Tool orientation optimization for 3 + 2-axis CNC machining of sculptured surface

Zhu Yu [a], Chen Zhi-Tong [a,*], Ning Tao [a], Xu Ru-Feng [b]

[a] School of Mechanical Engineering and Automation, Beihang University, Beijing 100191, People's Republic of China
[b] School of Mechanical Engineering, Shandong University of Technology, Zibo 255049, People's Republic of China

ARTICLE INFO

ABSTRACT

This paper presented an optimization method to select a tool orientation for machining a sculptured surface by the 3+2-axis machining strategy. The optimization method could select the tool orientation for the maximum average strip width in 3+2-axis machining. The method could also be used to determine the workpiece setup for general 3-axis machining. The average strip width estimation method was presented as well. Quasi-feasible sectors containing the optimal tool orientation could be found according to the projection planes and the normal vectors of sample points. And the method can find the optimal tool orientation based on projection planes. A freeform surface was parted into 9 sub-surfaces firstly, and then the presented method was applied on those sub-surfaces to determine the optimal tool orientations. The tool paths were generated with the optimized tool orientations and used to mill the sub-surfaces without interference. The method presented could also be applied on the trimmed surface, the surface with a boss, and the blade on a blisk. The machining results indicate that our method can improve machining efficiency through reducing the number of tool paths for 3 + 2-axis sculptured surface machining.

© 2016 Elsevier Ltd. All rights reserved.

1. Introduction

Sculptured surface, also called freeform surface, has been widely used in aerospace and mold industries. 3+2-axis machining uses the three linear motion axes during cutting and fixes the two rotary axes in a particular gesture. It could be treated as a special 5-axis machining strategy. This strategy has high rigidity and low motion error during machining. Generally, fillet-end cutters could be used with this strategy to mill the sculptured surface.

1.1. Research motivation

A lot of researches on 5-axis machining have been performed to improve the quality and the efficiency of sculptured surface machining [1–13]. Compared with the researches on tool orientation optimization for 5-axis machining, the publications [14–21] on 3 + 2-axis sculptured surface machining strategy are not enough. Especially, few researches take the tool orientation optimization into consideration for 3 + 2-axis machining.

Generally, the cost of using a 5-axis machine tool is much higher than that of using a 3+2-axis machine tool. And the 3+2-axis machining could improve the rigidity during machining and increase efficiency comparing with the traditional 3-axis machining. The purpose of our research is to apply the 3 + 2-axis machining strategy to mill the sculptured surface with high efficiency and low cost. Previous researches always focus on surface partitioning. But the detail for choosing the setup gesture for each sub-surface has not been studied enough. This paper will present an optimization method to find a suitable tool orientation for machining a sub-surface with a given torus cutter.

The principle for choosing tool orientation is improving the machining efficiency. Generally, the machining strip width and the feed rate have deep influence on machining efficiency. Redonnet et al. [10] and Lee [13] have illustrated that the productivity could be improved through expanding the effective cutter radius or sweep curve for end milling of freeform surface. It suggests the strip width could influence the machining efficiency. At the same time, the feed rate is always constrained by material, tool life and machining surface roughness. In this paper, it is assumed that the feed rate is constrained at a certain value. Then the purpose of this paper is to study the strategy to improve machining efficiency through enlarging the strip width. Therefore, the principle for tool

* Corresponding author. Tel.: +86 010 82339151; fax: +86 010 82339151.
E-mail address: ztchen@buaa.edu.cn (Z.T. Chen).

orientation optimization would be finding the tool orientation corresponding to the maximum machining strip width.

1.2. Application field

The method presented could be used in finding an optimal tool orientation for a sub-surface with 3 + 2-axis machining strategy, which is a complement for traditional 3 + 2-axis machining. It could be used in fixed axis machining and index axis machining. It could also be used to decide the workpiece setup for general 3-axis machining.

1.3. Research approach

This paper presents a strategy to optimize the tool orientation for 3 + 2-axis machining. The method chooses a tool orientation that is used to generate tool paths for machining the entire surface with a wider strip width. This paper gives a method that estimates the average strip width for a sculptured surface by a series of sample points, and it defines the quasi-feasible sector (QFS) for tool orientation optimization. A given surface should be distributed into plenty of grid points firstly, and then sample points would be selected from those points. Projection planes will be defined, and then the QFS domain for each plane will be found. The optimal tool orientation for those planes will be searched in those QFS regions. We find the best tool orientation from each projection plane. Then those tool orientations will be compared to find the optimal tool orientation. This method is a global optimization method, and the optimal tool orientation could be used to generate tool paths by the iso-parametric method. This method can improve the machining efficiency for 3 + 2-axis machining.

The rest of the paper is organized as follows. Firstly, the overview of 5-axis machining and 3 + 2-axis machining is introduced in Section 2. Then in Section 3, the method for estimating the average machining strip width of a surface is presented. Section 4 proposes the method to find the quasi-feasible sector (QFS) in which the optimal tool orientation will be searched. The tool orientation optimization method based on multiple projection planes is presented in Section 5. Some practical examples are given in Section 6, and then followed by the conclusions and discussion in Section 7.

2. Literature review

2.1. Tool orientation optimization for 5-axis machining of freeform surface

Much work in 5-axis machining has been done on the optimization of tool orientations for making the cutter osculate with the surface at each cutter contact point [2,3], avoiding collisions between the surface and the cutter [4] and satisfying the geometric and dynamic constraints [5–8]. In order to optimize the tool orientation, we should detect the cutting shape in machining firstly, which could be used to compute the strip width at each tool position. Plakhotnik and Lauwers [9] proposed a method to compute the swept section in each cutter location to predict the real shape of the removed materials. Redonnet et al. [10] gave an analytical solution to compute the cutter effective radius by projecting the tool envelope profile into the plane that was perpendicular to the feed direction and take a curve parallel to an ellipse to be the actual cutting shape. Lin et al. [11] found the boundary of a machining band with a distributed machining surface, and then optimized the tool positions in the next tool path for maximizing the machining strip width. Fan and Ball [12] developed the tool orientation optimization method to maximize the machining strip width. Lee [13] studied the tool positioning method and the method to compute

the instantaneous cutting profile from a geometrical standpoint for 4-axis and 5-axis machining with the end milling cutter, and presented that it was possible to maximize machining efficiency through optimizing the tool orientation to fit the instantaneous cutting profile with the local surface shape. Their method could improve machining efficiency through broadening the strip width, which is one of the hotspots in the research of 5-axis machining.

2.2. 3 + 2-axis machining strategy of freeform surface

However, the 3 + 2-axis machining, which is an alternative method to traditional 5-axis machining, has not been studied adequately. Generally a sculptured surface is subdivided into several sub-surfaces firstly. Then each sub-surface is milled individually with a unique tool orientation. Suh and Lee [14] firstly proposed this method to reduce the number of interaction axes when milling a freeform surface. A rotary/tilt table and a three-axis CNC machine tool were used to adjust the setup of the part and mill the relevant subarea. This theory was also called additional-axis machining technology [15].

Gray et al. [16] made a comparison between 3-axis machining and 5-axis machining, and concluded that the 3-axis machine with an additional rotary/tilt table would improve the surface finish. Chen et al. [17] systematically investigated a 3 + 2-axis machining strategy, and then applied the clustering method and the Voronoi method to divide a sculptured surface into several patches. The tool paths and the setup for each patch were calculated afterward. Roman et al. [18] explored a surface partitioning method and optimized the number of sub-divisions to reduce the machining time for 3 + 2-axis machining. Gray et al. [19] adapted the 5-axis AIM algorithm into 3 + 2-axis machining by optimizing the tool orientation for each path. Flores [20] summarized the surface partitioning method for 3+2-axis machining and proved that 3+2-axis machining could reduce machining times comparing with 5-axis machining using the "Sturz" method. Bi et al. [21] employed the accessibility cone to calculate the safe tool length and generate the collision free tool paths for 3 + 2-axis machining with a ball end cutter.

In 3 + 2-axis machining, few studies have been done on the tool orientation optimization for improving machining efficiency. Many researchers have chosen a fixed tool orientation without optimizing it. The 3 + 2-axis Arc Insert Method, proposed by Gray et al. [19], used a projection plane to find a fixed tool orientation for a path. This method chose the tool orientation from the plane defined by the average normal vector and the average feed direction. However this plane may not contain the finest tool orientation for machining the entire surface. Roman et al. [18] used the projection of the normal vectors to find the tool orientation for 3 + 2-axis machining, as shown in Fig. 1. Their method projected normal vectors onto **YOZ** plane and chose a vector outside the normal cone but near the boundary of the cone to be the target tool orientation. In this case, the generated tool paths would have a wider strip width when machining the location where the normal vector is close to the **Z** axis. But the strip width would be extremely narrow when the normal vector deviates significantly from the **Z** axis, as shown in Fig. 1. That is because this vector may not be the optimal tool orientation in this plane, and they have not taken the impact of the gesture of the projection plane into consideration.

3. Average strip width estimation

The tool orientation optimization is based on estimating the average strip width for a freeform surface. The method in this section applies the strip widths computed at a series of specially selected points to estimate the average strip width for the entire surface. The basis for the method includes the tool positioning method, the sample points selecting method and the average strip width estimating method.
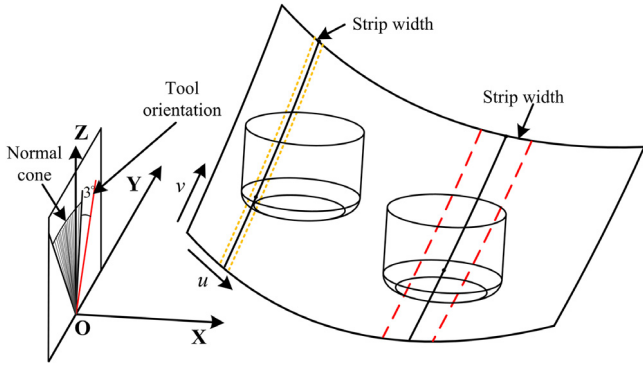
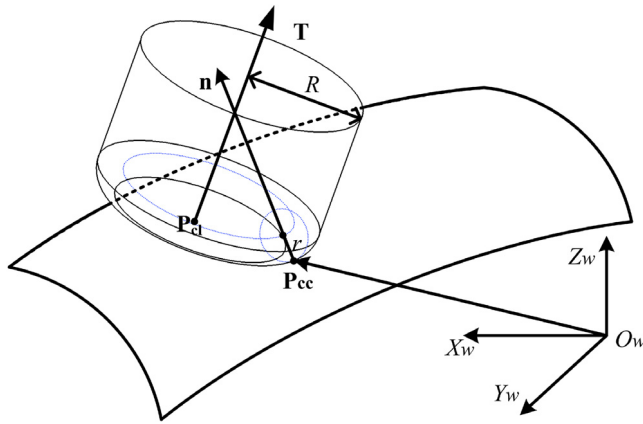**Fig. 1.** The non-optimized tool orientation has large impact on the machining strip width in different regions.

Fig. 2. Calculating tool position with a given tool orientation.

3.1. Tool position computation with a given tool orientation

The tool positioning method is the basis for estimating the average strip width. The tool orientation should maintain a stationary gesture during machining. Cutter locations can be computed using cutter contact points and the given tool orientation.

In Fig. 2, $\mathbf{P}_{cc}$ is the cutter contact point, $\mathbf{n}$ is the unit normal vector, R is the radius of tool flute, and r is the cutter fillet radius. $\mathbf{T}$ is the unit vector along the given tool orientation. The cutter location $\mathbf{P}_{cl}$ (the bottom center of the tool) in the workpiece coordinate can be calculated as:

$$\mathbf{P}_{cl} = \mathbf{P}_{cc} + r \cdot \mathbf{n} + (R - r) \cdot \frac{(\mathbf{T} \times \mathbf{n}) \times \mathbf{T}}{|(\mathbf{T} \times \mathbf{n}) \times \mathbf{T}|} - r \cdot \mathbf{T}. \quad (1)$$

3.2. Sample points selection

We employ the surface partitioning method to select sample points for representing the original surface. A surface should be divided into patches firstly, and then the centroids of patches will be selected to be the sample points for the surface. The strip width computed at a centroid should be able to represent the average strip width in the corresponding patch. The total number of sample points should be under control.

The selection algorithm divides a surface into several patches through iteratively using a BSP (binary space partitioning) method that based on clustering analysis. Each time the BSP method is used, it will part a patch into two patches through three steps. Firstly, the *clustering method* will be used to find two centrals in the patch. Then a boundary between these two points should be identified appropriately. Finally, the patch will be parted by this boundary,
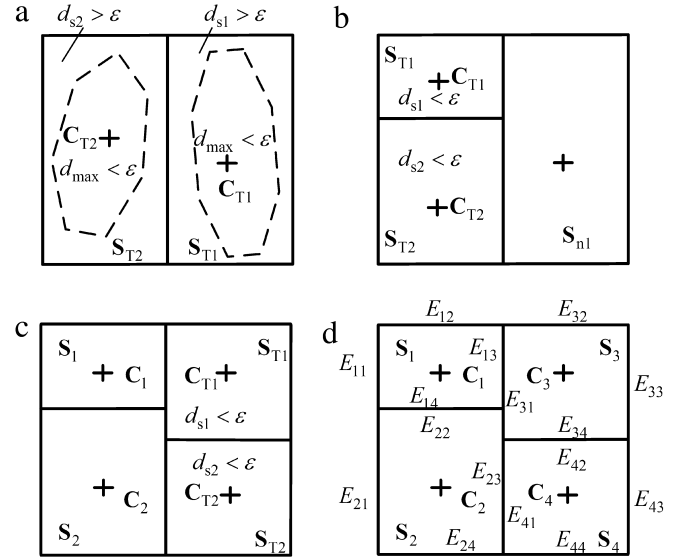

Fig. 3. The sample points selection process in the *uv*-parametric domain.

and the new centroids of two patches will be recalculated by clustering again.

A surface should evenly and densely be dispersed into grid points before surface partitioning. A grid point is defined as $\mathbf{P} = (x, y, z, i, j, k, k_1, k_2)$. $\mathbf{P}_{cc} = (x, y, z)$ is the location of a grid point in workpiece coordinate, $\mathbf{P}_n = (i, j, k)$ is the unit normal vector at a grid point, and k_1 and k_2 are the two principal curvatures at a grid point, which compose a vector $\mathbf{P}_k = (k_1, k_2)$. The reason for choosing those features for surface partition is that those features are directly related to calculate the machining width at a cutter contact point. Each feature of the data should be normalized before the sample points selection.

The distance between two points $\mathbf{P}_1$ and $\mathbf{P}_2$, which indicates the deviation on those features, is defined as:

$$d_{12} = w_{cc} \cdot |\mathbf{P}_{1cc} - \mathbf{P}_{2cc}| + w_n \cdot |\mathbf{P}_{1n} - \mathbf{P}_{2n}| + w_k \cdot |\mathbf{P}_{1k} - \mathbf{P}_{2k}| \quad (2)$$

where w_{cc}, w_n and w_k are the weights of the vectors $\mathbf{P}_{cc}$, $\mathbf{P}_n$ and $\mathbf{P}_k$ respectively. The weights should be used to adjust the number and the shape for the parted patches. Those weights could be chosen through experience and trial from the region [0, 1]. The purposes for adjusting those weights are decreasing the amount of parted patches and reducing the slender patches. According to experience, the weight w_{cc} has a larger effect on the amount of parted patches when the surface fluctuates slightly. And when the surface fluctuates strongly, a small value of w_{cc} and two greater value of w_n and w_k would reduce the number of narrow-long patches. After those weights are determined, the surface would be parted into patches. And the centroid of each patch can adequately represent all grid points in the patch, when the maximum distance between grid points and the centroid point is under a tolerance.

Fig. 3 shows the process that divides a surface into four patches. The dispersed grid points compose the data set $\mathbf{S}$, which represents the original surface. $\mathbf{S}$ is regarded as a surface and also a data set. The data set $\mathbf{S}$ is divided into two parts $\mathbf{S}_{T1}$ and $\mathbf{S}_{T2}$ by clustering firstly. Then a boundary, which is an *uv*-parametric curve between the two centroids $\mathbf{C}_{T1}$ and $\mathbf{C}_{T2}$, is identified to part the surface into two patches (two data sets). The new data sets are assessed by computing the maximal distances d_{s1} and d_{s2} between grid points and the relevant kernels of the data sets. Fig. 3(a) shows that the maximal distances of patches $\mathbf{S}_{T1}$ and $\mathbf{S}_{T2}$ are larger than the given tolerance ε. So the $\mathbf{S}_{T1}$ and the $\mathbf{S}_{T2}$ should be recorded as $\mathbf{S}_{n1}$ and $\mathbf{S}_{n2}$ for further partitioning. Fig. 3(b) shows that the patch $\mathbf{S}_{n2}$ is parted into two parts, and the maximal distances of the data sets
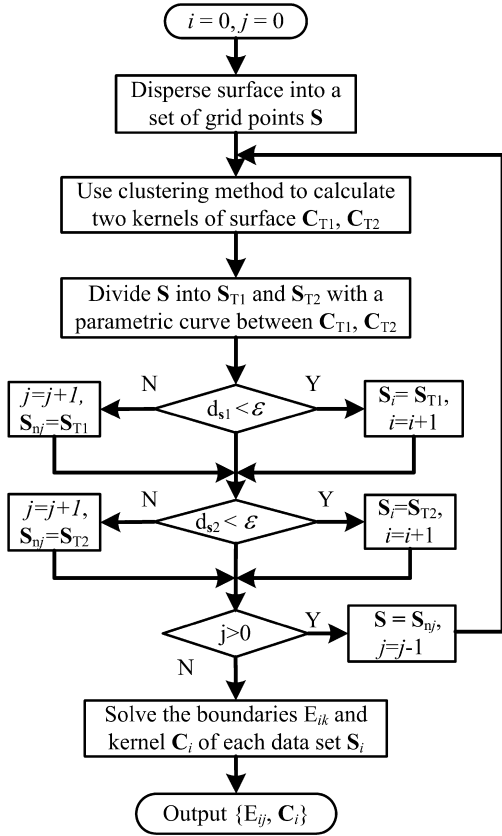
Fig. 4. The sample points selection method.

S_{T1} and S_{T2} are less than the given tolerance ε. Those patches will be recorded as S_1 and S_2. Fig. 3(c) shows the that the patch S_{n2} is parted into two parts and recorded as S_3 and S_4. Then the centers and the boundaries of those patches are computed and shown in Fig. 3(d). $E_{ik}(k = 1, 2, 3, 4)$ are the boundaries of the ith patch, which have a corresponding centroid C_i. Those center points are the sample points.

The flowchart of this algorithm is shown in Fig. 4. And the algorithm works as follows.

Step 1: Input and disperse the original surface. Record the surface by a data set S. The initial $i = 0, j = 0$.

Step 2: Divide the surface S into S_{T1} and S_{T2}. Compute the distances d_{s1} and d_{s2}.

Step 3: If d_{s1} is less than ε, then the patch S_{T1} will be recorded as $S_i, i = i + 1$ and go to *Step* 4. Otherwise $j = j + 1$, and the patch S_{T1} will be recorded as S_{nj}, go to *Step* 4.

Step 4: If d_{s2} is less than ε, then the patch S_{T2} will be recorded as $S_i, i = i + 1$ and go to *Step* 5. Otherwise $j = j + 1$, and the patch S_{T2} will be recorded as S_{nj}, go to *Step* 5.

Step 5: If j is positive, then the data of patch S_{nj} will be given to the patch S, $j = j - 1$ and go to *Step* 2. Otherwise go to Step 6.

Step 6: Compute the boundaries and kernels of the patches recorded by S_i. Then output those boundaries and kernels.

3.3. Average strip width estimation

The average strip width for machining the entire surface can be estimated with a given tool orientation. Since each sample point represents a patch of the surface in the aspect of location, unit normal and curvatures, the strip width computed at the sample point could be treated as the average strip width for the represented patch. Then we use the strip width computed at a
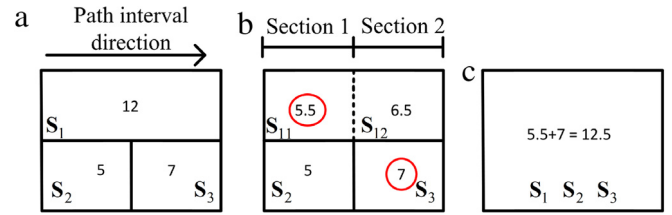

Fig. 5. Strip width computation for a surface parted into 3 patches.

sample point to calculate the tool path number for milling the patch along the path interval direction. After that, the total path number for the machining surface could be computed. Fig. 5 shows the process computing the total path number for a region parted into three patches. The path numbers for the three patches should be computed firstly. Then the surface should be divided into two sections by the boundaries between S_2 and S_3. Since S_1 spans over the two sections, its tool path number should be allocated to two sections according to the proportion of the areas of S_{11} and S_{12}. For each section, we choose the largest number to be the path number in this section. At last, the path numbers in two sections are added together to be the total path number for a surface.

Through computing the error between the cutter and the machining surface, we could solve the machining strip width w_i at the ith sample point. The error will be computed by the method presented in [22], which could also be applied to detect the local interference. The strip width is defined in the original uv-parametric domain.

The path number n_i of this patch is computed as:

$$n_i = (B_{\max} - B_{\min})/w_i \tag{3}$$

where $B_{\max}$ and $B_{\min}$ are the upper and lower boundaries of the patch in uv-parametric domain along the path interval direction respectively.

The estimation algorithm could solve the total path number for machining the entire surface after computing the path numbers in all patches. In Fig. 6(a), path numbers could be calculated for all patches. We should firstly identify a bunch of sections by different boundaries, and then compute the necessary path number for each section, as shown in Fig. 6(b). The total path number can be calculated by accumulating the necessary path numbers by sections. When we compute the number of total paths for machining the entire surface, each patch should be detected and identified which section or sections it belongs to. If the ith patch stretches over the tth section, it should be parted by relevant section boundaries, and then the tool path number could be computed by Eq. (4). This patch is the jth patch that stretches over the tth section.

$$n_{tj} = n_i \cdot (S_{t\ \max} - S_{t\ \min})/(B_{i\ \max} - B_{i\ \min}) \tag{4}$$

where $B_{i\ \max}$ and $B_{i\ \min}$ are respectively the upper and the lower boundaries along the path interval direction of the patch represented by the ith sample point, $S_{t\ \max}$ and $S_{t\ \min}$ are the maximal and the minimal values of the tth section respectively. Then the tool path number in the tth section is defined as:

$$n_{st} = \max\{n_{tj}\} \quad j \in [1, m], \ j \in N \tag{5}$$

where m is the total number of the patches that stretch over the tth section. The average strip width is:

$$\overline{w} = 1/\left(\sum_{t=1}^{M} n_{st}\right) \tag{6}$$

where M is the total number of the sections.

(a) Tool path number in each region. (b) Find path number in each section.
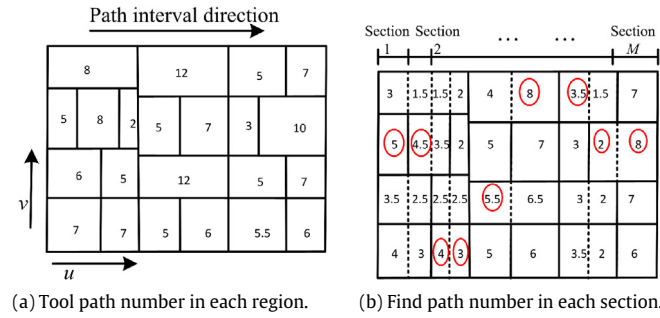
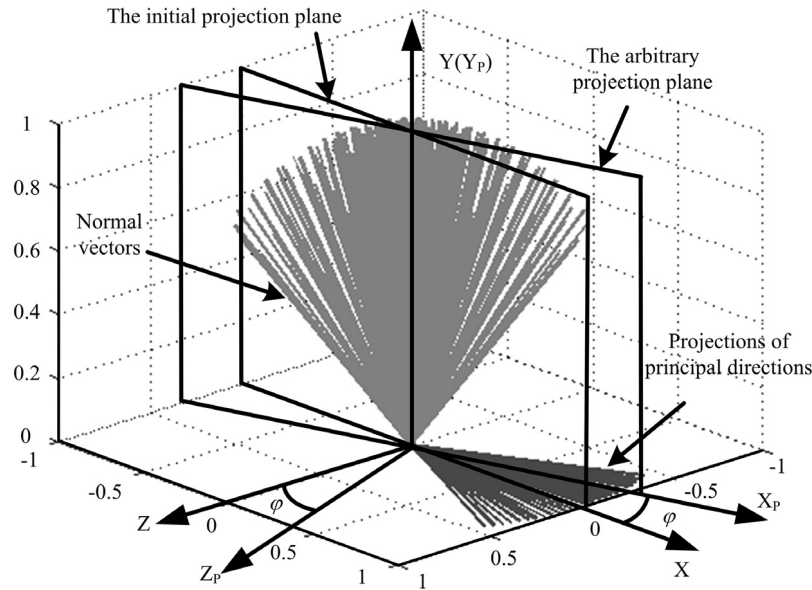Fig. 6. Strip width computation for the entire surface.

Fig. 7. The initial projection plane and the arbitrary projection plane.

4. The quasi-feasible sector (QFS) on a projection plane

In this section we use a projection plane to find a quasi-feasible sector (QFS) domain. This domain contains the vectors that could be used as the tool orientation to generate tool paths for $3 + 2$-axis machining. The QFS domain would be used to optimize the tool orientation in next section.

4.1. Projection plane definition

Projection planes, which are defined by two axes $\mathbf{X_P}$ and $\mathbf{Y_P}$, are used to optimize the tool orientation, shown in Fig. 7. It could be founded by adjusting the gesture of the initial projection plane. The initial projection plane is defined by the average unit normal and the average principal direction. The y-axis of the initial projection plane, $\mathbf{Y}$, is the average normal direction. The plane perpendicular to the y-axis will be used to project the principal directions of minimum principal curvatures of all sample points on. Then the average orientation of the projected principal directions is defined as the x-axis of the initial projection plane, $\mathbf{X}$.

The initial projection plane rotates around the $\mathbf{Y_P}$-axis with the roll angle φ to get an arbitrary projection plane, Fig. 7. Since the plane is defined by two axes, the projection plane can be calculated as follows:

$$\begin{cases} \mathbf{X_P} = \mathbf{X} \cdot \cos \varphi - \mathbf{Z} \cdot \sin \varphi \\ \mathbf{Z_P} = \mathbf{X_P} \times \mathbf{Y_P}. \end{cases} \tag{7}$$

Then a bunch of projection planes could be established based on different values of the roll angle φ, as shown in Fig. 8.
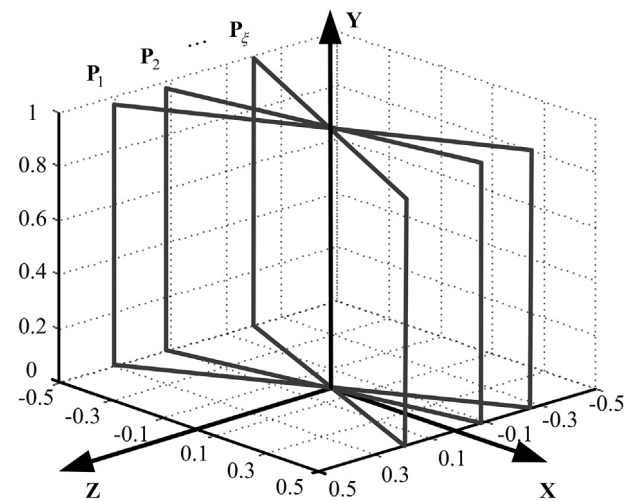

Fig. 8. A bunch of projection planes.

4.2. QFS domain in a projection plane

After a projection plane is constructed, we can project the normal vectors of all sample points onto the plane, Fig. 9. $\mathbf{B}_1$ and $\mathbf{B}_2$ are the boundaries of the sector, and $\mathbf{B}_L$ is defined as the limiting boundary of the tool orientation in this plane. $\mathbf{B}_L$ is perpendicular to $\mathbf{B}_1$ and lies in the first quadrant of the plane. The boundaries are applied to define a quasi-feasible sector (QFS) domain in the
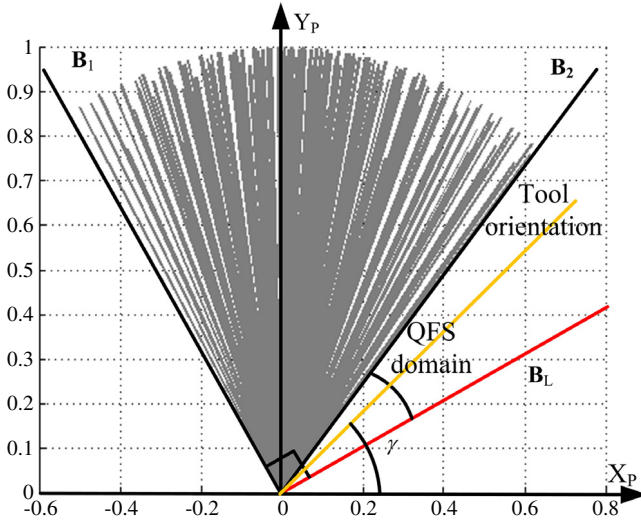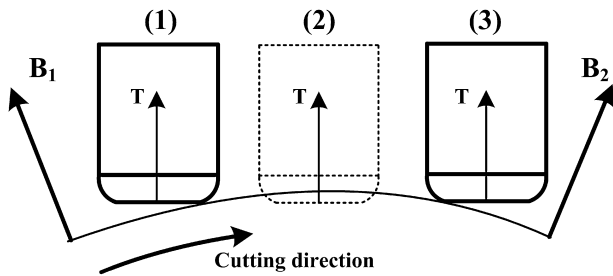
Fig. 9. The normal vectors projected on a plane.

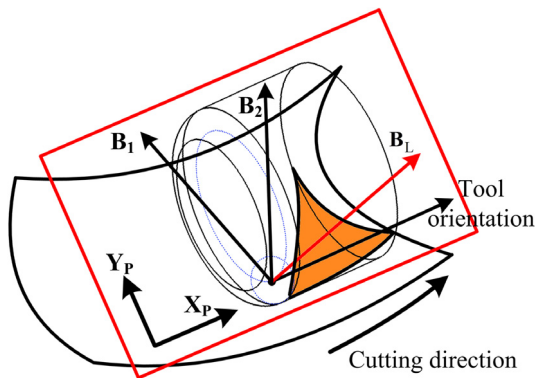**Fig. 10.** Tool orientation lies inside the projection sector.

Fig. 11. Tool orientation lies outside of the QFS domain.

plane, which is the sector area between $\mathbf{B}_L$ and $\mathbf{B}_2$. The feasible tool orientation lies in this domain.

Fig. 10 shows the case that the tool orientation lies between $\mathbf{B}_1$ and $\mathbf{B}_2$. In one path, there will be some tool positions using the front region of torus to mill the surface and some tool positions using the back region of torus to cut the surface. The cutter will gouge the machining surface between two neighboring tool positions if the contact point of the cutter changes from the front region to the back region of the torus.

There will be interference between a cutter and a surface at some cutter locations, when the vector lying between the boundary $\mathbf{B}_L$ and the $\mathbf{X}_P$-axis of the projection plane is used as the tool orientation, Fig. 11. The shadow region illustrates the collision between a fillet end cutter and a concave surface.

Consequently, the tool orientations choosing from a projection plane should lie in QFS region. This region is called the quasi-

feasible sector (QFS) domain, inside which the tool orientation can be selected and optimized. The local and global collision would be avoided by computing the error at every sample point during the optimization process.

5. Tool orientation optimization based on projection planes

After finding the QFS domain for every projection plane, we could establish a model to optimize the tool orientation for milling a surface with the maximal average strip width. The optimal tool orientation will be found from the QFS domains. Since different projection planes correspond to different QFS domains. The optimization method has two steps. The best tool orientation in each projection plane should be searched firstly. And then we could compare those tool orientations and identify the optimal tool orientation.

5.1. Tool orientation optimization model

The optimal tool orientation should be selected by the principle of maximum the average strip width without generating the local and global interference.

$$
\begin{cases}
\mathbf{T}_{opt} = \left\{ \mathbf{T} | w(\mathbf{T}) = \max\{w(\mathbf{T})\}, \ \mathbf{T} \in R^3, \ i^2 \right. \\
\qquad \left. + j^2 + k^2 = 1, k > 0 \right\} \\
\text{s.t.} \\
\{\varepsilon_{\min}(\mathbf{S}, C_{EM}, \mathbf{T}) \geq 0 | \varepsilon_{\min}(\mathbf{S}, C_{EM}, \mathbf{T}) \\
\quad = \min\{\varepsilon(\mathbf{P}_{cc}, \mathbf{S}, C_{EM}, \mathbf{T})\}, \mathbf{P}_{cc} \in \mathbf{S}\} \\
\{\xi_{\min}(\mathbf{S}, \mathbf{S}_I, C_{EM}, \mathbf{T}) \geq 0 | \xi_{\min}(\mathbf{S}, \mathbf{S}_I, C_{EM}, \mathbf{T}) \\
\quad = \min\{\xi(\mathbf{P}_{cc}, \mathbf{S}_I, C_{EM}, \mathbf{T})\}, \mathbf{P}_{cc} \in \mathbf{S}\}
\end{cases}
\tag{8}
$$

where C_{EM} is the end mill cutter, $\varepsilon(\mathbf{P}_{cc}, \mathbf{S}, C_{EM}, \mathbf{T})$ is the minimum distance between a cutter and the machining surface at a cutter contact point $\mathbf{P}_{cc}$, and $\xi(\mathbf{P}_{cc}, \mathbf{S}_I, C_{EM}, \mathbf{T})$ is the minimum distance between a cutter and the interference surfaces at a cutter contact point $\mathbf{P}_{cc}$. When $\varepsilon_{\min} \geq 0$ and $\xi_{\min} \geq 0$, the surface could be machined by the tool orientation without generating interference.

However, the searching space is too large in Eq. (8), and there are too many cutter locations on the surface needed to be solved. In order to improve the computation efficiency, we use a bunch of QFS regions to narrow the original searching space, $\{\mathbf{T} \in R^3 | i^2 + j^2 + k^2 = 1, \ k > 0\}$. The sample points selected by the method presented in Section 3 are employed to replace the original surface $\mathbf{S}$. Then the optimization model would be:

$$
\begin{cases}
\mathbf{T}_{opt} = \left\{ \mathbf{T} | w(\mathbf{T}) = \max\{\overline{w}(\mathbf{T})\}, \mathbf{T} \in R_t^{QFS}, t \in [1, N_\Pi] \right\} \\
\text{s.t.} \\
\{\varepsilon_{\min}(\mathbf{S}, C_{EM}, \mathbf{T}) \geq 0 | \varepsilon_{\min}(\mathbf{S}, C_{EM}, \mathbf{T}) \\
\quad = \min\{\varepsilon(\mathbf{P}_{cc}, \mathbf{S}, C_{EM}, \mathbf{T})\}, \mathbf{P}_{cc} \in \mathbf{C}\} \\
\{\xi_{\min}(\mathbf{S}, \mathbf{S}_I, C_{EM}, \mathbf{T}) \geq 0 | \xi_{\min}(\mathbf{S}, \mathbf{S}_I, C_{EM}, \mathbf{T}) \\
\quad = \min\{\xi(\mathbf{P}_{cc}, \mathbf{S}_I, C_{EM}, \mathbf{T})\}, \mathbf{P}_{cc} \in \mathbf{C}\}
\end{cases}
\tag{9}
$$

where R_t^{QFS} is the QFS region in the tth projection plane, N_Π is the total number of projection planes, $\mathbf{C}$ is the set of all sample points selected, $\overline{w}$ is the estimation strip width that is applied to replace the value of the strip width computed by generating the tool paths. The computation efficiency is improved significantly, because the number of sample points is much less than that of cutter contact points, and the space in QFS regions is just a small part of the original space.

5.1.1. The optimization in a QFS domain

Since a projection plane is defined by its x-axis, y-axis, the normal vectors projected on the plane, $\mathbf{n}_{proj}^i = (x_{proj}^i, y_{proj}^i)$, can

be calculated as:

$$\mathbf{n}_{\mathrm{proj}}^{i} = \mathbf{n}_{i}^{T} \cdot [\mathbf{X}_{\mathrm{P}}, \mathbf{Y}_{\mathrm{P}}]. \tag{10}$$

The angle between $\mathbf{n}_{\mathrm{proj}}^{i}$ and the positive direction of x-axis is:

$$\alpha^{i} = \arctan(y_{\mathrm{proj}}/x_{\mathrm{proj}}). \tag{11}$$

The boundaries $\mathbf{B}_1$ and $\mathbf{B}_2$ correspond to the maximal and minimal values of α^i respectively. Any tool orientation can be computed by the inclination angle γ which is in the range of $[\min\{\alpha^i\}, \max\{\alpha^i\} - \pi/2]$, Fig. 9. The tool orientation, $\mathbf{T} = (i, j, k)^{\mathrm{T}}$, can be solved as:

$$\mathbf{T} = \mathbf{T}(\gamma) = [\mathbf{X}_{\mathrm{P}}, \mathbf{Y}_{\mathrm{P}}] \cdot \begin{bmatrix} \cos\gamma \\ \sin\gamma \end{bmatrix}. \tag{12}$$

The average strip width, $\overline{w}$, could be calculated by the method in Section 3.

Since the strip width at one tool position only depends on the tool axis $\mathbf{T}$, the tool orientation in a projection plane, $\mathbf{T}_\Pi$, can be optimized by solving the problem:

$$\begin{aligned}
\mathbf{T}_\Pi &= \left\{ \mathbf{T} | w(\mathbf{T}) = \max\{\overline{w}(\mathbf{T})\}, \mathbf{T} \in R_\Pi^{\mathrm{QFS}} \right\} \\
&= \{ \mathbf{T} | w(\mathbf{T}) = \max\{\overline{w}(\mathbf{T})\}, \mathbf{T} = \mathbf{T}(\gamma), \\
&\quad \gamma \in [\min\{\alpha^i\}, \max\{\alpha^i\} - \pi/2] \}.
\end{aligned} \tag{13}$$

The tool axis is the dependent variable of the inclination angle γ, the optimum tool orientation in the QFS domain can be find through solving the Eq. (13).

5.1.2. The optimization between different projection planes

Since a projection plane is defined by the roll angle φ, the optimal tool orientation would be found by solving the following optimization model:

$$\begin{aligned}
\mathbf{T}_{\mathrm{opt}} &= \{ \mathbf{T} | w(\mathbf{T}) = \max\{\overline{w}(\mathbf{T}_\Pi)\}, \Pi = \Pi(\varphi), \varphi = \varphi(t), \\
&\quad t \in [1, N_\Pi] \}
\end{aligned} \tag{14}$$

where Π is a projection plane computed by the roll angle φ. And φ could be determined through appropriately defining the maximum, minimum and step increment of φ. Since the plane containing the optimal tool orientation is close to the initial projection plane. The enumeration method with a small step length could be used to determine every projection plane. And then the optimal tool axis could be selected from the tool orientations optimized in different projection planes.

5.2. Tool orientation optimization in a projection plane

The process to find the optimal tool orientation in a projection plane, $\mathbf{T}_\Pi$, is illustrated in Fig. 12. After finding the QFS domain in a projection plane, we could traverse the vectors in the QFS domain to solve the optimization problem in Eq. (13). A tool orientation can be computed by γ, the step increment Δ is used to compute each γ, $\alpha_{\mathrm{limit}} = \max\{\alpha^i\} - \pi/2$, and w_Π records the maximal average strip width according to the tool orientation $\mathbf{T}_\Pi$.

The algorithm works as follows.

Step 1: The initial γ equals to $\min\{\alpha^i\}$, and the initial w_m equals to zero.

Step 2: The value of γ is decreased by Δ. If γ is larger than α_{limit}, go to *Step* 3. Otherwise go to *Step* 5.

Step 3: Compute tool orientation $\mathbf{T}$ with the value of γ. If the tool orientation satisfies the constraints of machine tool geometry, go to *Step* 4. Otherwise go to *Step* 2.
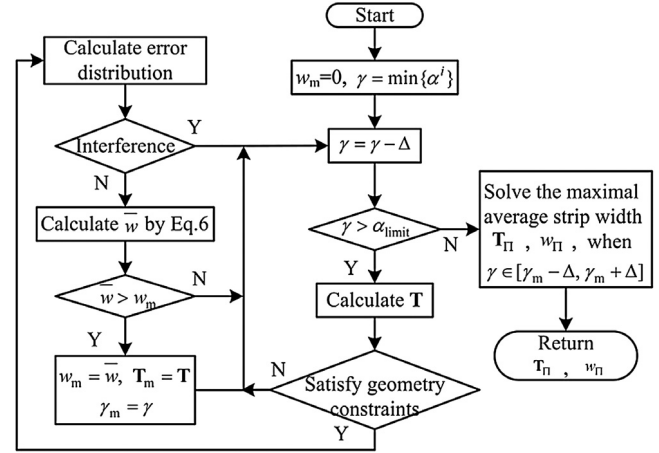

Fig. 12. Optimizing the tool orientation in a projection plane.

Step 4: And then figure out the error at each sample point for detecting interference. If the cutter gouges the surface at a sample point, go to *Step* 2. Otherwise go to *Step* 5.

Step 5: Calculate the average strip width $\overline{w}$. Compare $\overline{w}$ with w_m, the wider value will be recorded by w_m, and the incline angle is recorded by γ_m. Then go to *Step* 2.

Step 6: The *approximate golden section method* will be used to find the best γ in the range of $[\gamma_m - \Delta, \gamma_m + \Delta]$ with a given tolerance.

Step 7: Output the optimal tool orientation, $\mathbf{T}_\Pi$, in a projection plane and the average machining width, w_Π.

The geometry constraints of machine tool relate to the motion ranges of the two rotary axes and the global interference. The motions of two rotary axes can be computed by each tool orientation $\mathbf{T}$. If the two rotary motions are in the feasible ranges respectively, the tool orientation could be applied in the optimization algorithm. These constraints could also be used to avoid global-interference. The tool position at each sample point can be computed with the tool orientation $\mathbf{T}$. Then the distances between the cutter and the interference surfaces will be calculated at every sample point to detect the existence of global interference. The tool orientation could be used only when all tool positions are interference-free.

5.3. Tool orientation optimization based on projection planes

Fig. 13 is the flowchart for solving the optimization problem in Eq. (14). The optimal tool orientation in one projection plane is selected by the method in Section 5.2. In order to solve the optimization model, we should restrict the roll angle φ in an appropriate range. $\varphi_{\min}$ and $\varphi_{\max}$ are the limitation values of the roll angle φ. Δ_φ is the step length of φ. $\mathbf{T}_{\max}$ is the optimal tool orientation corresponding to the maximal average strip width $w_{\max}$.

The algorithm works as follows.

Step 1: Find the initial projection plane, define the initial φ and $w_{\max}$.

Step 2: Compute the projection plane $\Pi(\varphi)$ by the roll angle φ. Then find the best tool orientation $\mathbf{T}_\Pi$ in the plane $\Pi(\varphi)$ using the algorithm in 5.2, and estimate the average strip width w_Π. Go to *Step* 3.

Step 3: Compare the value of w_Π with $w_{\max}$, the wider value will be recorded by $w_{\max}$, and the tool axis corresponding to $w_{\max}$ would be recorded by $\mathbf{T}_{\max}$. Then go to *Step* 4.

Step 4: Update the value of φ with the step increment Δ_φ. If φ is larger than $\varphi_{\max}$, then go to Step 5. Otherwise go to Step 2.

Step 5: Output the optimal tool orientation $\mathbf{T}_{\max}$ and the maximal average strip width $w_{\max}$.

The essential of this method is searching the space in workpiece coordinate system to find the optimal setup for a surface. The QFS region is promoted to narrow the scope for the searching process.

5.4. Extending the method to 4 + 1-axis machining

The projection plane method could be used to find the optimal gesture of the fixed rotational axis for 4 + 1-axis machining. The 4 + 1-axis machine tool has two rotational axes. One rotational axis could rotate continuously, the other one will be fixed in a posture during machining. In order to illustrate the method using projection plan to find the posture of the fixed rotatable axis, we assume that the continuously rotational axis is the A-axis, and the fixed axis is the B-axis.

The posture of the B-axis could be confirmed by the projection plane for 4 + 1-axis machining. However, the projection plane is defined by X-axis and Z-axis, and it cannot change its gesture. Since the 4 + 1-axis machining strategy could adjust its tool orientation at every tool position. The method to estimate the strip width and the method to find the QFS domain have to be modified.

5.4.1. The tool positioning method for 4 + 1-axis machining

A tool position could be figured out for 4 + 1-axis machining by the angle of B-axis, B, the yaw angle, ϕ, and the cutter contact point, $\mathbf{P}_{cc}$. Traditionally, the tool position for 5-axis machining can be solved by the incline angle λ and the yaw angle ϕ in local coordinate system.

$$\mathbf{R}(A) \cdot \mathbf{R}(B) \cdot [0, 0, 1]^T = \mathbf{M}_{WL} \cdot \mathbf{M}(\phi) \cdot \mathbf{M}(\lambda) \cdot [0, 0, 1]^T \quad (15)$$

where $\mathbf{R}(A)$ and $\mathbf{R}(B)$ are the rotational transfer matrixes in workpiece coordinate system, $\mathbf{M}(\phi)$ and $\mathbf{M}(\lambda)$ are the rotational transfer matrixes in local coordinate system, $\mathbf{M}_{WL}$ is a transform matrix. The relationship between A and B could be established with the incline angle λ and the yaw angle ϕ [13]. Then the relationship that treats the motion of B-axis as a given constraint is as follows:

$$(A, \lambda) = f(B, \phi). \quad (16)$$

If the angles of B and ϕ are given, the angles of A and λ could be solved by the Eq. (16). Then the tool position can be solved.

5.4.2. The method of computing machining strip width at each sample point

Since the 4 + 1-axis machining could adjust the tool orientation at every tool position using the continuously rotatable axis. The tool orientation should be optimized at every sample point. The tool orientation is determined by B and ϕ. Since the angle of B will be given before computing tool positions, the value of ϕ should be optimized at each tool position for the maximum machining strip width w_i. Then the strip width w_i will be applied in Eq. (3) to solve the average machining strip width for the entire surface, which will be applied to select the optimal angle of B.

5.4.3. The method of computing the fixed rotary axis

The method that optimizes the angle of B for 4 + 1-axis machining is the extension of the projection plane method presented before. But the projection plane should be defined as the XOZ plane in the workpiece coordinate system, which cannot change its gesture. The other difference between the optimization method for 3 + 2-axis machining and the method for 4 + 1-axis machining is the angle α^i computed in Eq. (11). It should be modified as:
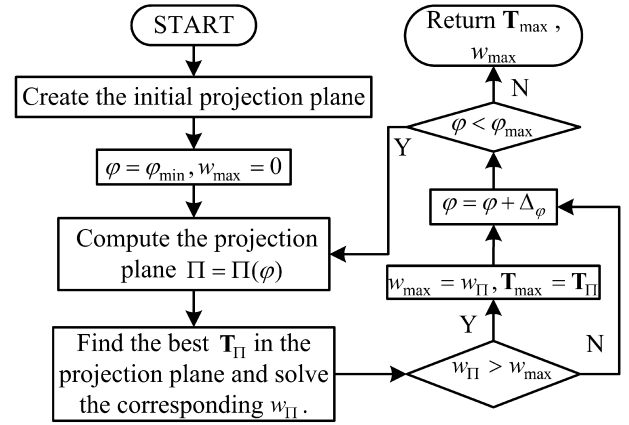
$$\alpha^i = \arccos x_{ni} \quad (17)$$

Fig. 13. Flowchart of tool orientation optimization.

where x_{ni} is the component of the normal vector $\mathbf{n}_i$ on x-axis in workpiece coordinate system. Then the method in Section 5.2 could be used to solve the optimal angle γ which will be used as the fixed angle of the B-axis.

6. Machining example

The proposed algorithm was implemented by using C programming language on the platform of UG Open/API. The selected tool orientation would be applied to generate tool paths for machining the surface using 3 + 2-axis machining strategy. The method presented was to find the optimal tool axis, which would be fixed during machining a surface or a region. A freeform surface with complex curvature variation and great surface fluctuation, shown in Fig. 14, was used to demonstrate the validity of our method. The surface partition method was used to divide the original surface into 9 sub-surfaces firstly. Then the optimum tool orientations would be used to generate tool paths for milling the surface. The tool paths were tested and verified on a 5-axis machine tool, DMG dmu 60t, with a single spindle head (B-axis) and a single rotation table (C-axis). The two rotary axes were locked during machining.

6.1. Sample points selection

In this step, some parameters should be given firstly. The cutter used had a diameter of 12 mm and a corner radius of 1 mm. The tolerance ε equaled to 0.1. And the weights for sample points selection method were given in Table 1. The sample points for surface $\mathbf{S}_4$ and the region represented by the point were shown in Fig. 15. The sample points of the other 8 surfaces could be selected similarly.

6.2. The effectiveness of strip width estimation method

The surface $\mathbf{S}_4$ and 39 tool orientations were used to testify the availability and the efficiency of the strip width estimation method. Those tool orientations were selected from the projection plane when φ equals to $-5°$. The tool orientations were solved by different values of γ. Fig. 16 showed the estimated average strip widths and the average strip widths which were computed through generating tool paths.

Fig. 16 indicates that the average strip width computed by the estimation method is close to the actual value. And the variation of the estimated strip width is almost same as the variation of the actual value. For exhibiting the effectiveness of our estimation method, we chose three tool orientations from Fig. 16. And the corresponding computation results were demonstrated in Table 2.
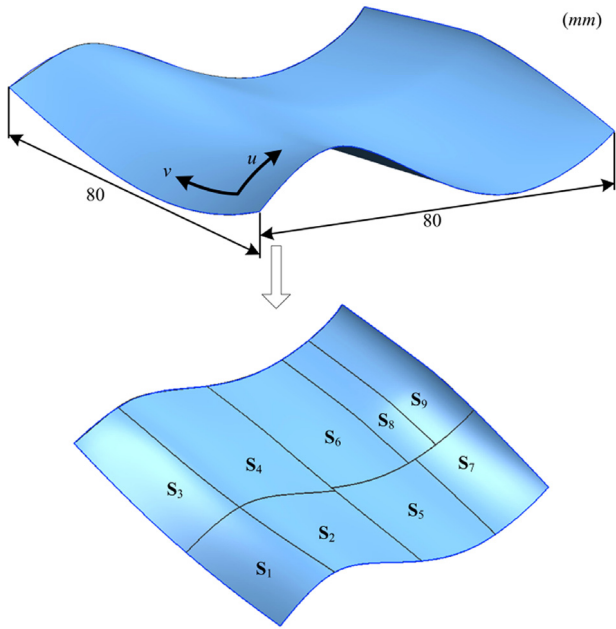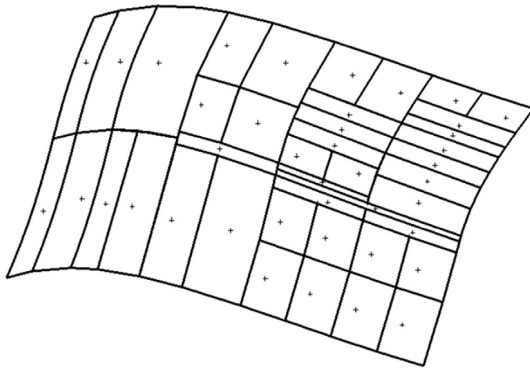
Fig. 14. The original machining surface.

Fig. 15. Selecting sample points from the surface S_4.

Table 1
The parameters for sample points selection.

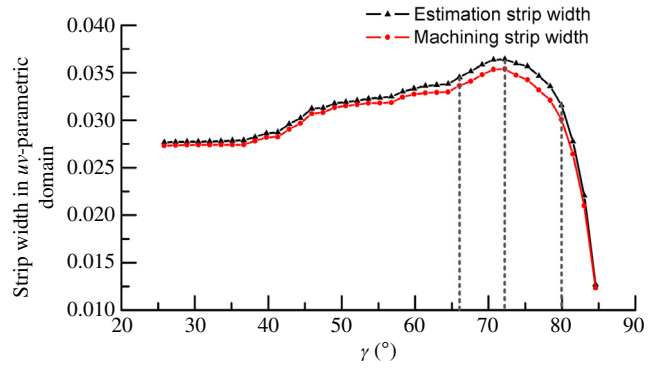| Surface	The weights of different factors		
	w_{cc}	w_n	w_k
S_1	0.25	0.25	0.5
S_2	0.2	0.4	0.4
S_3	0.33	0.33	0.33
S_4	0.17	0.5	0.33
S_5	0.2	0.4	0.4
S_6	0.17	0.5	0.33
S_7	0.2	0.4	0.4
S_8	0.17	0.33	0.5
S_9	0.14	0.43	0.43

Fig. 16. Comparison between the estimation strip width and the actual strip width.

It shows that the error between the estimating value and the actual value is negligibly small and the maximum relative error is less than 2%.

6.3. Tool orientations optimization and paths generation for nine sub-surfaces

Fig. 17(a) showed the projection plane that contained the optimal tool orientation. The QFS domain of this projection plane was obtained, and then the optimal tool orientation was found, as shown in Fig. 17(b). The maximum cusp height value was 0.02 mm for tool paths generation.

Two groups of tool paths were generated to display the difference between the tool orientation found by our method and the tool orientation used in traditional 3 + 2-axis machining. Fig. 18(a) showed the tool paths generated using the optimal tool orientation. Fig. 18(b) showed the tool paths that computed by the method presented in [18]. Fig. 19 showed the strip widths of the tool paths generated by those two methods. The results indicate that the average strip width calculated by our method is larger than that computed by the method presented in [18] (see Table 3).

The machining results of 9 sub-surfaces were shown in Fig. 20. The results indicate that this method could optimize the tool orientation for an entire or a region of freeform surface in 3 + 2-axis machining or fixed axis machining. It could also be used to determine the workpiece setup for machining a freeform surface by a 3-axis machine.

The surface cannot be machined by 3-axis machining strategy with the torus cutter. But it could be machined on 3-axis machine tool with a ball end cutter. It could also be milled by the 5-axis machining strategy and the fixed axis machining strategy with the torus cutter. The tool paths with those strategies were generated respectively. And the computing results were shown in Table 4. The results indicate that the 5-axis machining strategy has the shortest total path length. The tool path length of the 3 + 2-axis machining strategy with our method to find the tool orientation was 1.457 times of the path length of 5-axis machining strategy. The machining times for finishing cutting the surface by those four strategies were computed with the same feedrates, and then the values were listed in Table 4. The cutting speed was 3000 mm/min
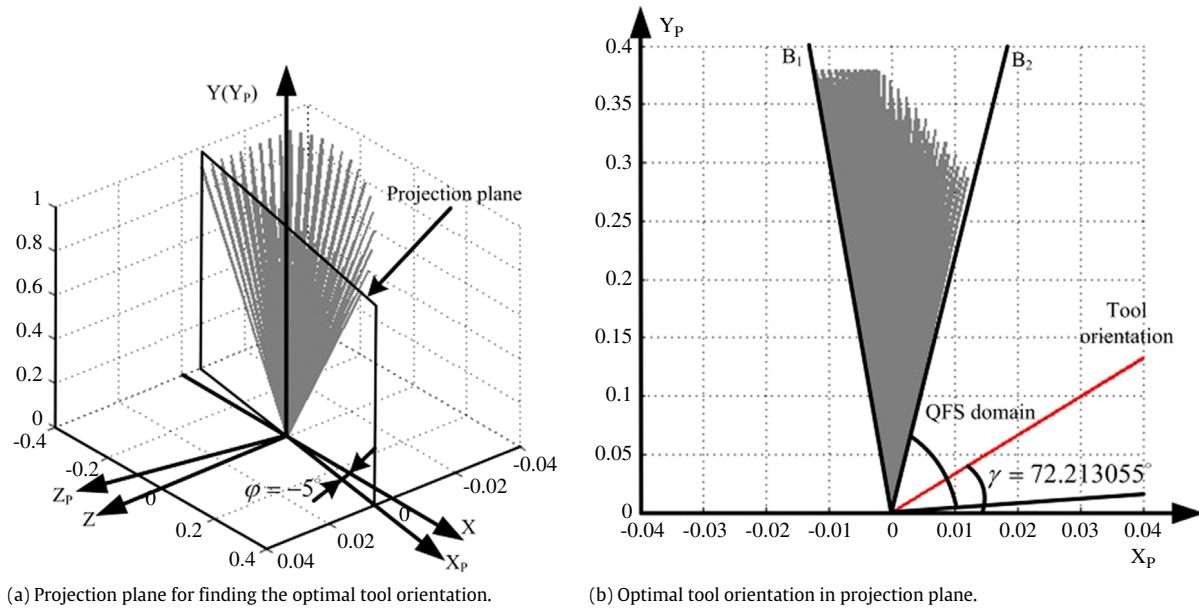
Table 2
The effectiveness of two methods in computing the average strip width.

	$\gamma = 66.03019°$		$\gamma = 72.213055°$		$\gamma = 79.941636°$	
	$\mathbf{T} = (0.359632, 0.367225, 0.857794)$		$\mathbf{T} = (0.36956, 0.264966, 0.890628)$		$\mathbf{T} = (0.375916, 0.13295, 0.917067)$	
	Strip width estimation	Tool paths generation	Strip width estimation	Tool paths generation	Strip width estimation	Tool paths generation
Strip width	0.034522195	0.0336397	0.036417175	0.0353815	0.031561425	0.0300705
Computing time (s)	0.346	16.126	0.351	15.654	0.353	19.05
Computing error	0.000882		0.001036		0.001491	

(a) Projection plane for finding the optimal tool orientation.

(b) Optimal tool orientation in projection plane.

Fig. 17. Tool orientation optimization result.

(a) Tool paths generated using the method in the paper.

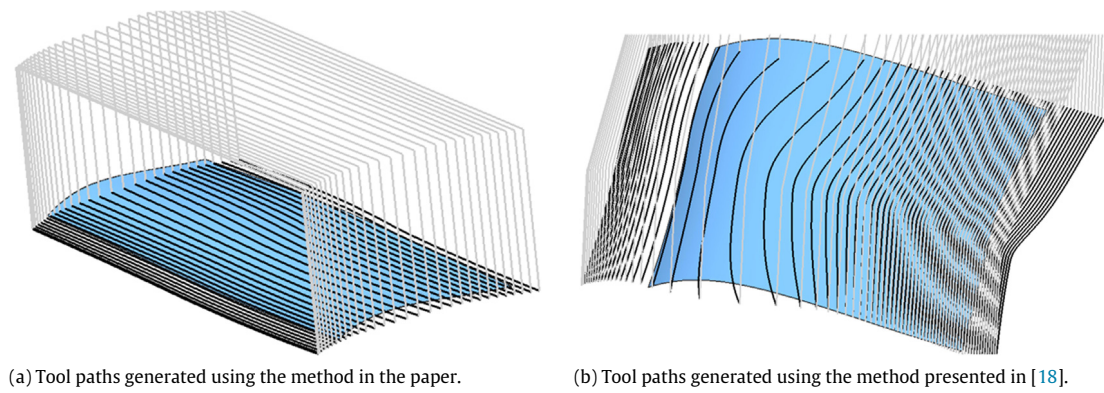(b) Tool paths generated using the method presented in [18].

Fig. 18. Tool paths generated by two different methods.

Table 3
The different parameters and computing results for the 9 sub-surfaces.

Methods	Surface	Number of the sample points	Too orientation	Tool path number
The method in this paper	S_1	55	(−0.416374, 0.316032, 0.852500)	19
	S_2	48	(0.635474, 0.019791, 0.771869)	31
	S_3	37	(−0.114940, −0.127445, 0.985163)	35
	S_4	40	(0.36956, 0.264966, 0.890628)	30
	S_5	37	(0.167221, 0.236284, 0.957187)	19
	S_6	29	(0.211053, 0.276318, 0.937606)	10
	S_7	60	(−0.414398, 0.126486, 0.901263)	14
	S_8	50	(−0.201413, 0.227219, 0.952788)	5
	S_9	42	(−0.479950, 0.340633, 0.808466)	10
The method presented in [18]	S_4	×	(0.295012, 0.076967, 0.951938)	71

Table 4
Comparison of the 4 machining strategies.

	5-axis machining strategy	Fixed axis machining strategy	3 + 2-axis machining strategy	3-axis machining with ball-end cutter
Total path length (mm)	3927.03	8976.06	5720.1	11684.1
Total machining time (s)	194.58	443.79	359.3	576.1

during finish machining, and the spindle speed was 7500 rpm. The machining efficiency of 3 + 2-axis machining is lower than 5-axis machining strategy currently. But since the 3 + 2-axis machining has high rigidity. It is possible to apply a higher federate while using the 3 + 2-axis machining strategy in the future.

6.4. Applying the method on trimmed surface

This method could also be used on trimmed surface machining for finding the optimum tool axis or workpiece setup. The only difference lies in the process of finding the sample points from
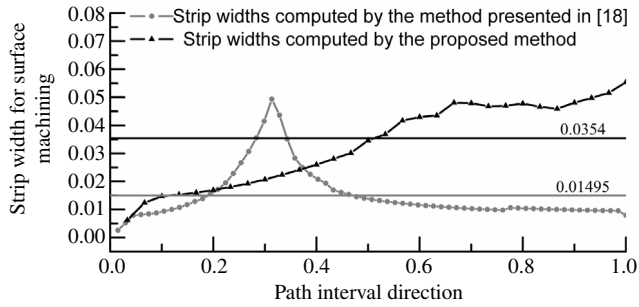
Fig. 19. Comparison of the tool paths generated by different methods.
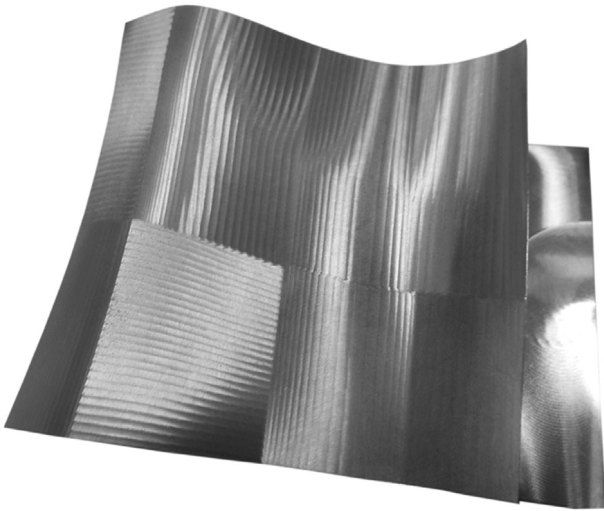

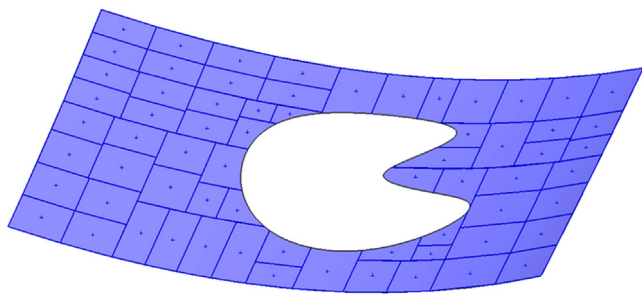**Fig. 20.** The machining result of the entire surface.

Fig. 21. The trimmed surface and the corresponding sample points.

the surface. The problem could be easily solved by modifying the Eq. (2) when computing the distance between two points, and by correctly identifying the region represented by every sample point. A trimmed surface and the sample points selected from the surface were given in Fig. 21.

The weights for selecting sample points should be carefully chosen for the trimmed surface shown in Fig. 21. The weight w_{cc} was 0.334, the weight w_n was 0.283 and the weight w_k was 0.383. The given tolerance ε equaled to 0.1.

After determining the sample points, we could employ the projection plane method to find the optimum tool axis and generate tool paths for this surface. In this example, the torus cutter used had a diameter of 16 mm and a corner radius of 2 mm. And the maximum cusp height value was 0.02 mm. The optimal tool orientation was $(-0.535191, 0.475039, 0.698505)$, and the tool
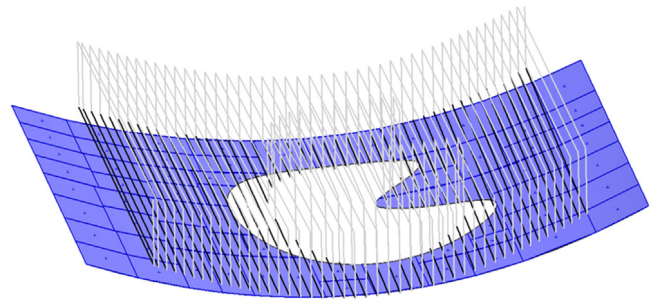

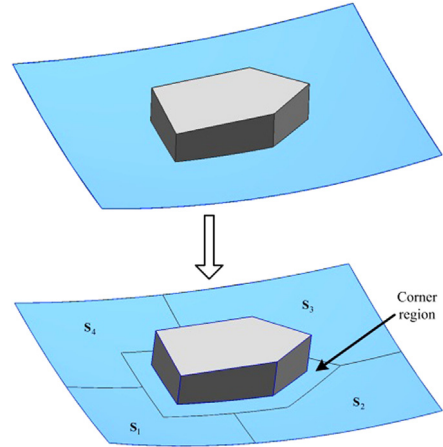**Fig. 22.** The tool paths generated for the trimmed surface.

Fig. 23. The model and surface partitioning for a freeform surface with a boss.

path number was 41. The tool paths generated were shown in Fig. 22. The result indicates that our method could be applied to find the optimal tool axis for the trimmed surface.

6.5. Applying the method on surface with bosses

The method could be applied on the surface with a boss. As shown in Fig. 23, a freeform surface should be parted by computing the distance between the point on surface and the boss. The surface in Fig. 23 was parted into 4 sub-surfaces firstly. Then our method could be used to find the best tool orientation for each sub-surface. During optimization process, the local interference and the global interference should be detected since there is a boss. The corner region of the boss cannot be machined by the torus cutter. This region should be clean-up by a ball-end cutter, and will be studied in the future.

The tool paths for each sub-surface were shown in Fig. 24. In this example, the torus cutter used had a diameter of 10 mm and a corner radius of 1 mm. And the maximum cusp height value was 0.02 mm. For sample points selection, the weight w_{cc} was 0.6, the weight w_n was 0.2 and the weight w_k was 0.2. The given tolerance ε equaled to 0.1. Table 5 showed the optimal tool orientation and the corresponding tool path number for every sub-surface.

If a surface with several bosses needs to be machined, the surface should be divided by the bosses firstly. Then each sub-surface could be parted and milled as the example in this section.

6.6. Applying the method on a blade surface of a blisk

The method could be applied on the blade surface of a blisk. As shown in Fig. 25, a blisk with 16 blades was used to demonstrate the validity of the proposed method. The shortest distance between the adjacent blades was 13.4 mm. The blade

Table 5
The computing results for the 4 sub-surfaces.

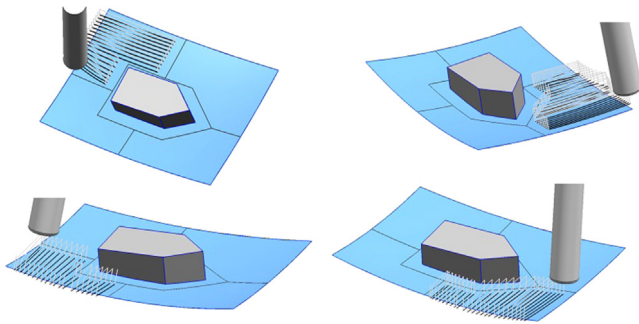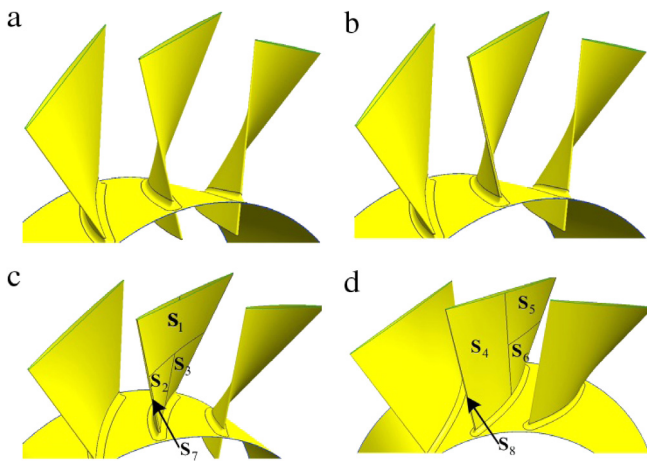| Surface	Too orientation	Tool path number	Average strip width
S_1	(0.144147, 0.405891, 0.902482)	19	0.05389
S_2	(−0.018188, 0.258719, 0.965782)	24	0.04228
S_3	(−0.028162, −0.093619, 0.99521)	22	0.04381
S_4	(0.222612, 0.064526, 0.972769)	17	0.06019

Fig. 24. The tool paths computed for the four sub-surfaces.

Fig. 25. The model and surface partitioning for a blade of a blisk.

Fig. 26. The tool paths computed for two sub-surfaces.

surface should be parted into the suction surface, the pressure surface, the leading edge and the trailing edge firstly. Then the suction surface and the pressure surface should be divided into several sub-surfaces. The original blade was parted into 8 sub-surfaces, and then our method could be used to find the best tool orientation for each sub-surface. During optimization process, the local interference and the global interference should be detected. The root and the fillet corner region of the blade cannot be machined by the torus cutter. The clean-up tool paths generation method for 3 + 2-axis machining will be studied in the future.

The tool paths for two typical sub-surfaces were shown in Fig. 26. In this example, the torus cutter used had a diameter of 10 mm and a corner radius of 1 mm. And the maximum cusp height value was 0.02 mm. For sample points selection, the weight w_{cc} was 0.7, the weight w_n was 0.15 and the weight w_k was 0.15. The given tolerance ε equaled to 0.1. Table 6 showed the optimal tool orientation and the corresponding tool path number for every sub-surface. The searching range of φ should be expanded when searching the optimal tool orientation for the blade on a blisk, since it is easy to generate interference during machining.

The method presented could be used to determine the tool orientation for a freeform surface no matter whether the surface is trimmed. It could be used to complement the traditional 3 + 2-axis machining strategy. The surface partition method presented by Chen [17] could be used to part the surface into several patches.
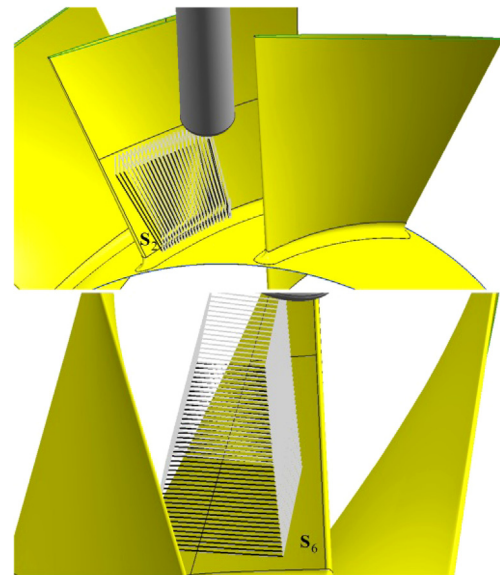
Then our method could be used to find the optimal tool orientation for each patch to acquire a wider average strip width.

7. Conclusions and discussion

This paper has introduced a novel tool orientation optimization method for 3 + 2-axis machining using a fillet-end cutter. The key issues to tool orientation optimization were establishing an optimization model and finding the QFS domain. This paper established the tool orientation optimization model based on the sample points selection method and the average strip width estimation method. The algorithm was presented to optimize the tool orientation for maximizing the average strip width. The sample points selection method and the average strip width estimation method could also be used to solve other problems.

The algorithms implemented on a freeform surface indicate that the proposed method could find a better tool orientation for generating more efficient tool paths. And the method could also be used on the trimmed surface, the surface with bosses and the blade surface of a blisk. The sample points selected by the surface partitioning method could represent the original surface. And the estimated average strip width was receivable. This research has emphasized on the optimization of the tool orientation for the entire surface machining rather than optimizing the tool orientation at one cutter location. This work would promote the availability and efficiency of 3 + 2-axis machining.

Since the original surface is parted into several sub-surfaces for 3 + 2-axis machining, a tiny visible mark will be left on the border of two neighboring patches after machining. But the mark would not exceed the tolerance generally, when the linear axis of a machine tool has a high precision, the rigidity of a part fulfills the requirement, and the cutter has no wear. If the undercut does exist in the edge of a sub-surface, the edge should be machined again using the tool orientation optimized before. The undercut usually

Table 6
The computing results for the 8 sub-surfaces of a blade on a blisk.

Surface	Too orientation	Tool path number	Average strip width
S_1	(−0.735501, 0.654352, 0.175674)	34	0.02951
S_2	(−0.230669, −0.153449, 0.960856)	25	0.04084
S_3	(0.116362, −0.516818, 0.84815)	34	0.02968
S_4	(0.543971, −0.068807, 0.836278)	33	0.03077
S_5	(−0.262985, 0.80952, 0.524896)	38	0.02667
S_6	(−0.223237, 0.463047, 0.85776)	41	0.02439
S_7	(−0.287711, 0.746861, 0.599517)	12	0.08387
S_8	(0.257789, −0.917348, 0.303344)	12	0.08472

occurs on the border of the sub-surface that has a very irregular shape. So the irregular shaped sub-surface should be avoided to eliminate such a situation. When the 3 + 2-axis machining strategy is applied to machine a part with low rigidity, the rigidity of the part should be improved, and the machining sequences should be carefully arranged. Besides that, the polishing machine and the vibratile tumbling machine could be used after milling process when the requirement of surface quality is very strict. The machining mark would be removed on those machines.

Since machining in a patch-by-patch way requires that the cutter has to enter and exit from the middle of the surface frequently. The efficiency of 3 + 2-axis machining is lower than that of 5-axis machining currently. The machining time listed in Table 4 shows the efficiency of four machining strategies with the same cutting speed and approaching speed. However, the cost of a 3 + 2-axis machine would be much cheaper than that of a 5-axis machine, and the rigidity of the 3 + 2-axis machining strategy is higher. In order to make full use of the advantages, we should improve the efficiency of 3 + 2-axis machining strategy to narrow the gap with 5-axis machining strategy. This paper focuses on optimizing the tool orientation to improve the efficiency for machining every sub-surface. Since the number of sub-surfaces would influence the efficiency of 3 + 2-axis machining, our future research would be improving the efficiency through reducing the number of sub-surfaces. Besides that, the future research could focus on improving the efficiency through improving the cutting speed to make the most of the advantages of the 3 + 2-axis machining strategy.

However, this research has not taken tool size, cutting force, vibration and other issues into consideration. Further research should extend this work and optimize the tool orientation for other goals and different cutters, and the method to improve the computation speed should be studied as well. The strategy to appropriately employ the 3 + 2-axis machining is still an open question.

Acknowledgments

References

[1] Lasemi A, Xue D, Gu P. Recent development in CNC machining of freeform surfaces: A state-of-the-art review. Comput-Aided Des 2010;42(7):641–54.
[2] Zhu L, Lu Y. Geometric conditions for tangent continuity of swept tool envelopes with application to multi-pass flank milling. Comput-Aided Des 2015;59:43–9.
[3] Farouki RT, Han CY, Li S. Inverse kinematics for optimal tool orientation control in 5-axis CNC machining. Comput Aided Geom Design 2014;31(1):13–26.
[4] Tang TD. Algorithms for collision detection and avoidance for five-axis NC machining: A state of the art review. Comput-Aided Des 2014;51:1–17.
[5] Hu P, Tang K. Improving the dynamics of five-axis machining through optimization of workpiece setup and tool orientations. Comput-Aided Des 2011;43(12):1693–706.
[6] Chen DZ, Misiołek E. Computing feasible toolpaths for 5-axis machines. Theoret Comput Sci 2012;447:13–25.
[7] Farouki RT, Li S. Optimal tool orientation control for 5-axis CNC milling with ball-end cutters. Comput Aided Geom Design 2013;30(2):226–39.
[8] Zou Q, Zhang J, Deng B, Zhao J. Iso-level tool path planning for free-form surfaces. Comput-Aided Des 2014;53:117–25.
[9] Plakhotnik D, Lauwers B. Computing of the actual shape of removed material for five-axis flat-end milling. Comput-Aided Des 2012;44(11):1103–14.
[10] Redonnet J, Djebali S, Segonds S, Senatore J, Rubio W. Study of the effective cutter radius for end milling of free-form surfaces using a torus milling cutter. Comput-Aided Des 2013;45(6):951–62.
[11] Lin Z, Fu J, Shen H, Gan W. A generic uniform scallop tool path generation method for five-axis machining of freeform surface. Comput-Aided Des 2014; 56:120–32.
[12] Fan J, Ball A. Flat-end cutter orientation on a quadric in five-axis machining. Comput-Aided Des 2014;53:126–38.
[13] Lee YS. Mathematical modeling using different endmills and tool placement problem for 4-and 5-axis NC complex surface machining. Int J Prod Res 1998; 36(3):785–814.
[14] Suh S, Lee J. Five-axis part machining with three-axis CNC machine and indexing table. ASME J Manuf Sci Eng 1998;120:120–8.
[15] Suh SH, Lee JJ, Kim SK. Multiaxis machining with additional-axis NC system: Theory and development. Int J Adv Manuf Technol 1998;14(12):865–75.
[16] Gray P, Bedi S, Ismail F, Rao N, Morphy G. Comparison of 5-axis and 3-axis finish machining of hydroforming die inserts. Int J Adv Manuf Technol 2001; 17(8):562–9.
[17] Chen Z, Dong Z, Vickers GW. Automated surface subdivision and tool path generation for 3 1/2 1/2-axis CNC machining of sculptured parts. Comput Ind 2003;50(3):319–31.
[18] Roman A, Bedi S, Ismail F. Three-half and half-axis patch-by-patch NC machining of sculptured surfaces. Int J Adv Manuf Technol 2006;29(5):524–31.
[19] Gray PJ, Ismail F, Bedi S. Arc-intersect method for $3\frac{1}{2}\frac{1}{2}$-axis tool paths on a 5-axis machine. Int J Mach Tools Manuf 2007;47(1):182–90.
[20] Roman FA. Surface partitioning for 3+2-axis machining [dissertation]. Waterloo, ON, Canada: University of Waterloo; 2007.
[21] Bi Q, Ding H, Wang Y. Safe and short tool length generation for 3+2 axis NC machining of a ball-end cutter using graphics hardware. In: Xiong C, et al., editors. Intelligent robotics and applications. ICIRA 2008: Proceedings of the first international conference on intelligent robotics and applications; 2008 Oct 15–17; Wuhan, China. Berlin: Springer; 2008. p. 348–55.
[22] Engeli M, Waldvogel J, Schnider T. Method for processing work pieces by removing material. United States patent US 6485236B1. 2002 November 26.