it straightforward to bound deviations from the original polyline by manipulating control points and tension parameters, and their local support simplifies real-time implementations.

In this study we propose a Catmull–Rom (CR) spline-based C^3 continuous tool-path smoothing method for five-axis milling. Tool orientations provided by CAM software in the WCS are first transformed to the machine coordinate system (MCS) via inverse kinematics. Tool tip position and orientation are then smoothed independently using an adjustable Catmull–Rom (ACR) spline method. Control

points are used to constrain local deviation errors, and these errors are further reduced via a simple, efficient optimization. Finally, the remaining linear segments are replaced by ACR splines so that tool tip position and orientation are synchronized with the tool-tip displacement while preserving C^3 continuity. Importantly, the control points for these splines can be selected directly without additional computation, substantially reducing processing time.

The remainder of this paper is organized as follows. Section 2 introduces the ACR spline-based corner smoothing method for tool tip position and orientation, including a brief overview of the ACR spline. Section 3 presents a deviation-error optimization method to constrain the smoothing-induced errors in tool tip position and orientation. Section 4 describes the synchronization method that aligns tool tip position and orientation with tool-tip displacement. Section 5 provides numerical simulations that validate the proposed method, and Section 6 concludes the study.

2. Analytical C^3 continuous corner smoothing method

The motion control of NC milling is realized by driving the rotation or translation of each joint axis in MCS, while the existing commercial CAM software generally generates the tool path in WCS. For five-axis machine tools, the drive commands $\mathbf{q} = \left[X,\ Y,\ Z,\ R_A,\ R_C\right]^{\mathrm{T}}$ in MCS are utilized to control the motion of machining, while the tool path is made up of tool-tip position $\mathbf{p} = \left[x,\ y,\ z\right]^{\mathrm{T}}$ and tool orientation $\mathbf{O} = \left[O_i,\ O_j,\ O_k\right]^{\mathrm{T}}$ in WCS. The transformation between these two coordinate systems are realized through the forward and inverse kinematics models. In this study, the table-tilting five-axis machine shown in Fig. 1 is taken as an example for analysis.

The forward kinematics transformation from MCS to WCS can be calculated by using Denavit-Hartenberg matrix:

$$\begin{aligned}
x &= -\cos(R_C)X - \cos(R_A)\sin(R_C)Y + \sin(R_A)\sin(R_C)Z \\
&\quad - \sin(R_A)\sin(R_C)L_{ya}\,, \\
y &= \sin(R_C)X - \cos(R_A)\sin(R_C)Y + \sin(R_A)\cos(R_C)Z \\
&\quad - \sin(R_A)\cos(R_C)L_{ya}\,, \\
z &= \sin(R_A)Y + \cos(R_A)Z - \cos(R_A)L_{ya} - L_{ac}\,, \\
O_i &= \sin(R_A)\sin(R_C), \\
O_j &= \sin(R_A)\cos(R_C), \\
O_k &= \cos(R_A),
\end{aligned} \tag{1}$$

where L_{ya} and L_{ac} are offsets determined by the geometry of the rotary table. The smoothing of tool orientation in the tool path smoothing method developed in this study is carried out in MCS, so it is necessary to transform the generated tool path by CAM from WCS to MCS through the inverse kinematics transformation firstly, which can be obtained by solving Eq. (1):

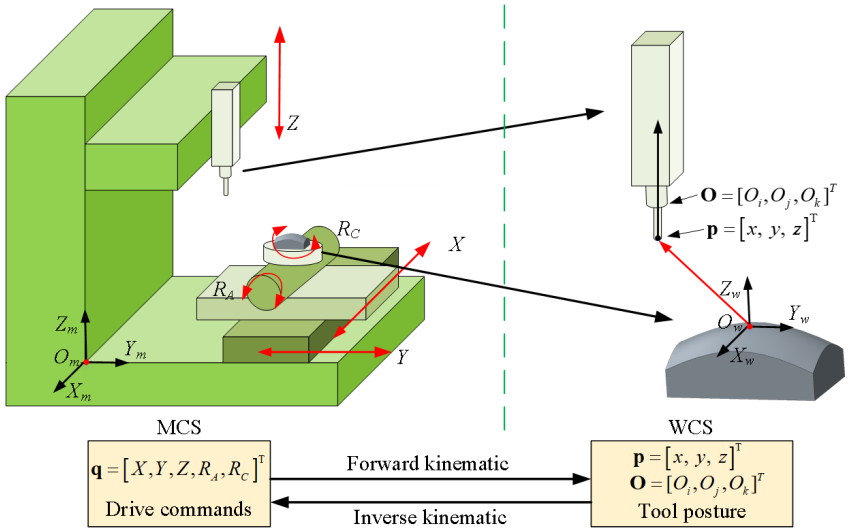


Fig. 1. Coordinate transformation of table-tilting five-axis machine

$$
\begin{aligned}
R_A &= \arccos(O_k),\\
R_C &= \arctan(O_i, O_j),\\
X &= -\cos(R_C)x + \sin(R_C)y,\\
Y &= -\cos(R_A)\sin(R_C)x - \cos(R_A)\cos(R_C)y + \sin(R_A)z + \sin(R_A)L_{ca},\\
Z &= -\sin(R_A)\sin(R_C)x + \sin(R_A)\cos(R_C)y + \cos(R_A)z\\
&\quad + \cos(R_A)L_{ca} + L_{ya}.
\end{aligned}
\tag{2}
$$

Fig. 2 illustrates the flow chart of the developed tool path smoothing method based on the ACR spline.

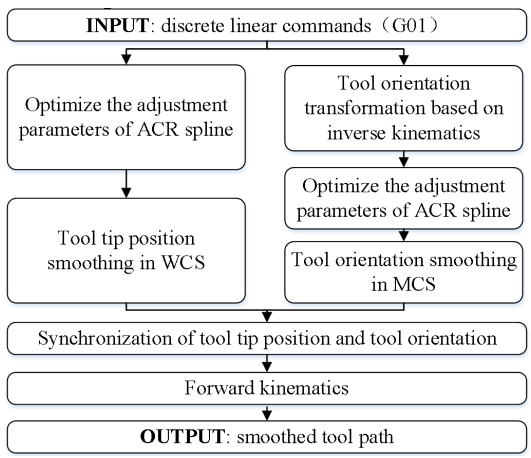


Fig. 2. CFlow chart of the developed tool path smoothing method

In this process, the tool tip position $\mathbf{p} = \left[x,\ y,\ z\right]^{\mathrm{T}}$ and tool orientation $\mathbf{\Psi} = \left[R_A,\ R_C\right]$ are independently smoothed within WCS and MCS, respectively. The maximum deviations introduced by local smoothing for both the tool tip position and orientation are constrained within predetermined limits. Finally, synchronization between the tool tip position and tool orientation is achieved by replacing the remaining linear segments with ACR splines.

2.1. Introduction of the ACR spline

In this subsection, the ACR spline, derived from the quasi-Catmull–Rom spline described in [36], is presented. The ACR spline is designed to achieve C^3 continuity while effectively controlling deviation errors and preventing the formation of cusps and self-intersections. It should be emphasized that avoidance of cusps and self-intersections is not an exclusive or automatic property of any single spline family. Classical polynomial B-splines and NURBS can also be designed to avoid such defects via appropriate knot placement and control-point design. The ACR formulation used here was chosen for three practical reasons: (1) interpolation property – ACR interpolates the specified control points which simplifies preservation of key corner points; (2) local support – each curve segment depends only on four control points, enabling local modification without global reparameterization; (3) analytical C^3 basis – this facilitates explicit derivation of synchronization constraints used in Section 4. At the same time, ACR introduces two shape parameters $\kappa,\ \lambda \in R$ that require careful selection. In this work, control points are constructed according to Eqs. (12)–(16) and κ, λ are chosen by the optimization procedure in Section 3. Under these construction and parameter-selection rules, the ACR curves used in our simulations avoid cusps and self-intersections while meeting the prescribed deviation tolerances (see Table 2 and Section 5).

For a series of control points (x_i, y_i, z_i), where $i = 0, 1, \ldots, n$, the CR spline within the interval $x_i \in [x_i, x_{i+1}]$ can be expressed as:

$$\begin{aligned} R_{y,i}(x) &= \sum_{j=0}^{3} b_j \left(\frac{x - x_i}{\Delta x}\right) y_{i+j-1}\,, \\ R_{z,i}(x) &= \sum_{j=0}^{3} b_j \left(\frac{x - x_i}{\Delta x}\right) z_{i+j-1}\,, \quad i = 1, 2, \ldots, n-2, \end{aligned} \tag{3}$$

where $\Delta x = x_{i+1} - x_i$, $i = 0, 1, \ldots, n-1$; $b_j(t)$ denote three basis functions related to the spline in this interval. In order to realize C^3 continuity and adjustable of ACR spline, the degree of basis functions is set to seven, and two additional parameters are introduced to them. The term degree is used to denote the polynomial degree, whereas order is reserved for the approximation order (i.e., order = degree + 1 in the approximation-theory sense). The polynomials in Eq. (5) are septic, that is, they have degree 7 and therefore approximation order 8. To avoid ambiguity, this

paper uses degree exclusively for polynomial degree and order only when referring to approximation order. And then the following basis functions can be obtained:

$$\begin{aligned} b_0(t) &= \kappa l_0(t) + \lambda m_0(t), \\ b_1(t) &= \kappa l_1(t) + \lambda m_1(t) + n_1(t), \\ b_2(t) &= -\kappa l_0(t) + \lambda m_2(t) + (1 - n_1(t)), \\ b_3(t) &= -\kappa l_1(t) - \lambda \left(m_0(t) + m_1(t) + m_2(t)\right), \end{aligned} \tag{4}$$

where $t \in [0,\ 1]$, $\kappa,\ \lambda \in R$ are two additional adjustment parameters and

$$\begin{aligned} l_0(t) &= -t+20t^4-45t^5+36t^6-10t^7, \quad l_1(t) = 15t^4-39t^5+34t^6-10t^7, \\ m_0(t) &= t^2-10t^4+20t^5-15t^6+4t^7, \quad m_1(t) = -2t^2+25t^4-54t^5+43t^6-12t^7, \\ m_2(t) &= t^2-20t^4+48t^5-41t^6+12t^7, \quad n_1(t) = 1-35t^4+84t^5-70t^6+20t^7. \end{aligned} \tag{5}$$

It is not difficult to obtain from Eq. (4) that $\sum_{i=1}^{3} b_i(t) = 1$ and $b_i(1-t) = b_{3-i}(t)$, which verifies the unitary and symmetry of the constructed basis functions, respectively; besides, the following relationship can be obtained from Eq. (4) to verify the interpolate property and continuity of the ACR spline:

$$\begin{aligned} &b_0(0) = 0, \quad b_1(0) = 1, \quad b_2(0) = 0, \quad b_3(0) = 0, \\ &b_0(1) = 0, \quad b_1(1) = 0, \quad b_2(1) = 1, \quad b_3(1) = 0, \end{aligned} \tag{6}$$

$$\begin{aligned} &b_0'(0) = -\kappa, \quad b_1'(0) = 1, \quad b_2'(0) = \kappa, \quad b_3'(0) = 0, \\ &b_0'(1) = 0, \quad b_1'(1) = -\kappa, \quad b_2'(1) = 0, \quad b_3'(1) = \kappa, \end{aligned} \tag{7}$$

$$\begin{aligned} &b_0''(0) = 2\lambda, \quad b_1''(0) = -4\lambda, \quad b_2''(0) = 2\lambda, \quad b_3''(0) = 0, \\ &b_0''(1) = 0, \quad b_1''(1) = 2\lambda, \quad b_2''(1) = -4\lambda, \quad b_3''(1) = 2\lambda, \end{aligned} \tag{8}$$

$$\begin{aligned} &b_0'''(0) = 0, \quad b_1'''(0) = 0, \quad b_2'''(0) = 0, \quad b_3'''(0) = 0, \\ &b_0'''(1) = 0, \quad b_1'''(0) = 0, \quad b_2'''(1) = 0, \quad b_3'''(0) = 0. \end{aligned} \tag{9}$$

The interpolate property of ACR spline can be verified by adding two control points (x_{-1}, y_{-1}, z_{-1}) and $(x_{n+i}, y_{n+i}, z_{n+i})$ to Eq. (3):

$$\begin{aligned} &R_{y,i}(x_i) = y_i\,, \quad R_{y,i}(x_{i+1}) = y_{i+1}\,, \\ &R_{z,i}(x_i) = z_i\,, \quad R_{z,i}(x_{i+1}) = z_{i+1}\,, \quad i = 1, 2, \ldots, n-1 \end{aligned} \tag{10}$$

which means that ACR splines always interpolates these given control points.

The C^3 continuity of ACR spline can be derived by combining Eq. (3) and Eqs. (7)–(9):

$$
\begin{aligned}
R'_{y,i}(x_{i+1}) &= \frac{\kappa}{\Delta x}(y_{i+3} - y_{i+1}) = R'_{y,i+1}(x_{i+1}),\\
R'_{z,i}(x_{i+1}) &= \frac{\kappa}{\Delta x}(z_{i+3} - z_{i+1}) = R'_{z,i+1}(x_{i+1}),\\
R''_{y,i}(x_{i+1}) &= \frac{2\lambda}{\Delta x^2}(y_{i+1} - 2y_{i+2} + y_{i+3}) = R''_{y,i+1}(x_{i+1}),\\
R''_{z,i}(x_{i+1}) &= \frac{2\lambda}{\Delta x^2}(z_{i+1} - 2z_{i+2} + z_{i+3}) = R''_{z,i+1}(x_{i+1}),\\
R'''_{y,i}(x_{i+1}) &= 0 = R'''_{y,i+1}(x_{i+1}), \quad R'''_{z,i}(x_{i+1}) = 0 = R'''_{z,i+1}(x_{i+1}),
\end{aligned}
\tag{11}
$$

Since two additional parameters κ and λ are incorporated into the basis function presented in Eq. (4), the shape of the ACR spline can be modified by adjusting these parameters without altering the associated control points. Moreover, as demonstrated in Eq. (3), each ACR spline function $R_i(x)$ is influenced by only four control points $(x_{i+j}, y_{i+j}, z_{i+j})$, where $j = 1, 2, 3$. Consequently, modifying any one control point will affect the shape of only the four ACR spline segments associated with it, thereby confirming the local support property of the ACR spline.

2.2. Scope and collision-avoidance considerations

The smoothing/interpolation procedure presented in this paper is formulated as a post-processing step that operates on an input tool path. The method assumes that the input path (sequence of tool poses) has been generated by a prior planning stage and is collision-free and reachable by the target machine. The proposed algorithm preserves interpolation at specified key poses and enforces explicit bounds on positional and orientation deviation (see Section 2.3 and 2.4), so that the smoothed path remains within a prescribed geometric tolerance of the original path. Under these conditions, and provided that the original path has a clearance larger than the chosen positional tolerance, the smoothed trajectory will not introduce new collisions with the workpiece or fixtures.

Nevertheless, full processing pipelines in practice require an explicit collision-checking step. Two practical integration options are: (i) offline verification: after smoothing, the path is sampled densely and checked against the CAD model of the workpiece and fixtures using standard collision detection libraries (collision of tool swept volume or tool-tip geometry), and inverse-kinematics / reachability checks are performed to ensure the required poses are achievable without joint-limit or singularity violations; (ii) constrained smoothing: collision-avoidance constraints (positional deviation bounds, orientation limits or explicit inequality constraints derived from clearance margins) are incorporated into the smoothing optimization so that the solver enforces collision-related limits during path generation.

We discuss implementation choices and trade-offs below and recommend applying a collision verification pass in the post-processing stage before NC code generation.

2.3. Tool tip position corner smoothing method

In this subsection, the tool tip position corner $\angle \mathbf{P}_{i-1}\mathbf{P}_i\mathbf{P}_{i+1}$ depicted in Fig. 3 is selected as an illustrative example of the developed tool tip smoothing method, where $\mathbf{P}_{i-1}$, $\mathbf{P}_i$ and $\mathbf{P}_{i+1}$ are three endpoints of the adjacent linear segments. The ACR spline described in Section 2.1 is employed to smooth the tool tip position at this transition corner, while ensuring that the deviation remains within the specified error tolerance ε_p.

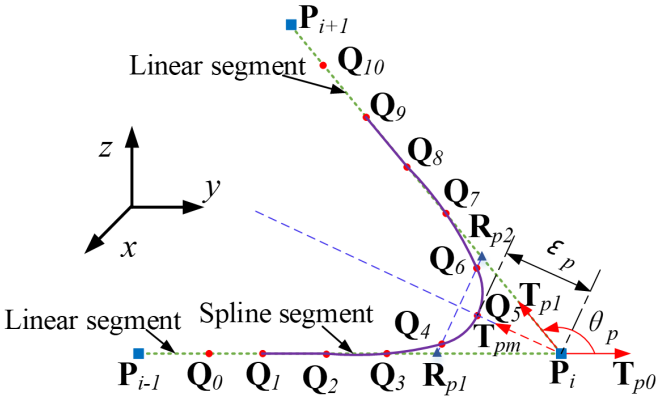


Fig. 3. Corner smoothing of tool tip position

In order to ensure the tangential continuity of the constructed ACR spline and the original line segment, considering the continuity and local property of the constructed CR spline, the control points $\mathbf{Q}_0$–$\mathbf{Q}_3$ and $\mathbf{Q}_7$–$\mathbf{Q}_{10}$ are located on the linear segments $\overrightarrow{\mathbf{P}_{i-1}\mathbf{P}_i}$ and $\overrightarrow{\mathbf{P}_i\mathbf{P}_{i+1}}$, respectively. Besides, to realize the geometric symmetry of the inserted ACR splines with respect to the bisector of the angle formed by adjacent position segments, and to avoid cusps and self-intersections at the same time, the following control points are constructed:

$$
\begin{aligned}
&\mathbf{Q}_5 = \mathbf{P}_i + \mathbf{T}_{pm}L_p\,, \quad \mathbf{Q}_3 = \mathbf{P}_i - \frac{2\mathbf{T}_{p0}L_p}{\sin(\theta_p/2)}\,, \quad \mathbf{Q}_7 = \mathbf{P}_i + \frac{2\mathbf{T}_{p1}L_p}{\sin(\theta_p/2)}\,,\\
&\mathbf{Q}_2 = 1.5\mathbf{Q}_3 - 0.5\mathbf{P}_i\,, \quad \mathbf{Q}_8 = 1.5\mathbf{Q}_7 - 0.5\mathbf{P}_i\,, \quad \mathbf{Q}_1 = 2\mathbf{Q}_3 - \mathbf{P},\\
&\mathbf{Q}_9 = 2\mathbf{Q}_7 - \mathbf{P}_{ii}\,, \quad \mathbf{Q}_0 = 2.5\mathbf{Q}_3 - 1.5\mathbf{P}_i\,, \quad \mathbf{Q}_{10} = 2.5\mathbf{Q}_7 - 1.5\mathbf{P}_i\,,\\
&\mathbf{Q}_4 = 0.9\mathbf{R}_{p1} + 0.1\mathbf{R}_{p2}\,, \quad \mathbf{Q}_6 = 0.1\mathbf{R}_{p1} + 0.9\mathbf{R}_{p2}\,,
\end{aligned}
\tag{12}
$$

where

$$\begin{aligned}
&\mathbf{T}_{p0} = \overrightarrow{\mathbf{P}_{i-1}\mathbf{P}_i} \Big/ \left\|\overrightarrow{\mathbf{P}_{i-1}\mathbf{P}_i}\right\|, \quad \mathbf{T}_{p1} = \overrightarrow{\mathbf{P}_i\mathbf{P}_{i+1}} \Big/ \left\|\overrightarrow{\mathbf{P}_i\mathbf{P}_{i+1}}\right\|, \\
&\mathbf{T}_{pm} = (\mathbf{T}_{p1} - \mathbf{T}_{p0}) / \|\mathbf{T}_{p1} - \mathbf{T}_{p0}\|, \\
&\mathbf{R}_{p1} = \mathbf{Q}_3/3 + 2\left(\mathbf{P}_i - \mathbf{T}_{p0} L_p / \sin(\theta_2)/2\right)/3, \\
&\mathbf{R}_{p2} = \mathbf{Q}_7/3 + 2\left(\mathbf{P}_i - \mathbf{T}_{p2} L_p / \sin(\theta_2)/2\right)/3, \\
&L_p \leqslant \varepsilon_p,
\end{aligned} \tag{13}$$

where $\theta_p = \arccos(\mathbf{T}_{p0} \bullet \mathbf{T}_{p1})$.

Since each linear segment is shared by two corners (except of the first and the last linear segments), therefore, one third of the segment is retained to adjust the synchronization of the tool tip position and tool orientation, that is, the linear segment used to construct ACR spline should be less than one third of the total length of the segment:

$$\left\|\overrightarrow{\mathbf{Q}_0\mathbf{P}_i}\right\| \leqslant \left\|\overrightarrow{\mathbf{P}_{i-1}\mathbf{P}_i}\right\|/3, \quad \left\|\overrightarrow{\mathbf{P}_i\mathbf{Q}_{10}}\right\| \leqslant \left\|\overrightarrow{\mathbf{P}_i\mathbf{P}_{i+1}}\right\|/3. \tag{14}$$

Substituting Eq. (12) into Eq. (14) and combining with the error tolerance constraint of $L_p \leqslant \varepsilon_p$, the final constraints of L_p can be obtained as:

$$L_p = \min\left(\varepsilon_p, \frac{\left\|\overrightarrow{\mathbf{P}_{i-1}\mathbf{P}_i}\right\| \sin(\theta_p/2)}{12\|\mathbf{T}_{p0}\|}, \frac{\left\|\overrightarrow{\mathbf{P}_i\mathbf{P}_{i+1}}\right\| \sin(\theta_p/2)}{12\|\mathbf{T}_{p1}\|}\right). \tag{15}$$

2.4. Tool orientation corner smoothing method

In this subsection, the tool orientation corner $\angle\boldsymbol{\Psi}_{i-1}\boldsymbol{\Psi}_i\boldsymbol{\Psi}_{i+1}$ shown in Fig. 4 is chosen as an example to illustrate the developed tool orientation smoothing method, where $\boldsymbol{\Psi}_{i-1}$, $\boldsymbol{\Psi}_i$ amd $\boldsymbol{\Psi}_{i+1}$ are three end points of the adjacent linear

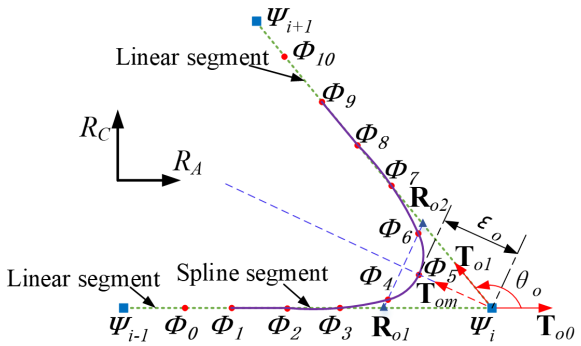


Fig. 4. Corner smoothing of tool orientation

segment. The ACR spline described in Section 2.1 is utilized to smooth the tool orientation at this transition corner under the error tolerance ε_p.

Similar to the method of determining control points of tool tip spline described in last subsection, the control points of the tool orientation ACR spline can be determined by the following equations:

$$\begin{aligned}
&\boldsymbol{\Phi}_5 = \boldsymbol{\Psi}_i + \mathbf{T}_{om} L_o\,, \quad \boldsymbol{\Phi}_3 = \boldsymbol{\Psi}_i - \frac{2\mathbf{T}_{o0} L_o}{\sin(\theta_0/2)}\,, \quad \boldsymbol{\Phi}_7 = \boldsymbol{\Psi}_i + \frac{2\mathbf{T}_{o1} L_o}{\sin(\theta_0/2)}\,,\\
&\boldsymbol{\Phi}_2 = 1.5\boldsymbol{\Phi}_3 - 0.5\boldsymbol{\Psi}_i\,, \quad \boldsymbol{\Phi}_8 = 1.5\boldsymbol{\Phi}_7 - 0.5\boldsymbol{\Psi}_i\,, \quad \boldsymbol{\Phi}_1 = 2\boldsymbol{\Phi}_3 - \boldsymbol{\Psi}_i\,,\\
&\boldsymbol{\Phi}_9 = 2\boldsymbol{\Phi}_7 - \boldsymbol{\Psi}_i\,, \quad \boldsymbol{\Phi}_0 = 2.5\boldsymbol{\Phi}_3 - 1.5\boldsymbol{\Psi}_i\,, \quad \boldsymbol{\Phi}_{10} = 2.5\boldsymbol{\Phi}_7 - 1.5\boldsymbol{\Psi}_i\,,\\
&\boldsymbol{\Phi}_4 = 0.9\mathbf{R}_{o1} + 0.1\mathbf{o2}\,, \quad \boldsymbol{\Phi}_6 = 0.1\mathbf{R}_{o1} + 0.9\mathbf{o2}\,,
\end{aligned} \tag{16}$$

where

$$\begin{aligned}
&\mathbf{T}_{o0} = \overrightarrow{\boldsymbol{\Psi}_{i-1}\boldsymbol{\Psi}_i} \Big/ \left\|\overrightarrow{\boldsymbol{\Psi}_{i-1}\boldsymbol{\Psi}_i}\right\|, \quad \mathbf{T}_{o1} = \overrightarrow{\boldsymbol{\Psi}_i\boldsymbol{\Psi}_{i+1}} \Big/ \left\|\overrightarrow{\boldsymbol{\Psi}_i\boldsymbol{\Psi}_{i+1}}\right\|,\\
&\mathbf{T}_{om} = (\mathbf{T}_{o1} - \mathbf{T}_{o0}) / \|\mathbf{T}_{o1} - \mathbf{T}_{o0}\|,\\
&\mathbf{R}_{o1} = \boldsymbol{\Phi}_3/3 + 2\left(\boldsymbol{\Psi}_i - \mathbf{T}_{o0} L_o/\sin(\theta_0/2)\right)/3,\\
&\mathbf{R}_{o2} = \boldsymbol{\Phi}_7/3 + 2\left(\boldsymbol{\Psi}_i - \mathbf{T}_{o1} L_o/\sin(\theta_0/2)\right)/3,\\
&L_o \leqslant \varepsilon_o\,,
\end{aligned} \tag{17}$$

where $\theta_o = \arccos(\mathbf{T}_{o0} \bullet \mathbf{T}_{o1})$.

Besides, one third of the segment should be retained to adjust the synchronization of the tool tip position and tool orientation:

$$\left\|\overrightarrow{\boldsymbol{\Phi}_0\boldsymbol{\Psi}_i}\right\| \leqslant \left\|\overrightarrow{\boldsymbol{\Psi}_{i-1}\boldsymbol{\Psi}_i}\right\|/3, \quad \left\|\overrightarrow{\boldsymbol{\Psi}_i\boldsymbol{\Phi}_{10}}\right\| \leqslant \left\|\overrightarrow{\boldsymbol{\Psi}_i\boldsymbol{\Psi}_{i+1}}\right\|/3. \tag{18}$$

Substituting Eq. (16) into Eq. (18) and combining with the error tolerance constraint of $L_o \leqslant \varepsilon_o$, the constraints of L_o can be obtained as:

$$L_o = \min\left(\varepsilon_o, \frac{\left\|\overrightarrow{\boldsymbol{\Psi}_{i-1}\boldsymbol{\Psi}_i}\right\|\sin(\theta_0/2)}{12\|\mathbf{T}_{o0}\|}, \frac{\left\|\overrightarrow{\boldsymbol{\Psi}_i\boldsymbol{\Psi}_{i+1}}\right\|\sin(\theta_0/2)}{12\|\mathbf{T}_{o1}\|}\right). \tag{19}$$

In practical application, the tool orientation error ε_{ow} should be constrained in WCS, as shown in Fig. 5, while the error tolerance parameter ε_o utilized in Eq. (19) is constrained in the MCS. Therefore, the relationship between ε_o and ε_{ow} should be established.

The deviation of the tool orientation vector induced by the local smoothing in the WCS, as shown in Fig. 5, should satisfy the following relationship:

$$\Delta\mathbf{O} \leqslant 2\sin(\varepsilon_{ow}/2). \tag{20}$$

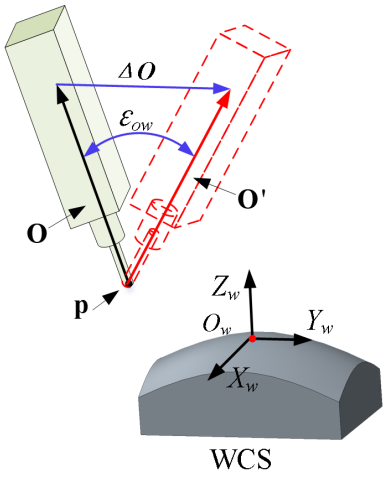


Fig. 5. Tool orientation error in the WCS

Since the deviation tolerance of tool orientation vector is generally small, the linear approximate relationship between the tool orientation deviation in WCS and the joint rotation deviation in MCS can be obtained:

$$\Delta O \approx \mathbf{J} \bullet \boldsymbol{\varepsilon}_o \,, \tag{21}$$

where $\mathbf{J}$ is the Jacobian matrix of the tool orientation and can be obtained by differentiating Eq. (1):

$$\mathbf{J} = \begin{bmatrix} \cos(R_A)\sin(R_C) & \sin(R_A)\cos(R_C) \\ \cos(R_A)\cos(R_C) & -\sin(R_A)\sin(R_C) \\ -\sin(R_A) & 0 \end{bmatrix}. \tag{22}$$

Since the tool orientation error ΔO is defined as the minimum deviation between the original tool orientation and the smoothed tool orientation, it can be obtained that:

$$\Delta O \leqslant \|\mathbf{J} \bullet (\boldsymbol{\Psi}_i - \boldsymbol{\Phi}_5)\| \,. \tag{23}$$

Substitute Eq. (14) and Eq. (20) into the above equation:

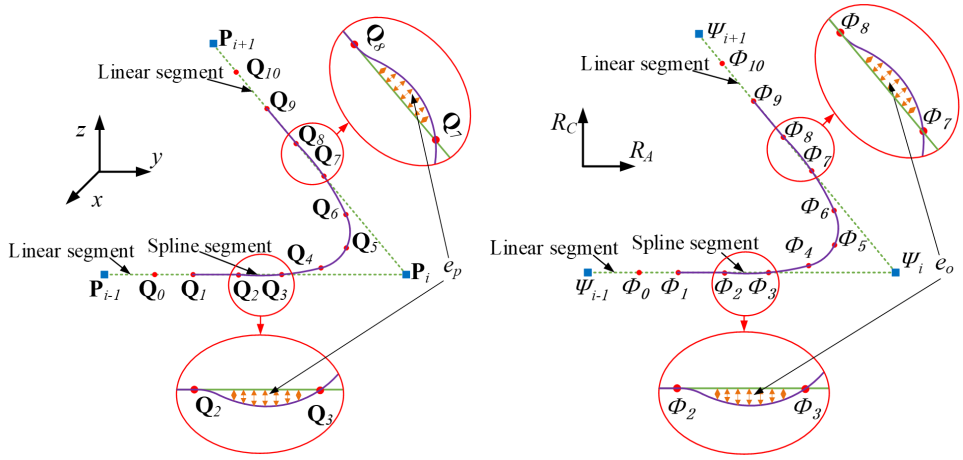
$$\varepsilon_o \leqslant \frac{2\sin(\varepsilon_{ow}/2)}{\|\mathbf{J}\,\mathbf{T}_{om}\|} \,. \tag{24}$$

Substitute it into Eq. (21), the final constraints of L_o can be obtained as:

$$L_o = \min\left(\frac{2\sin(\varepsilon_{ow}/2)}{\|\mathbf{J}\,\mathbf{T}_{om}\|}, \frac{\left\|\overrightarrow{\boldsymbol{\Psi}_{i-1}\boldsymbol{\Psi}_i}\right\| \sin(\theta_0/2)}{12\|\mathbf{T}_{o0}\|}, \frac{\left\|\overrightarrow{\boldsymbol{\Psi}_i\boldsymbol{\Psi}_{i+1}}\right\| \sin(\theta_0/2)}{12\|\mathbf{T}_{o1}\|} \right). \tag{25}$$

3. Constraint of the tool tip position and tool orientation error

Based on the local support property of the ACR spline, it can be inferred that, when an ACR spline is used to smooth a transition corner, not only does a smoothing error (ε_p and ε_o) arise, but a deviation between the spline curve and the original linear segment (e_p and e_o) is also induced, as illustrated in Fig. 6. Consequently, it is essential to control this deviation error by adjusting the spline's shape. In this subsection, an analytical optimization method is developed to minimize deviation errors e_p and e_o by optimizing the adjustment parameters κ and λ.



(a) Tool tip position (b) Tool orientation

Fig. 6. The smoothed tool tip position and tool orientation

Taking the segments between control points $\mathbf{Q}_2 = (x_2, y_2, z_2)$ and $\mathbf{Q}_3 = (x_3, y_3, z_3)$ as an example, the error between the spline segment and the original linear segment can be defined as:

$$e_p(\kappa, \lambda) = \int_{x_2}^{x_3} \left(R_{y,2}(x) - y(x)\right)^2 + \left(R_{z,2}(x) - z(x)\right)^2 \mathrm{d}x, \tag{26}$$

where $R_{y,2}(x)$ and $R_{z,2}(x)$ are ACR spline functions in the interval $x \in [x_2, x_3]$ with control points $\mathbf{Q}_1$, $\mathbf{Q}_2$, $\mathbf{Q}_3$ and $\mathbf{Q}_4$; $y(x)$ and $z(x)$ are the functions of linear segment between $\mathbf{Q}_2$ and $\mathbf{Q}_3$, which can be expressed as:

$$y(x) = y_2 + \frac{(y_3 - y_2)(x - x_2)}{x_3 - x_2}, \quad z(x) = z_2 + \frac{(z_3 - z_2)(x - x_2)}{x_3 - x_2}. \tag{27}$$

Substituting Eq. (4) into Eq. (3), the ACR spline functions $R_{y,2}(x)$ and $R_{z,2}(x)$ in the interval $x \in [x_2, x_3]$ can be rewritten as:

$$\begin{aligned} R_{y,2}(x) &= A_{y,2}(x)\kappa + B_{y,2}(x)\lambda + C_{y,2}(x), \\ R_{z,2}(x) &= A_{z,2}(x)\kappa + B_{z,2}(x)\lambda + C_{z,2}(x), \end{aligned} \tag{28}$$

where

$$\begin{aligned} A_{y,2}(x) &= l_0(t)y_1 + l_1(t)y_2 - l_0(t)y_3 - l_1(t)y_4\,, \\ B_{y,2}(x) &= m_0(t)y_1 + m_1(t)y_2 + m_2(t)y_3 - \big(m_0(t) + m_1(t) + m_2(t)\big)y_4\,, \\ C_{y,2}(x) &= n_1(t)y_2 + \big(1 - n_1(t)\big)y_3\,, \\ A_{z,2}(x) &= l_0(t)z_1 + l_1(t)z_2 - l_0(t)z_3 - l_1(t)z_4\,, \\ B_{z,2}(x) &= m_0(t)z_1 + m_1(t)z_2 + m_2(t)z_3 - \big(m_0(t) + m_1(t) + m_2(t)\big)z_4\,, \\ C_{z,2}(x) &= n_1(t)z_2 + \big(1 - n_1(t)\big)z_3\,, \end{aligned} \tag{29}$$

where $t = (x - x_2)/(x_3 - x_2)$. The following optimization model can be derived by substituting Eq. (28) into Eq. (26):

$$\begin{aligned} &\min\ e_p(\kappa, \lambda) = D_1\kappa^2 + D_2\lambda^2 + D_3\kappa\lambda + D_4\kappa + D_5\lambda + D_6 \\ &\text{s.t.}\quad \kappa,\ \lambda \in R, \end{aligned} \tag{30}$$

where

$$\begin{aligned} D_1 &= \int_{x_2}^{x_3} \Big(\big(A_{y,2}(x)\big)^2 + \big(A_{z,2}(x)\big)^2 \Big)\,\mathrm{d}x, \\ D_2 &= \int_{x_2}^{x_3} \Big(\big(B_{y,2}(x)\big)^2 + \big(B_{z,2}(x)\big)^2 \Big)\,\mathrm{d}x, \\ D_3 &= \int_{x_2}^{x_3} \big(A_{y,2}B_{y,2} + A_{z,2}B_{z,2} \big)\,\mathrm{d}x, \\ D_4 &= \int_{x_2}^{x_3} \big(A_{y,2}(x)\big(C_{y,2}(x) - y(x)\big) + \big(A_{z,2}(x)\big(C_{z,2}(x) - z(x)\big)\big) \big)\,\mathrm{d}x, \\ D_5 &= \int_{x_2}^{x_3} \big(B_{y,2}(x)\big(C_{y,2}(x) - y(x)\big) + \big(B_{z,2}(x)\big(C_{z,2}(x) - z(x)\big)\big) \big)\,\mathrm{d}x, \\ D_6 &= \int_{x_2}^{x_3} \Big(\big(C_{y,2} - y(x)\big)^2 + \big(C_{z,2} - z(x)\big)^2 \Big)\,\mathrm{d}x. \end{aligned} \tag{31}$$

It is straightforward to show that $e_p(\kappa, \lambda) \geq 0$. Consequently, the minimum of $e_p(\kappa, \lambda)$ can be obtained by solving the following equation. A complete derivation is given in Appendix A.

$$\frac{\partial e_p(\kappa, \lambda)}{\partial \kappa} = 0, \quad \frac{\partial e_p(\kappa, \lambda)}{\partial \lambda} = 0. \tag{32}$$

The following equations can be obtained by combining Eq. (32) with Eq. (30):

$$D_1\kappa + D_3\lambda = -D_4\,, \quad D_1\kappa + D_2\lambda = -D_5\,. \tag{33}$$

The optimal adjustment parameters κ and λ corresponding to minimum deviation error e_p can be obtained by solving Eq. (33).

4. Synchronization of the tool orientation and tool tip position

Although the tool tip position and tool orientation at the corner have been smoothed independently in previous sections, if the rate of change of the tool orientation relative to the tool tip displacement does not satisfy C^3 continuity at the junction between the smoothed spline segment and the remaining linear segment, discontinuities or abrupt variations in velocity, acceleration, and jerk may occur. Such inconsistencies can lead to similar irregularities in the kinematic responses – velocity, acceleration, and jerk – of each machine tool joint.

To guarantee the synchronization of the change rate of smoothed tool orientation and the speed of tool tip displacement, the developed ACR spline is utilized to replace the remaining position and orientation linear segments after smoothing, as shown in Fig. 7, where $\mathbf{Q}_{0,i}$, $\mathbf{Q}_{1,i}$, $\mathbf{Q}_{2,i}$ and $\mathbf{Q}_{3,i}$ are control points of the smoothed position curves of the i-th corner, $\mathbf{Q}_{7,i-1}$, $\mathbf{Q}_{8,i-1}$, $\mathbf{Q}_{9,i-1}$ and $\mathbf{Q}_{10,i-1}$ are control points of the smoothed position curves of the $(i-1)$-th corner, the linear segment

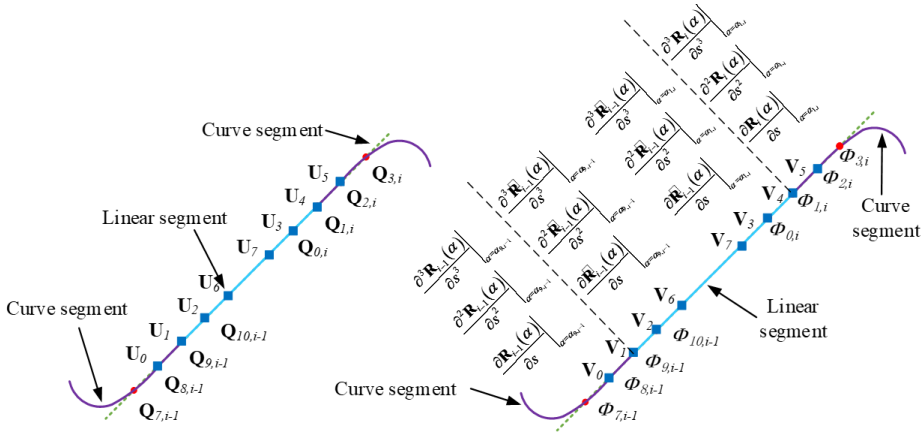


Fig. 7. Synchronization of the tool tip position and orientation

between $\mathbf{Q}_{10,i-1}$ and $\mathbf{Q}_{0,i}$ is the remaining position linear segment between two corners. Similarly, $\mathbf{\Phi}_{0,i}$, $\mathbf{\Phi}_{1,i}$, $\mathbf{\Phi}_{2,i}$ and $\mathbf{\Phi}_{3,i}$ are control points of the smoothed orientation curves of the i-th corner, $\mathbf{\Phi}_{7,i-1}$, $\mathbf{\Phi}_{8,i-1}$, $\mathbf{\Phi}_{9,i-1}$ and $\mathbf{\Phi}_{10,i-1}$ are control points of the smoothed orientation curves of the $(i-1)$-th corner, the linear segment between $\mathbf{\Phi}_{10,i-1}$ and $\mathbf{\Phi}_{0,i}$ is the remaining orientation linear segment between two corners.

To ensure that the rate of change of the smoothed tool orientation is synchronized with the tool tip displacement, the developed ACR spline is employed to replace the remaining linear segments for both tool tip position and tool orientation after smoothing, as illustrated in Fig. 7. In this figure, $\mathbf{Q}_{0,i}$, $\mathbf{Q}_{1,i}$, $\mathbf{Q}_{2,i}$ and $\mathbf{Q}_{3,i}$ denote the control points $\mathbf{Q}_{8,i-1}$ of the smoothed position curve of the i-th corner, $\mathbf{Q}_{7,i-1}$, $\mathbf{Q}_{8,i-1}$, $\mathbf{Q}_{9,i-1}$, and $\mathbf{Q}_{10,i-1}$ denote the control points of the smoothed position curves of the $(i-1)$-th corner, the linear segment between $\mathbf{Q}_{10,i-1}$ and $\mathbf{Q}_{0,i}$ represents the remaining position segment between two corners. Similarly, $\mathbf{\Phi}_{0,i}$, $\mathbf{\Phi}_{1,i}$, $\mathbf{\Phi}_{2,i}$ and $\mathbf{\Phi}_{3,i}$ denote the control points of the smoothed orientation curve of the i-th corner, $\mathbf{\Phi}_{7,i-1}$, $\mathbf{\Phi}_{8,i-1}$, $\mathbf{\Phi}_{9,i-1}$ and $\mathbf{\Phi}_{10,i-1}$ are control points of the smoothed orientation curves of the $(i-1)$-th corner, and the linear segment between $\mathbf{\Phi}_{10,i-1}$ and $\mathbf{\Phi}_{0,i}$ represents the remaining orientation segment between adjacent corners.

Owing to the local support property of the ACR spline, the curve segment between control points $\mathbf{Q}_{1,i}$ and $\mathbf{Q}_{2,i}$ is influenced solely by $\mathbf{Q}_{0,i}$, $\mathbf{Q}_{1,i}$, $\mathbf{Q}_{2,i}$ and $\mathbf{Q}_{3,i}$, all of which lie on the original linear segment; as a result, this curve segment coincides exactly with the original linear segment. Similarly, the curve segments between control points $\mathbf{Q}_{8,i-1}$ and $\mathbf{Q}_{9,i-1}$, $\mathbf{\Phi}_{1,i}$ and $\mathbf{\Phi}_{2,i}$, $\mathbf{\Phi}_{8,i-1}$ and $\mathbf{\Phi}_{9,i-1}$ are also coincident with the corresponding portions of the original linear segment. Therefore, C^3 continuity between the remaining linear segment and the smoothed curve segment is achieved, provided that the control points of the remaining segment are located on the original linear segment.

In this study, the developed ACR splines are employed to replace the remaining linear segments. For simplicity, the control points $\mathbf{Q}_{8,i-1}$, $\mathbf{Q}_{9,i-1}$, $\mathbf{Q}_{10,i-1}$, $\mathbf{Q}_{0,i}$, $\mathbf{Q}_{1,i}$ and $\mathbf{Q}_{2,i}$ of the two adjacent curve segments are adopted as the control points for the remaining position linear segments and are subsequently redefined as $\mathbf{U}_0$, $\mathbf{U}_1$, $\mathbf{U}_2$, $\mathbf{U}_3$, $\mathbf{U}_4$ and $\mathbf{U}_5$, as illustrated in Fig. 7. Similarly, the control points $\mathbf{\Phi}_{8,i-1}$, $\mathbf{\Phi}_{9,i-1}$, $\mathbf{\Phi}_{10,i-1}$, $\mathbf{\Phi}_{0,i}$, $\mathbf{\Phi}_{1,i}$ and $\mathbf{\Phi}_{2,i}$ of the two adjacent curve segments are used as the control points for the remaining orientation linear segments, and are redefined as $\mathbf{V}_0$, $\mathbf{V}_1$, $\mathbf{V}_2$, $\mathbf{V}_3$, $\mathbf{V}_4$ and $\mathbf{V}_5$.

In addition to these constraints, ensuring the synchronization and C^3 continuity of the tool orientation relative to the tool tip position requires that the first-, second-, and third-order derivatives of the tool orientation $\mathbf{R}(\alpha)$ with respect to the tool tip displacement are equal at the junction between the smoothed spline and the remaining linear segment. As shown in Fig. 7, at the intersection point $\mathbf{V}_i$, the derivatives of the tool orientation at the curve endpoint with respect to the

displacement of its tool tip are given by:

$$\begin{aligned}
\left.\frac{\partial \mathbf{R}_{i-1}(\alpha)}{\partial s}\right|_{\alpha=\alpha_{9,i-1}} &= \left(\frac{\partial \mathbf{R}_{i-1}(\alpha)}{\partial \alpha}\frac{\partial \alpha}{\partial s}\right)_{\alpha=\alpha_{9,i-1}},\\
\left.\frac{\partial^2 \mathbf{R}_{i-1}(\alpha)}{\partial s^2}\right|_{\alpha=\alpha_{9,i-1}} &= \left(\frac{\partial^2 \mathbf{R}_{i-1}(\alpha)}{\partial \alpha^2}\left(\frac{\partial \alpha}{\partial s}\right)^2 + \frac{\partial \mathbf{R}_{i-1}(\alpha)}{\partial \alpha}\frac{\partial^2 \alpha}{\partial s^2}\right)_{\alpha=\alpha_{9,i-1}},\\
\left.\frac{\partial^3 \mathbf{R}_{i-1}(\alpha)}{\partial s^3}\right|_{\alpha=\alpha_{9,i-1}} &= \left(\frac{\partial^3 \mathbf{R}_{i-1}(\alpha)}{\partial \alpha^3}\left(\frac{\partial \alpha}{\partial s}\right)^3 + 3\frac{\partial^2 \mathbf{R}_{i-1}(\alpha)}{\partial \alpha^2}\frac{\partial \alpha}{\partial s}\frac{\partial^2 \alpha}{\partial s^2}\right.\\
&\quad \left. + \frac{\partial \mathbf{R}_{i-1}(\alpha)}{\partial \alpha}\frac{\partial^3 \alpha}{\partial s^3}\right)_{\alpha=\alpha_{9,i-1}}.
\end{aligned} \tag{34}$$

At the endpoint of the linear segment, the derivatives of the tool orientation with respect to the displacement of its tool tip are as follows:

$$\begin{aligned}
\left.\frac{\partial \tilde{\mathbf{R}}_{i-1}(\alpha)}{\partial s}\right|_{\alpha=\alpha_{9,i-1}} &= \left(\frac{\partial \tilde{\mathbf{R}}_{i-1}(\alpha)}{\partial \alpha}\frac{\partial \alpha}{\partial s}\right)_{\alpha=\alpha_{9,i-1}},\\
\left.\frac{\partial^2 \tilde{\mathbf{R}}_{i-1}(\alpha)}{\partial s^2}\right|_{\alpha=\alpha_{9,i-1}} &= \left(\frac{\partial^2 \tilde{\mathbf{R}}_{i-1}(\alpha)}{\partial \alpha^2}\left(\frac{\partial \alpha}{\partial s}\right)^2 + \frac{\partial \tilde{\mathbf{R}}_{i-1}(\alpha)}{\partial \alpha}\frac{\partial^2 \alpha}{\partial s^2}\right)_{\alpha=\alpha_{9,i-1}},\\
\left.\frac{\partial^3 \tilde{\mathbf{R}}_{i-1}(\alpha)}{\partial s^3}\right|_{\alpha=\alpha_{9,i-1}} &= \left(\frac{\partial^3 \tilde{\mathbf{R}}_{i-1}(\alpha)}{\partial \alpha^3}\left(\frac{\partial \alpha}{\partial s}\right)^3 + 3\frac{\partial^2 \tilde{\mathbf{R}}_{i-1}(\alpha)}{\partial \alpha^2}\frac{\partial \alpha}{\partial s}\frac{\partial^2 \alpha}{\partial s^2}\right.\\
&\quad \left. + \frac{\partial \tilde{\mathbf{R}}_{i-1}(\alpha)}{\partial \alpha}\frac{\partial^3 \alpha}{\partial s^3}\right)_{\alpha=\alpha_{9,i-1}}.
\end{aligned} \tag{35}$$

Since the control points related to the ACR spline between control points $\mathbf{\Phi}_{8,i-1}$ and $\mathbf{\Phi}_{9,i-1}$, $\mathbf{V}_1$ and $\mathbf{V}_2$ are all located on the original linear segment. Therefore, it can be obtained from Eq. (11) that, in order to realize the synchronous of the tool tip position and tool orientation related the tool tip displacement, it is necessary to insert four additional control points $\mathbf{U}_6$, $\mathbf{U}_7$ and $\mathbf{V}_6$, $\mathbf{V}_7$, as shown in Fig. 7, and the following formulas must be satisfied:

Since the control points associated with the ACR spline between control points $\mathbf{\Phi}_{8,i-1}$ and $\mathbf{\Phi}_{9,i-1}$ (and similarly between $\mathbf{V}_1$ and $\mathbf{V}_2$) lie entirely on the original linear segment, it follows from Eq. (11) that, in order to achieve synchronization between the tool tip position and tool orientation with respect to the tool tip displacement, two additional control points – denoted here as $\mathbf{U}_6$, $\mathbf{U}_7$ ($\mathbf{V}_6$, $\mathbf{V}_7$) – must be inserted, as shown in Fig. 7. Furthermore, the following conditions must be satisfied:

$$\begin{aligned}
\left\|\overrightarrow{\mathbf{U}_0\mathbf{U}_1}\right\| &= \left\|\overrightarrow{\mathbf{U}_1\mathbf{U}_2}\right\| = \left\|\overrightarrow{\mathbf{U}_2\mathbf{U}_6}\right\|, \quad \left\|\overrightarrow{\mathbf{U}_5\mathbf{U}_4}\right\| = \left\|\overrightarrow{\mathbf{U}_4\mathbf{U}_3}\right\| = \left\|\overrightarrow{\mathbf{U}_3\mathbf{U}_7}\right\|,\\
\left\|\overrightarrow{\mathbf{V}_0\mathbf{V}_1}\right\| &= \left\|\overrightarrow{\mathbf{V}_1\mathbf{V}_2}\right\| = \left\|\overrightarrow{\mathbf{V}_2\mathbf{V}_6}\right\|, \quad \left\|\overrightarrow{\mathbf{V}_5\mathbf{V}_4}\right\| = \left\|\overrightarrow{\mathbf{V}_4\mathbf{V}_3}\right\| = \left\|\overrightarrow{\mathbf{V}_3\mathbf{V}_7}\right\|
\end{aligned} \tag{36}$$

and then the following equations can be obtained

$$\begin{aligned}
\left.\frac{\partial \mathbf{R}_{i-1}(\alpha)}{\partial s}\right|_{\alpha=\alpha_{9,i-1}} &= \left.\frac{\partial \tilde{\mathbf{R}}_{i-1}(\alpha)}{\partial s}\right|_{\alpha=\alpha_{9,i-1}}, \\
\left.\frac{\partial^2 \mathbf{R}_{i-1}(\alpha)}{\partial s^2}\right|_{\alpha=\alpha_{9,i-1}} &= \left.\frac{\partial^2 \tilde{\mathbf{R}}_{i-1}(\alpha)}{\partial s^2}\right|_{\alpha=\alpha_{9,i-1}} = 0, \\
\left.\frac{\partial^3 \mathbf{R}_{i-1}(\alpha)}{\partial s^3}\right|_{\alpha=\alpha_{9,i-1}} &= \left.\frac{\partial^3 \tilde{\mathbf{R}}_{i-1}(\alpha)}{\partial s^3}\right|_{\alpha=\alpha_{9,i-1}} = 0.
\end{aligned} \tag{37}$$

It should be noted that both adjustment parameters κ and λ used in constructing the remaining ACR spline are set to zero in order to reduce the computational burden.

5. Numerical simulation

In this section, the effectiveness of the developed C^3-continuous tool-path smoothing method based on ACR splines is evaluated. The test tool path, shown in Fig. 8, comprises four straight segments and three transitional corners. Relevant parameters are listed in Table 1. Tolerances for tool-tip position and tool-direction deviations were specified in millimeters and radians, respectively. For comparison, the proposed method was benchmarked against the PH-spline–based smoothing method [30], and the curves before and after optimization were also examined. Fig. 9 presents the comparisons of tool-tip position and tool direction after smoothing.

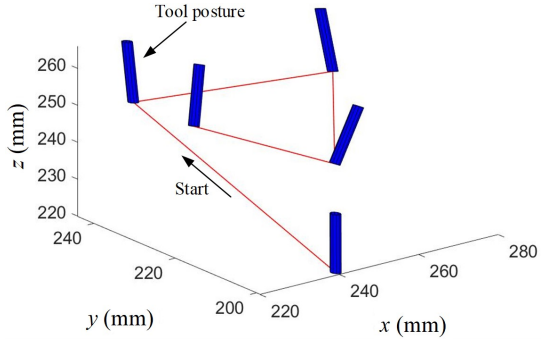
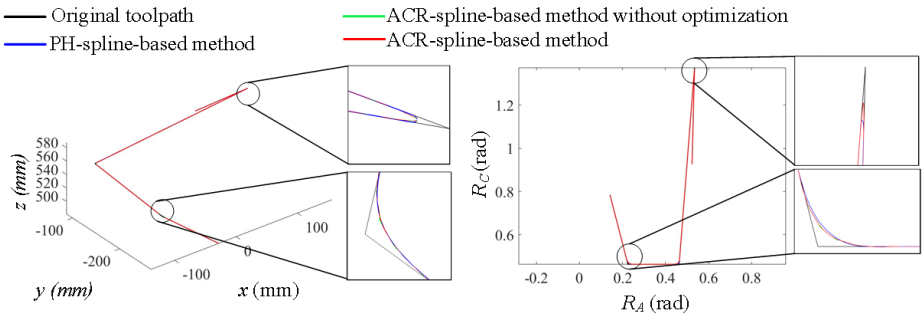


Fig. 8. The testing tool path utilized in numerical simulation

Fig. 9 clearly demonstrates the effectiveness of the developed method and the necessity of the optimization step. The unoptimized curve (green) exhibits pronounced jitter, which is eliminated in the optimized curve (red). Moreover, the deviation errors of the curves produced by the developed method are less than or equal to those from the PH-spline–based approach, further confirming the method's superior accuracy.

Table 1. Parameters of the testing tool path

Number of corners	Number of tool path	Tool tip position (x, y, z) (mm)	Tool orientation (O_i, O_j, O_k)
1	1	(240, 200, 220)	(0.1, 0.1, 0.9899)
	2	(230, 240, 250)	(0.1, 0.2, 0.9747)
2	3	(270, 230, 252.2)	(0.2, 0.4, 0.8944)
3	4	(250, 210, 242)	(0.5, 0.1, 0.8602)
		(235, 230, 246.8)	(0.4, 0.3, 0.8660)



(a) Tool tip position (b) Tool orientation

Fig. 9. The smoothed tool tip position and tool orientation

The deviation errors for the tool tip position and tool orientation in WCS, measured against the preset tolerances, are presented in Table 2. It is evident that the maximum deviation errors for both the tool tip position ε_p and orientation ε_o do not exceed the prescribed tolerances. Moreover, since the tool tip position e_p and orientation e_o are controlled by adjustment parameters κ and λ, Table 2 confirms that e_p and e_o are sufficiently small to ensure that the discrepancies between the smoothed tool path and the original tool path remain within acceptable limits (the corresponding adjustment parameter values are provided in parentheses).

Table 2. Tool-tip position and orientation deviations (ε_p = 0.8 mm, ε_o = 0.01 rad)

Number of corners		1	2	3
Tool tip position	ε_p (mm)	0.8000	0.8000	0.8000
	e_p (mm^3) (κ, λ)	5.4858e-06 (0.478, –5.412e-04)	2.3160e-06 (0.511, –1.448e-04)	6.4518e-05 (0.477, 6.106e-04)
Tool orientation	ε_o (rad)	0.0100	0.0100	0.0100
	e_o (rad^3) (κ, λ)	1.0114e-11 (0.477, 0.006)	1.7870e-36 (–2.300e-15, –1.164e-15)	2.0254e-12 (0.463, –0.002)

To further verify the effectiveness of the developed five-axis NC milling tool path smoothing method based on the ACR spline under different error tolerances, the deviation error thresholds for the tool tip position and orientation were reset to ε_p = 1.2 mm and ε_o = 0.005 rad, respectively. As shown in Table 3, the maximum deviation errors for both the tool tip position ε_o and orientation e_o remain below the newly preset tolerances. Furthermore, both e_p and e_o are still sufficiently small to guarantee that the discrepancies between the smoothed tool path and the original tool path are maintained within acceptable limits.

Table 3. Tool-tip position and orientation deviations (ε_p = 1.2 mm, ε_o = 0.005 rad)

Number of corners		1	2	3
Tool tip position	ε_p (mm)	1.2000	1.2000	1.2000
	e_p (mm^3) (κ, λ)	1.856e-05 (0.478, –8.139e-04)	7.818e-06 (0.511, –2.172e-04)	2.1709e-04 (0.477, 9.146e-04)
Tool orientation	ε_o (rad)	0.0050	0.0050	0.0050
	e_o (rad^3) (κ, λ)	1.2656e-12 (0.477, 0.003)	7.2878e-38 (–4.078e-13, –2.823e-16)	2.5308e-13 (0.463, –0.001)

In Section 4, it is deduced that the synchronization errors of the tool orientation relative to the tool tip displacement at the junctions between the curve splines and the linear splines are always zero. To validate this deduction, the first-, second-, and third-order synchronization errors are computed at these junctions, as illustrated in Fig. 10 and Fig. 11. For comparison, the synchronization errors of the PH-spline based smoothed tool path obtained using the method described in [30] is also evaluated on the same test tool path. The results clearly demonstrate that

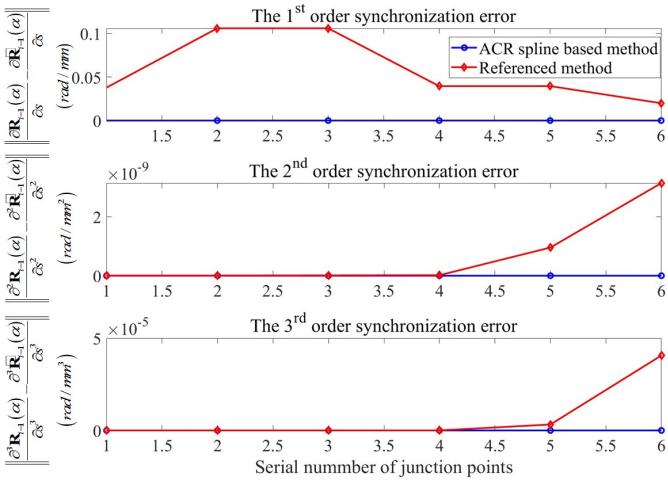


Fig. 10. Orientation-position synchronization errors of the tool (ε_p = 0.8 mm, ε_o = 0.01 rad)

the synchronization errors (at the first, second, and third orders) for the tool path smoothed by the developed ACR-spline-based method are all equal to zero, whereas the corresponding errors for the method in [30] are non-zero. This confirms that the developed method achieves C^3 continuous synchronization of the tool orientation with respect to the tool tip displacement.

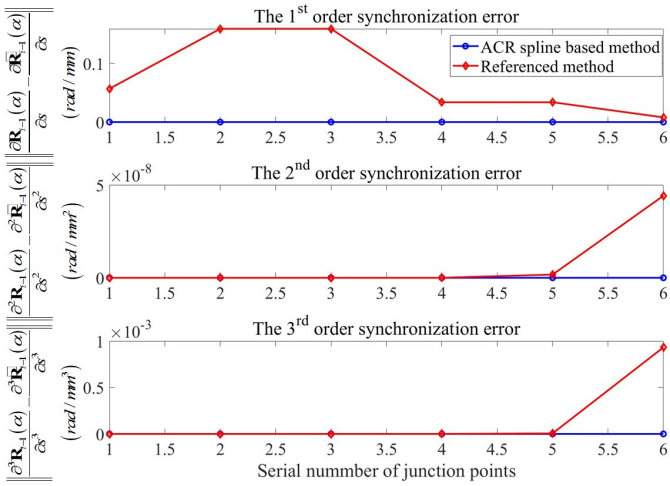
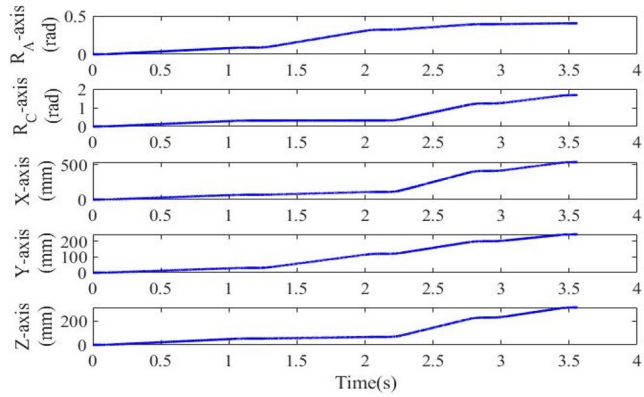


Fig. 11. Orientation-position synchronization errors of the tool (ε_p = 1.2 mm, ε_o = 0.005 rad)

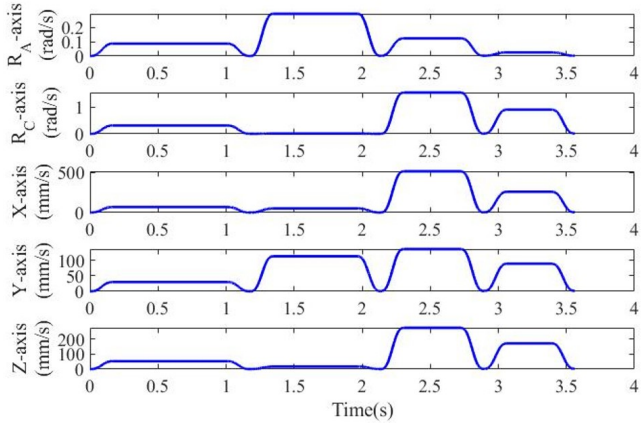
Since the developed ACR-spline-based C^3 continuous corner smoothing method achieves third-order continuity for both the tool tip position and tool orientation, it mathematically guarantees continuous jerk commands. For validation, a C^3 continuous cubic acceleration profile [37] is employed to interpolate the smoothed tool path produced by the developed method under error tolerances of ε_p = 1.2 mm and ε_o = 0.005 rad. The maximum velocity, acceleration, and jerk are set to 50 mm/s, 2000 mm/s^2, and 10 000 mm/s^3, respectively. As shown in Fig. 12, the kinematic profiles – including displacement, velocity, acceleration, and jerk – for each axis are smooth and continuous up to the jerk level, further verifying the effectiveness and practicability of the developed path smoothing method.

The present formulation enforces continuity of the third derivative (jerk) across segment joins but does not explicitly minimize the magnitude of the jerk. In many mechanical applications small jerk magnitudes are desirable to reduce dynamic excitation. The proposed framework can be extended to include jerk-magnitude minimization by adding an integral cost of the form:

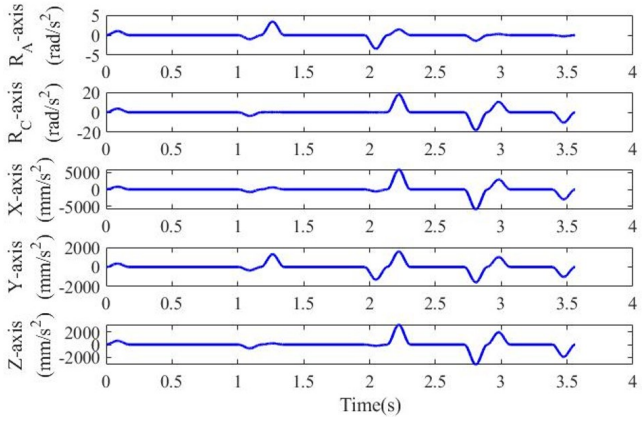
$$J_{\text{jerk}} = \int_{t_0}^{t_1} \left\| r^{(3)}(t) \right\| \mathrm{d}t \tag{38}$$



(a) Displacement of each axis

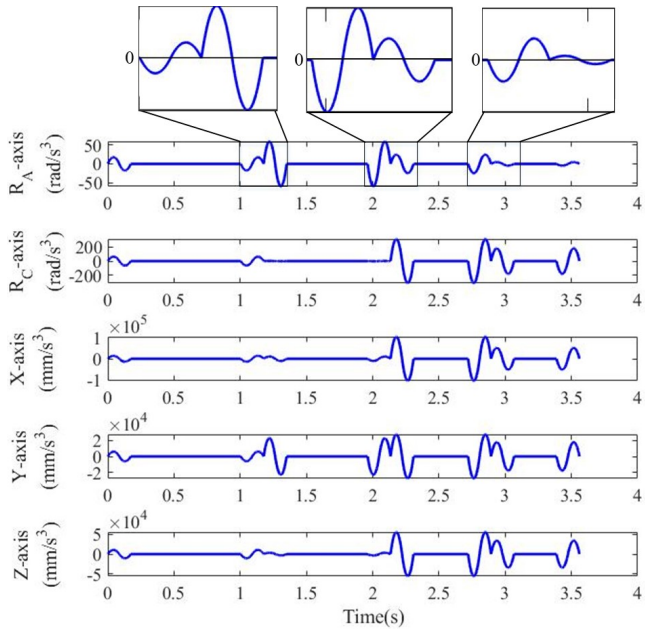


(b) Velocity of each axis



(c) Accelerate of each axis

Fig. 12



(d) Jerk of each axis

Fig. 12. Kinematic profiles of each axis of machine tool

to the optimization objective (where $r(t)$ is the vector-valued curve). When the curve is represented in the chosen basis as $r(t) = \sum_i c_i \phi_i(t)$, the third derivative is linear in the coefficients and the integral above can be written in quadratic form $c^{\mathrm{T}} Q c$ with

$$Q_{ij} = \int_{t_0}^{t_1} \phi_i^{(3)}(t)\, \phi_j^{(3)}(t)\, \mathrm{d}t. \tag{39}$$

Therefore, if the basis functions are fixed (i.e., no nonlinear shape parameters), jerk-magnitude minimization leads to a convex quadratic objective and can be solved efficiently under linear constraints (e.g., deviation bounds, interpolation/synchronization linear constraints). If the basis depends nonlinearly on shape parameters (κ, λ), the resulting objective is nonlinear in those parameters and standard nonlinear solvers (e.g., sequential quadratic programming) are applicable. In practice one may form a weighted multi-objective

$$\min \alpha J_{\text{deviation}} + \beta J_{\text{jerk}} \tag{40}$$

and tune the weight β to trade off geometric fidelity against dynamic smoothness. For transparency we include below a discrete implementation recipe and recommended metrics to evaluate jerk reduction.

6. Conclusions

High-order smoothness of tool paths is essential for computational efficiency and dynamic performance in multi-axis CNC milling. This study presents an ACR spline–based method that attains C^3-continuous tool-path smoothing. By analyzing control points at the junctions of discrete linear segments, the method constrains corner deviation analytically – without iterative computation. Any residual deviations are minimized via a simple, effective optimization. Replacing the remaining linear segments with ACR splines synchronizes tool-tip position and orientation with tool-tip displacement under C^3 continuity, without additional computational cost. Numerical simulations show the method meets maximum deviation tolerances and achieves third-order synchronization of orientation with displacement, yielding smooth jerk transitions. This approach streamlines computation while improving real-world machining performance: reduced computational overhead enables faster processing and lower production costs in high-precision environments. Its fully analytical, non-iterative nature also facilitates integration with adaptive control systems and opens pathways for further advances in dynamic performance and efficiency for future CNC milling applications.

Appendix A. Proof that Eq. (30) defines a convex quadratic and closed-form minimizer

Define the vector-valued functions on the interval $[x_2, x_3]$

$$\mathbf{A}(x) = \begin{bmatrix} A_{y,2}(x) \\ A_{z,2} \end{bmatrix}, \quad \mathbf{B}(x) = \begin{bmatrix} B_{y,2}(x) \\ B_{z,2} \end{bmatrix}, \quad \mathbf{R}(x) = \begin{bmatrix} C_{y,2}(x) - y(x) \\ C_{z,2} - z(x) \end{bmatrix}.$$

Using the definitions from Eq. (31), let:

$$D_1 = \int_{x_2}^{x_3} \|\mathbf{A}(x)\|^2 \,\mathrm{d}x, \qquad D_2 = \int_{x_2}^{x_3} \|\mathbf{B}(x)\|^2 \,\mathrm{d}x,$$

$$D_3 = \int_{x_2}^{x_3} \mathbf{A}(x) \bullet \mathbf{B}(x) \,\mathrm{d}x, \qquad D_2 = \int_{x_2}^{x_3} \mathbf{A}(x) \bullet \mathbf{R}(x) \,\mathrm{d}x,$$

$$D_5 = \int_{x_2}^{x_3} \mathbf{B}(x) \bullet \mathbf{R}(x) \,\mathrm{d}x, \qquad D_6 = \int_{x_2}^{x_3} \|\mathbf{R}(x)\|^2 \,\mathrm{d}x,$$

Then the error function (Eq. (30)) can be written in the quadratic form:

$$e_p(\kappa, \lambda) = D_1\kappa^2 + 2D_3\kappa\lambda + D_2\lambda^2 + 2D_4\kappa + 2D_5\lambda + D_6$$

which equivalently, in matrix notation:

$$e_p(\kappa, \lambda) = \mathbf{v}^{\mathrm{T}}\mathbf{M}\,\mathbf{V} + 2\mathbf{b}^{\mathrm{T}}\mathbf{v} + c$$

with

$$\mathbf{v} = \begin{bmatrix} \kappa \\ \lambda \end{bmatrix}, \quad \mathbf{M} = \begin{bmatrix} D_1 & D_3 \\ D_3 & D_2 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} D_4 \\ D_5 \end{bmatrix}, \quad c = D_6\,.$$

Convexity/positive (semi)definiteness: The quadratic part satisfies the integral identity:

$$\mathbf{v}^{\mathrm{T}}\mathbf{M}\,\mathbf{V} = D_1\kappa^2 + 2D_3\kappa\lambda + D_2\lambda^2 = \int_{x_2}^{x_3} \|\boldsymbol{\kappa}\mathbf{A}(x) + \lambda\mathbf{B}(x)\|^2 \geq \mathrm{d}x \geq 0.$$

Hence $\mathbf{M}$ is positive semidefinite and e_p is a convex quadratic in (κ, λ).

Moreover, by the Cauchy–Schwarz inequality:

$$D_3^2 = \left(\int \mathbf{A} \bullet \mathbf{B}\right)^2 \leqslant \left(\int \|\mathbf{A}|^2\right)\left(\int \|\mathbf{B}\|^2\right) = D_1 D_2$$

and equality holds if and only if $\mathbf{A}(x)$ and $\mathbf{B}(x)$ are almost everywhere linearly dependent. Therefore, in the typical nondegenerate case where $\mathbf{A}(x)$ and $\mathbf{B}(x)$ are not almost everywhere proportional, we have:

$$\det \mathbf{M} = D_1 D_2 - D_3^2 > 0$$

so $\mathbf{M}$ is positive definite. In this nondegenerate case $e_p(\kappa, \lambda)$ is strictly convex and its graph is an elliptic paraboloid (a "bowl" with a unique global minimizer). If $D_1 D_2 - D_3^2 = 0$ (degenerate case), the quadratic form is only semidefinite and the minimizer may be non-unique or may not exist as an isolated point.

Closed-form minimizer (consistency with Eq. (32)): Set the gradient to zero:

$$\nabla_v e_p = 2\mathbf{M}\,\mathbf{v} + 2\mathbf{b} \;\Rightarrow\; \mathbf{M}\,\mathbf{v} = -\mathbf{b}.$$

When $\det \mathbf{M} \neq 0$ (nondegenerate case), the unique solution is

$$\mathbf{v}^* = \begin{bmatrix} \kappa^* \\ \lambda^* \end{bmatrix} = -\mathbf{M}^{-1}\mathbf{b}$$

writing the components explicitly,

$$\kappa^* = \frac{D_3 D_5 - D_2 D_4}{D_1 D_2 - D_3^2}, \quad \lambda^* = \frac{D_3 D_4 - D_1 D_5}{D_1 D_2 - D_3^2}\,.$$

This closed-form result is identical to solving the linear system in Eq. (32).

The minimal value attained at $\mathbf{v}^*$ equals:

$$e_p(\kappa^*, \lambda^2) = c - \mathbf{b}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{b} = D_6 - \frac{D_2 D_4^2 - 2D_3 D_4 D_5 + D_1 D_5^2}{D_1 D_2 - D_3^2}$$

provided $D_1 D_2 - D_3^2 \neq 0$.

Degenerate case remark: If $D_1 D_2 - D_3^2 = 0$, then $\mathbf{M}$ is singular (semi-definite). In that case the quadratic part has a zero direction; minimization must be handled by checking compatibility of $\mathbf{b}$ with the column space of $\mathbf{M}$. Concretely:

If $-\mathbf{b}$ lies in the column space of $\mathbf{M}$, there are infinitely many minimizers (a line of minimizers); choose the one with smallest norm if desired.

If $-\mathbf{b}$ is not in the column space, the gradient equation has no solution and the minimum is attained at the boundary of the feasible set (or the quadratic is unbounded below along the zero-direction depending on linear terms).

Conclusion: The displayed derivation justifies that Eq. (30) defines a convex quadratic in (κ, λ); under the mild nondegeneracy condition $D_1 D_2 - D_3^2 > 0$ it is strictly convex (an elliptic paraboloid), and the minimizer is given by the explicit formula above (consistent with Eq. (32)).

References

[1] K. Erkorkmaz and Y. Altintas. High speed CNC system design. Part I: jerk limited trajectory generation and quintic spline interpolation. *International Journal of Machine Tools and Manufacture*, 41(9):1323–1345, 2001. doi: 10.1016/S0890-6955(01)00002-5.

[2] Y. Liu, M. Wan, X.B. Qin, Q.B. Xiao, and W.H. Zhang. FIR filter-based continuous interpolation of G01 commands with bounded axial and tangential kinematics in industrial five-axis machine tools. *International Journal of Mechanical Sciences*, 169:105325, 2020. doi: 10.1016/j.ijmecsci.2019.105325.

[3] Q. Bi, J. Shi, Y. Wang, L. Zhu, and H. Ding. Analytical curvature-continuous dual-Bézier corner transition for five-axis linear tool path. *International Journal of Machine Tools and Manufacture*, 91:96–108, 2015. doi: 10.1016/j.ijmachtools.2015.02.002.

[4] Y. Sun, Y. Zhao, Y. Bao, and D. Guo. A novel adaptive-feedrate interpolation method for NURBS tool path with drive constraints. *International Journal of Machine Tools and Manufacture*, 77:74–81, 2014. doi: 10.1016/j.ijmachtools.2013.11.002.

[5] Q. Liu and T. Huang. Inverse kinematics of a 5-axis hybrid robot with non-singular tool path generation. *Robotics and Computer-Integrated Manufacturing*, 56:140–148, 2019. doi: 10.1016/j.rcim.2018.11.006.

[6] K. Erkorkmaz. Efficient fitting of the feed correction polynomial for real-time spline interpolation. *Journal of Manufacturing Science and Engineering*, 137(4):044501, 2015. doi: 10.1115/1.4030300.

[7] D. Li, W. Zhang, W. Zhou, T. Shang, and J. Fleischer. Dual NURBS path smoothing for 5-axis linear path of flank milling. *International Journal of Precision Engineering and Manufacturing*, 19(12):1811–1820, 2018. doi: 10.1007/s12541-018-0209-6.

[8] Q. Hu, Y. Chen, X. Jin, and J. Yang. A real-time C^3 continuous local corner smoothing and interpolation algorithm for CNC machine tools. *Journal of Manufacturing Science and Engineering*, 141(4):041004, 2019. doi: 10.1115/1.4042606.

[9] Y. Zhang, P. Ye, H. Zhang, and M. Zhao. A local and analytical curvature-smooth method with jerk-continuous feedrate scheduling along linear toolpath. *International Journal of Precision Engineering and Manufacturing*, 19(10):1529–1538, 2018. doi: 10.1007/s12541-018-0180-2.

[10] Q. Hu, Y. Chen, J. Yang, and D. Zhang. An analytical C^3 continuous local corner smoothing algorithm for four-axis computer numerical control machine tools. *Journal of Manufacturing Science and Engineering*, 140(5):051003, 2018. doi: 10.1115/1.4039312.

[11] Q.B. Xiao, M.Wan, Y. Liu, X.B. Qin, and W.H. Zhang. Space corner smoothing of CNC machine tools through developing 3D general clothoid. *Robotics and Computer-Integrated Manufacturing*, 64:101949, 2020. doi: 10.1016/j.rcim.2020.101949.

[12] X. Yang, Y. You, and W.A. Yang. Simultaneous optimization of curvature and curvature variation for tool path generation in high-speed milling of corners. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 44(2):68, 2022. doi: 10.1007/s40430-022-03360-x.

[13] W. Fan, C.H. Lee, and J.H. Chen. A realtime curvature-smooth interpolation scheme and motion planning for CNC machining of short line segments. *International Journal of Machine Tools and Manufacture*, 96:27–46, 2015. doi: 10.1016/j.ijmachtools.2015.04.009.

[14] H. Zhao, L. Zhu, and 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*, 65:88–98, 2013. doi: 10.1016/j.ijmachtools.2012.10.005.

[15] R.T. Farouki. Construction of G2 rounded corners with Pythagorean-hodograph curves. *Computer Aided Geometric Design*, 31(2):127–139, 2014. doi: 10.1016/j.cagd.2014.02.002.

[16] X. Beudaert, S. Lavernhe, and C. Tournier. 5-axis local corner rounding of linear tool path discontinuities. *International Journal of Machine Tools and Manufacture*, 73:9–16, 2013. doi: 10.1016/j.ijmachtools.2013.05.008.

[17] J. Shi, Q. Bi, L. Zhu, and Y. Wang. Corner rounding of linear five-axis tool path by dual PH curves blending. *International Journal of Machine Tools and Manufacture*, 88:223–236, 2015. doi: 10.1016/j.ijmachtools.2014.09.007.

[18] X. Beudaert, P. Y. Pechard, and C. Tournier. 5-Axis tool path smoothing based on drive constraints. *International Journal of Machine Tools and Manufacture*, 51(12):958–965, 2011. doi: 10.1016/j.ijmachtools.2011.08.014.

[19] A. Hashemian, P. Bo, and M. Barton. Reparameterization of ruled surfaces: toward generating smooth jerk-minimized toolpaths for multi-axis flank CNC milling. *Computer-Aided Design*, 129:102868, 2020. doi: 10.1016/j.cad.2020.102868.

[20] X. Beudaert, S. Lavernhe, and C. Tournier. Feedrate interpolation with axis jerk constraints on 5-axis NURBS and G1 tool path. *International Journal of Machine Tools and Manufacture*, 57:73–82, 2012. doi: 10.1016/j.ijmachtools.2012.02.005.

[21] Y. Jin, Q. Bi, and Y. Wang. Dual-Bezier path smoothing and interpolation for five-axis linear tool path in workpiece coordinate system. *Advances in Mechanical Engineering*, 7(7):1687814015595211, 2015. doi: 10.1177/1687814015595211.

[22] J. Huang, X. Du, and L.M. Zhu. Real-time local smoothing for five-axis linear toolpath considering smoothing error constraints. *International Journal of Machine Tools and Manufacture*, 124:67–79, 2018. doi: 10.1016/j.ijmachtools.2017.10.001.

[23] X. Zhao, H. Zhao, S. Wan, X. Li, and H. Ding. An analytical decoupled corner smoothing method for five-axis linear tool paths. *IEEE Access*, 7:22763–22772, 2019. doi: 10.1109/ACCESS.2019.2898703.

[24] X. Huang, F. Zhao, T. Tao, and X. Mei. A novel local smoothing method for five-axis machining with time-synchronization feedrate scheduling. *IEEE Access*, 8:89185–89204, 2020. doi: 10.1109/ACCESS.2020.2992022.

[25] J. Yang, Q. Qi, A. Adili, and H. Ding. An analytical tool path smoothing algorithm for robotic machining with the consideration of redundant kinematics. *Robotics and Computer-Integrated Manufacturing*, 89:102768, 2024. doi: 10.1016/j.rcim.2024.102768.

[26] A. Yuen, K. Zhang, and Y. Altintas. Smooth trajectory generation for five-axis machine tools. *International Journal of Machine Tools and Manufacture*, 71:11–19, 2013. doi: 10.1016/j.ijmachtools.2013.04.002.

[27] Y. Zhang, T. Wang, J. Dong, P. Peng, Y. Liu, and R. Ke. An analytical G3 continuous corner smoothing method with adaptive constraints adjustments for five-axis machine tool. *The International Journal of Advanced Manufacturing Technology*, 109(3):1007–1026, 2020. doi: 10.1007/s00170-020-05402-x.

[28] S. Tulsyan and Y. Altintas. Local toolpath smoothing for five-axis machine tools. *International Journal of Machine Tools and Manufacture*, 96:15–26, 2015. doi: 10.1016/j.ijmachtools.2015.04.014.

[29] J. Yang and A. Yuen. An analytical local corner smoothing algorithm for five-axis CNC machining. *International Journal of Machine Tools and Manufacture*, 123:22–35, 2017. doi: 10.1016/j.ijmachtools.2017.07.007.

[30] Q. Hu, Y. Chen, X. Jin, and J. Yang. A real-time C^3 continuous tool path smoothing and interpolation algorithm for five-axis machine tools. *Journal of Manufacturing Science and Engineering*, 142(4):041002, 2020. doi: 10.1115/1.4046091.

[31] S.K. Sharma, R.K. Duvedi, S. Bedi, and S. Mann. A method for generating multiple solutions for multipoint five-axis tool positioning. *The International Journal of Advanced Manufacturing Technology*, 100:2509–2520, 2019. doi: 10.1007/s00170-018-2871-3.

[32] R.T. Farouki and S. Li. Optimal tool orientation control for 5-axis CNC milling with ball-end cutters. *Computer Aided Geometric Design*, 30:226–239, 2013. doi: 10.1016/j.cagd.2012.11.003.

[33] B. Ezair and G. Elber. Automatic generation of globally assured collision free orientations for 5-axis ball-end tool-paths. *Computer-Aided Design*, 102:171–181, 2018. doi: 10.1016/j.cad.2018.04.011.

[34] J.Z. Chichell, A. Rečková, M. Bizzarri, and M. Barton. Collision-free tool motion planning for 5-axis CNC machining with toroidal cutters. *Computer-Aided Design*, 173:103725, 2024. doi: 10.1016/j.cad.2024.103725.

[35] R.D. Bergeron, T.D. DeRose, and B.A. Barsky. Geometric continuity, shape parameters, and geometric constructions for Catmull–Rom splines. *ACM Transactions on Graphics (TOG)*, 7(1):1–41, 1988. doi: 10.1145/42188.42265.

[36] J. Li and C. Liu. The C^3 Quasi Catmul–Rom spline function with parameters. *Mathematica Numerica Sinica*, 40(1):96–106, 2018. doi: 10.12286/jssx.2018.1.96. (in Chinese)

[37] K. Erkorkmaz. *Optimal trajectory generation and precision tracking control for multi-axis machines*. Ph.D. thesis, University of British Columbia, Canada, 2004.