Int J Adv Manuf Technol (2017) 91:1369–1379
DOI 10.1007/s00170-016-9821-8

ORIGINAL ARTICLE

Constructing smooth tool orientation field based on radial basis function for 5-axis machining

Kun-Tao Huang[1] · **Zhenyu Zhang**[1] · **Hu Gong**[1] · **Z. J. Li**[2] · **F. Z. Fang**[1] · **Dongfang Wang**[3]

Received: 25 July 2016 / Accepted: 21 November 2016 / Published online: 7 December 2016

Abstract In order to improve the surface quality and dynamics for 5-axis finish machining, new algorithms to construct smooth tool orientation field function are proposed in this paper. Given several surface points and corresponding key tool axes, two methods are studied for open surfaces and half-open surfaces: (a) radial basis function (RBF) interpolation algorithm and (b) RBF-spline approximation algorithm. The two methods can ensure smoothness of all tool axes on the whole surface instead of a single tool path. Finally, several cases are studied to verify algorithms' effectiveness.

Keywords Tool orientation field function · 5-axis machining · Radial basis function · Spline

1 Introduction

5-axis machining technology has been widely used in machining parts with complex shape. Due to high flexibility of 5-axis machine tool, tool path generation of 5-axis machining has higher complexity than traditional 3-axis machining. In academic and industrial community, several issues are still focused on, such as gouge-free, collision-free, smooth tool path generation, singular axis avoiding, and cutter selection [1–3].

The gouge problems in -axis machining are classified into local gouge, rear gouge, and global gouge (collision) problems. For solving local gouge and rear gouge problems, an effective method is to select an appropriate size cutter and a suitable tool orientation [4–7] matching the curvature of surfaces. Local gouge and rear gouge always exist in a saddle point and concave point. For global collision problems, Lee et al. [8] provide feasibility cone checking method to confirm suitable cutter and tool orientation. More researches [9–13] focus on how to calculate feasible C-space of tool orientation. Based on two rotary degrees of freedom, tilt angle φ, and yaw angle θ, C-space of tool orientation means the space spanned by the two variables [14]. To avoid collision, it is ensured that tool orientation of each point lies in the respective feasible C-space.

Singular problem is another important problem existing in the inverse kinematical transformation (IKT) process [15]. In the vicinity of a singular axis, a small change of the tool orientation in workpiece coordinates system (WCS) would lead to an abrupt change of kinematic axes in machine coordinates system (MCS). This will lead to unsmooth tool orientation changes. To solve this problem, Affouard et al. [15] pointed out singular problem will cause positioning uncertainties in 5-axis machine tools. They adopted a dual B-spline format to describe tool path and proposed a method to deform the tool path, so that the tool orientation does not traverse the singular cone. Based on the domain of admissible orientation (DAO) concept, Castagnetti et al. [16] proposed a method to optimize tool path. The singularity can be detected with DAO for AC-type or BC-type machine tools. Sørby [17] presented a post-processing algorithm to avoid the singular

✉ Hu Gong
gonghu2012@gmail.com

[1] State Key Laboratory of Precision Measuring Technology and Instruments, Tianjin University, Tianjin Shi 300072, China

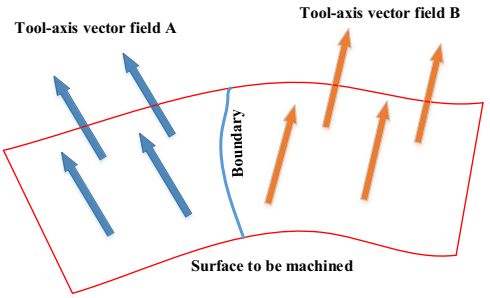
[2] Tianjin Key Laboratory of High Speed Cutting and Precision Machining, Tianjin University of Technology and Education, Tianjin 300222, China

[3] State Key Laboratory of Applied Optics, Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China

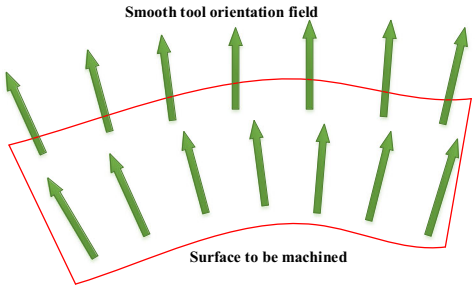
problem. The algorithm modifies the exact inverse kinematics to give robustness to singularities at the expense of a small tool orientation deviation. Lin et al. [18, 19] look into the irregular-machined surface textures in the process of avoiding 5-axis singularities. A PSO method is intergraded into the tool orientation translation method. To avoid a singular problem, a singular cone is built that tool path can be adjusted to avoid the cone. After optimization, the singular problem is avoided, and the surface textures are controlled.

Poor smoothness of tool path leads to discontinuous move of machine tools, which may increase machining time and destroy surface quality. Fleisig [20] adopted a near arc-length parameterized quintic polynomial spline to interpolate position spline and adopted a near arc-length parameterized quintic spherical Bezier spline to interpolate orientation spline. Smooth motion is accomplished with an orientation re-parameterization spline. The tool path is C^3 continuous and ensures constant feed and limited angular acceleration. Ho [21] described tool orientation by employing quaternion to smooth the tool orientation and reduce machining error. Masatoshi [22] and Jean [23] used dual spline to describe tool path and interpolated the two spline curves using polynomial interpolation algorithm and NURBS algorithm. Yuen [24] interpolate the tool path by quintic spline to ensure C^3 continuous. The non-linear relationship between spline parameters and locations along the tool path is approximated with ninth order and seventh order feed correction splines for position and orientation respectively.

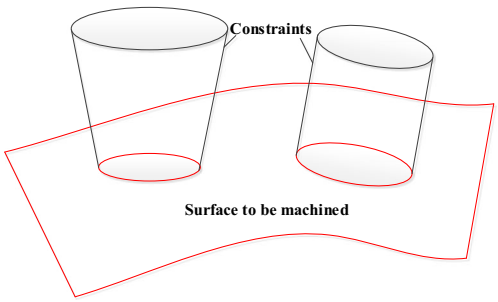
Errors in 5-axis machining process cause deviations in tool position and orientation from NC code, which will consequently affect geometric accuracy of the machined surface [25, 26]. Therefore, abrupt changes of tool orientations between two adjacent tool paths will worsen surface quality. As shown in Fig. 1a, the surface to be machined is divided into two regions with different tool orientation fields. After machining, machining trace will be left at the boundary between two regions. In a machining environment, many error sources cause kinematic errors, like assembly inaccuracies of machine tool servo axes, cutter shape error, thermo-mechanical errors, loads, and dynamic forces [27], and it is hard, even impossible to remove or compensation such errors. To achieve the goal of improving the surface quality, we hope that any adjacent tool postures have smooth transition to keep the smooth variation of error of the machine tool. Therefore, we provide a new methodology to construct smooth tool orientation field for 5-axis machining in this paper, as shown in Fig. 1b, especially for surface with gouge-free constraints shown in Fig. 1c. In Section 2, algorithms of constructing tool orientation field are illustrated. In Section 3, case study and experimental study are conducted, and conclusion close this paper in Section 4.



(a) Discontinuous tool orientation field



(b) Continuous tool orientation field



(c) Surface with constraints

Fig. 1 The description of the problem. **a** Discontinuous tool orientation field. **b** Continuous tool orientation field. **c** Surface with constraints

2 The algorithms of constructing tool orientation field

2.1 The basic idea of constructing tool orientation field

Many researches tried to generate gouge-free 5-axis tool path automatically [8, 12, 28]. The main algorithm process is shown as below. (1) Sample the surface into several points. (2) Calculate feasible C-space (FC-space) of each point. (3) Based on FC-space of each point, design an algorithm to determine tool orientation of each point. However, two disadvantages should be noticed. (1) The expression of FC-space is discrete, which leads to high storage space and long calculating time. (2) It is complicated to design such an algorithm to calculate and

optimize tool orientations. Therefore, an interactive tool path generation algorithm will be proposed in this section.

The entire tool path generation algorithm is shown as Fig. 2. First, key surface points and key tool axes should be selected. The basic principle is to restrict tool axis at some positions with very high possibility of collision. To detect these areas, Liu et al. [12] present a method of tool posture global collision-free area generation for 5-axis machining point clouds. For obtaining precise collision-free area efficiently, discrete inclination angles and their collision-free rotation angle ranges are calculated to construct the area in two dimensional configuration space (C-space), which avoids mapping obstacles to C-space or searching boundaries. And to confirm such key points and axes, human–computer interaction can be conducted. Second, tool orientation field function (TOFF) is calculated. With the reason that TOFF calculation is to solve linear equation systems, failure may occur if the matrix is ill-conditioned and singular. So checking whether the output value is available is necessary. If exceptions occur in solving equation systems, the quantity or positions of key points should be adjusted, or algorithm arguments should be modified. Third, tool path can be generated based on surface points and TOFF. Since we are not sure that the calculated tool path is gouge free, so the gouge check process is necessary. When gouge is detected, key surface points and key tool axes should also be adjusted or added. Then TOFF should be recalculated. Repeat the three steps until gouge is not detected. With the quantity of key points increasing, the risk that exceptions occur in TOFF calculation rises up. So step *d* is necessary. If TOFF calculation fails and positions of key points are all rational, the whole process is crashed. To improve the calculation, two steps play key roles. One is how to select key surface points and key tool axes, the other one is to design a TOFF calculation algorithm.

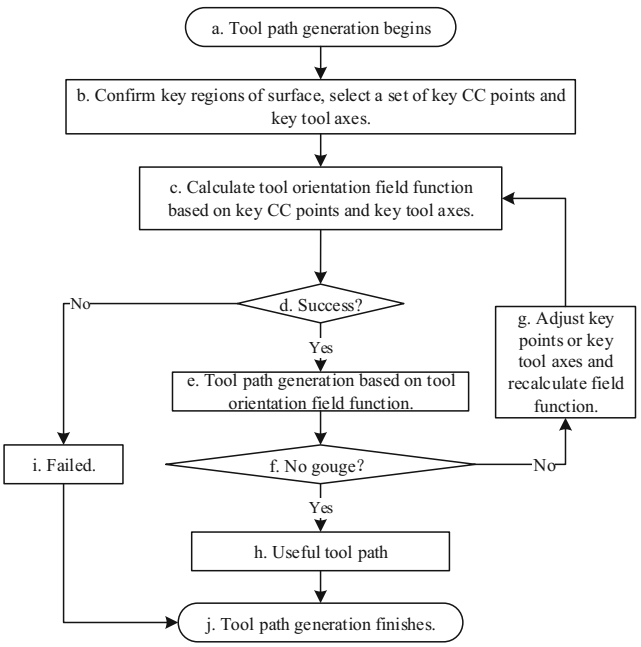


Fig. 2 Flow chart of interactively tool path generation

In this study, the main goal is just to construct smooth and gouge-free function $\boldsymbol{T}(u, v)$, which is step *c* in the flow chart. We assume that key surface points and key tool axes have been determined, so TOFF calculation can be conducted. For a surface $\boldsymbol{S}(u, v)$, let $\boldsymbol{CC}_m(m = 0\ldots M)$ denote the key surface points, $\boldsymbol{T}_m(m = 0\ldots M)$ denote the key tool axes, $(u_m, v_m)(m = 0\ldots M)$ denote parameters at key surface points, and $\boldsymbol{T}(u, v)$ denote TOFF.

2.2 The algorithm of constructing TOFF

Given scattered key tool axes on the surface, a fitting algorithm based on multi-dimensional data points is to be designed to calculate TOFF. Fitting problem can be solved by two main algorithms based on whether TOFF pass key tool axes exactly.

The first one is interpolation algorithm. The multi-dimensional interpolation problem is encountered in computer graphic, reverse engineering, image processing, etc. [29]. Several algorithms are proposed by previous researches. Franke R [30] summarized them and classified into six types, that is, inverse distance-weighted methods, rectangle-based blending methods, triangle-based blending methods, finite element-based methods, Foley's methods, and global basis function type methods. The second one is approximation algorithm. If the output of interpolation algorithm cannot be accepted, an approximation function can be adopted. Giving a tolerance, the distance between the given points and the function is required to be smaller than the tolerance.

In this paper, radial basis function interpolation (RBF interpolation) is adopted, which is one of global basis function type methods. For methods mentioned in the previous paragraph, rectangle-based blending requires that the input data should be on the parametric grid. Other methods are with complex processes. RBF method belongs to inverse distance-weighted methods. The advantage is that the process is simple and the interpolated function is with enough continuity with the base function selection. And the disadvantage is when the quantity of input data is large, long computational time will be consumed and solving failure will occur. In our study, the quantity of key points is usually not large, so the disadvantage is inhibited. On the other hand, to approximate the input data, RBF-spline approximation algorithm is proposed subsequently.

Two types of surfaces will be discussed with different TOFF calculating algorithms, open surface and half-open surface. As shown in Fig. 1a, the boundary of two parameters *u* and *v* are both open, which is called open surface. For half-open surface, one parameter direction is closed. Here, we assume that *u* direction is open and *v* direction is closed. The TOFF should be

smooth on the closed iso-v boundary line. It is necessary to design a different TOFF calculation algorithm.

2.2.1 Construct TOFF for open surface

RBF interpolation algorithm As introduced in Section 2.1, key tool axes are $\boldsymbol{T}_m$, surface points are $\boldsymbol{CC}_m$,. The RBF interpolation formula is given as

$$\boldsymbol{T}(u,v) = \sum_{m=0}^{M} \boldsymbol{c}_m \phi(d_{mi}) = \sum_{m=0}^{M} \boldsymbol{c}_m \phi(\|\boldsymbol{CC}_m - \boldsymbol{CC}_i\|), \quad (1)$$

where $\boldsymbol{CC}_\mathrm{i}$ is the point with parameters (u, v) of the surface. $D_\mathrm{mi} = \|\boldsymbol{CC}_m - \boldsymbol{CC}_\mathrm{i}\|$ is distance between $\boldsymbol{CC}_i$ and key surface points $\boldsymbol{CC}_\mathrm{m}$. $\phi(\cdot)$ denotes a function of distance. For the definition of distance $\|\cdot\|$, it is not unique. In open surface, $\|\cdot\|$ is suitable for defining geodesic distance, or linear distance for simplification.

The coefficient $\boldsymbol{c}_\mathrm{m}$ has three components, the same as tool axis, and the components are independent. To calculate $\boldsymbol{c}_\mathrm{m}$, Eq. (2) is built, which contains 3 linear equations.

$$\sum_{m=0}^{M} \boldsymbol{c}_m \phi(\|\boldsymbol{CC}_m - \boldsymbol{CC}_i\|) = \boldsymbol{T}_i, i = (0...M). \quad (2)$$

The next step is to select radial basis function. To ensure the systems of linear equations is solvable, the matrix $\boldsymbol{\Phi}$ constructed by $\phi(\cdot)$ should be invertible. The two typical radial basis functions meet this condition.

(1) Exponential Gaussian distribution function: $\phi(t) = exp.(-(a \cdot t)^2)$, $a \in \mathbf{R}_+$.
(2) Rational inverse multi-quadric distribution function: $\phi(t) = (c^2 + t^2)^{-\beta}$, $c \in \mathbf{R}$, $\beta \in \mathbf{R}_+$.

To confirm the type and coefficients selection, two different basis functions with different coefficients are shown in Fig. 3. The main characteristics of basis function are as follows. (a) $\phi(0) = 1$ and $\phi'(0) = 0$. (b) $\phi(t)$ is attenuated, and when t tends to positive infinite, $\phi(t)$ tends to 0. (c) The matrix $\boldsymbol{\Phi}$ is diagonally dominant matrix. When the attenuation rate is large enough, the matrix $\boldsymbol{\Phi}$ is far from ill-conditioned and singular, which raises the possibility to solve the system of linear equations. Gauss basis function and IMQ basis function both satisfy with these characteristics mentioned above. However, if the basis function attenuates too fast, the smoothness of the function is declined. Therefore, IMQ basis function is more suitable than Gauss basis function. The coefficient a can be used for adjusting attenuation rate. A fast attenuation rate declines smoothness of output function, while a small attenuation rate reduces stability of solving process. For key points selection, when distance between two nearest key points is too small, a will have to be small that may worsen smoothness of the whole TOFF. Therefore, when selecting key surface points, the density should be uniform if possible on the surface.

RBF-spline approximation algorithm In some occasions, RBF interpolation cannot get satisfactory smooth function. A spline interpolation process following RBF interpolation will be proposed in this section, which is called RBF-spline approximation algorithm, as shown in Fig. 4. The algorithm is shown as follows. (1) Interpolate the original key points with RBF interpolation algorithm to get RBF function. (2) Sample the interpolated function by given constant parameter interval (Δu, Δv) to get grid points of RBF function, called grid points. (3) Re-interpolate the RBF grid points to get a new function. In steps 2 and 3, it is obvious that the new function does not go through the original key points accurately. For a given tolerance, this approximation algorithm improves the smoothness of the function.

To calculate the grid points $\{\boldsymbol{T}_\mathrm{ij}\}$, B-spline interpolation algorithm is adopted [31]. Given number of sample points N_u and N_v along u direction and v direction, so (Δu, Δv) can be calculated as follows:

$$(\Delta u, \Delta v) = \left(\frac{u_\mathrm{max} - u_\mathrm{min}}{N_u}, \frac{v_\mathrm{max} - v_\mathrm{min}}{N_v} \right),$$

assuming degrees of B-spline function along u direction and v direction are $p_u = p_v = p \le 3$. Since the RBF grid points are uniformly sampled, the parameters u_i and v_i are equally spaced, and the knot vector $\boldsymbol{U}$ and $\boldsymbol{V}$ are also equally spaced, that is shown in Eq. (3) and ($\boldsymbol{V}$ is the same as $\boldsymbol{U}$).

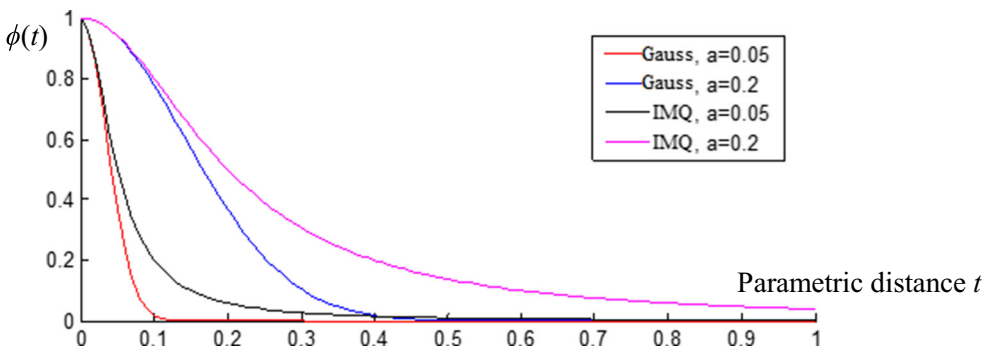


Fig. 3 Different radial basis function

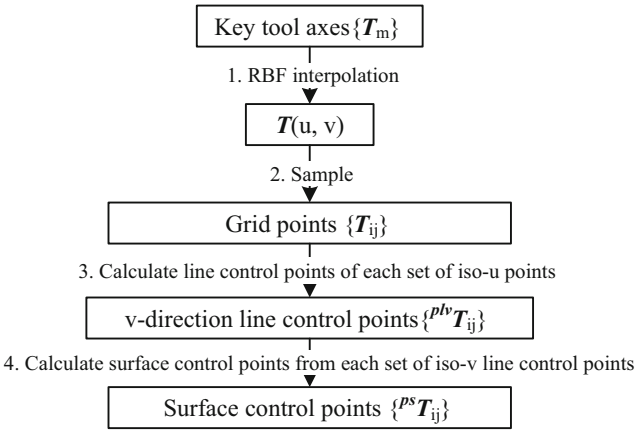


Fig. 4 RBF-spline approximation algorithm flow chart

$$\begin{cases} u_i = \dfrac{i}{N_u}, i = 0, 1, ..., N_u \\ \boldsymbol{U} = \left\{ \overline{u}_0, \overline{u}_1, ..., \overline{u}_{n+p+1} \right\} \\ \overline{u}_0 = \cdots = \overline{u}_p = 0, \overline{u}_{n+1} = \cdots = \overline{u}_{n+p+1} = 1 \\ \overline{u}_{j+p} = \dfrac{j}{n-p+1}, j = 1, 2, ..., n-p \end{cases} . \tag{3}$$

The surface control points $\{^{ps}\boldsymbol{T}_{ij}\}$ can be calculated by solving systems of linear equations as Eq. (4).

$$\begin{cases} \sum_{j=0}^{N_u} B_{j,p}(u_i)^{ps} T_{ij} = {}^{plv}T_{ij}, \quad i = 0...N_u \\ \sum_{i=0}^{N_v} B_{i,p}(v_i)^{plv} T_{ij} = T_{ij}, \quad j = 0...N_v \end{cases} . \tag{4}$$

Confirmed $\{^{ps}\boldsymbol{T}_{ij}\}$, the TOFF can be calculated by B-spline surface calculating rules.

2.2.2 Construct TOFF for half-open surface

RBF interpolation algorithm One main exception of half-open surface is that geometric space is continuous on some parametric boundary while the parametric space is not-continuous. Without loss of generality, let u is parameter of latitude and v is that of longitude, so latitude curve of the earth is iso-v curve. The two v direction boundaries are on the same longitude curve, so change of the tool orientation should be smooth on the v direction boundaries. The TOFF calculation algorithm mentioned above has to be improved to adjust this case.

Allowing for that (u, v) parameters in two directions are with different metrologies, two weights are adopted. A cylinder that contains the surface with the smallest diameter and the smallest height is constructed as shown in Fig. 5. The diameter is D, and the height is H. The ratio between two weights can be obtained, that is, $mu{:}mv = H{:}\pi D$.

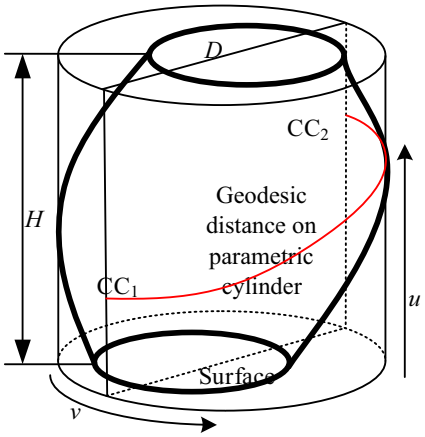


Fig. 5 Distance definition on parametric cylinder of half-open surface

For half-open surface TOFF calculation, the input is similar to that of open surface calculation. Equations (1) and (2) are still available. The parametric space is mapped to the smallest containing cylinder. The distance $\|\cdot\|$ is defined as geodesic distance on the cylinder.

The other question is how to select basis function. Two drawbacks of two basis functions in open surface are obvious. First, discontinuity occurs in v parameter boundary. Second, when a certain CC_1 point moves on the cylinder, distance between CC_1 and CC_2 is abruptly changed that will cause first-order discontinuity of TOFF. A new basis function is designed to adapt to TOFF calculation for half-open surface.

Before basis function $\phi(t)$ is designed, firstly, we need to normalize v parameter, and transform u parameter with the same ratio. The $\phi(t)$ is required that when $t > 0.5$, $\phi(t) = 0$. It is reasonable because CC_1 is totally not affected by CC_2 when CC_1 is far enough from CC_2. Therefore, $\phi(t)$ can be designed as Eq. (5).

$$\begin{cases} g(t) = \sum_{i=0}^{N} c_i t^i \\ \phi(t) = \begin{cases} 0.5(\cos(\pi \cdot g(t/a)) + 1) & 0 < t \le a \\ 0 & t > a \end{cases} \quad a < 0.5 \end{cases} . \tag{5}$$

Let $g(0) = 0, g(a) = 1, g^{(i)}(a) = 0$ ($i = 1..N-1$). When $N = 2$, $g(t) = 2t - t^2$. It can be proved that $\phi(t)$ is with the third order at

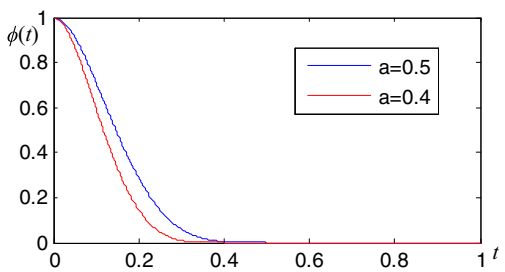
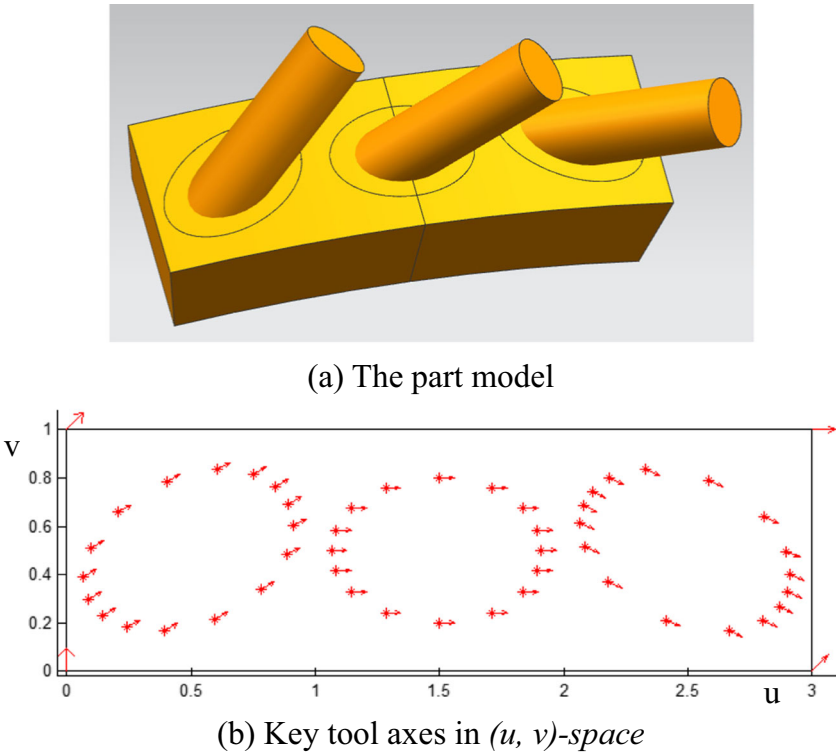


Fig. 6 Radial basis function in half-open surface tool orientation interpolation



(a) The part model

(b) Key tool axes in *(u, v)-space*

Fig. 7 The model of case 1. **a** The part model. **b** Key tool axes in (*u*, *v*) *space*

point $t = a$. To achieve higher order continuity, N can be larger. The functions with different coefficient a are shown in Fig. 6. Attenuation rate is determined by a.

RBF-spline approximation algorithm The main process of RBF-spline approximation is similar to the process in Fig. 4. However, due to the closed V direction, step 3 is different. Closed uniform spline interpolation algorithm is adopted in step 3.

The main difference between closed uniform spline interpolation and traditional open spline interpolation is that we should use a different method to confirm control points and knot vector $\boldsymbol{V}$. In closed uniform spline interpolation, the first p control points are the same as the last p control points. So the quantity of v direction control points is $N_v + p$ and $P_{Nv} = P_0$, $P_{Nv+1} = P_1$,…, $P_{Nv+p-1} = P_{p-1}$. Knot vector $\boldsymbol{V}$ is $\boldsymbol{V} = \{-p/N_v, -(p-1)/N_v, \ldots, -1/N_v, 0, 1/N_v, \ldots, (N_v-1)/N_v, 1, 1 + 1/N_v, \ldots, 1 + p/N_v\}$. So $\{{}^{plv}\boldsymbol{T}_{ij}\}$ can be calculated by Eq. (6).

$$\sum_{j=i}^{p+1} B_{j,p}(t_i)^{plv}T_{kj} = T_{ki}, \quad i = 0...N_v \quad k = 0...N_u\,. \tag{6}$$

2.2.3 Evaluation indexes

To analyze the effectiveness of different algorithms, two indexes are proposed. The first one is used for evaluating TOFF smoothness, and the second one is for evaluating fitting precision.

For smoothness evaluation, first, let us see how to evaluate smoothness of 1D function. For a 1D function, the smaller the stretching energy is, the smoother the function is. The stretching energy is the integral of square of the first differential, which is expressed as Eq. (7) [32].

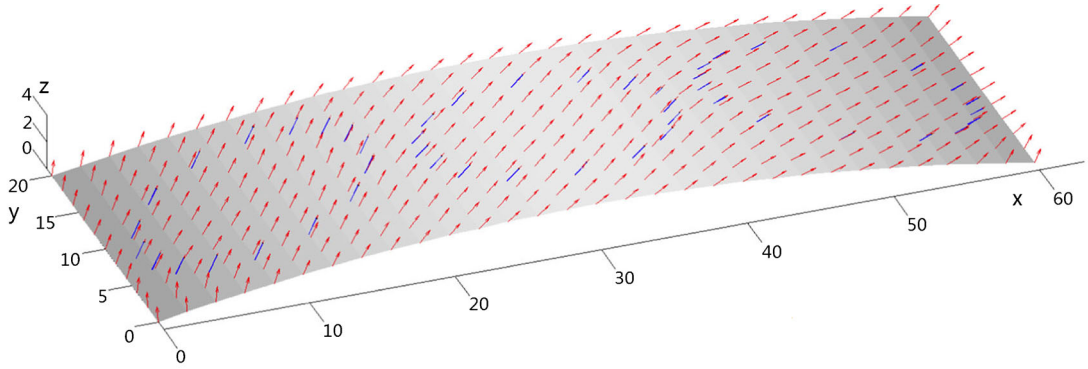
$$S_{LC} = \int_0^1 \left(\frac{1}{2}\cdot\frac{dA(t)}{dt}\cdot\frac{dA(t)}{dt}\right)dl = \int_0^1 \left(\frac{1}{2}\cdot A_t{}^2\right)dt. \tag{7}$$

For 2D function, a total differential operator is replaced with the first differential in 1D function.

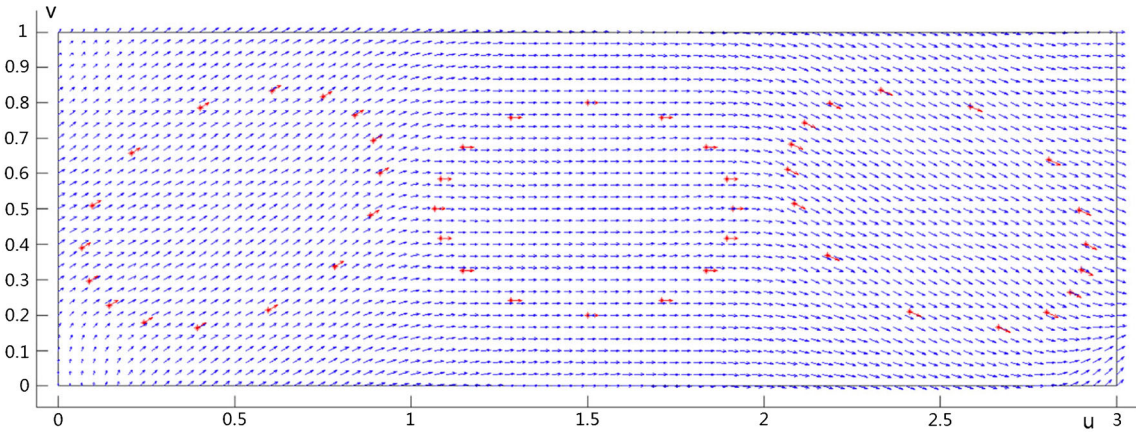
$$S_{TC} = \int_0^1\int_0^1 \frac{1}{2}\left(T_u(u,v)du + T_v(u,v)dv\right)^2. \tag{8}$$

The integral cannot be calculated, S_{TC} is converted to S_{TD}.

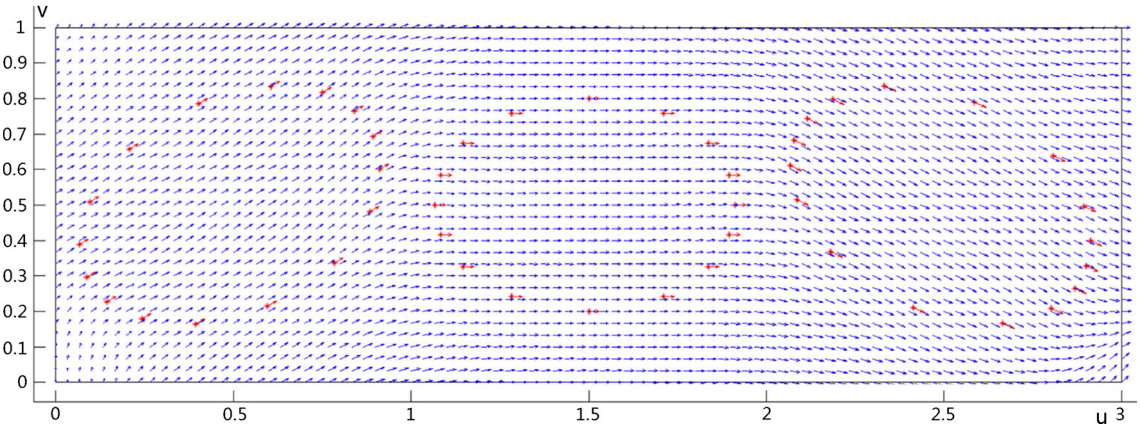
$$S_{TD} = \frac{1}{N_u-2}\frac{1}{N_v-2}\sum_{i=1}^{N_u-2}\sum_{j=1}^{N_v-2}\frac{1}{8}\left(\begin{array}{l}(T((\mathrm{i}+1)\Delta u,\mathrm{j}\Delta \mathrm{v})-T((\mathrm{i}-1)\Delta u,\mathrm{j}\Delta \mathrm{v}))\Delta u+\\ (T(\mathrm{i}\Delta u,(\mathrm{j}+1)\Delta \mathrm{v})-T(\mathrm{i}\Delta u,(\mathrm{j}-1)\Delta \mathrm{v}))\Delta \mathrm{v}\end{array}\right)^2. \tag{9}$$



(a) Tool axis field of Case 1



(b) RBF algorithm



(c) RBF-spline algorithm

Fig. 8 Interpolating results of case 1. **a** Tool axis field of case 1. **b** RBF algorithm. **c** RBF-spline algorithm

The smaller S_{TD} is, the smoother $\boldsymbol{T}(u, v)$ is.

For TOFF-fitting precision with key tool axes, fitting precision accuracy index I_{fa} is calculated. Allowing for the complexity in calculating angle between two vectors, the inner product is calculated. The relationship between inner product between two vectors and angle is shown at Eq. (10).

$$\boldsymbol{T}_1 \cdot \boldsymbol{T}_2 = \cos\langle \boldsymbol{T}_1, \boldsymbol{T}_2 \rangle \approx 1 - \frac{\langle \boldsymbol{T}_1, \boldsymbol{T}_2 \rangle^2}{2}. \tag{10}$$

Table 1 Condition of interpolating algorithm in case 1

Algorithm	RBF	RBF-spline
Parameters	$\phi(t) = (a^2 + t^2)^{-1}$ $a = 0.8$	RBF calculation parameters are the same as RBF alg. The grid interval: $N_u = 30$, $N_v = 10$
Calculation result	Figure 8b	Figure 8c
S_{TD} ($N_{uD} = 900$, $N_{vD} = 300$)	0.9227	0.9056
I_{fa}	1	0.999982

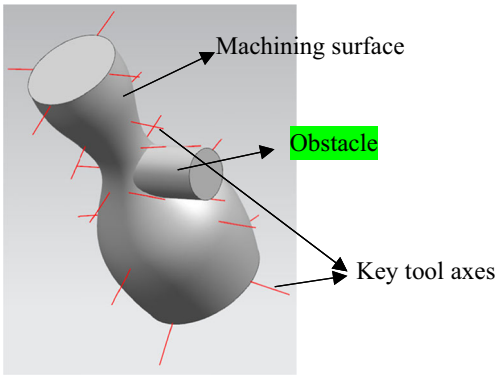
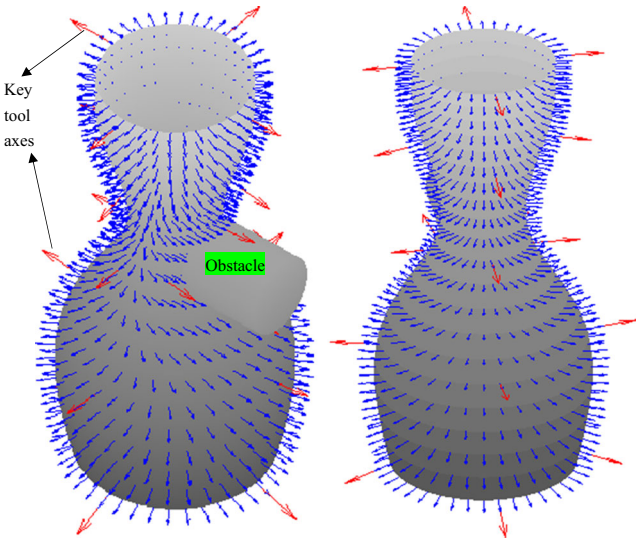


Fig. 9 Key tool axes of bowling



(a) Front view (b) Back view

Fig. 10 Bowling TOFF calculation result. **a** Front view. **b** Back view

Table 2 Bowling TOFF calculating parameters

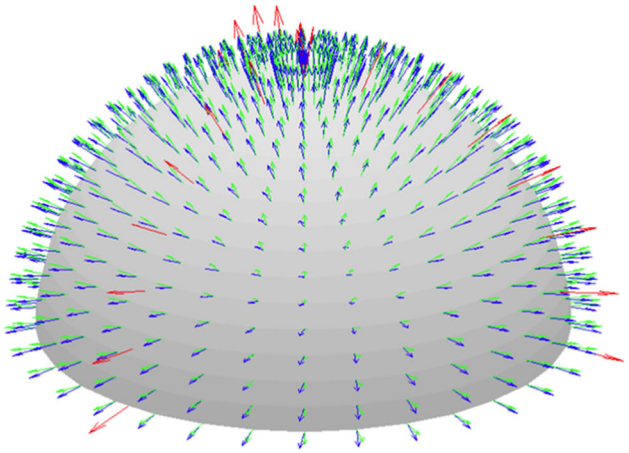
Algorithm	Parameter	S_{TD} ($N_{uD} = 60$, $N_{vD} = 100$)	I_{cl_si}
RBF alg.	$\phi(t)$ is (5). $a = 0.4$	19.9887	1
RBF-spline $N_u = 30$, $N_v = 20$		19.9079	0.999999999986

mu:*mv* = 0.646:1

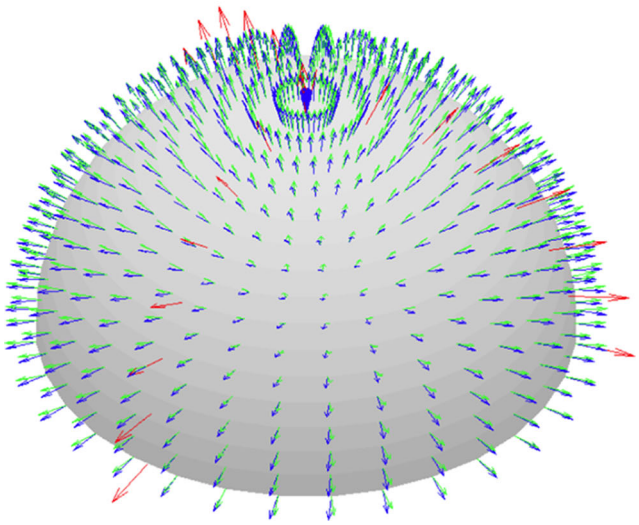
Therefore, I_{fa} can be expressed at Eq. (11).

$$I_{fa} = \frac{\sum_{i=0}^{q} (\boldsymbol{T}_{cl}(i\Delta t)\cdot\boldsymbol{T}_{si}(\mathrm{P}_{cl}(i\Delta t)))}{q+1}. \quad (11)$$

The closer to 1 I_{fa} is, the higher fitting precision is.



(a) RBF-Spline Algorithm



(b) RBF Algorithm

Fig. 11 Interpolating results of case 3. **a** RBF-spline algorithm. **b** RBF algorithm

Table 3 Condition of interpolating algorithm in case 3

Algorithm	RBF	RBF-spline
Parameters	$\phi(t)$ is (5). $a = 0.4$	RBF calculation parameters are the same as RBF alg. The grid interval: $N_u = 8$, $N_v = 32$
S_{TD} ($N_{uD} = 80$, $N_{vD} = 480$)	9.35568	9.30947
I_{fa}	1	0.9999972

3 Case study

To verify the algorithms, three cases are calculated below. The first case represents open surface with obstacles, the second one represents half-open surface, and the third one is for machining experiments.

Case 1: The part is shown in Fig. 7a. The surface is expressed as Eq. (12). Three different-oriented oblique cylinders are obstacles to be avoided. Three rotary axes are parallel to (2, 1, 2), (2, 0, 2), and (2, −1, 2); passing points (9.93390, 10.00000, 1.57440), (30.00000, 10.00000, 2.83766), and (50.06610, 10.00000, 1.57440). Key points contains two parts. One is sampled from the boundary of the cylinders, while the other part is four corners. For key tool axes, the first part is parallel to rotary axes, and the second part is (0 1 4), (1 1 1), (1 1 4), and (1 0 1) on parameter points (0, 0), (1, 0), (0, 1), and (1, 1), shown in Fig. 7b.

$$\begin{cases} x(u,v) = 160\sin((2u-1)\theta) + 160\sin(\theta) \\ y(u,v) = 20v \\ z(u,v) = 160\cos((2u-1)\theta) - 160\cos(\theta) \end{cases}, \tag{12}$$

where $\theta = \arcsin(30/160)$.

TOFF calculating results with different algorithms are shown in Fig. 8. From Fig. 8b, c, the two tool orientation fields look very similar, so the indexes are calculated in Table 1. It is obvious that using RBF algorithm, I_{fa} is 1, but S_{TD} is smaller than that of RBF-spline. To analyze different densities of RBF grid points using RBF-spline, another sample density is that $N_x = 45$, $N_y = 15$. The calculated indexes show that with RBF grid data increasing, S_{TD} is declining, and I_{fa} is closer to 1.

Case 2: As shown in Fig. 9, a bowling surface traversed with a cylinder obstacle is one example for half-open surface TOFF calculation. The bowling surface is processed, and the cylinder is obstacle. The red lines are key tool axes. Due to the similar calculating result by RBF and RBF-spline, just the result calculated by RBF-spline is shown in Fig. 10. The conditions and TOFF calculated parameters are shown in Table 2.

Case 3: To machine revolution surfaces or quasi-revolution surfaces, spiral machining is adopted. Usually, parameters of surfaces are similar to latitude and longitude of earth. A half-sphere is selected for both calculating and machining. Key points are distributed uniformly on latitude and longitude curves. Quantities of key tool axes on latitude and longitude are signed as N_{uKey} and N_{vKey}. And to select key tool axes, the tilt angles are 10° larger than latitude of points location, and the yaw angles are the same as longitude as Fig. 11. The reason that tilt angles are 10° larger than latitude is that the singular axis and the tool tip are ought to be kept away. The TOFF calculating results using RBF and RBF-spline are shown in Fig. 11a, b. The green arrows are the normal of surface, and the blue ones are TOFF. The condition and TOFF calculated parameter are shown in Table 3.

The half-sphere machining is conducted. The material is aluminum alloy. The sphere is $\phi 20$, and the $\phi 4$ ball-end milling tool is used for finish milling. To generate tool path, CC points are calculated first.

$$\begin{cases} u = t \\ v = \mathrm{mod}(200*t) \end{cases} \quad t \in [0, 1) .$$

(a) Region division strategy (b) Our paper's strategy

Fig. 12 Region division strategy VS smooth tool orientation field strategy. **a** Region division strategy. **b** Our paper's strategy

$$\begin{cases} x = R*\sin(\pi u(t)/2)*\cos(2\pi v(t)) \\ y = R*\sin(\pi u(t)/2)*\sin(2\pi v(t)) \\ z = R*\cos(\pi u(t)/2) \\ N_{\text{uKey}} = 11, N_{\text{vKey}} = 41 \end{cases}.$$

CL points are calculated by Eq. (13).

$$CL = CC + R{\cdot}N_s(CC) - R{\cdot}T_{\text{toolAxis}}. \quad (13)$$

The algorithm is used RBF algorithm.

As shown in Fig. 12, the machining trace is cleared up.

Depth of cut is 2 mm, feed rate is 0.05 mm/r, spindle speed is 20,000 r/min.

4 Conclusion

In order to improve surface finish quality, we developed an algorithm system of constructing smooth tool orientation field function for 5-axis machining. For given key tool axes and key surface points, TOFF calculating algorithm is proposed. Allowing for diversity in surface parameterization, algorithms for open surface and half-open surface was studied separately. Usually, RBF-spline approximation algorithm is adopted widely because it can obtain a smoother TOFF. When a TOFF calculation is with high risk of gouge and collision due to high-orientation accuracy requirement, RBF interpolation can be adopted. These algorithms can help us obtain a smooth tool orientation planning in tool path generation. With suitable tool location planning strategies, the tool path can generate surface with qualified surface finish. Finally, a machining experiment is conducted to verify algorithms' effectiveness.

Acknowledgements The authors appreciate the support from the Funds for International Cooperation and Exchange of the National Natural Science Foundation of China (Grant no. 51320105009), the National Natural Science Foundation of China (Grant no. 51275344), and the State Key Laboratory of applied optics of P. R. China.

References

1. Balasubramaniam M, Ho S, Sarma S, Adachi Y (2002) Generation of collision-free 5-axis tool paths using a haptic surface[J]. Comput Aided Des 34(4):267–279
2. Ding S, Mannan MA, Poo AN (2004) Oriented bounding box and octree based global interference detection in 5-axis machining of free-form surfaces[J]. Comput Aided Des 36(13):1281–1294
3. Lasemi A, Xue DY, Gu PH (2010) Recent development in CNC machining of freeform surfaces: a state-of-the-art review. Comput Aided Des 42(7):641–654
4. Lee YS (2010) Non-isoparametric tool path planning by machining strip evaluation for 5-axis sculptured surface machining[J]. Comput Aided Des 30(7):559–570
5. Gong H, Cao LX, Liu J (2008) Second order approximation of tool envelope surface for 5-axis machining with single point contact[J]. Comput Aided Des 40(5):604–615
6. Rao A, Sarma R (2000) On local gouging in five-axis sculptured surface machining using flat-end tools. Comput Aided Des 32(7):409–420
7. Gray P, Bedi S, Ismail F (2003) Rolling ball method for 5-axis surface machining. Comput Aided Des 35(4):347–357
8. Lee YS, Chang TC (1996) Automatic cutter selection for 5-axis sculptured surface machining. Int J Prod Res 34(4):977–998
9. Jensen CG (2008) A three-dimensional configuration-space method for 5-axis tessellated surface machining. Int J Comput Integr Manuf 21(5):550–568
10. Choi BK, Kim DH, Jerard BJ (1997) C-space approach to tool-path generation for die and mould machining. Comput Aided Des 29(9): 657–669
11. Morishige K (1997) Collision-free tool path generation using 2-dimensional C-space for 5-axis control machining. Int J Adv Manuf Technol 13(6):393–400
12. Liu W, Zhang JW, Zhu SM, Zhang CC, Yuan TJ (2016) Efficient tool posture global collision-free area generation for 5-axis point clouds machining. International J Adv Manuf Technol:1–11
13. Morisige K, Kase K, Takeuchi Y (1999) Tool path generation using C-space for 5-axis control machining. J Manuf Sci Eng 121(1): 144–149
14. Jun CS, Cha K, Lee YS (2003) Optimizing tool orientations for 5-axis machining by configuration-space search method. Comput Aided Des 35(6):549–566
15. Affouard A, Duc E, Lartigue C, Langeron JM, Bourdet P (2004) Avoiding 5-axis singularities using tool path deformation[J]. Int J Mach Tools Manuf 44(4):415–425
16. Castagnetti C, Duc E, Ray P (2008) The domain of admissible orientation concept: a new method for five-axis tool path optimisation. Comput Aided Des 40(9):938–950
17. Sørby K (2007) Inverse kinematics of five-axis machines near singular configurations[J]. Int J Mach Tools Manuf 47(2):299–306
18. Lin Z, Fu J, Shen H, Gan W (2014) Non-singular tool path planning by translating tool orientations in c-space. Int J Adv Manuf Technol 71(9–12):1835–1848
19. Lin Z, Fu J, Yao X, Sun Y (2015) Improving machined surface textures in avoiding five-axis singularities considering tool orientation angle changes[J]. International Journal of Machine Tools & Manufacture 98:41–49
20. Fleisig RV, Spence AD (2001) A constant feed and reduced angular acceleration interpolation algorithm for multi-axis machining. Comput Aided Des 33(1):1–15
21. Ho MC, Hwang YR, Hu CH (2003) Five-axis tool orientation smoothing using quaternion interpolation algorithm. Int J Mach Tools Manuf 43(12):1259–1267
22. Morikawa M, Ishida T, Teramoto K, Takeuchi Y (2006) 5-axis control tool path generation using curved surface interpolation. JSME International Journal Series C 49(4):1209–1214
23. Langeron JM, Duc E, Lartigue C, Bourdet P (2004) A new format for 5-axis tool path computation, using Bspline curves. Comput Aided Des 36(12):1219–1229
24. Yuen A, Zhang K, Altintas Y (2013) Smooth trajectory generation for five-axis machine tools. Int J Mach Tools Manuf 71:11–19
25. Uddin MS, Ibaraki S, Matsubara A, Matsushita T (2009) Prediction and compensation of machining geometric errors of five-axis machining centers with kinematic errors[J]. Precis Eng 33(2):194–201
26. Ibaraki S, Sawada M, Matsubara A, Matsushita T (2010) Machining tests to identify kinematic errors on five-axis machine tools[J]. Precis Eng 34(3):387–398
27. Schwenke H, Knapp W, Haitjema H, Weckenmann A, Schmitt R, Delbressine F (2008) Geometric error measurement and compensation of machines—an update[J]. CIRP Annals-Manufacturing Technology 57(2):660–675
28. Wang N, Tang K (2007) Automatic generation of gouge-free and angular-velocity-compliant five-axis toolpath. Comput Aided Des 39(10):841–852

29. Boissonnat JD, Cazals F (2000) Smooth surface reconstruction via natural neighbour interpolation of distance functions. Proceedings of the sixteenth annual symposium on Computational geometry. ACM
30. Franke R (1982) Scattered data interpolation: tests of some methods. Math Comput 38(157):181–200
31. Piegl L, Tiller W (2012) The NURBS book. Springer Science & Business Media.
32. Yan CY, Lee CH, Yang JZ (2012) Three-axis tool-path B-spline fitting based on preprocessing, least square approximation and energy minimization and its quality evaluation. Mod Mach (MM) Sci J