Dong He
Department of Mechanical and
Aerospace Engineering,
Hong Kong University of Science and Technology,
Clear Water Bay, Kowloon, Hong Kong
e-mail: dong.he@connect.ust.hk

Yamin Li
Department of Mechanical and
Aerospace Engineering,
Hong Kong University of Science and Technology,
Clear Water Bay, Kowloon, Hong Kong
e-mail: ylifm@connect.ust.hk

Zhaoyu Li
Department of Mechanical and
Aerospace Engineering,
Hong Kong University of Science and Technology,
Clear Water Bay, Kowloon, Hong Kong
e-mail: zlidh@connect.ust.hk

Kai Tang[1]
Department of Mechanical and
Aerospace Engineering,
Hong Kong University of Science and Technology,
Clear Water Bay, Kowloon, Hong Kong
e-mail: mektang@ust.hk

Geodesic Distance Field-Based Process Planning for Five-Axis Machining of Complicated Parts

A critical task in multi-pass process planning for the five-axis machining of complicated parts is to determine the intermediate surfaces for rough machining. Traditionally, the intermediate surfaces are simply parallel Z-level planes, and the machining is of the simplest three-axis type. However, for complicated parts, this so-called Z-level method lacks flexibility and causes isolated islands on layers, which require extraneous air movements by the tool. Moreover, the in-process workpiece machined according to the Z-level method suffers from the staircase effect, which often induces unstable dynamic problems on the tool-spindle system. In this paper, we propose a new method of planning a five-axis machining process for a complicated freeform solid part. In our method, the intermediate surfaces are no longer planar but curved, and they are intrinsically influenced by the convex hull of the part. The powerful algebraic tool of geodesic distance field is utilized to generate the desired intermediate surfaces, for which collision-free five-axis machining tool paths are then planned. In addition, we propose a novel idea of alternating between the roughing and finishing machining operations, which helps improve the stiffness of the in-process workpiece. Ample physical cutting experiments are performed, and the experimental results convincingly confirm the advantages of our method. [DOI: 10.1115/1.4048956]

Keywords: five-axis machining, geodesic distance field, volume decomposition, process planning, tetrahedron model, CAD/CAM/CAE, computer-integrated manufacturing

1 Introduction

Process planning is pivotal in the multi-stage machining of complicated parts, such as molds, art sculptures, and impellers [1]. A multi-stage machining starts from a blank and converges to the eventual design surface of the part, which can be roughly categorized into rough and finish machining. For rough machining, a series of intermediate machining layers (MLs) are defined in the volume between the raw stock and the finishing part, on which the tool paths are then planned. After the rough machining, the processed workpiece becomes a near-net-shape, and the residual material on this near-net-shape is removed by a final cut (i.e., the finish machining) with a much conservative depth-of-cut for better surface quality.

Conventionally, in the rough machining stage, the material is removed by the so-called Z-level method on a three-axis machine tool, wherein the volume to be machined is sliced by equal interval planes perpendicular to the Z-axis. However, for complicated shapes, this parallel planar pattern lacks flexibility and causes isolated islands on some layers, which require extraneous air movements by the tool. Moreover, the in-process workpiece machined according to the Z-level method has a staircase effect, which often induces unstable dynamic problems on the tool-spindle system.

To resolve the issues of the Z-level method, some surface-based methods such as offsetting [2–5] or morphing [6–12] are suggested for rough machining. These methods are performed on five-axis platforms because curved interface layers need more degrees-of-freedom for a complicated workpiece. Zhu et al. [2] proposed an offsetting-based tool path generation method for rough machining on a mesh surface model. The 3D morphing technique proposed by Lefebvre and Lauwers [8,9] can be utilized to generate the intermediate surfaces of rough machining, where the model is represented by the boundary representation data structure (BREP)

instead of a mesh. Some other process planning methods have also been proposed for complicated parts, targeting certain specialized applications such as the machining of impellers and blisks [13,14]. Unfortunately, these process planning methods are not customized for complicated features like concave or genus-*n* structures in general, where collision and other types of geometric issues may occur and become more difficult to handle. Specifically, for the offsetting method, the self-intersection of tool paths or intermediate surfaces posts to be a difficult geometric problem to resolve; while for various morphing based methods, redundant tool paths typically are generated and the depth-of-cut is nonuniform.

As machining or printing is a volumetric operation, recently volume-based methods for process planning have also been proposed. For example, Dai et al. [15] proposed a multi-axis support-free volume printing method based on the voxel model, where the concept of *accumulation field* for the printing model is established and the printing layers are obtained by slicing this field. On the contrary, in our recent work [16], we presented a voxel-based machining process planning method by constructing a *subtraction field* to reflect the material removal process and generate the intermediate machining layers. Another voxel-based CAM system reported in Refs. [17–23] obtained the intermediate layers by expanding the boundary of the part's voxel model and then planned tool paths on these intermediate surfaces. Although this system is able to handle some complicated parts and the self-intersection issue is mostly circumvented, the generated intermediate surfaces are often inaccessible to the tool for deep concave features. In general, the voxel model provides a convenient channel for model processing. However, as it is a zero-order approximation of model, it needs much more computation time and storage for geometry representation than some other volumetric representations such as a tetrahedral mesh. Additionally, the boundary of the voxel model must be smoothed before the tool paths can be planned. Also, in a voxel model, as layers are generated by tracing adjacent voxels, their thickness is restricted by the size of the voxel and the variation of thickness could be significant.

In this paper, as inspired by the concept of subtraction field from our recent work [16], we propose an algorithm for generating

---

machining layers based on the subtraction field as well as their accompanying five-axis tool paths for machining an arbitrary complex solid part from a raw blank to the eventual design surface. Fundamentally different from Ref. [16] though, while in Ref. [16] the subtraction field is constructed on a voxel model, we construct the subtraction field by establishing a geodesic distance field (GDF) embedded on the tetrahedron mesh of the volume to be removed. The high-order continuity of the geodesic distance field ensures that the generated machining layers will be smooth (as they are the iso-surfaces of a field), thus eliminating the need for the computationally expensive smoothing operation on every machining layer before its tool path is planned. Moreover, the proposed method enjoys two distinct advantages over the voxel-based approach [16]:

(i) The voxel size in Ref. [16] is determined by the minimum depth-of-cut, which could be very small when compared with the overall size of the part (e.g., 0.05 mm versus 400 mm), thus requiring an extremely large amount of computer RAM to run the computer program, which may be handicapped by the limited RAM of the computer. The proposed method does not have this problem as it is independent of the depth-of-cut.

(ii) In the voxel-based approach, the intermediate machining layer is obtained by interpolating the two jagged voxel layers into a smooth surface, whose thickness is obviously nonuniform, as illustrated in Fig. 1(a). In the contrast, our method first constructs a smooth geodesic distance-based field, as shown in Fig. 1(b), in which the vertices that have equal iso-values (black dots) are then interpolated to form iso-surfaces (red and blue curves), whose uniformness in layer thickness is ensured.

The paper is organized as follows. In Sec. 2, first, we give a detailed account of how to construct a geodesic distance field in a tetrahedral mesh based on the heat diffusion method. Next, we describe how to efficiently compute a polyhedral isosurface in a geodesic distance field. In Sec. 3, we present our two-stage machining strategy—first rough machining the volume between the raw blank and the convex hull (CH) of the part and then rough + finish machining the volume between the convex hull of the part and the design surface, with the curved machining layers for the two stages exactly being the iso-surfaces of the established geodesic distance field (the subtraction field). Then, we present our algorithm of generating collision-free five-axis tool paths for machining the curved layers, with a new and core idea of alternating between roughing and finishing machining, so to improve the stiffness of the in-process workpiece. Finally, the experimental results are presented in Sec. 4, followed by the conclusion in Sec. 5.

2 Volume Slicing Based on Geodesic Distance Field

To slice a volume into a set of MLs, we will define a subtraction field, whose iso-surfaces naturally represent a sequence of machining. In this paper, based on the tetrahedral modeling

representation, a geodesic distance field is established to facilitate the construction of this subtraction field. First, the CH and a safe box (SB) of the part are constructed as the boundary surface for tetrahedralization. Then, this tetrahedral volume is decomposed into machining layers based on the geodesic distance field defined on the volume. Similarly, we define the geodesic distance field between CH and the part surface and generate the machining layers accordingly. The pipeline of the algorithm is illustrated in Fig. 2, while the details will be provided in the following sections.

2.1 Division of the Volume and the Two-Stage Machining Process. The key to tetrahedralization is to decide a boundary surface to form a watertight volume. A common method is to directly use a bounding box (BB) of the part and its design surface for tetrahedralization and slicing. However, as the original BB is tangent to the part, it would induce separated volumes and even non-manifold edges, making it difficult to correctly slice the part. Thus, the original BB is offset by a sufficiently small distance to form a SB which defines a connected and orthogonal volume that strictly contains the design surface, as shown in step (1) in Fig. 2. Nevertheless, the machining layers generated in the volume between the safe box and the design surface are irregular if the subtraction field starts from the safe box. This is because the machining layers are only morphed from the box, without considering the shape of the part. On the other hand, if the subtraction field starts from the design surface, the tool accessibility becomes a major concern in the concave regions of the part.

To resolve these two mutually conflicting issues, we propose to insert an intermediate surface where the subtraction field starts, so as to balance between the orthogonal shape of SB and the freeform design surface. The CH of the part is then a natural candidate and adopted by us. The intermediate surface CH divides the volume between SB and the design surface into two connected volumes, i.e., volume V_{out} between SB and CH and volume V_{in} between CH and the design surface. Accordingly, we will use two different stages to machine the entire part, i.e., the first stage for machining volume V_{out} which will be entirely rough machining and the second stage for machining volume V_{in} which will include both rough and finish machining.

2.2 Construction of the Geodesic Distance Filed and Its Induced Subtraction Field. As mentioned earlier, our subtraction field is based on a geodesic distance field defined in the volume. Before we begin the construction of the geodesic distance field, it is assumed that the two volumes V_{out} and V_{in}, i.e., the one between the safe box and CH and the other between CH and the design surface, are already tetrahedralized, as shown in Figs. 2(b) and 2(e). In our implementation, we used a built-in tetrahedralization function in MATLAB to realize this task. We use Crane's heat method [24] to construct the geodesic distance field, where the geodesic lines start from a heat source intuitively (a.k.a. the boundary condition of the heat diffusion equation). In our case, the heat source is exactly the convex hull of the part, i.e., CH.
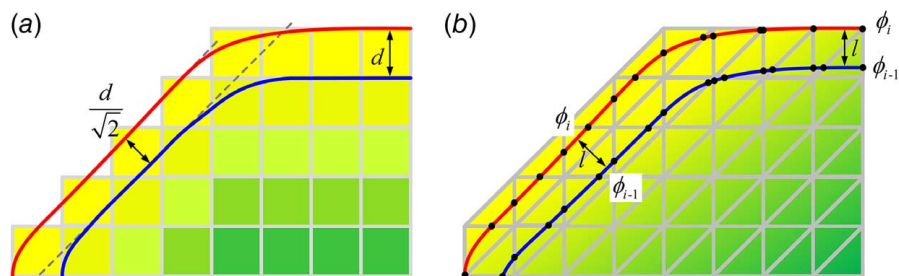

Fig. 1 2D illustration of (a) discrete voxel-based subtraction field in Ref. [16] and (b) the proposed smooth geodesic distance-based field with respective iso-surfaces (represented by the red and blue curves) (Color version online.)
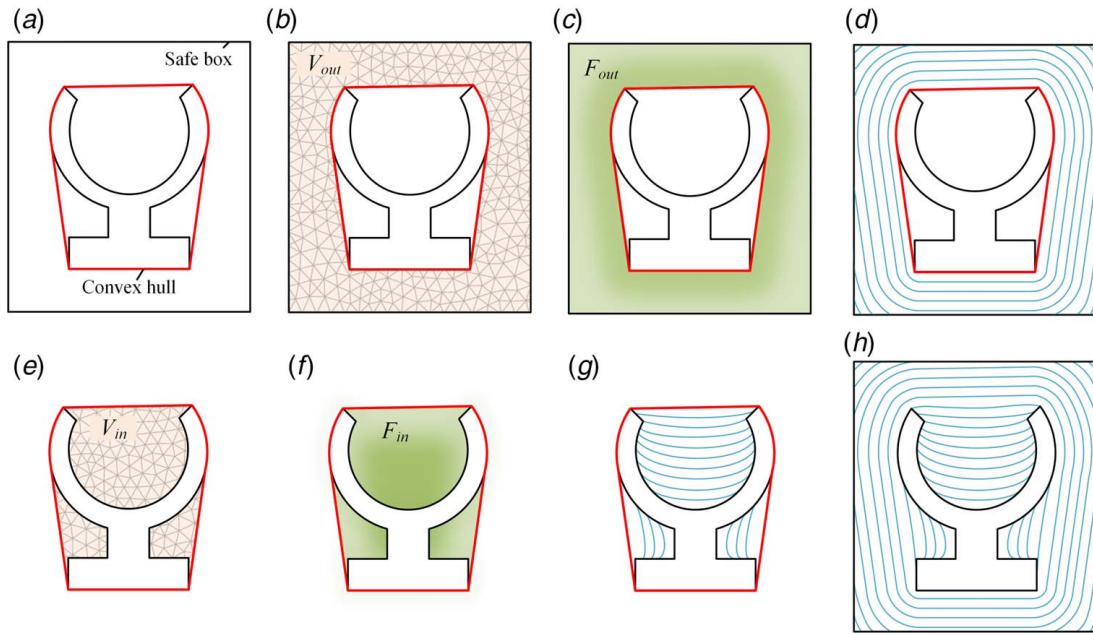
Fig. 2 The pipeline of the proposed volume slicing method: (*a*) boundary surfaces, (*b*) tetrahedral V_{out}, (*c*) subtraction field F_{out}, (*d*) sliced layers in F_{out}, (*e*) tetrahedral V_{in}, (*f*) subtraction field F_{in}, (*g*) sliced layers in F_{in}, and (*h*) merged machining layers

To begin with, a temperature scalar field $u(x, y, z)$ is established by solving the following heat diffusion equation:

$$\rho \frac{\partial u}{\partial t} = \Delta u \qquad (1)$$

where $\rho(x, y, z)$ is a nonuniform density function that is to be determined depending on the position, so to attain more freedom for the shape of the geodesic distance field. For example, in a homogeneous concave region, as the speed of heat flow is uniform, if $\rho(x, y, z)$ is constant, the iso-surfaces would fail to follow the shape of the design surface, as shown in Fig. 3(*a*). On the other hand, if $\rho(x, y, z)$ is large near the boundary, the heat flow near the boundary would be slower and the iso-surfaces would be bent. These resulting MLs will then have a larger tool accessibility and make full use of five-axis milling.

For numerical computation, the heat diffusion equation (1) is discretized by the backward Euler method:

$$(\mathbf{I} - t\mathrm{P}^{-1}\Delta)\boldsymbol{u}_t = \boldsymbol{u}_0 \qquad (2)$$

where $\mathbf{I} \in \mathbb{R}^{|V|\times|V|}$ and $\mathbf{P} \in \mathbb{R}^{|V|\times|V|}$ denote the identity matrix and a diagonal matrix containing the vertex density ρ_i, respectively; $\boldsymbol{u}_0$ is the initial temperature field vector; and $\boldsymbol{u}_t$ is the temperature field vector at time instant t. In order to solve Eq. (2) on a tetrahedral mesh, a discrete Laplacian operator $(Lu)_i$ for vertex i is needed

$$(Lu)_i = \frac{1}{V_i} \sum_{j \in N(i)} w_{ij}(u_i - u_j) \qquad (3)$$

where V_i is the vertex volume that is one-fourth the volume of all the tetrahedrons incident on vertex i, $N(i)$ is the set of vertices immediately adjacent to vertex i, and w_{ij} is a scalar weight assigned to edge (i, j). We can rewrite the discrete Laplacian operator in Eq. (3) as the Laplacian matrix $\mathbf{L}$ as

$$\mathbf{L} = \mathbf{V}^{-1}\mathbf{L}_c \qquad (4)$$

where $\mathbf{V} \in \mathbb{R}^{|V|\times|V|}$ is a diagonal matrix containing the vertex volumes; $\mathbf{L}_c \in \mathbb{R}^{|V|\times|V|}$ is the cotangent operator whose elements
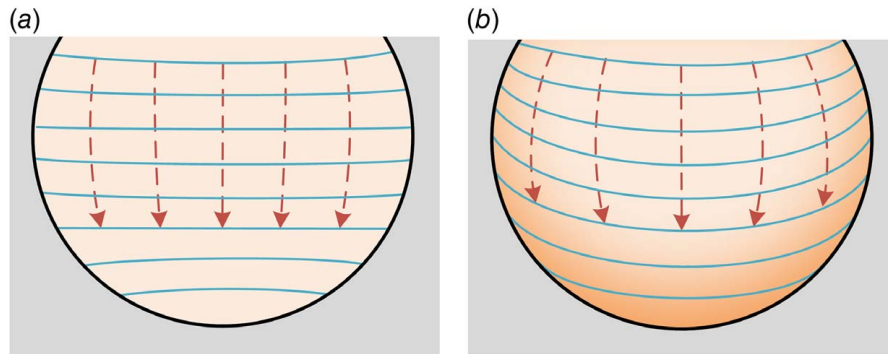

Fig. 3 The density distribution (orange colour), the heat flow (arrows) and iso-surfaces (blue lines): (*a*) homogeneous solid and standard iso-surfaces and (*b*) large density near the design boundary and bent iso-surfaces (Color version online.)
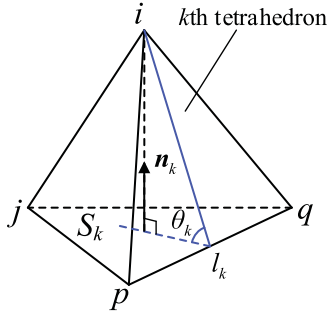
Fig. 4 Illustration of the discrete volumetric Laplacian weight in tetrahedron T_{ijpq}; θ_k is the dihedral angle of edge l_k; S_k and n_k are the area and the normal vector of the face jpq, respectively

L_{ij} can be expressed as

$$L_{ij} = \begin{cases} -\sum_{v_k \in N(i)} w_{ik}, & i = j \\ w_{ij}, & j \in N(i) \\ 0, & \text{otherwise} \end{cases} \qquad (5)$$

The remaining work is to calculate the weight defined on edge (i, j), w.r.t. its m adjacent tetrahedrons [25]

$$w_{ij} = \frac{1}{6} \sum_{k=1}^{m} l_k \cot(\theta_k) \qquad (6)$$

where k is the index of the tetrahedron adjacent to edge (i, j); for the kth tetrahedron, l_k is the length of the opposite edge (p, q) to which edge (i, j) is against, while θ_k is the dihedral angle of edge (p, q), as illustrated in Fig. 4.

Combining Eq. (2) with Eq. (4), we have

$$(\mathbf{I} - t\mathbf{M}^{-1}\mathbf{L}_c)u_t = u_0 \qquad (7)$$

where $\mathbf{M} = \mathbf{VP} = \mathbf{PV}$ is the diagonal matrix containing the mass at each vertex i. Now, the temperature field u_t can be obtained by solving Eq. (7), while the time-step t is estimated by $t = h^2$, where h is the average length of edges [24].

After the calculation of the temperature scalar field u, the temperature gradient ∇u inside any tetrahedron (i, j, p, q) is calculated by [25]

$$\nabla u = \begin{bmatrix} (v_i - v_q) \\ (v_j - v_q) \\ (v_p - v_q) \end{bmatrix}^{-1} \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} u_i \\ u_j \\ u_p \\ u_q \end{bmatrix} \qquad (8)$$

where v_i, v_j, v_p, and v_q are the vertices' coordinates. The key to the Crane's heat method [24] is to normalize the gradient ∇u to an $\mathbf{X}$ as the vector field of geodesic distance in a three-manifold:

$$\mathbf{X} = \frac{\nabla u}{|\nabla u|} \qquad (9)$$

Finally, the geodesic distance field $\phi \in \mathbb{R}^{|V| \times 1}$ for all the vertices can be obtained by solving the Poisson equation

$$\mathbf{L}_c \phi = \nabla \cdot \mathbf{X} \qquad (10)$$

For practical implementation, the above-integrated divergence of the obtained vector field $\mathbf{X}$ with vertex i is discretized by

$$\text{Div}(\mathbf{X}) = \nabla \cdot \mathbf{X} = -\frac{1}{3} \sum_{k \in N(i)} S_k n_k \cdot \mathbf{X} \qquad (11)$$

where $k \in N(i)$ denotes the set of tetrahedrons immediately adjacent to vertex i, and S_k and n_k are the area and the normal vector of the opposite triangular face to vertex i in the kth tetrahedron, respectively, as shown in Fig. 4.

This geodesic distance field ϕ is thus exactly the sought subtraction field. Specific to our case, by setting the convex hull CH as the heat source and propagating the heat outwards and inwards separately, we obtain two subtraction fields F_{out} and F_{in}, as illustrated in Figs. 2(c) and 2(f), which are, respectively, embedded in the volume between the safe box SB and the convex hull CH and that between CH and the design surface.

2.3 Construction of the Iso-Geodesic Distance Surfaces.
After the GDF is obtained on the tetrahedrons' vertices, a series of iso-surfaces of GDF can be interpolated to slice the target volume. For an arbitrary edge (i, j) with two vertices v_i and v_j on a tetrahedron and a query iso-geodesic distance ϕ, an interpolation point exists on the edge if and only if the geodesic distances at the two vertices satisfy

$$(\phi - \phi_i)(\phi - \phi_j) < 0 \qquad (12)$$

where ϕ_i and ϕ_j are the field values of v_i and v_j, respectively. The unique corresponding vertex on edge (i, j) that realizes Eq. (11) is

$$v = (|\phi_i - \phi|v_j + |\phi_j - \phi|v_i)/|\phi_i - \phi_j| \qquad (13)$$

Then, all the valid edges that meet Eq. (11) can be identified and their corresponding vertices v's collected.

Besides this vertex list, we also need a list of faces that will connect the interpolation points into triangles. According to the adjacent relationship between edges and tetrahedrons, we can select all the involved tetrahedrons. Each valid tetrahedron may contain either three interpolation points (Fig. 5(a)) or four interpolation points (Fig. 5(b)). In the first case, one triangular face is uniquely defined in one tetrahedron. In the second case, however, two triangular faces are defined by adding an edge along one of the two diagonals of the quadrilateral. Therefore, the face list can be obtained by visiting each valid tetrahedron.

Conceivably, the sizes of the thus obtained interpolating triangles are in general nonuniform. We then apply the well-known isotropic remeshing algorithm [26] to improve the mesh quality, wherein the boundary conformation constraint is introduced to preserve the boundary of the iso-surfaces.

By performing the above isosurface generation procedure in both subtraction fields F_{out} and F_{in}, we will have generated MLs in both volumes V_{out} and V_{in}, as illustrated in Figs. 2(d) and 2(g), which actually can be merged into a single partially ordered sequence of MLs, as shown in Fig. 2(h), which begins at the safe box and ends on the design surface. Finally, as the actual raw blank is usually a right BB that is strictly inside the safe box, the MLs in V_{out} that are beyond the raw stock BB will be removed or trimmed. Figure 6 shows some cross-sectional views of the heat graph of GDFs F_{out} and F_{in} generated by the algorithm given in Sec. 2.2, as well as the MLs trimmed by BB. Originally, the geodesic distance in the first stage starts from CH, which means the geodesic distance value of any point near CH is very small. However, to reflect the actual machining sequence, the GDF of F_{out} is flipped
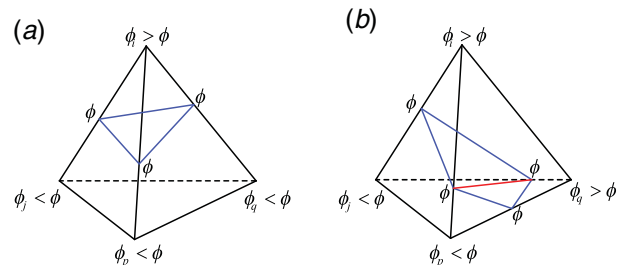

Fig. 5 Illustration of interpolation points on the edges of a tetrahedron and their triangular faces: (a) three interpolation points and (b) four interpolation points
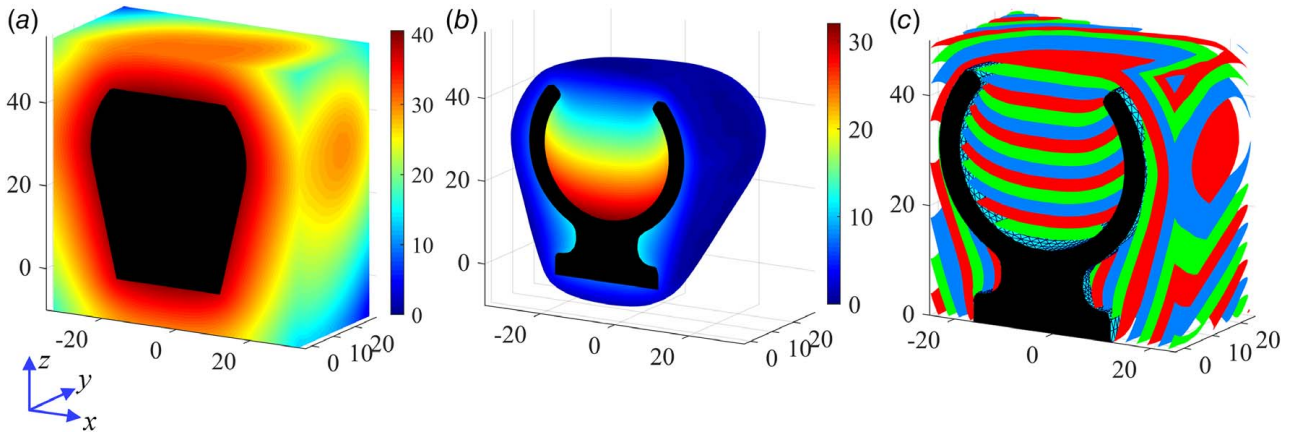
Fig. 6 The geodeic distance filed of (a) F_{out}, (b) F_{in}, and (c) is the sliced machining layers with different colors

and starts from the outside (i.e., the raw stock surface BB), as shown in Fig. 6(a).

3 Machining Process Planning

With the help of the established geodesic distance fields, we have now defined a partially ordered series of MLs that can then be used to machine the part from the raw stock BB to its design surface. The remaining task is then to plan a proper machining process that will mainly comprise the following sub-tasks: (1) determine the sequence of machining; (2) for each ML, generate a suitable sequence of cutter contact (CC) curves; and (3) for every sample point on each CC curve, select a collision-free tool orientation.

3.1 Machining Sequence Planning. In Sec. 2, we have generated a partially ordered sequence of MLs, as illustrated in Fig. 2(h). Of these MLs, only those that overlap with the final design surface will belong to the *finish machining* operation, while the rest (the majority) belong to the rough machining operation. Actually, only the second stage will involve finish machining while the first stage is solely for the rough machining. As machining is a time-continuous process, we must assign a total ordering among the MLs. Specific to the second stage, the sequence of MLs is actually not unique. Conventionally, finish machining is performed at last after all the rough machining is completed. Nevertheless, for a thin-walled workpiece, after the rough machining, the near-net-shape will have very low stiffness, especially at the free end, which will induce large deformation or even chattering. To mitigate this problem, a new machining strategy [16] is employed in the second stage: the rough machining and finish machining are performed alternatingly from CH to the design surface.

This alternating sequence is determined as follows. As schematically illustrated in Fig. 7, some regions of the design surface will emerge (the red regions) after the rough machining by one or more machining layers. At this point, e.g., Fig. 7(a), the stiffness of the in-process workpiece is relatively higher than that of the finished part. Thus, these red regions will be finish machined in advance before the next round of rough machining (e.g., Figs. 7(b)–7(d)). By adopting this alternating strategy rather than waiting for the entire rough machining operation to complete before the finish machining, we aim at increasing the stiffness of the in-process workpiece, as to be validated by our experiments in Sec. 4.

The planning of the entire sequence of rough and finish machining is illustrated in Fig. 8. In the first stage, all the MLs perform rough machining, starting from the outmost layer r_1 inwards and ending on the intermediate layer $r_i = $ CH. Importantly, no tool collision check is necessary since all the MLs are convex and outside CH. Then, in the second stage, suppose that the design surface is
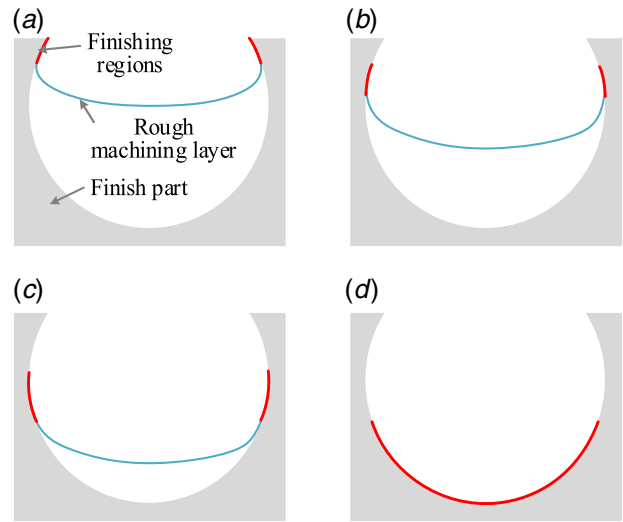

Fig. 7 Finish machining regions corresponding to each machining layer (2D illustration)

first exposed at ML r_j, after the rough machining of ML r_{j+1}, the finish machining layer f_1 is inserted between r_{j+1} and r_{j+2}. Similarly, the finish machining layer f_2 is inserted between the next two rough MLs, and so forth.

It must be pointed that, although the MLs for rough machining (e.g., the blue regions in Fig. 7) are already available, that for finish machining in the second stage (e.g., the red regions in Fig. 7) are not known beforehand and hence need to be segmented from the design surface. Fortunately, the geodesic distance of each vertex of V_{in} has been calculated and the inner boundary surface of V_{in} is precisely the design surface. Refer to Fig. 9, we can group each triangle on the design surface easily by checking its vertices' geodesic distances. The facets whose vertices' geodesic distances fall in $[\phi_{i-1}, \phi_i]$ will be selected for constructing the current finish machining layer f_i, which will then be arranged in the alternation sequence, as shown in Fig. 8.

3.2 Tool Path Planning on Machining Layers. Now that we have determined a total ordering of MLs for both rough and finish machining, the final task is then to generate a proper five-axis tool path to machine each of these MLs. As stated in the previous section, the machining layers for rough machining are sliced by the heat-based GDF embedded in the divided volumes V_{in} and V_{out}. For consistency and generality, the CC curves of a tool path on an ML are represented by iso-geodesic distance contours
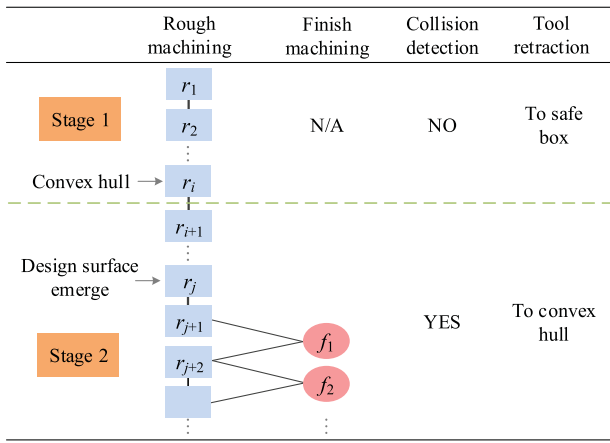
Fig. 8 The alternating sequence between rough and finish machining with the instructions of collision detection and tool retraction
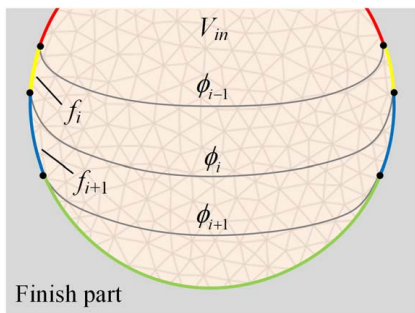

Fig. 9 Segmented patches f_i and f_{i+1} on the design surface for finish milling (2D illustration)

(IGDC) on the ML and these contours are also constructed by the heat-based method. The details can be found in Crane's paper [24]. The key to the path filling is to select the heat source on the surface of the ML where the GDF on the surface originates. The IGDCs are then constructed by interpolating the triangle mesh according to the obtained scalar GDF. This step is similar to Sec. 2.3, but the difference is that here we linearly interpolate triangles rather than tetrahedrons, and the eventual geometry to form is an embedded curve rather than a surface.

Due to the different machining requirements for roughing and finishing, we employ different machining strategies for the two. For any connected rough machining patch, as the cutting force is typically large, we want its CC curves to have nearly constant Z-coordinates, which will maximize the usage of the heavier and much more powerful linear XY table while minimize that of the much weaker B or C rotary tables. Accordingly, the heat source for generating the iso-geodesic contours is at the bottom of that individual patch (i.e., the lowest point of the patch is assigned with $\phi = 0$), as illustrated in Fig. 10(a). The smooth geodesic distance field on this individual rough machining patch is then constructed, and the IGDCs are interpolated according to the geodesic distance value ϕ to form a nearly parallel pattern, as shown Fig. 10(a). The IGDCs of different ϕ's are then sampled and connected to form a smoothed series of zig-zag CC curves, as shown in Fig. 10(b), where different colors represent different layers of rough machining patches. For the finish machining operation, since all the CC curves are on the design surface, we set the bottom of the entire workpiece as the heat source and simply propagate the IGDCs from the bottom to the whole surface only once, as shown in Fig. 11(a). Then, we allocate these closed IGDCs to each segmented finish machining patch (see Sec. 3.1), on which zig-zag CC curves are then connected, as shown in Fig. 11(b).

3.3 Tool Orientation Planning and Collision Avoidance. For any CC curve generated on a machining layer, regardless of rough or finish machining, our final task is to assign proper tool orientations to the sample points on the CC curve, thus forming a continuous five-axis tool path. A minimum criterion for the tool orientation is that the tool must not collide with the in-process workpiece or any fixture. In the first stage, as the MLs and CH are both convex, assuming only considering the in-process workpiece as the obstacle, the choice for the tool orientation is trivial—as long as the tool is above the tangent plane at the CC point. Nonetheless, considering certain minimum machining requirements, the following validity criteria are adopted by us for the selection of tool orientation at a CC point p:

(1) The tool axis should fall into an accessible conical ring, whose axis is the normal vector of the ML at p and the tile angle range is [α min, α max] (0 deg < α min < α max < 90 deg).
(2) Considering the physical constraints from the rotary table, the Z component of the tool axis (in the workpiece coordinate system) should be larger than a minimum positive value for safety reasons.
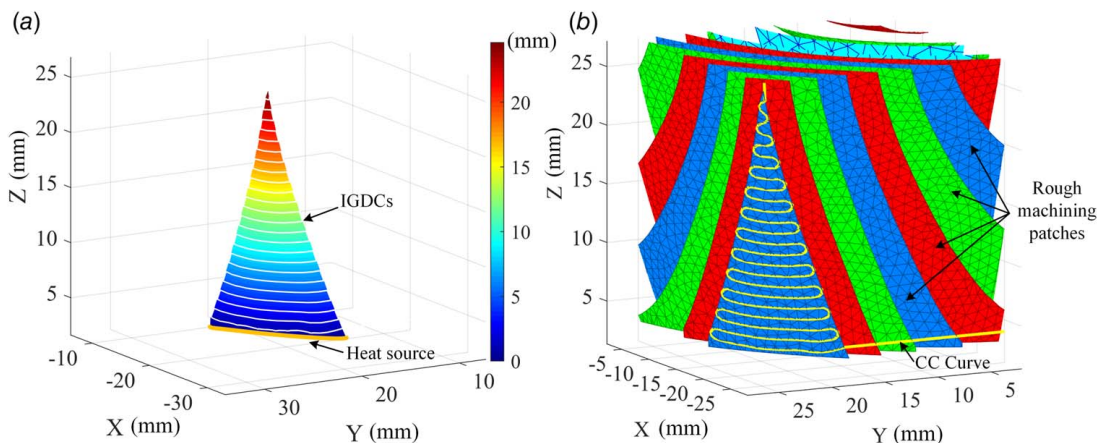

Fig. 10 Generation of tool paths on the rough machining patches: (a) the geodesic distance field on one of the patches with its IGDCs starting from the heat source at the bottom and (b) the trimmed sliced patches and one exemplary connected zig-zag CC curve (yellow). Different colors represent different layers of rough machining patches. (Color version online.)
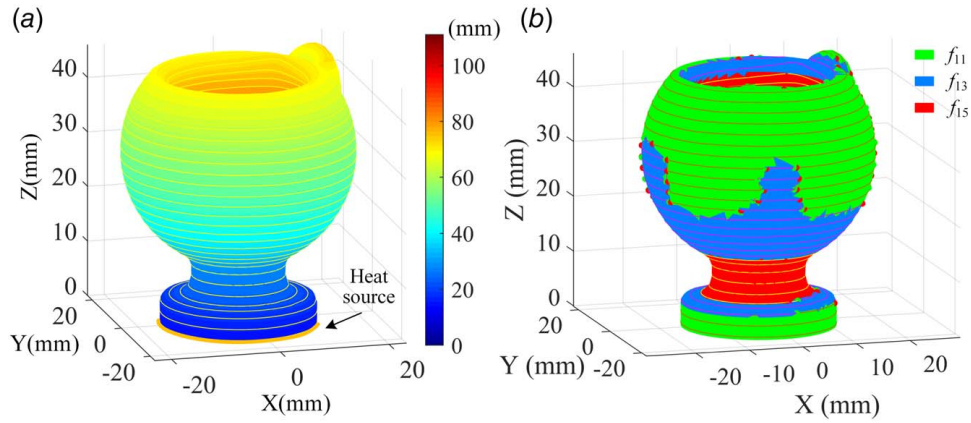
Fig. 11 Generation of tool paths on the finish machining patches: (*a*) the geodesic distance field on the design surface with its IGDCs starting from the heat source at the bottom and (*b*) the partitioned sub-surfaces f_{11}, f_{13} and f_{15} as well as the connected zig-zag tool paths. Different colors represent different segments of the finish machining regions.

(3) The angle between the tool axes of adjacent CC points should be capped by a threshold to ensure a smooth change of tool postures.

(4) With all the above conditions satisfied, the tilt angle α should be as close as possible to the prescribed optimal tilt angle α*.

In the second stage, however, apart from the aforementioned criteria, the collision avoidance must be considered since in general the in-process workpiece is concave. As collision avoidance is not the focus of this work, we will only consider the ball-end tool, which enjoys the huge benefit of the independence between the reference point of the tool (i.e., the center of the bottom hemisphere) and the tool axis, and the collision check can be converted to a line-visibility problem at the reference point of the tool.

In this paper, the tool orientation for each CC point is computed based on an improved greedy scheme [16], which avoids calculating all the accessible regions for every CC point, thereby saving a large amount of computing time. Let $\{p_1, p_2, \ldots, p_i, \ldots\}$ be the sample points on the current CC curve to be processed, with each p_i associated with a unique surface normal n_i of the machining layer. First, each CC point p_i is offset outward to a point c_i along n_i with a distance r, where r should be at least as large as the tool radius [1]. Then the initial tool orientation of the first cutter location (CL) point c_1 is assigned according to the validity criteria. For every ensuing CL point c_i, its tool axis T_i remains the same as the previous one as long as it passes the validity criteria (1) and (2) and the collision-free criterion (only for the second stage); otherwise, the current tool orientation should be locally updated. In this situation, a localized search is performed in a cone centered on the current tool orientation T_i with an angle of $\beta < \alpha_{max}$, whereas the search procedure is implemented by visiting points that are uniformly sampled on part of a hemisphere [27]. This can reduce the computation cost as compared with the search in the entire accessible conical ring (i.e., validity criterion (1)). After the localized search, among the candidate orientations, we simply pick the one closest to the optimal tilt angle α*.

Tool retraction is always required when the tool moves from one ML to another, or sometimes (though very rare) between consecutive zig-zag CC curves on the same ML. The specific strategy for deciding the tool retraction between two CC points P_a and P_b depends on the stage of the machining. For the first stage, the tool is retracted to the safe box, as schematically shown in Fig. 12: the tool retracts from the current CC point P_a along the tool axis T_a, intersects one of the safe planes at P'_a, moves in this safe plane to cross over the obstacle, reaches a relay point P'_b, and finally approaches the target CC point P_b. For the second stage, the retraction path is iteratively refined for the tool to collision-freely reach CH. As illustrated in Fig. 13, the lines of tool axes T_a and T_b intersect with CH at P'_a and P'_b, respectively. We connect them to form the segment $\overline{P'_a P'_b}$ and let P_c be the
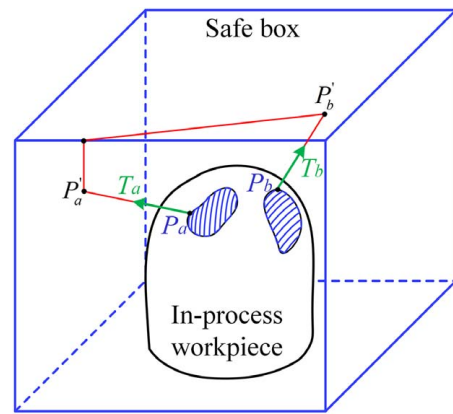

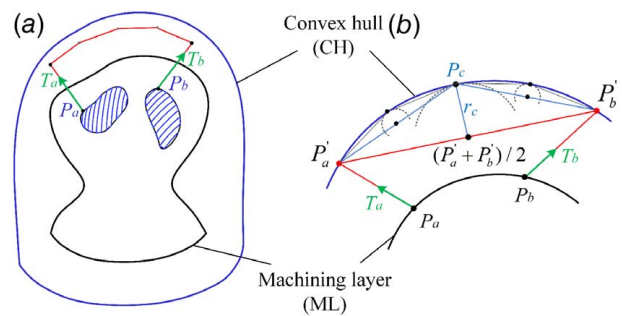**Fig. 12 Retraction path in the first stage between the in-process workpiece and the safe box**

Fig. 13 Retraction path in the second stage: (*a*) convex hull of the part and (*b*) the refinement of retraction path along the hull (2D illustration)

vertex on CH that is closest to the midpoint $(P'_a + P'_b)/2$. This is equivalent to a sphere that is centered at the midpoint and internally tangent to CH at P_c; so, the shortest distance is the corresponding radius r_c. The line segment $\overline{P'_a P'_b}$ can be selected as part of the valid retraction path if it does not intersect CH and the shortest distance r_c is below the given threshold ε. However, if $\overline{P'_a P'_b}$ intersect CH or $max(r_c) > \varepsilon$, the midpoint $(P'_a + P'_b)/2$ will be *dragged* to the hull, which means point P_c will be inserted in the retraction path between P'_a and P'_b. The tool orientation at P_c can be simply the vector pointing from the midpoint to P_c after the validation check. Then, the two new segments $\overline{P'_a P_c}$ and $\overline{P_c P'_b}$ are introduced
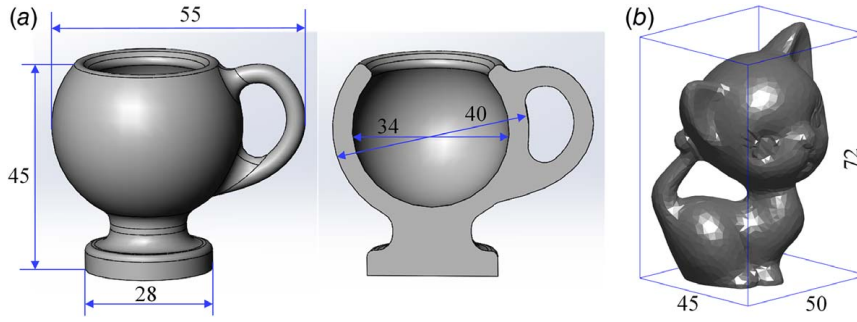
Fig. 14 The two sculptured test parts: (a) the cup model and (b) the kitten model

and a new round of iteration is performed until the segment no longer intersects CH and $max(r_c) \leq \varepsilon$. The entire procedure of tool retraction path determination is outlined in Algorithm 1.

Algorithm 1 Tool retraction path determination algorithm

Algorithm 1　　Tool retraction path determination.

Input: the convex hull CH, the safe box SB, the CC points P_a and P_b, and the tool orientations T_a and T_b
Output: The retraction path list P
1　Determine the machining stage = 1 or 2
2　**if** stage == 1 **then**
3　　$P'_a, P'_b = RayMeshIntersect$ (SB, $[P_a, P_b]$, $[T_a, T_b]$)
4　　$P = MoveOnBox$ (SB, P'_a, P'_b)
5　**End**
6　**if** stage == 2 **then**
7　　$P'_a, P'_b = RayMeshIntersect$(CH, $[P_a, P_b]$, $[T_a, T_b]$)
8　　$Insert$ (&P, $[P'_a, P'_b]$)
9　　**while**
10　　　$P_m = GetMidpointList$ (P)
11　　　$P_c, r_c = FindNearestPointList$ (CH, P_m)
12　　　**if** $RayMeshIntersect$ (CH, $P[0:-2]$, $P[1:-1]-P[0:-2]$) ! $= \varnothing$ ||
　　　　$max(r_c) > \varepsilon$
13　　　　$Insert$ (&P, P_c)
14　　　**else**
15　　　　**break**
16　　　**end**
17　　**end**
18　**end**

---

4 Experimental Validation and Discussion

The proposed methodology is tested on three exemplary parts. To assess the generality of the proposed methodology, we select two sculptured parts with complicated features such as concave cavities and of genus-one. The third test part is a thin-walled blade, which is used to demonstrate the capability of increasing the stiffness of the in-process workpiece by our method. The algorithms of the proposed methodology are implemented by us in MATLAB and the computer program is run on a PC with Intel Core i7-4790 CPU @ 3.60 GHz. The physical machining experiments are performed on a five-axis machine tool of model JD GR200_A10H.

4.1 Machining of Two Complex Sculptured Parts.
Figure 14 shows the two sculptured test parts, both of which have concave cavities and are of genus-one. The machining parameters of the two parts are listed in Table 1. The generated CL paths and the actual machining processes of the two parts are shown in Figs. 15 and 16, respectively. The interval of the geodesic distance in the subtraction field is set to be 2 mm for the construction of iso-surfaces, while the path step-over distances for roughing and finishing are respectively set to be 1 mm and 0.5 mm. In Figs. 15(a) and

Table 1　Parameters of machining

Parameter	Value
Machine tool	JD GR200_A10H, BC table
Diameter of ball-end cutter (mm)	4
Path step over of the roughing machining (mm)	1
Path step over of the finishing machining (mm)	0.5
Nominal machining layer thickness (mm)	2
Blank material	Wood
Blank dimensions (mm)	Cup: $50 \times 50 \times 50$, kitten: $55 \times 50 \times 80$
Safe box dimensions (mm)	Cup: $60 \times 60 \times 60$, kitten: $65 \times 60 \times 90$
Number of tetrahedrons in V_{out}	Cup: 237,553, kitten: 342,932
Number of tetrahedrons in V_{in}	Cup: 78,686, kitten: 80,289
Maximum geodesic distance of F_{out} (mm)	Cup: 40.5, kitten: 45.7
Maximum geodesic distance of F_{in} (mm)	Cup: 32.0, kitten: 20.1
Mean tetrahedron edge length (mm)	Cup: 2.1, kitten: 2.1
Number of machining layers	Cup: 23, kitten: 25

16(a), we show a few sliced MLs generated in the first stage, which are gradually machined off from the raw blank to CH, and the corresponding tool paths. These tool paths contain some segments of tool retraction to the safe box SB in order to avoid potential collisions with the in-process workpiece. In the second stage, however, the tool retracts to the convex hull CH except for the start and end CL points, as shown in Figs. 15(b) and 16(b). Figures 15(c) and 16(c) show some snapshots of the physical machining processes, respectively, for the two test parts, which clearly demonstrate that, while all the machining in the first stage pertains only to roughing, in the second stage the roughing and finishing operations are performed alternatingly to shape the in-process workpiece to its eventual design surface.

The proposed method is then compared with the benchmarking Z-level method, in which a set of parallel planes are selected to form the MLs whose normal vectors align with the Z-axis and the interval is 2 mm, as shown in Fig. 17. It can be seen that the benchmarking method is unable to deal with concave regions where the surface normal vector changes drastically with respect to the Z-axis. Specifically, as shown in Figs. 17(a) and 17(c), because the tool is restricted to be above the current (planar) layer, it may be unable to avoid collision with the part (and other obstacles if any). In the contrast, in our method, as demonstrated in Figs. 17(b) and 17(d), because the iso-surfaces originate from the convex hull of the part, the collision-free zone for tool orientation is much enlarged. Another issue of the benchmarking method is the severe staircase effect on the in-process workpiece, which aggravates the cutting stability in the ensuing machining processes.
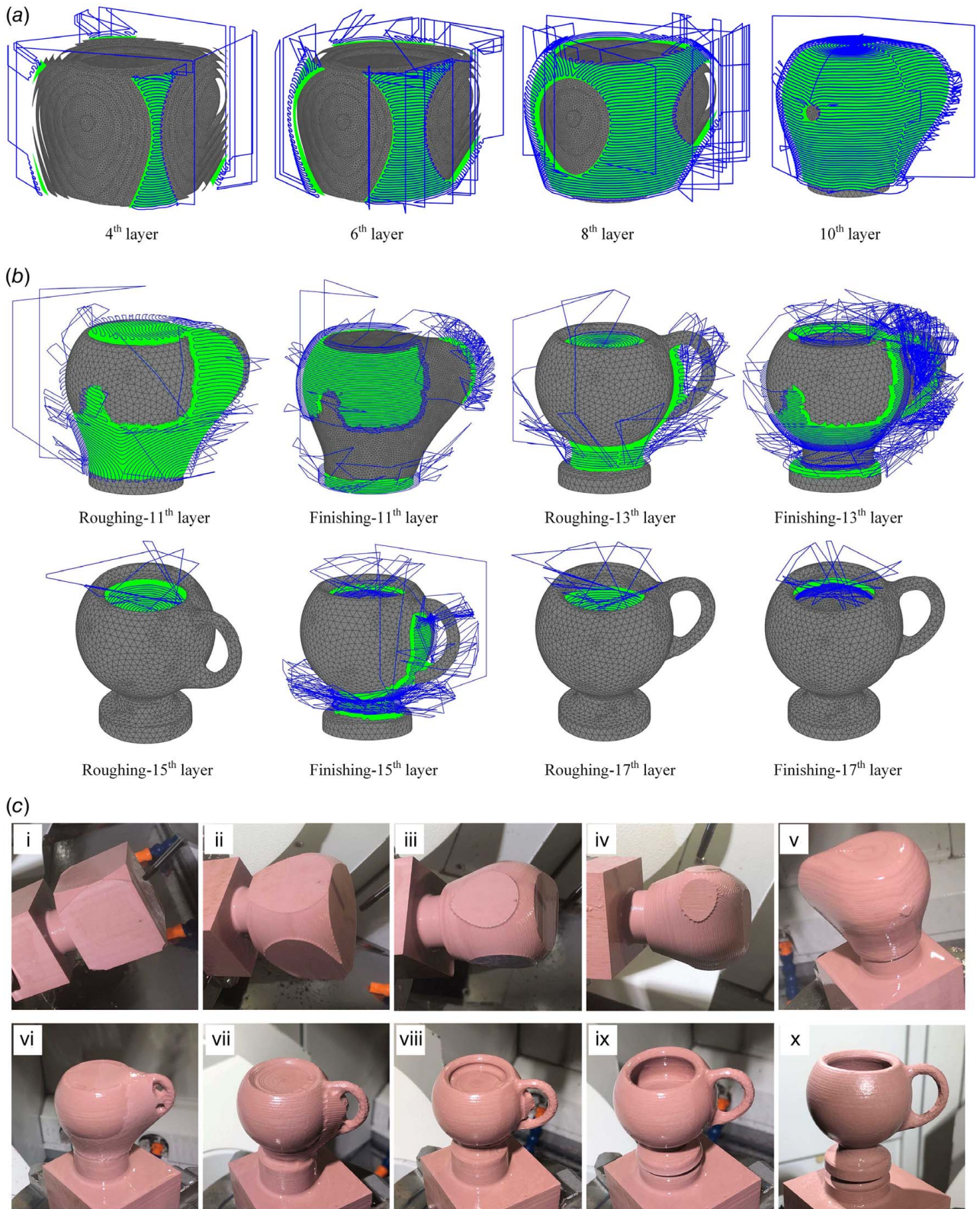
Fig. 15 Machining of the cup model: (*a*) CL paths of the first stage, (*b*) CL paths of the second stage, and (*c*) the actual machining process. Green surfaces are current machining layers. (Color version online.)

The time complexities of various operations of our algorithm as well as the actually recorded amounts of running time are listed in Table 2. The construction of the convex hull is based on the quick hull algorithm which has a time complexity of $O(n\log n)$ [28], where n is the number of input mesh vertices. The tetrahedralization is implemented using an MATLAB built-in function. The subtraction field generation algorithm described in Sec. 2.2 is based on a linear diffusion equation [24], so its time complexity is only $O(n)$, i.e., it is only linear with the number of vertices. Solving this linear system can be further expedited by means of parallel computing, as
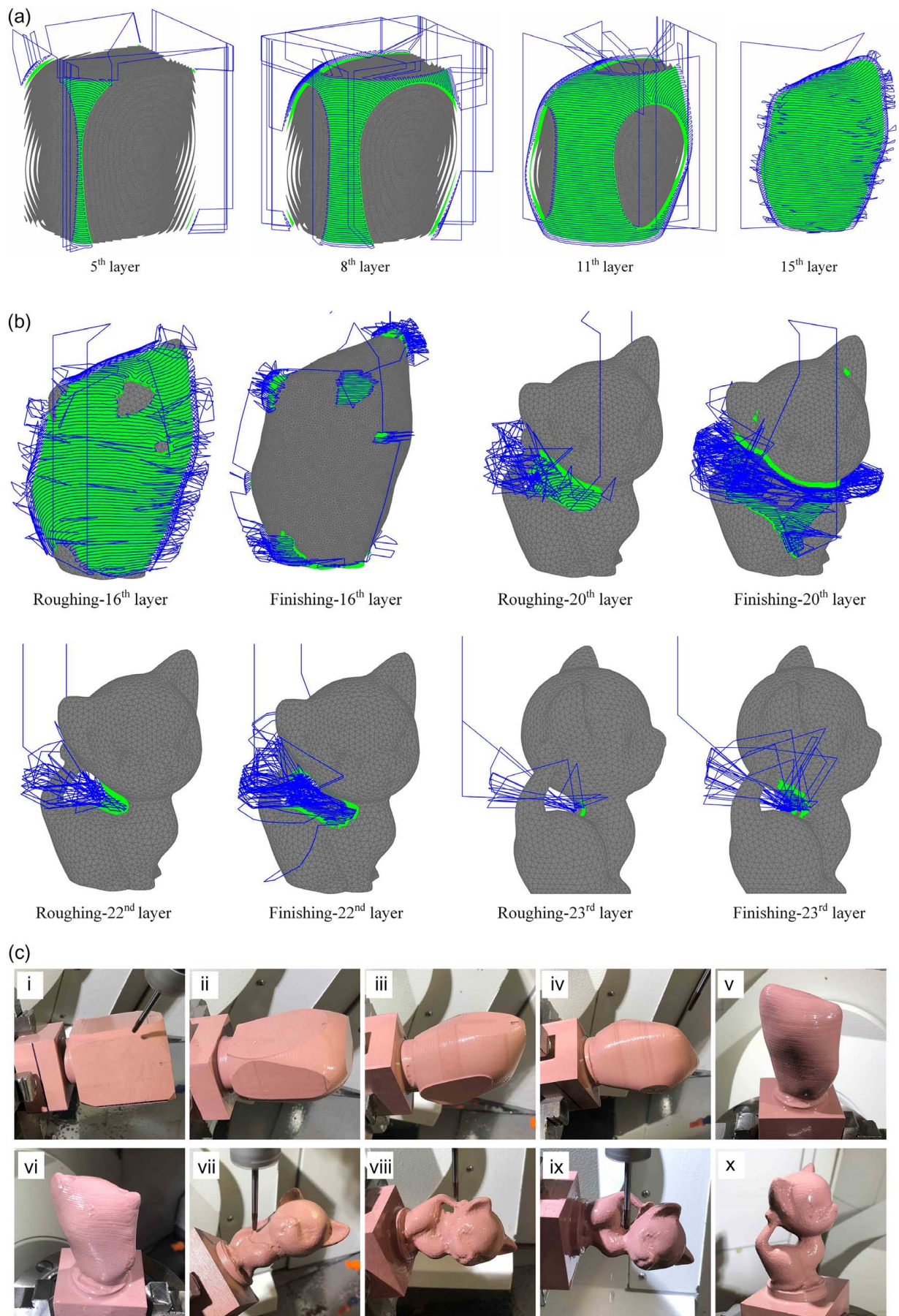
Fig. 16   Machining of the kitten model: (*a*) CL paths of the first stage, (*b*) CL paths of the second stage, and (*c*) the actual machining process. Green surfaces are current machining layers. (Color version online.)
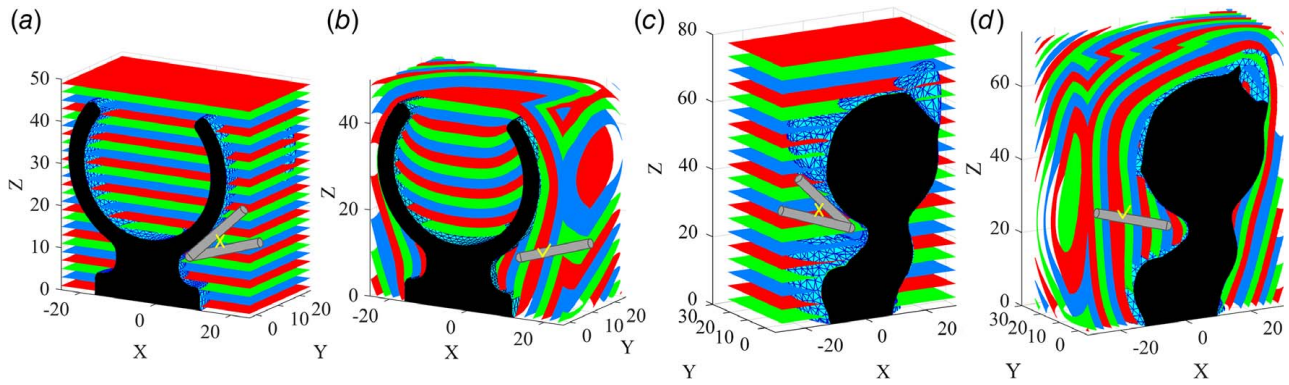
Fig. 17 Comparison between the benchmarking Z-level method (*a*) and (*c*) and our method (*b*) and (*d*)

compared to the voxel-based construction of the subtraction field in Ref. [16] which can only be achieved by sequentially traversing the voxel nodes. The time complexity of the algorithm of iso-surfaces interpolation is $O(n*m)$, where n and m are the number of layers and the average number of mesh triangles of each layer. The isotropic remeshing operation has the time complexity of at most $O(n^2)$, where n is the number of mesh triangles. For finishing surface partitioning, the time complexity is $O(n*m)$, where n and m are the number of segmented faces and the average number of mesh triangles of each face. The time complexity of the generation of iso-geodesic tool paths on each sliced ML is $O(n*m*k)$, where n, m, and k are the number of triangles, the number of CC curves, and the number of MLs, respectively. The process of tool orientation planning takes up the most running time (3806 s and 4253 s, respectively, for the two parts), which has the time complexity of $O(n*m)$, where n and m are the number of CC points and the number of mesh triangles, respectively. It should be noted that as our method can generate intrinsic smooth iso-surfaces, the smoothing operation (as required in Ref. [16]) is no longer needed.

Table 2 Time complexities of the algorithms and running time of the first two tests

Process	Time complexity	Running time (s)	
		Cup	Kitten
Convex hull generation	$O(n\log n)$	0.13	0.14
Tetrahedralization	MATLAB	Out: 3; in: 10	Out: 5; in: 11
Subtraction field generation	$O(n)$	Out: 6; in: 2	Out: 10; in: 2
Iso-surfaces interpolation	$O(n*m)$	Out: 12; in: 4	Out: 18; in: 4
Isotropic remeshing	$O(n^2)$	Out: 10; in: 3	Out: 15; in: 3
Finishing surface partitioning	$O(n*m)$	0.18	0.13
Tool path planning	$O(n*m)$	63	131
Tool orientation planning	$O(n*m)$	3806	4253

4.2 Machining of a Thin-Walled Blade. In the third test, a thin-walled blade model shown in Fig. 18 is machined based on both the proposed method and the conventional uniform depth-of-cut method (i.e., the benchmarking method), and the comparison is made. As shown in the cross-sectional views in Fig. 19(*a*), under the benchmarking method, the volume between the design surface and the raw blank is sliced into parallel MLs on both sides of the blade. This is accomplished by offsetting the design surface outwards by a uniform cutting depth. The widely used iso-planar method is then used to generate a *zig-zag* tool

path of each ML (Fig. 19(*c*)) to form the entire tool path as shown in Fig. 19(*d*). The layer thickness (i.e., the cutting depth) is 1 mm for the roughing and 0.5 mm for the final finish machining. Under the benchmarking method, in the roughing stage, the tool first removes the material from one side of the part and then switches to the other side. The final finish cutting is not performed until the entire rough machining is complete. On the other hand, under our method, as shown in the cross-sectional views in Fig. 20(*b*), the finish cutting is performed when the rough machining is only *partially* complete: in the second stage, finish machining
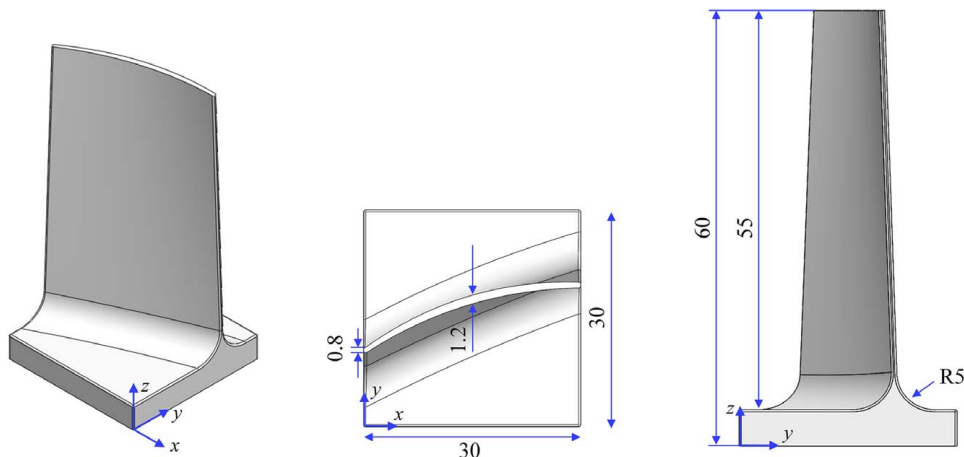

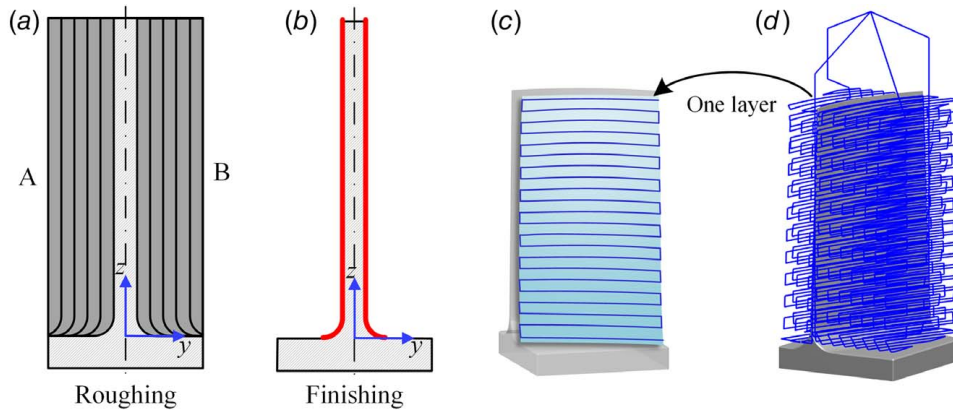**Fig. 18 Dimensions of the thin-walled blade model**

Fig. 19 Machining of the blade by the benchmarking method: (*a*) the sliced parallel MLs (cross-sectional view), (*b*) the finishing layer (cross-sectional view), (*c*) the iso-planar toolpath on one ML of (*a*), and (*d*) the entire toolpath of both roughing and finishing stage

operations (indicated by the red color) are inserted into rough machining operations. The main disadvantages of the benchmarking method are the large cutter-yield and the potential chattering, both due to the poor stiffness of the in-process workpiece. On the contrary, the MLs from our method consciously respect the intrinsic shape of the part (i.e., its convex hull), which results in a significant improvement of the stiffness of the in-process workpiece. The machining parameters and the pertinent statistics data of the two machining tests are given in Table 3.

Figures 21(*a*) and 21(*b*) show some photos of the actually machined parts according to our method and the benchmarking method, respectively, while some snapshots of the in-process workpiece by our method are given in Fig. 21(*c*), which correspond to the simulated result in the bottom row of Fig. 20(*b*). By examining Figs. 21(*a*) and 21(*b*), we observe that the machined blade surface by the benchmarking method has severe oblique chatter marks due to low stiffness while that by our method has a clear normal machined pattern. The ostensive curved gear marks on the blade surface in
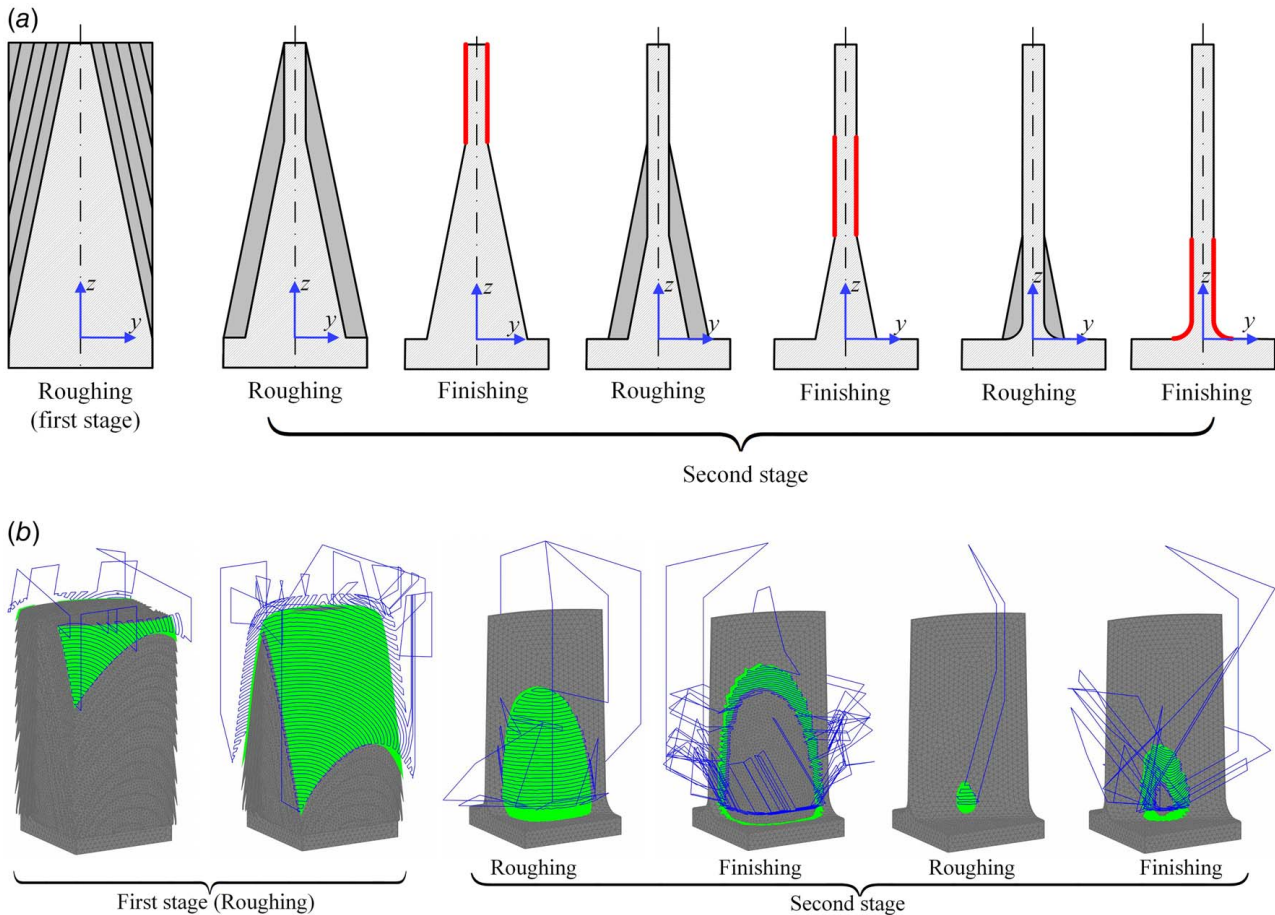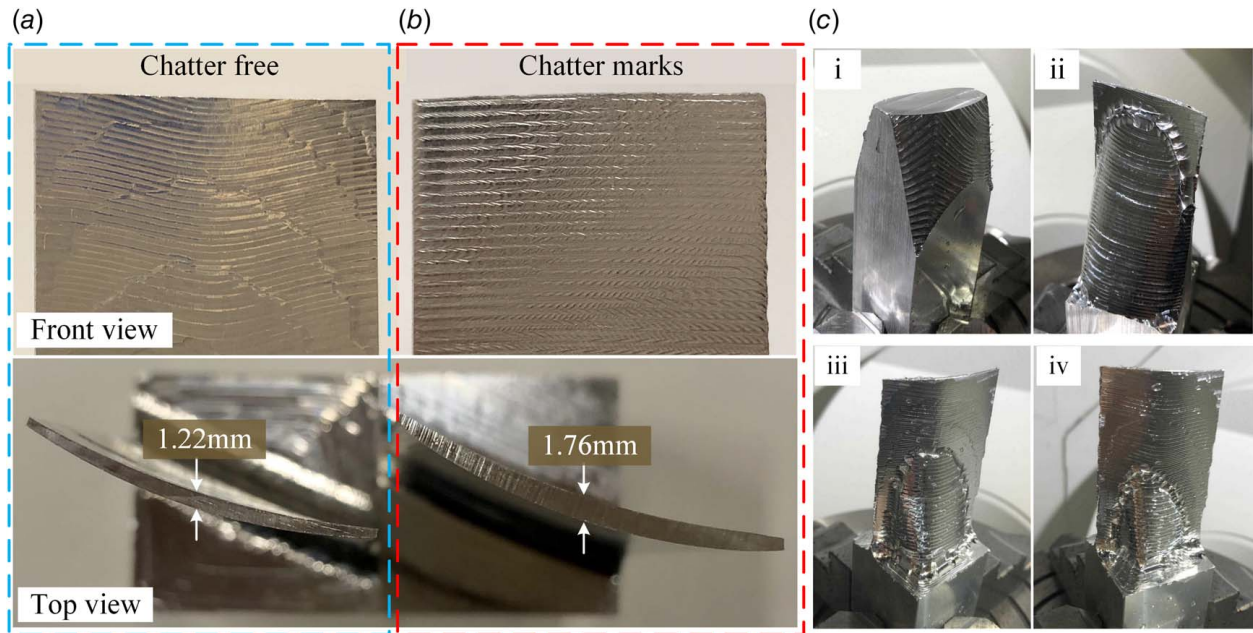

Fig. 20 Machining of the blade by the proposed method: (*a*) the sketch of the oblique MLs as well as the alternation sequence between roughing and finishing and (*b*) the actual CL paths of the cutting experiment

Table 3 Machining parameters and statistics data

Parameters	Benchmarking method	Our method
Blank material	Aluminum	Aluminum
Diameter of the ball-end cutter (mm)	6	6
Feed rate (mm/min)	300	300
Spindle speed (r/min)	6000	6000
Path step over for the roughing machining (mm)	1	1
Path step over for the finishing machining (mm)	0.5	0.5
Dimensions of the blank (mm)	$30 \times 30 \times 60$	$30 \times 30 \times 60$
Dimensions of the safe box (mm)	$40 \times 40 \times 80$	$40 \times 40 \times 80$
Depth-of-cut for the roughing machining (mm)	1	1
Depth-of-cut for the finishing machining (mm)	0.5	<1

Fig. 21 The machined blades: (a) by the proposed method, (b) by the benchmarking method (after repeating the finish machining operation twice), and (c) the in-process workpiece by our method

Table 4 The measured thickness and the cumulative cutting CL length

Parameters	Benchmarking method			Our method
	After 1st finishing (no repeat)	After repeat finishing once	After repeat finishing twice	
Thickness in the middle of the top edge (mm)	2.02	1.88	1.76	1.22
Cumulative cutting CL length (mm)	64,877	73,628	82,379	83,150

Fig. 21(a) have a clear explanation: those are exactly the boundaries of the segmented finish machining regions. In our cutting tests, to save the machining time, the step-over distance is set at 0.5 mm (for the finish machining). If a smaller step-over distance is used (as would be in any real machining), the gear marks will be much less obvious.

To quantify the effect of cutter-yield improved by our method, the thickness in the middle of the top edge of the machined blade is measured and compared with the design thickness 1.20 mm. As indicated in Fig. 21(a), our method obtains a decently accurate thickness (1.22 mm). On the contrary, for the benchmarking method, the result is much worse. Specifically, the measured thickness right after the finish machining is 2.02 mm, obviously due to

the severe cutter-yield. Following the common practice in machining, we then repeat once and then twice the finish machining operation (i.e., executing the same G-codes of the final finish machining tool path), and the measured thickness is, respectively, 1.88 mm and 1.76 mm (Fig. 21(b)). In Table 4, we list the measured thicknesses as well as the corresponding cumulative cutting CL lengths. It is worth pointing out that, in the case of repeating the finish machining operation twice by the benchmarking method, while the amounts of the total cutting time by our method and the benchmarking method are almost the same (83,150 mm versus 82,379 mm in cutting length), our machining accuracy is much higher than the benchmarking method (i.e., 1.22 mm versus 1.76 mm). Actually, under the benchmarking method, even after repeating three more times

of the finish machining operation, the measured thickness can only be lowered to 1.47 mm, still much worse than ours of 1.22 mm.

5 Conclusion

In this paper, we have presented a new process planning methodology for five-axis machining of parts with complicated features or weak structures. For the roughing operation, rather than simply taking Z-level parallel planes as intermediate machining layers, we have proposed to use curved machining layers, which will eliminate the severe staircase effect on the in-process workpiece, as suffered by the Z-level method, and hence significantly stabilize the cutting force during the machining and reduce the susceptibility to dynamic problems such as chattering. To facilitate the determination of the desired curved machining layers, we have given an elaborate algorithm of constructing a geodesic distance field embedded in the volume-to-remove whose iso-surfaces are then naturally used as the machining layers, which are assured to be smooth and have a uniform thickness. Additionally, aiming at improving the stiffness of the in-process workpiece, and also facilitated by the prescribed geodesic distance field, we have proposed a new machining strategy of alternating between the roughing and finishing machining. Finally, collision-free five-axis tool paths are devised to machine the generated machining layers. As a necessary component of this research work, we have carried out physical machining experiments on three representative freeform parts and the results have given a positive confirmation on the intended benefits of the proposed methodology.

Regarding the future work, first, it is found through our experiments that there are a lot of tool retractions generated by our method. This is mainly because currently we rely on the level-first-traversal (LFT) scheme when deciding the order of the machining layers—the machining layers with the same geodesic distance will be machined first before the others. As the machining layers of the same geodesic distance are disjointed and can be quite far away from each other, and their traversal order is currently arbitrary, the total length of the tool retractions could be sizeable. As a future work, we will first explore the depth-first-traversal (DFT) scheme—after a machining layer is machined, the tool moves to the machining layer immediately beneath it (with a smaller geodesic distance), which thus will not require a tool retraction. However, on the negative side, DFT tends to worsen the collision situation. Therefore, a comprise or optimization between LFT and DFT is needed. Second, while the convex hull of the part is a natural selection of the intermediate surface for defining the geodesic distance field (and consequently the machining layers), other choices are worth a trial, or we may even design an intermediate surface that is adaptive to the shape of the part, so to improve the collision situation or other aspects of the tool path such as reducing its acceleration or jerk.

Acknowledgment

Conflict of Interest

There are no conflicts of interest.

Data Availability Statement

The datasets generated and supporting the findings of this article are obtainable from the corresponding author upon reasonable request. The authors attest that all data for this study are included in the paper.

References

[1] Tang, T. D., 2014, "Algorithms for Collision Detection and Avoidance for Five-Axis NC Machining: A State of the art Review," Comput.-Aided Des., **51**, pp. 1–17.

[2] Zhu, J., Tanaka, T., and Saito, Y., 2007, "A Rough Cutting Model Generation Algorithm Based on Multi-Resolution Mesh for Sculptured Surface Machining," J. Adv. Mech. Des. Syst. Manuf., **1**(5), pp. 628–639.

[3] Inui, M., 2003, "Fast Inverse Offset Computation Using Polygon Rendering Hardware," Comput.-Aided Des., **35**(2), pp. 191–201.

[4] Tang, K., Cheng, C. C., and Dayan, Y., 1995, "Offsetting Surface Boundaries and 3-Axis Gouge-Free Surface Machining," Comput.-Aided Des., **27**(12), pp. 915–927.

[5] Takeuchi, Y., Sakamoto, M., Abe, Y., Orita, R., and Sata, T., 1989, "Development of a Personal CAD/CAM System for Mold Manufacture Based on Solid Modeling Techniques," CIRP Ann., **38**(1), pp. 429–432.

[6] Chen, L., Hu, P., Luo, M., and Tang, K., 2018, "Optimal Interface Surface Determination for Multi-Axis Freeform Surface Machining with Both Roughing and Finishing," Chin. J. Aeronaut., **31**(2), pp. 370–384.

[7] Chen, L., Li, Y., and Tang, K., 2017, "Variable-depth Multi-Pass Tool Path Generation on Mesh Surfaces," Int. J. Adv. Manuf. Technol., **95**(5–8), pp. 2169–2183.

[8] Lefebvre, P. P., and Lauwers, B., 2005, "3D Morphing for Generating Intermediate Roughing Levels in Multi-Axis Machining," Comput.-Aided Des. Applic., **2**(1–4), pp. 115–123.

[9] Lauwers, B., and Lefebvre, P., 2006, "Five-axis Rough Milling Strategies for Complex Shaped Cavities Based on Morphing Technology," CIRP Ann., **55**(1), pp. 59–62.

[10] Huang, B., 2013, "A Unified Approach for Integrated Computer-Aided Design and Manufacturing," UCLA, Los Angeles, CA.

[11] Gan, W.-F., Fu, J.-Z., Shen, H.-Y., and Lin, Z.-W., 2014, "A Morphing Machining Strategy for Artificial Bone," J. Zhejiang Univ., Sci., A, **15**(3), pp. 157–171.

[12] Narayanaswami, R., and Pang, J., 2003, "Multiresolution Analysis as an Approach for Tool Path Planning in NC Machining," Comput.-Aided Des., **35**(2), pp. 167–178.

[13] Young, H.-T., Chuang, L.-C., Gerschwiler, K., and Kamps, S., 2004, "A Five-Axis Rough Machining Approach for a Centrifugal Impeller," Int. J. Adv. Manuf. Technol., **23**(3–4), pp. 233–239.

[14] Ming, L., Ce, H., and Hafeez, H. M., 2019, "Four-axis Trochoidal Toolpath Planning for Rough Milling of Aero-Engine Blisks," Chin. J. Aeronaut., **32**(8), pp. 2009–2016.

[15] Dai, C., Wang, C. C., Wu, C., Lefebvre, S., Fang, G., and Liu, Y.-J., 2018, "Support-free Volume Printing by Multi-Axis Motion," ACM Trans. Graph., **37**(4), pp. 1–14.

[16] Li, Y., Tang, K., and Zeng, L., 2020, "A Voxel Model-Based Process-Planning Method for Five-Axis Machining of Complicated Parts," ASME J. Comput. Inf. Sci. Eng., **20**(4), p. 041012.

[17] Tarbutton, J., Kurfess, T. R., Tucker, T., and Konobrytskyi, D., 2013, "Gouge-free Voxel-Based Machining for Parallel Processors," Int. J. Adv. Manuf. Technol., **69**(9–12), pp. 1941–1953.

[18] Collins, J. S., 2018, "Digital Twin Volume Registration for Voxel-Based Closed-Loop Machining Systems," Georgia Institute of Technology, Atlanta, GA.

[19] Ameur, A., 2017, "Voxel-based Tool Sequence Optimization for 5-Axis Machining Using High Performance Computing," Georgia Institute of Technology, Atlanta, GA.

[20] Lynn, R., Dinar, M., Huang, N., Collins, J., Yu, J., Greer, C., Tucker, T., and Kurfess, T., 2018, "Direct Digital Subtractive Manufacturing of a Functional Assembly Using Voxel-Based Models," ASME J. Manuf. Sci. Eng., **140**(2), p. 021006.

[21] Lynn, R., Contis, D., Hossain, M., Huang, N., Tucker, T., and Kurfess, T., 2017, "Voxel Model Surface Offsetting for Computer-Aided Manufacturing Using Virtualized High-Performance Computing," J. Manuf. Syst., **43**, pp. 296–304.

[22] Kurfess, T., Lynn, R., Saleeby, K., Tucker, T., and Saldana, C., 2018, "Multi-Axis Voxel-Based CNC Machining of Centrifugal Compressor Assemblies," American Helicopter Society Forum 74.

[23] Yu, J., Lynn, R., Tucker, T., Saldana, C., and Kurfess, T., 2017, "Model-free Subtractive Manufacturing From Computed Tomography Data," Manuf. Lett., **13**, pp. 44–47.

[24] Crane, K., Weischedel, C., and Wardetzky, M., 2013, "Geodesics in Heat," ACM Trans. Graph., **32**(5), pp. 1–11.

[25] Liao, S.-H., Tong, R.-F., Dong, J.-X., and Zhu, F.-D., 2009, "Gradient Field Based Inhomogeneous Volumetric Mesh Deformation for Maxillofacial Surgery Simulation," Comput. Graph., **33**(3), pp. 424–432.

[26] Botsch, M., and Kobbelt, L., 2004, "A Remeshing Approach to Multiresolution Modeling," Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, pp. 185–192.

[27] Li, Y., Zeng, L., Tang, K., and Xie, C., 2020, "Orientation-point Relation Based Inspection Path Planning Method for 5-Axis OMI System," Rob. Comput. Integr. Manuf., **61**, p. 101827.

[28] Barber, C. B., Dobkin, D. P., and Huhdanpaa, H., 1996, "The Quickhull Algorithm for Convex Hulls," ACM Transactions on Mathematical Software (TOMS, **22**(4), pp. 469–483.