



# Five-axis iso-error numerical control tool path generation for flat-end tool machining sculptured surface

Wei Liu<sup>1,2,3,4</sup> · Lv-Yang Fan<sup>1</sup> · Qi-Xin Zhu<sup>1</sup> · Shu-Mei Zhu<sup>1</sup> · Tian-Li Wang<sup>1</sup> · Feng Tang<sup>1</sup>

Received: 23 November 2020 / Accepted: 7 January 2022 / Published online: 26 January 2022  
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2022

## Abstract

The quality of numerical control (NC) tool path directly affects the machining precision and efficiency. As a basic parameter of tool path generation, step error represents the difference between tool path and machined surface, which is closely relevant to the distance (named step length) between adjacent cutter contacting (CC) points along feed direction. The step error along feed direction in machining process is actually the maximum error between tool envelope surface and CC curve. Ideally, step error is expected equal to the maximum allowable value so as to obtain theoretical maximum step length and least CC points. For this purpose, this paper presents a method of iso-error tool path generation for five-axis sculptured surface machining with flat-end tool. Tool envelope surface of flat-end tool is irregular and hard to represent exactly because of rotational axes. Discrete bottom circles of flat-end tool are used to replace tool envelope surface and calculate step error iteratively. A step error calculation algorithm of a tool location is presented to calculate the minimum distance between a tool bottom circle and CC curve, and used to obtain the point on CC curve with maximum non-linear error. Another step error calculation algorithm of a point on CC curve is presented to search the maximum step error between the two points corresponding to the maximums of linear and non-linear errors. The maximum step error is the real step error between two adjacent CC points. Finally, as step length, the parameter increment between adjacent CC points is adjusted to make step error in the defined range, and iso-error CC points are obtained to generate iso-error tool path. Two step error calculation algorithms are major innovations in this paper. A typical free-form surface and a blade are taken as examples to generate tool path. Compared with the other methods, the presented method can generate tool path with uniform step error and much less CC points.

**Keywords** Tool path generation · Iso-error · Step error · Sculptured surface · Five-axis

## 1 Introduction

With the rapid development of aerospace, automobile and mold manufacturing industry, sculptured surface is more and more used in configuration design for better performance and esthetic feature. Numerical control (NC) machining is a main processing method of sculptured surface, and tool

path plays an important role in quality and efficiency of NC machining [1].

Iso-parameter, iso-planar and iso-scallop method are main ways of tool path generation for sculptured surface. Iso-parametric method [2–4] sets a constant value as  $u$  or  $v$  parameter increment to plan CC curves, and tool path is generated based on CC curves. Iso-planar method [5–8] makes a series of parallel planes intersect with surface, and the intersecting lines are taken as CC curves. Iso-planar method can be applicable to almost all surfaces and widely used in commercial CAM software. Iso-scallop method [9–13] plans tool path line by line and ensure the scallop height between two adjacent lines be the allowed maximum. Iso-scallop tool path has shorter length than other methods but needs more computing time.

A group of CC curves can be planned by the above three methods. Tool path is usually generated based on discrete CC points on CC curves one by one. As shown in Fig. 1,

✉ Wei Liu  
liuwei@usts.edu.cn

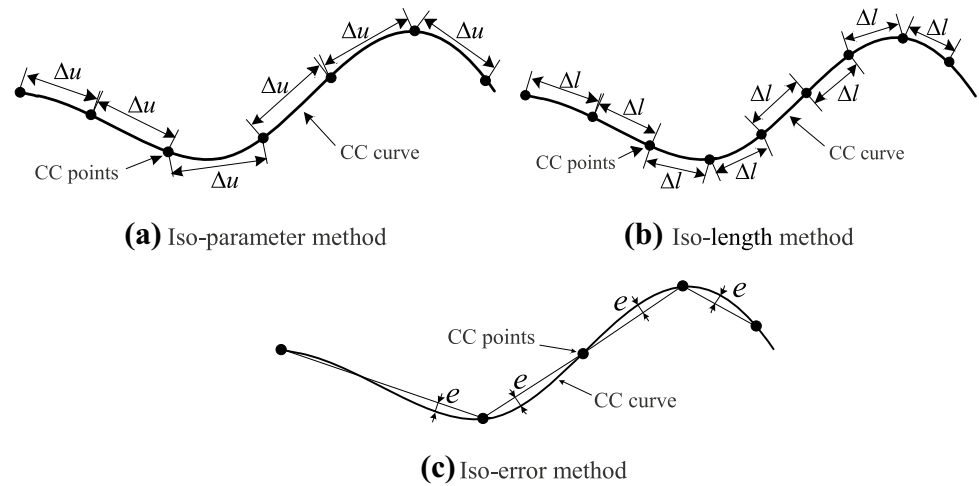
<sup>1</sup> College of Mechanical Engineering, Suzhou University of Science and Technology, Suzhou 215000, China

<sup>2</sup> Tianping College of Suzhou University of Science and Technology, Suzhou 215000, China

<sup>3</sup> School of Mechanical Science & Engineering, Huazhong University of Science and Technology, Wuhan 430000, China

<sup>4</sup> Wuxi Turbine Blade Limited Company, Wuxi 214000, China

**Fig. 1** Methods of CC point generation



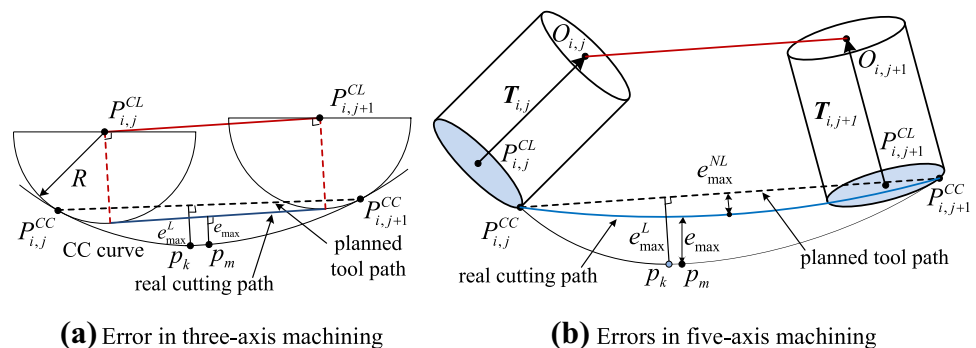
iso-parameter, iso-length and iso-error methods are main methods of CC point generation. As the name implies, iso-parameter method sets a constant value as parameter increment and CC points are calculated by incremental parameters in Fig. 1a. Similarly, iso-length method sets a constant arc or chord length as step length in Fig. 1b. Both methods have simple construction and efficient calculation.

As shown in Fig. 1c, iso-error method can generate CC points with uniformed error. The error between CC curve and line segments of CC points is usually named chord error or linear error. Sculptured surface generally has complex shape and uneven curvature. Constant parameter or length increment cannot generate uniformed chord errors, and chord errors between CC points may be greater than the allowed maximum. Line segments of CC points and chord error are not real cutting contours and machining error. When cutting tool moves from a CL point to the next one along the feed direction, tool envelope surface forms, which is the real cutting contour. The real machining error between two adjacent CC points is the maximum error between tool envelope surface and CC curve, which is named step error in this paper and affected by both step length and curve contour [14].

To make all step error not greater than the allowable maximum, the parameter and length increments in iso-parametric and iso-length methods need to be set conservative values. As a result, a lot of redundant CC points are generated, which is not beneficial to improve efficiency of tool path generation and NC machining. But greater step length may cause step error greater than the maximum. In order to minimize the number of CC points and meet the error requirement, iso-error method calculates suitable step length for every CC point to make step error equal or close to the allowed maximum.

Figure 2 shows errors between two adjacent CC points. All points (including CC, CL, points on CC curves), curves and errors are not in a plane because CC curves are three-dimensional and normal vectors of CC points are different. In three-axis machining, tool envelope surface of ball-end tool is a cylindrical surface along CL path, which is the real cutting contour or path. As shown in Fig. 2a, the step error  $e_{\max}$  is the maximum error between the cylindrical surface and CC curve, and is less than the chord error  $e_{\max}^L$  when the local CC curve between two adjacent CC points is concave. Conversely,  $e_{\max}$  is greater than  $e_{\max}^L$  when the local CC curve is convex.

**Fig. 2** Errors between two adjacent CC points



Compared with three-axis ball-end tool machining, because two rotating axes can make tool flexible enough, five-axis flat-end tool machining can contact surface more closely, which brings better machined surface quality and much larger cutting width.

The actual cutting contours of flat-end tool are also not the CC line segments but tool envelope surface with complicated and irregularly shape, and the error between them is called non-linear error. As shown in Fig. 2b, the step error  $e_{\max}$  is not the sum or difference between chord error  $e_{\max}^L$  and non-linear error  $e_{\max}^{NL}$ . Actually, the step error  $e_{\max}$  is the maximum error between tool envelope surface and CC curve. It is hard for flat-end tool to obtain accurate tool envelope surface and calculate the accurate value of step error directly, which is the main bottleneck in developing five-axis iso-error method.

Calculating accurate step error is the base of iso-error tool path generation. Many scholars have researched about these. Li et al. [14] presented a geometry-based error evaluation method and a tool path discretization method for five-axis sculptured surface machining. Lu et al. [15] estimated step length with the radius of normal curvature of the design surface at CC point along the feed direction. Du et al. [16] presented a third-order method that estimates the chord error using conical helices. Can and Ünüvar [17] used iterative algorithm in step length calculation to optimize the discretization of CC points. Xu et al. [18], redundant points are removed by maximizing each cutting step length using geodesic curvature matching, and the remaining fewer CC points can maintain the tolerance limits. Zhao et al. [19] presented a constant chord error method with variable step length. Golden section method is used to search the maximum chord error, which is more efficient and accurate than conventional mid-point method. Lin et al. [20] used an accurate chord evaluation method to calculate the next CC point based on the cutting simulation process. Dense grasses are planted on the tool path curve and they are cut when the tool moves. The length values of the grasses are considered as the machining error, and CC points with uniform chord error can be generated.

Fountas et al. [21] presented a methodology based on a virus-evolutionary genetic algorithm for multi-objective optimization including chord error, machining time, the number of cutter location points and so on. Zhang and Tang [22] used the finite element method (FEM) to minimize tool path length and machining time for ball-end tool. Ma et al. [23] constructed cutter location mesh units to optimize tool-path parameters for three-axis machining curved surface. The length values of mesh units are adjusted to meet error requirement by an iterative process, and the nodes of the units are taken as the cutter location points. For larger step length, Dong et al. [24] planed CC points laid on CC curve or the chord error offset curve when CC curve is concave or

concave. Segments connected by the adjacent CC points are between the two curves, and meet error requirement.

From the present research work, the error calculation is mainly based on curvature estimation algorithms [14–16] and iterative calculation algorithms [17–20, 23, 24]. Curvature estimation algorithms have an advantage in efficiency but a disadvantage on accuracy. Iterative calculation algorithms cost more calculation and obtain more accurate results than curvature estimation algorithms.

Most of the previous work focused on error calculation including chord and step error, which is the premise of generating iso-error tool path. However, how to calculate real step error efficiently and adjust step length to make step error meet the iso-error requirement further is the most significant bottleneck. Both the difficulties make main commercial CAM software hardly generate five-axis iso-error NC tool path methods, and few methods have been reported and applied.

The purpose of this study is to develop an iso-error tool path generation method for flat-end tool. This paper mainly focuses on how to calculate accurate step error efficiently and obtain maximum step length meeting iso-error requirement so as to reduce redundant CL points for NC machining. In Sect. 2.1, initial local gouging-free CL points and tool vectors are calculated. In order to calculate more accurate step error, step error is expressed as the maximum distance between the actual cutting path and CC curve, and calculated based on the geometrical relationship among tool location, CC curve and actual cutting path. Two step error calculation algorithms for specified tool location and CC point are presented in Sect. 2.2 respectively and used in iso-error CC point calculation in Sect. 2.3. Two examples are implemented in Sect. 3 to verify the feasibility of the algorithms. Conclusions are summarized in Sect. 4.

## 2 Iso-error tool path generation

This section mainly presents the process of calculating accurate step error and generating iso-error tool path. The interval size is simply planned by iso-parametric method. Initial CL points and tool vectors are calculated firstly in Sect. 2.1. In Sect. 2.2, accurate step error of every two adjacent tool locations is computed based on the geometric relationship among tool locations (including CL points and tool vectors), CC points and step error. Finally, in Sect. 2.3, step length is adjusted by changing CC parameters iteratively so as to compute iso-error CC point and iso-error tool path. This section takes the  $i$  th line CC curve  $CC_i$  and the  $j$  th,  $j + 1$  th CC points  $P_{i,j}^{CC}$  and  $P_{i,j+1}^{CC}$  as examples to describe all algorithms. The main calculation process is shown in Fig. 3. In the process, all CC curves are 3D parametric curves on the

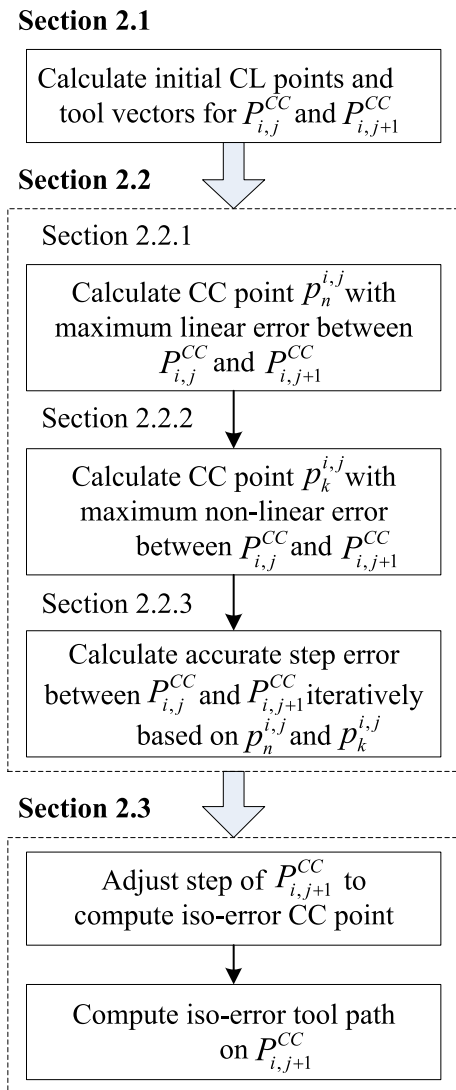


Fig. 3 Process of iso-error tool path generation

surface  $S$ . All the figures including CC and CL points just show basic geometrical relationship, and adjacent CC points and CL points are not on a plane.

### 2.1 Calculate initial CL points and tool vectors for CC curves

This section presents a calculation process of initial CL points and tool vectors for CC curve. Local short curves near CC points are approximately seen as arcs, and initial values of step length are calculated by chord error to establish discrete CC points. Then initial CL points and tool vectors are calculated by the inclination angle  $\alpha$ . If there is concave region in the machining surface,  $\alpha$  should be great enough so as to avoid local gouging [25]. Taking

the  $i$  th line CC curve  $CC_i$  as an example, the calculation process is as following.

**Step 1 obtain CC curve** Let  $u$  and  $v$  be the interval and feed directions of machined parametric surface  $S$ , respectively. The  $i$  th line CC curve  $CC_i$  can be obtained by Eq. (1).  $n$  is the total number of lines of tool path.  $u_{\min}$  and  $u_{\max}$  are minimum and maximum of  $u$  parameter, respectively.

$$\begin{cases} CC_i = S(u_i) \\ u_i = \frac{n-i}{n-1}u_{\min} + \frac{i-1}{n-1}u_{\max} \end{cases} \quad (1)$$

**Step 2 calculate discrete CC points on CC curve** Let the  $v$  range of  $CC_i$  be  $[v_{\min}, v_{\max}]$ . The first CC point has the parameter  $v_1 = v_{\min}$ . For the  $j$  th CC point  $P_{ij}^{CC}$ , the coordinate  $(x_{ij}^{CC}, y_{ij}^{CC}, z_{ij}^{CC})$ , curve length  $L_j$ , curvature  $k_j$ , tangent vector  $t_j$  and normal vector  $n_j$  can be calculated and expressed as Eq. (2).  $S_u$  and  $S_v$  are the tangent vectors of an arbitrary point on the surface along the directions of parameter  $u$  and  $v$ .

$$\begin{cases} P_{ij}^{CC}(x_{ij}^{CC}, y_{ij}^{CC}, z_{ij}^{CC}) = S(u_i, v_j) = CC_i(v_j) \\ L_j = L(v_j) \\ k_j = k(v_j) \\ t_j = t(v_j) \\ n = \frac{S_u \times S_v}{|S_u \times S_v|}, S_u = \frac{\partial S(u,v)}{\partial u}, S_v = \frac{\partial S(u,v)}{\partial v} \end{cases} \quad (2)$$

For the next CC point  $P_{ij+1}^{CC}$ , when the actual step error is equal to the allowed maximum  $e_{\max}$ , the step length is also the theoretical maximum. Considering the value of  $e_{\max}$  in finish machining is small and the curvature of surface changes smoothly, the local curve  $CC_{ij}$  between  $P_{ij}^{CC}$  and  $P_{ij+1}^{CC}$  is short in length and supposed iso-curvature in this section. As shown in Fig. 4, the chord error (linear error) between the segment line  $P_{ij}^{CC}P_{ij+1}^{CC}$  and the arc  $P_{ij}^{CC}P_{ij+1}^{CC}$  is set  $e_{\max}$ .  $P_{ij+1}^{CC}$  is the actual CC point on the curve  $CC_i$ . Based on the relationship,  $v_{j+1}$  of  $P_{ij+1}^{CC}$  can be

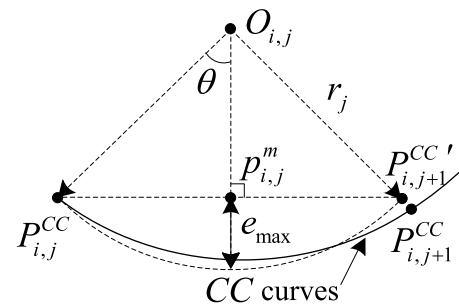


Fig. 4 Chord error calculation

calculated by Eq. (3) and the other data of  $P_{ij+1}^{CC}$  can be obtained by Eq. (2).

$$\begin{cases} v_{j+1} = L^{-1}(P_{ij}^{CC} P_{ij+1}^{CC'} + L(v_j)) \\ P_{ij}^{CC} P_{ij+1}^{CC'} = \frac{2\theta}{|k(v_j)|} \\ \theta = \arccos(1 - |k(v_j)| e_{\max}) \end{cases} \quad (3)$$

**Step 3 calculate initial CL points and tool vectors for CC points** Taking  $P_{ij}^{CC}$  as an example, the tangent vector  $t_j$  and normal vector  $n_j$  can be obtained by Eq. (2). A local coordinate centered at  $P_{ij}^{CC}$  can be established by  $y_j = t_j$ ,  $z_j = n_j$  and  $x_j = t_j \times n_j$ . As shown in Fig. 5, CL point and tool vector are established by the inclination angle  $\alpha$  and can be obtained by Eq. (4).

$$\begin{cases} T = n \cos \alpha + t \sin \alpha \\ P^{CL} = P^{CC} + R(t \times n) \times T \end{cases} \quad (4)$$

All CC points can be obtained by the above process. In Step 2, if  $P_{ij}^{CC}$  is an inflection point or near an inflection point, the curvature will be very small and  $P_{ij+1}^{CC}$  will have an abnormally large step length, which causes the actual step error greater than  $e_{\max}$ . In order to avoid this situation, every step error will be calculated accurately in Sect. 2.2 and step length will be adjusted to generate ios-error CC points and tool path in Sect. 2.3.

### 2.2 Compute step error of two adjacent tool locations

As shown in Fig. 6, when the flat-end tool moves along a line of tool path with changed rotation angles, the cutting circle on the tool bottom sweeps and builds non-uniform envelope

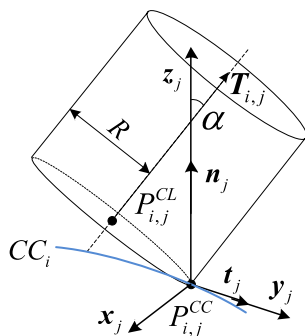


Fig. 5 CL point and tool vector

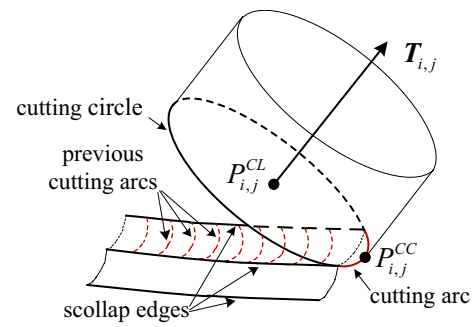


Fig. 6 Flat-end tool machining

surface on machined surface. Part of the cutting circle (cutting arc in Fig. 6) is the curve on the machined surface.

As shown in Fig. 7, the local curve  $CC_{ij}$  between  $P_{ij}^{CC}$  and  $P_{ij+1}^{CC}$  usually has smooth figure on sculptured surface. In this section, the maximum distance between  $CC_{ij}$  and the envelope surface is the wanted step error between two adjacent locations. In order to simply the calculation, a set  $\{p_k^{ij}\}$  of discrete points on  $CC_{ij}$  and the cutting circles on the tool bottom are obtained to replace  $CC_{ij}$  and the envelope surface in calculation.

Supposed  $p_m^{ij}$  on  $CC_{ij}$  has the maximum  $e_{\max}^{ij}$  of step error, the value of step error increases monotonically from  $P_{ij}^{CC}$  to  $p_m^{ij}$  and decreases monotonically from  $p_m^{ij}$  to  $P_{ij+1}^{CC}$ . Supposed  $p_n^{ij}$  and  $p_k^{ij}$  have the maximums of linear error and non-linear error, the wanted  $p_m^{ij}$  is located near  $p_n^{ij}$  and  $p_k^{ij}$ . This section firstly obtains  $\{p_k^{ij}\}$  on the curve  $CC_{ij}$  and calculates the point  $p_n^{ij}$  with the maximum  $e_{\max}^L$  of linear error in Sect. 2.2.1. Then a step error calculation method of specified tool location is presented in Sect. 2.2.2 to obtain the point  $p_k^{ij}$  with the maximum  $e_{\max}^{NL}$  of non-linear error. A step error calculation method for specified CC point is presented to calculate the step error  $e_n^{ij}$  of  $p_n^{ij}$  in Sect. 2.2.3. Finally, the maximum step error can be searched near  $p_n^{ij}$  and  $p_k^{ij}$ , which is the wanted step error between  $P_{ij}^{CC}$  and  $P_{ij+1}^{CC}$ .

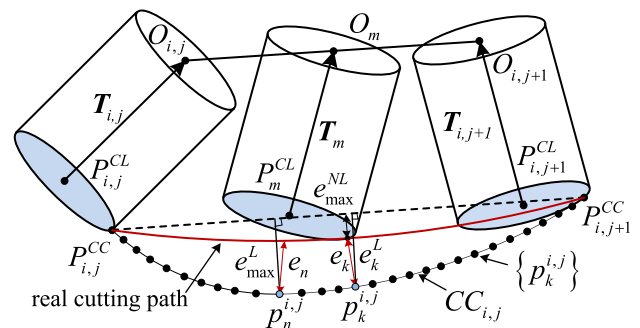


Fig. 7 Errors between two adjacent locations



### 2.2.1 Calculate CC points with maximum linear error on CC curve

The CC curve  $CC_{ij}$  between  $P_{ij}^{CC}$  and  $P_{ij+1}^{CC}$  can be obtained by Eq. (5).  $v_j, v_{j+1}$  are  $v$  parameters of  $P_{ij}^{CC}$  and  $P_{ij+1}^{CC}$ , respectively. Discrete points  $\{p_k^{ij}\}$  on the local curve  $CC_{ij}$  are calculated by Eq. (6) with the increment  $\Delta v$  to replace  $CC_{ij}$  in the calculation process, which can improve the calculation efficiency. The distance  $e_k^L$  between  $p_k^{ij}$  and the segment  $P_{ij}^{CC}P_{ij+1}^{CC}$  can be calculated by Eq. (7). Let  $p_n^{ij}$  in  $\{p_k^{ij}\}$  have the maximum  $e_{max}^L$ . As the chord error between the segment  $P_{ij}^{CC}P_{ij+1}^{CC}$  and  $CC_{ij}$ ,  $e_{max}^L$  is the wanted maximum of linear error.

$$CC_{i,j} = CC_i(v), v \in [v_j, v_{j+1}] \tag{5}$$

$$p_k^{ij} = CC_i(v_j + k \cdot \Delta v), k = 1, 2, 3... \tag{6}$$

$$e_k^L = \frac{\overline{P_{ij}^{CC}P_k^{ij}} \times \overline{P_k^{ij}P_{ij+1}^{CC}}}{\left| \overline{P_{ij}^{CC}P_{ij+1}^{CC}} \right|} \tag{7}$$

### 2.2.2 Calculate step error of a tool location between adjacent CL points

As shown in Fig. 7,  $O_{ij}$  and  $O_{ij+1}$  are top points of tool and seen as the location point of tool holder.  $O_m$  is the midpoint of  $O_{ij}O_{ij+1}$ . On the bottom circle of the tool location on  $O_m$ , there is a point which has a minimum distance to the curve  $CC_{ij}$ , and the minimum distance is the step error of tool location  $O_m$ . A geometric algorithm of step error calculation is presented for the distances between the point set  $\{p_k^{ij}\}$  and the bottom circle of tool. Taking the middle position  $O_m$  of  $O_{ij}$  and  $O_{ij+1}$  as an example, the detailed process of step error calculation is as following.

**Step 1 calculate tool data of middle position of two adjacent tool locations** As shown in Fig. 7, when tool moves from  $P_{ij}^{CL}$  to  $P_{ij+1}^{CL}$ , the tool holder moves in the straight line  $O_{ij}O_{ij+1}$ . The tool location on the midpoint  $O_m$  of  $O_{ij}O_{ij+1}$  has the maximum of non-linear error. The tool data of  $O_m$  can be calculated by Eq. (8), where  $L_T$  is the tool length.

$$\begin{cases} P_m^{CL} = O_m - L_T T_m \\ T_m = (T_{ij} + T_{ij+1})/2 \\ O_m = (O_{ij} + O_{ij+1})/2 \\ O_{ij} = P_{ij}^{CL} + L_T T_{ij} \\ O_{ij+1} = P_{ij+1}^{CL} + L_T T_{ij+1} \end{cases} \tag{8}$$

**Step 2 calculate initial step error for tool location  $O_m$ .** As shown in Fig. 8,  $B_k$  on the plane of tool bottom is the foot point of  $p_k^{ij}$ .  $C_k$  is the intersection point of  $P_m^{CL}B_k$  and  $\odot P_m^{CL}$ . The triangles  $\Delta p_k^{ij}B_kC_k$  and  $\Delta p_k^{ij}B_kP_m^{CL}$  are constructed, and the distance  $p_k^{ij}C_k$  can be calculated with  $p_k^{ij}B_k$  and  $B_kC_k$  by Eq. (9). In  $\Delta p_k^{ij}B_kC_k$ ,  $C_kB_k$  is the minimum distance between  $\odot P_m^{CL}$  and  $B_k$ , and  $p_k^{ij}B_k$  is the minimum distance between  $p_k^{ij}$  and the tool bottom plane. So,  $p_k^{ij}C_k$  is the minimum distance between  $p_k^{ij}$  and  $\odot P_m^{CL}$ . The maximum of the minimum distances of all points in  $\{p_k^{ij}\}$  is the initial step error of the tool location on  $O_m$ .

$$\begin{cases} p_k^{ij}C_k = \sqrt{p_k^{ij}B_k^2 + B_kC_k^2} \\ B_kC_k = |R - P_m^{CL}B_k| \\ P_m^{CL}B_k = \sqrt{P_m^{CL}p_k^{ij2} - p_k^{ij}B_k^2} \\ p_k^{ij}B_k = \frac{P_m^{CL}P_k^{ij} \cdot T_m}{|T_m|} \end{cases} \tag{9}$$

**Step 3 judge if the step error meets the precision requirement** Supposed that the distance  $d_k^{ij}$  of  $p_k^{ij}$  is the calculated step error in Step 2, the minimum difference  $\Delta d_{min}^k$  between  $d_{k-1}^{ij}$ ,  $d_{k+1}^{ij}$  and  $d_k^{ij}$  can be obtained by Eq. (10). If  $\Delta d_{min}^k$  is greater than the user-defined precision  $\Delta e$  of step error,  $d_k^{ij}$  cannot meet the precision requirement. The process goes to Step 4. If  $\Delta d_{min}^k$  is less than  $\Delta e$ ,  $d_k^{ij}$  can meet the precision requirement and is the wanted step error. The process ends. The flow chart of step error calculation is shown in Fig. 9.

$$\Delta d_{min}^k = \min(d_{k-1}^{ij} - d_k^{ij}, d_{k+1}^{ij} - d_k^{ij}) \tag{10}$$

**Step 4 obtain new CC points and calculate step error** Two midpoints  $p_{k-1}^m$  and  $p_k^m$  of  $p_{k-1}^{ij}p_k^{ij}$  and  $p_k^{ij}p_{k+1}^{ij}$  can be obtained by Eq. (11). The distances  $d_{k-1}^m, d_k^m$  from  $p_{k-1}^m, p_k^m$  to  $\odot P_m^{CL}$  can be calculated by Eq. (9). The maximum distance  $d_{max}^k$  is obtained by Eq. (12). Plug the distances into Step 3 and judges if  $d_{max}^k$  meets the precision requirement.

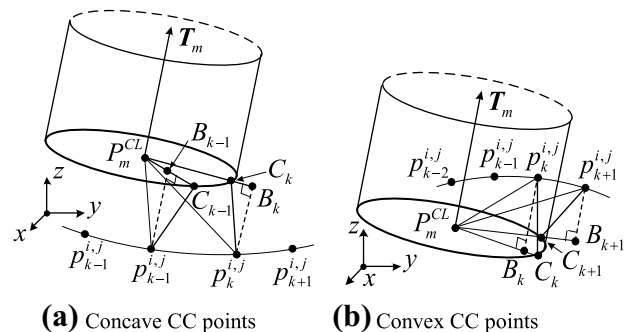


Fig. 8 Geometrical relationship between CC points and tool bottom circle

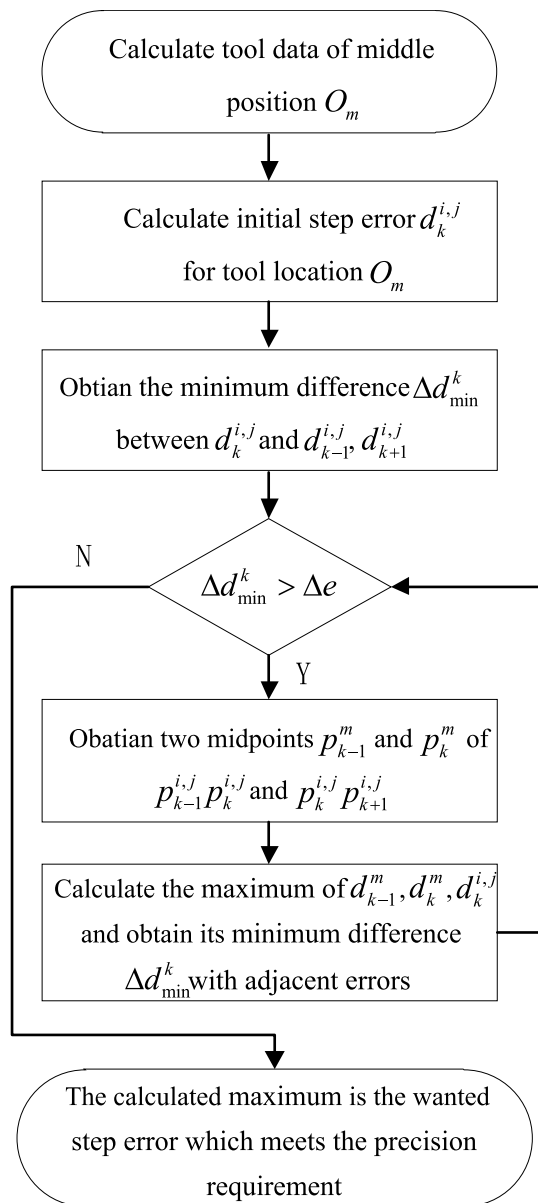


Fig. 9 Flow chart of calculating step error of tool location  $O_m$

$$\begin{cases} p_{k-1}^m = CC_i(v_{k-1}^m) \\ v_{k-1}^m = (v_{k-1}^{i,j} + v_k^{i,j})/2 \end{cases} \quad (11)$$

$$d_{\max}^k = \max(d_{k-1}^m, d_k^{i,j}, d_k^m) \quad (12)$$

2.2.3 Calculate step error of a point on CC curve

$p_n^{i,j}$  and  $p_k^{i,j}$  have the maximums of linear and non-linear errors, which can be calculated in Sects. 2.2.1 and 2.2.2. The wanted  $p_m^{i,j}$  with the maximum of step error is located near  $p_n^{i,j}$  and  $p_k^{i,j}$ . This section firstly calculates step error of  $p_n^{i,j}$ , and then search  $p_m^{i,j}$  by calculating step error with iterative parameter. The detailed process is as following.

**Step 1 Calculate tool location point based on CC point** As shown in Fig. 10,  $p_n^B$  and  $p_k^B$  on the segment  $P_{ij}^{CC} P_{i,j+1}^{CC}$  are foot points of  $p_n^{i,j}$  and  $p_k^{i,j}$ , respectively. The location coefficient  $k_n$  of  $p_n^B$  on  $P_{ij}^{CC} P_k^B$  is obtained by Eq. (13). Supposed that  $O_n$  has the same location coefficient  $k_n$  on  $O_{i,j} O_k$ ,  $O_n$  can be obtained by Eq. (14).

$$k_n = \frac{\overline{P_{ij}^{CC} P_n^B}}{\overline{P_{ij}^{CC} P_k^B}} \quad (13)$$

$$\overline{O_{i,j} O_n} = k_n \cdot \overline{O_{i,j} O_k} \quad (14)$$

**Step 2 calculate the initial step error of tool location** The CL point  $P_n^{CL}$  and tool vector  $T_n^{i,j}$  of  $O_n$  can be calculated by Eq. (15). The step error of tool location  $O_n$  can be calculated by the algorithm in Sect. 2.2.2 and is taken as the initial value.

$$\begin{cases} P_n^{CL} = O_n - LT_n^{i,j} \\ T_n^{i,j} = (1 - k_n)T_{i,j} + k_n T_k^{i,j} \end{cases} \quad (15)$$

**Step 3 calculate more accurate step error** Let  $k_{n+\Delta k} = k_n + \Delta k$  and  $k_{n-\Delta k} = k_n - \Delta k$ .  $\Delta k$  is the increment of location coefficient, which can be usually set from 0.05 to 0.1. Greater  $\Delta k$  makes calculation more efficient but lower precision. Same with  $k_n$ , tool locations of  $k_{n+\Delta k}$ ,  $k_{n-\Delta k}$  and their step error  $e_{n+\Delta k}$ ,  $e_{n-\Delta k}$  can be calculated by Step 1 and Step 2. If  $e_{n+\Delta k} > e_n$ , let  $k_{n+\Delta k}$  plus  $\Delta k$  to search greater step error. If  $e_{n-\Delta k} > e_n$ , let  $k_{n-\Delta k}$  minus  $\Delta k$  to search greater step error. If  $e_n > e_{n+\Delta k}$  and  $e_n > e_{n-\Delta k}$ ,  $e_n$  is the wanted more accurate step error.

2.3 Compute iso-error CC points and tool path

Section 2.2 presents the step error computation method of  $P_{ij}^{CL}$  and  $P_{i,j+1}^{CL}$ . Rename the step error of  $P_{ij}^{CL}$  and  $P_{i,j+1}^{CL}$  with  $e_{i,j}$ .  $e_{i,j}$  can hardly be exactly equal to the maximum allowable value  $e_{\max}$ . In this section, a range  $[e_{\max}^{low}, e_{\max}]$  is used to judge if  $e_{i,j}$  meets error requirement.  $e_{\max}^{low}$  is user-defined

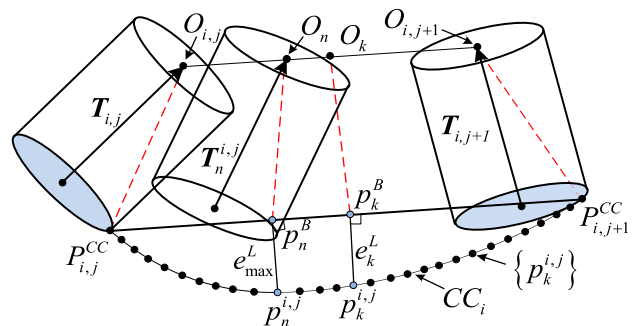
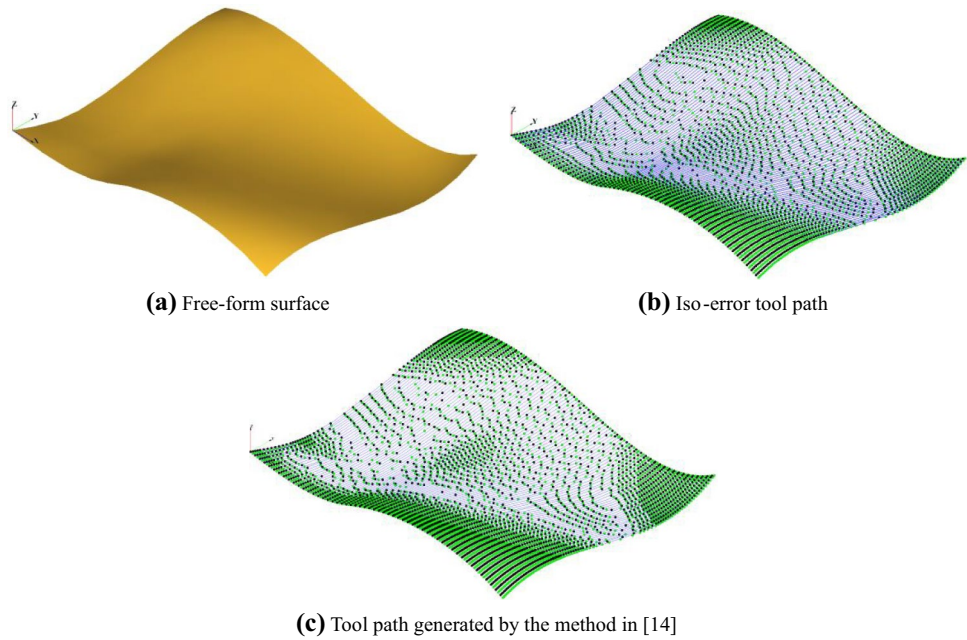


Fig. 10 Tool location on  $O_n$

**Fig. 11** Free-form surface and tool path



lower limit of  $e_{\max}$ , which is close to  $e_{\max}$  such as  $0.9 e_{\max}$ . If  $e_{i,j} \in [e_{\max}^{low}, e_{\max}]$ ,  $e_{i,j}$  will be seen as meeting error requirement and  $P_{i,j+1}^{CL}$  is the wanted iso-error CL point. Otherwise, the parameter of  $P_{i,j+1}^{CC}$  will be adjusted to make  $e_{i,j} \in [e_{\max}^{low}, e_{\max}]$ . The detailed process is as following.

**Step 1** If  $e_{i,j} \notin [e_{\max}^{low}, e_{\max}]$ , the step length needs to be adjusted and the process goes to Step 2. If  $e_{i,j} \in [e_{\max}^{low}, e_{\max}]$ , the process goes to Step 3.

**Step 2** Supposed step error and step length are linear relationship, the parameter increment  $\Delta v_{i,j+1}$  of CC point is obtained by Eq. (16). An increment coefficient  $k_e (k_e < 1)$  is added in Eq. (16) to adjust parameter increment, which can make the iterative process convergent. The adjusted CC point  $P_{i,j+1}^{CC} = CC_{i,j}(v_{i,j+1} + \Delta v_{i,j+1})$  is obtained to calculate new CL point  $P_{i,j+1}^{CL}$  and tool vector  $T_{i,j}$  with Eq. (4). And new step error  $e_{i,j}$  can be calculated by the methods in Sect. 2.2. The process goes to Step 1.

$$\Delta v_{i,j+1} = k_e \frac{e_{\max} - e_{i,j}}{e_{\max}} v_{i,j+1} \tag{16}$$

**Step 3** The step error  $e_{i,j}$  between  $P_{i,j+1}^{CL}$  and  $P_{i,j+1}^{CL}$  meets the requirement.  $P_{i,j+1}^{CL}$  and  $T_{i,j}$  are the wanted iso-error CL point and tool vector.

Same with  $P_{i,j+1}^{CL}$ , the subsequent CL points and tool vectors can also be calculated one by one. And all iso-error CL points and tool vectors of all CC curves can be obtained by the above algorithms.

### 3 Examples

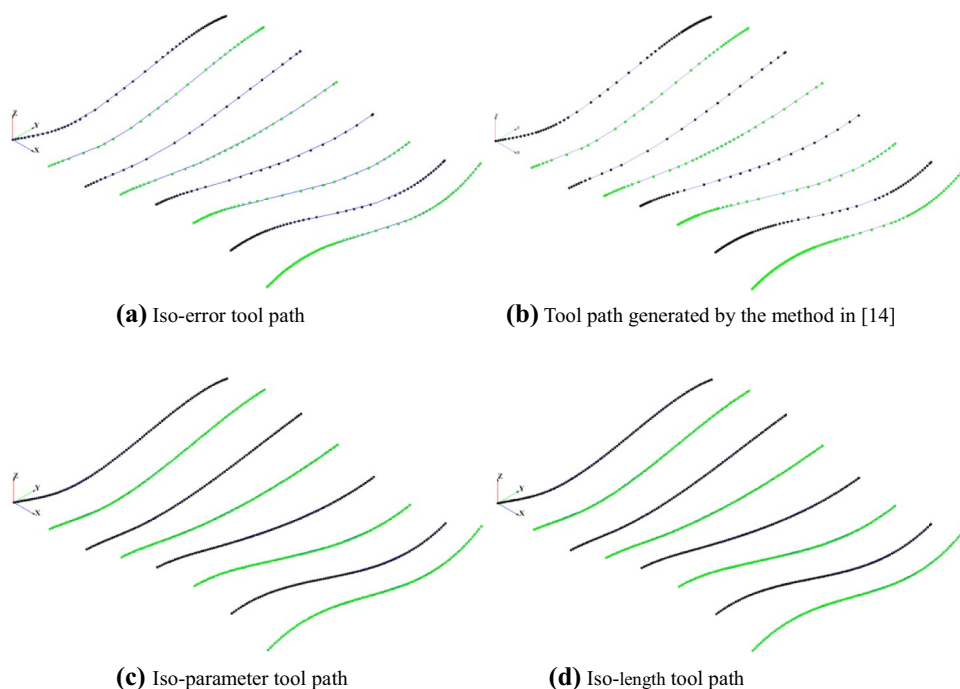
All the algorithms in Sect. 2 have been implemented in our tool path generation software based on Visual Studio 2017 and Opencascade.7.4.0. As shown in Figs. 11a and 15a, a typical free-form surface and a blade are selected for tool path generation to verify the feasibility and efficiency of the iso-error method presented in this paper, which contain concave, convex and large curvature regions. The information and machining parameters of two models are shown in Table 1. Besides, the increment of location coefficient  $\Delta k$  is 0.1, which is used to calculate step error for specified CC point. The increment

**Table 1** Information and machining parameters of two models

Model	Bounding box(mm)	Flat-end tool information			Step error		Number of lines of tool path
		Radius R(mm)	Length L(mm)	Inclination angle $\alpha$	Lower limit $e_{\max}^{low}$ (mm)	Upper limit $e_{\max}$ (mm)	
Free-form surface	120*100*31	5	100	20°	0.004	0.005	106
Blade	88*106*25	3	50	15°	0.0027	0.003	221



**Fig. 12** Eight lines of the free-form surface’s all tool path



coefficient  $k_e$  is 0.8, which is used to adjust parameter increment when searching iso-error CC points.

The free-form surface’s iso-error tool path generated by the iso-error method presented in this paper is shown in Fig. 11b. Another similar tool path generation method including geometry-based error evaluation method in [14], iso-parameter and iso-length methods are employed to generated tool path for comparison. Figure 11c shows the tool path generated by the method in [14]. The parameter and length increments in iso-parameter and iso-length methods are 0.01 and 1 mm, respectively.

Considering all lines of tool path are too dense to show clearly, eight lines of all tool path are evenly selected to show in Fig. 12 to compare the distribution of CC points on CC curves. The CC points on iso-error tool path and the tool path generated by the method in [14] are non-uniformly located on CC curves. The numbers of CC points of all tool path are shown in Table 2. The iso-error tool path has least CC points. Figure 13 shows the step errors of eight lines of all tool path. All the step errors of the iso-error tool path are in the defined range [0.004 mm, 0.005 mm]. Only the last step errors in every line are lower than 0.004 mm because the last CC points are the end point of CC curves. All the step errors of the tool path generated by the method in [14] are less than 0.005 mm. However, a few of them

are less than 0.004 mm. As shown in Figs. 13c, d, most step errors of iso-parameter tool path and iso-length tool path are out of the range [0.004 mm, 0.005 mm]. In practical application, smaller parameter and length increments should be defined to make step errors less than the upper limit, which will generate much more CC points than iso-error tool path. Figure 14 shows the parameter increments of the tool path generated by the iso-error method presented in this paper and the method in [14]. The parameter increments of iso-error tool path are generally greater, which means greater step length and makes the step errors are closer to the upper limit.

In another example, the blade is also performed by the same comparison with the free-form surface. Eight lines of four kinds of tool path is shown in Figs. 15b–e. The parameter and length increments are 0.113 and 0.1 mm, respectively. The numbers of CC points are shown in Table 3. The  $v$  parameter ranges [25, 34] and [58, 68] are lead and trail regions with much greater curvature. As shown in Fig. 16, the tool path generated by the methods in this paper and the paper [14] has no step errors greater than the upper limit, and Fig. 17 shows the parameter increments. Many step errors of iso-parameter tool path and iso-length tool path are out of the defined error range. A few step errors in the tool path generated by the method in [14] are less than the lower

**Table 2** Numbers of CC points on the free-form surface’s tool path generated by four methods

Method	Line number								All 106 lines	Meeting error
	1	16	31	46	61	76	91	106		
Iso-error in this paper	56	34	24	33	30	40	59	80	4,358	Yes
Method in [14]	70	37	25	39	35	48	73	97	5,155	Yes
Iso-parameter	100	100	100	100	100	100	100	100	10,600	No
Iso-length	103	103	102	102	103	104	104	104	10,924	No

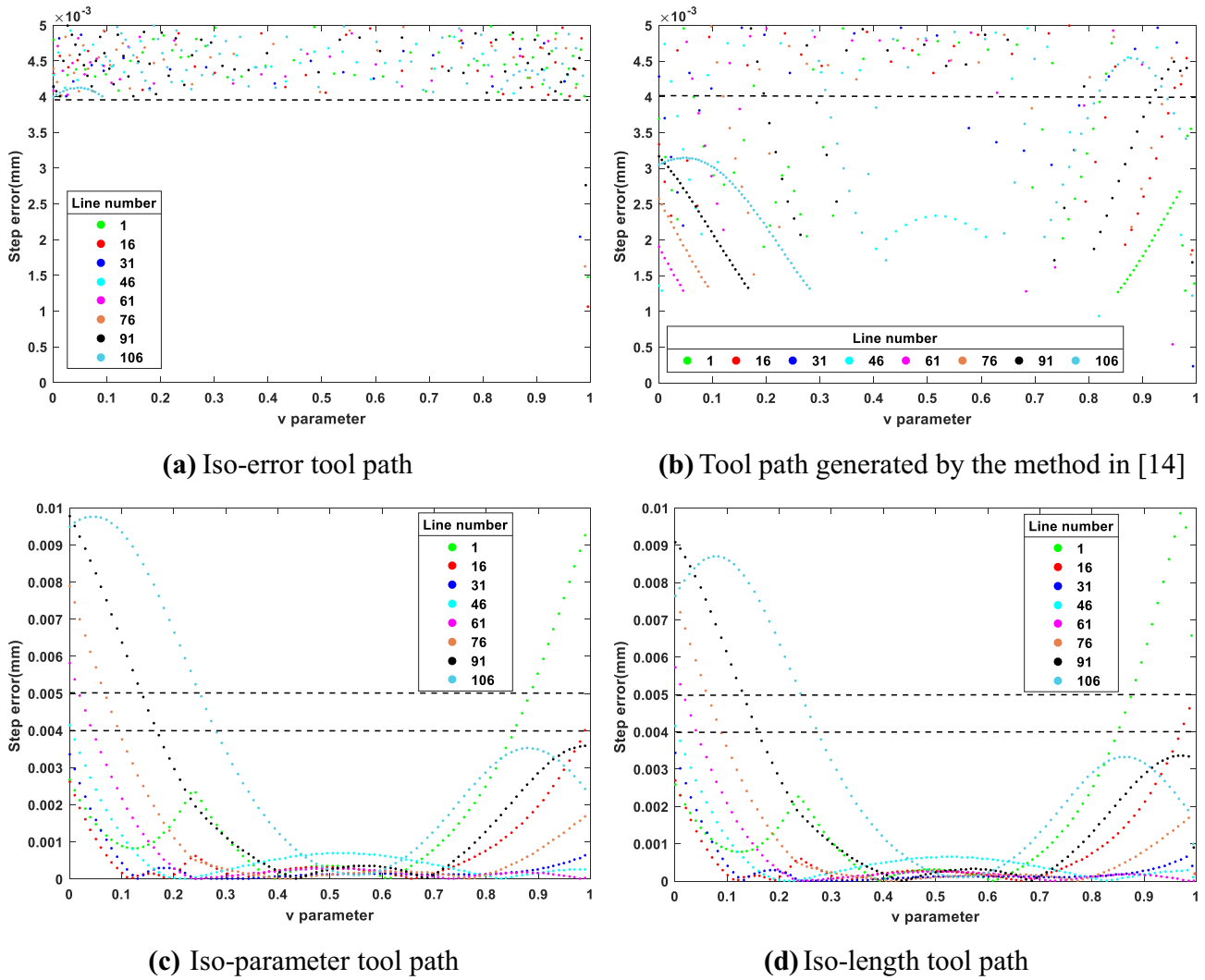


Fig. 13 Step error of eight lines of the free-form surface’s all tool path

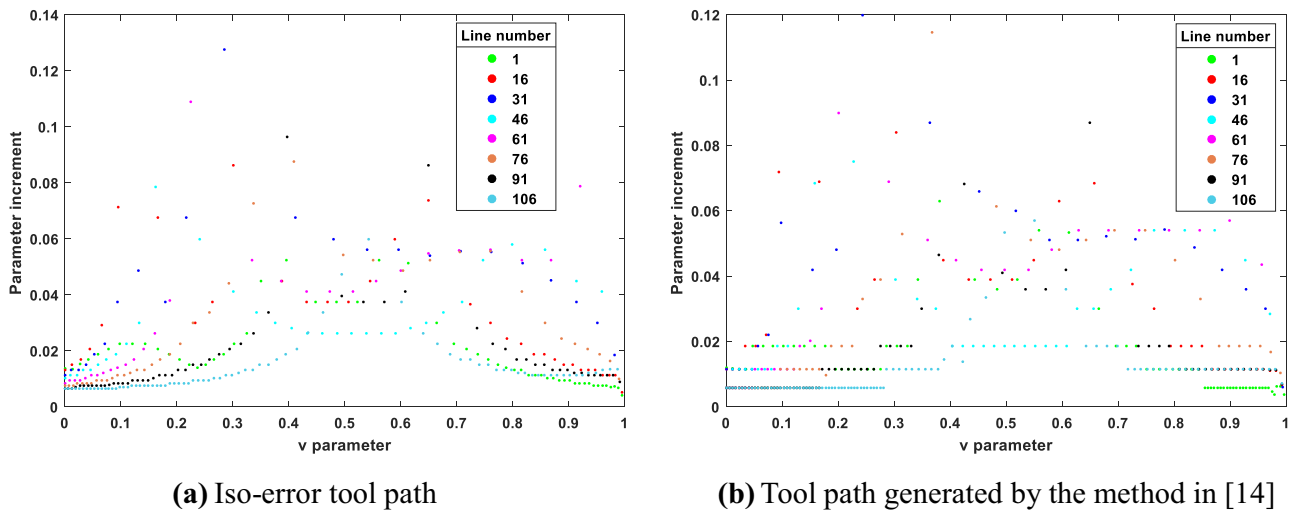


Fig. 14 Parameter increments of the free-form surface’s two kinds of tool path

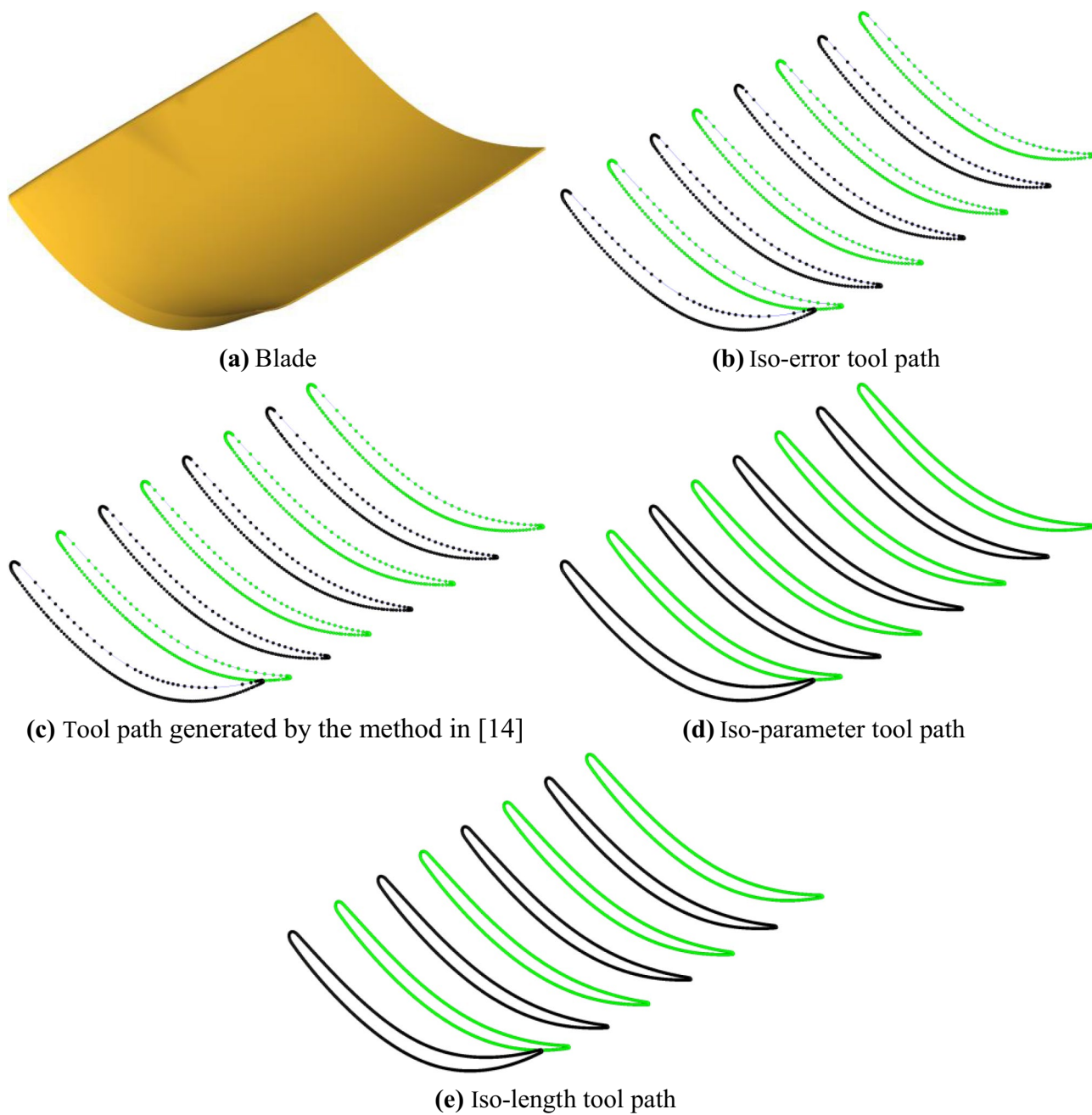
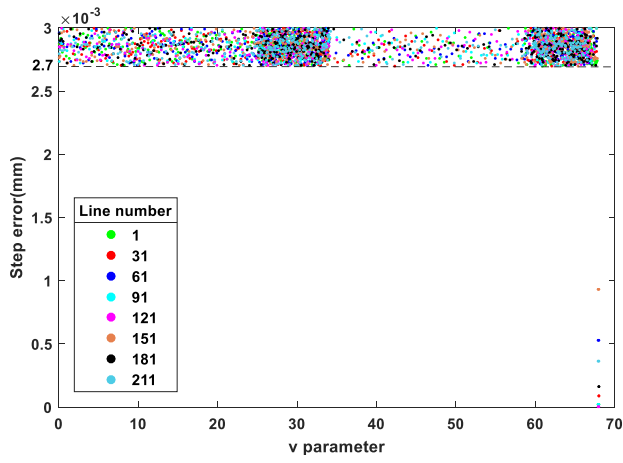


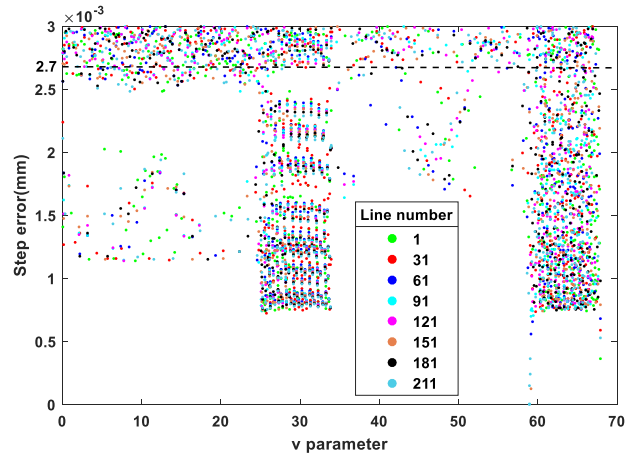
Fig. 15 Blade and eight lines of tool path generated by four methods

Table 3 Numbers of CC points on the blade’s tool path generated by four methods

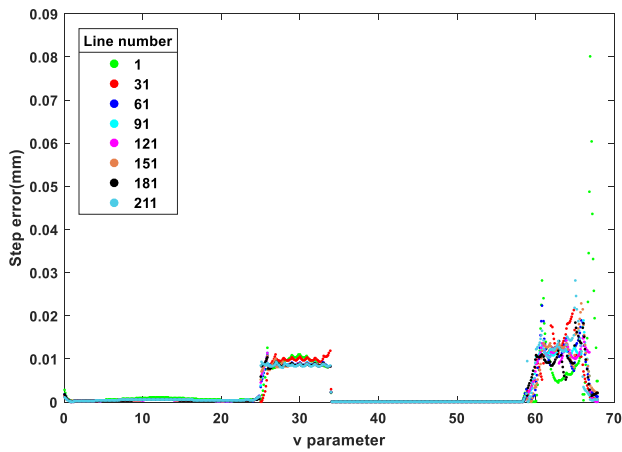
Method	Line number								All 221 lines	Meeting error
	1	31	61	91	121	151	181	211		
Iso-error in this paper	414	396	396	395	396	399	398	395	87,784	Yes
Method in [14]	521	502	494	501	498	502	501	511	110,947	Yes
Iso-parameter	600	600	600	600	600	600	600	600	132,600	No
Iso-length	654	585	573	568	568	571	575	592	128,200	No



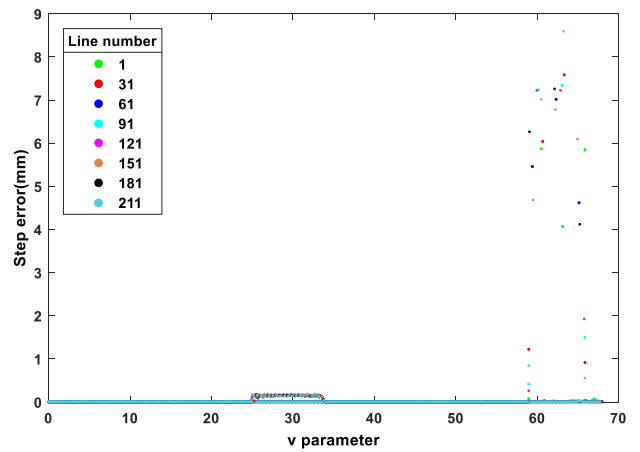
(a) Iso-error tool path



(b) Tool path generated by the method in [14]

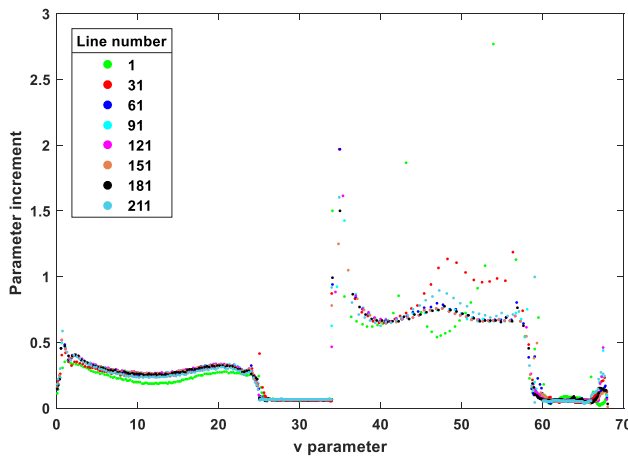


(c) Iso-parameter tool path

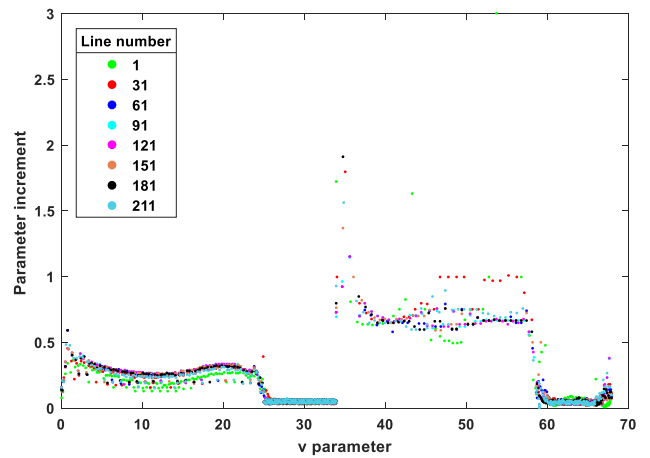


(d) Iso-length tool path blade's

Fig. 16 Step error of eight lines of the blade's all tool path



(a) Iso-error tool path



(b) Tool path generated by the method in [14]

Fig. 17 Parameter increments of the blade's two kinds of tool path



**Fig. 18** Machined blade

limit, especially in the lead and trail regions. Figure 18 shows the machined blade with the iso-error tool path.

Both examples have similar results. The tool path generated by iso-parameter and iso-length methods have many step errors greater than the defined upper limit, which cannot meet the error requirement and needs smaller parameter and length increments. Step errors of the tool path generated by the method in [14] are not greater than the defined upper limit. The method in this paper can generate iso-error tool path which has least CC points and all step errors in the defined range.

## 4 Conclusion

Calculating CL points and tool vectors based on discrete CC points is a main tool path generation method for sculptured surface. In order to void step error greater than the maximum allowance, step length is usually set conservative, which generate a lot of redundant CL points. For maximizing step length and minimizing redundant CC and CL points, this paper presents a five-axis iso-error tool path generation method for flat-end tool machining sculptured surface.

Because step error between adjacent tool locations is hard to calculate directly, two step error calculation algorithms of tool locations and CC points are presented respectively. For a tool location, the bottom circle of tool can be seen as contour of tool envelope surface. The minimum distance between the bottom circle and CC curve is the wanted step error of the tool location, which is obtained by calculating distances from CC points to the bottom circle iteratively. For a CC point, the minimum distance to tool envelope surface is the wanted step error of the CC point. There are different distances between the CC point and different tool locations. Discrete tool locations and bottom circles are obtained iteratively to replace tool envelope surface. The distances between the CC point and bottom circles are calculated to search the minimum distance, which is the step error of the CC point.

The real step error between two adjacent CC points can be calculated by the above two algorithms. In order to compute iso-error CC point, step length is adjusted by changing CC parameters iteratively until step error is in the defined range. Iso-error tool path are calculated based on iso-error CC points.

To meet the requirement of calculation precision, the algorithms in this paper use parameter iterations. However, iterative calculation costs a lot of time and is not an efficient way. Our future work will focus on improving efficiency on algorithms and calculation in order to reduce the time of iso-error tool path generation, which helps to the promotion of iso-error tool path.

**Authors' contributions** Wei Liu, Lv-Yang Fan, Qi-Xin Zhu and Shu-Mei Zhu designed the algorithms of step error in Sect. 2.2. Tian-Li Wang and Feng Tang designed the algorithms of initial CL points and tool vectors for CC curves in Sect. 2.1. Wei Liu and Shu-Mei Zhu designed the algorithms of iso-error CC points and tool path in Sect. 2.3. Wei Liu and Lv-Yang Fan carried out the examples in Sect. 3. Wei Liu wrote the paper. All authors discussed the results and revised the manuscript.

**Funding** The research is funded by Natural Science Foundation of Jiangsu Province (BK20210865), China Postdoctoral Science Foundation (2020M671604), the university science research project of Jiangsu Province (20KJB460025), the science and technology plan project of Suzhou City (SYG202043, SYG201816), the graduate research and innovation projects of Jiangsu Province (202010332012Z).

**Availability of data and material** Main data generated or analyzed during this study are included in this article. All data in this article are available from the corresponding author on reasonable request.

**Code availability** All code is available from the corresponding author on reasonable request.

## Declarations

**Ethics approval** Not applicable.

**Consent to participate** Not applicable.

**Consent for publication** Not applicable.

**Conflicts of interest/competing interests** The authors declare that they have no competing interests.

## References

1. Lasemi A, Xue D, Gu P (2010) Recent development in CNC machining of freeform surfaces: a state-of-the-art review. *Comput Aided Des* 42(7):641–654
2. Zou Q, Zhao JB (2013) Iso-parametric tool-path planning for point clouds. *Comput Aided Des* 45(11):1459–1468
3. He W, Lei M, Bin HZ (2009) Iso-parametric CNC tool path optimization based on adaptive grid generation. *Int J Adv Manuf Technol* 41(5–6):538–548



4. Sun YW, Guo DM, Jia ZY, Wang HX (2006) Iso-parametric tool path generation from triangular meshes for free-form surface machining. *Int J Adv Manuf Technol* 28(7–8):721–726
5. Ding S, Mannan MA, Poo AN, Yang DCH, Han Z (2003) Adaptive iso-planar tool path generation for machining of free-form surfaces. *Comput Aided Des* 35(2):141–153
6. Feng HY, Teng ZJ (2004) Iso-planar piecewise linear NC tool path generation from discrete measured data points. *Comput Aided Des* 37(1):55–64
7. Ding S, Mannan MA, Poo AN, Yang DCH, Han Z (2005) The implementation of adaptive iso planar tool path generation for the machining of free-form surfaces. *Int J Adv Manuf Technol* 26(7–8):852–860
8. Hu PC, Chen LF, Tang K (2017) Efficiency-optimal iso-planar tool path generation for five-axis finishing machining of freeform surfaces. *Comput Aided Des* 83(2):33–50
9. Suresh K, Yang DCH (1994) Constant scallop-height machining of free-form surfaces. *ASME J Eng Ind* 116(2):253–259
10. Lin ZW, Fu JZ, Shen HY, Gan WF (2014) A generic uniform scallop tool path generation method for five-axis machining of freeform surface. *Comput Aided Des* 56(11):120–132
11. Liu W, Zhu SM, Huang T, Zhou C (2020) An efficient iso-scallop tool path generation method for three-axis scattered point cloud machining. *Int J Adv Manuf Technol* 107(1–4):3471–3483
12. Liu W, Zhou LS, An LL (2012) Constant scallop-height tool path generation for three-axis discrete data points machining. *Int J Adv Manuf Technol* 63(1–4):137–146
13. Min C, Gao XS (2015) Iso-scallop tool-path generation of five-axis computer numerically controlled machining for cyclide splines. *Proc Inst Mech Eng B-J Eng* 229(7):1144–1156
14. Li HW, Tutunea-Fatan OR, Feng HY (2007) An improved tool path discretization method for five-axis sculptured surface machining. *Int J Adv Manuf Technol* 33(9–10):994–1000
15. Lu YA, Ding Y, Zhu LM (2016) Tool path generation via the multi criteria optimisation for flat-end milling of sculptured surfaces. *Int J Prod Res* 55(15):4261–4282
16. Du X, Huang J, Zhu LM, Ding H (2019) Third-order chord error estimation for freeform contour in computer-aided manufacturing and computer numerical control systems. *Proc Inst Mech Eng B-J Eng* 233(3):863–874
17. Can A, Ünüvar A (2010) A novel iso-scallop tool-path generation for efficient five-axis machining of free-form surfaces. *Int J Adv Manuf Technol* 51(9–12):1083–1098
18. Xu JH, Zhang SY, Tan JR, Liu XJ (2012) Non-redundant tool trajectory generation for surface finish machining based on geodesic curvature matching. *Int J Adv Manuf Technol* 62(9–12):1169–1178
19. Zhao ST, Zhao DB, Fu YY (2009) Algorithm of variable forward step size planning in freeform surface machining. *Int Conf Inf Autom* 618–622
20. Lin ZW, Fu JZ, Shen HY, Gan WF (2014) An accurate surface error optimization for five-axis machining of freeform surfaces. *Int J Adv Manuf Technol* 71(5–8):1175–1185
21. Fountas NA, Vaxevanidis NM, Stergiou CI, Benhadj-Djilali R (2017) A virus-evolutionary multi-objective intelligent tool path optimization methodology for 5-axis sculptured surface CNC machining. *Eng Comput-Germany* 33(3):375–391
22. Zhang K, Tang K (2016) Optimal five-axis tool path generation algorithm based on double scalar fields for freeform surfaces. *Int J Adv Manuf Technol* 83(9–12):1503–1514
23. Ma JW, Su WW, Jia ZY, Song DN, Hu GQ (2018) An optimization method of tool-path parameters for curved surface by construction of cutter location mesh units. *Int J Adv Manuf Technol* 94(1–4):701–710
24. Dong JL, Yu TB, Chen H, Li BC (2020) An improved calculation method for cutting contact point and tool orientation analysis according to the CC points. *Precis Eng* 61(1):1–13
25. Liu W, Zhang JW, Cao ZY, Zhu SM, Yuan TJ (2016) Direct 5-axis tool posture local collision-free area generation for point clouds. *Int J Adv Manuf Technol* 86(5–8):2055–2067

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.