Solid Freeform Fabrication 2021: Proceedings of the 32nd Annual International Solid Freeform Fabrication Symposium – An Additive Manufacturing Conference
Reviewed Paper

SUPPORT-FREE HOLLOWING FOR 3+2-AXIS ADDITIVE MANUFACTURING

Chen Lufeng*, Liu Ruosong

School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu, China

Abstract

Additive manufacturing (AM), also known as 3d printing, has become a hot topic in academia and industry in the past decades. For a typical layer-based additive manufacturing where the object is printed in a layer-by-layer fashion, the battle to reduce or even eradicate the support structure is always faced by researchers and industrial practitioners. The newly emerging multi-axis printing platform inspired by the five-axis machine tool opens new directions, such as surface quality improvement, support-free printing, etc. In this paper, we have presented a framework for the support-free hollowing of 3+2-axis printing. A suite of algorithms including curved skeleton extraction, print sequence optimization, hollowing generation, and print path planning is introduced. It is expected that the print efficiency will increase while the residue artifacts caused by the support structure on the contact surface can be ultimately eradicated.

1. Introduction

Additive manufacturing (AM), also known as 3d printing, has gained remarkable attention in academia and industry in the past decades [1]. It shows unrivaled advantages in fabricating complex models comparing with the conventional subtractive method, such as milling. The most commonly adopted AM technique is the layer-based printing scheme. The digital target model is discretized into a stacking of layers, each of which is manufactured following the layer-by-layer fashion.

For layer-based additive manufacturing, slicing plays a crucial role in decomposing the input mesh into a piling of intermediate layers [2]. The material is deposited layer upon layer until the target geometry is fabricated. The planar slicing is widely used by three-axis printers in a two-and-half manner where the nozzle exhibits planar movement during the material for each layer. The advantages are apparent: 1) the planar slicing is very computationally efficient, which can be achieved by parallel computing since the intersection of mesh and planes with different height is independent of each other [3]; 2) the collision between the platform and the building part is naturally avoided since the all the previously deposited material must lay below the current layer [4]. However, the prominent stair-step effect and the excessive support structure usage are two major battles for the 2.5-axis printing platform.

Thanks to the newly emerging multi-axis printer inspired by the five-axis CNC machining, the nozzle orientation can be adjusted during the printing process synchronizing with the linear movements. The multi-axis 3D printing is expected to alleviate the aforementioned issues. Previous works [5][6][7][8][9] have successfully shown the capability of multi-axis 3D printing by fabricating models without support structures and with enhanced print quality.

* Corresponding author. *E-mail address:* lchenar@connect.ust.hk

Planar slicer and curved layer slicer are two major approaches to fulfill the task. As the name implies, the planar slicer intersects the model with a planar plane. The resultant planar slice composes of one or multiples close regions. The 2.5-axis printing simplifies the situation by fixing the plane normal to +Z direction. The only variable left for the 2.5-axis slicing process is the slicing height. As for the curved layer slicing, Dai et al. [6] proposed a convex growing method to generate curved layers based on voxel representation of the original mesh. The clever convex-front approach of the former work naturally ensures collision-free multi-axis printing. Xu et al. [10] decomposed the mesh into curved layers, each of which was bounded by an iso-level contour extracted from the original mesh. Regarding the kinematics of the multi-axis printing platform, the continuous tilting of the nozzle head or the build plate tremendously drags down the printing efficiency.

The past experience showed that the material used for both support structures and inner fillings would drag down the print efficiency. To best take advantage of the multi-axis printing while preserving the superior properties of the planar slicing, in this paper, we propose a 3+2-axis process planning to decompose the model into a stacking of non-parallel slices with hollowing for the support-free printing to increase the print efficiency and to reduce material usage.

2. Notations and formulation

2.1. Variable-direction planar slicer (VPS)

In order to fabricate a mesh model Ω, the crux of the process planning is to decompose both Ω into a sequence of layers $\{S_i\}$. The operation is also known as the slicing process. The planar slicing is widely adopted in the 2.5-axis printing paradigm in which the target model is divided into a serial of parallel layers with planar planes. The plane used for slicing is known as the *planar slicer*. A general form of the planar slicer will be defined first.

Since a planar plane can be defined as a point p and a normal vector n, the corresponding planar slicer is denoted as $P = (p, \mathbf{n})$. The intersection between Ω and P results in one or multiple close contours on P, i.e., $S = \{s_i\} = \partial\Omega \cap P$. Model hollowing is a typical operation in the CAM process to increase printing efficiency by reducing the infill region inside the close contour. For the hollowing operation, both the mesh model Ω and the hollowed inner surface Ω^H are sliced by the planar slicer P. Therefore, the resultant build layer becomes a group of close regions composing of paired contours $S = \{(s_i, s_i^H)\}$. It is worth noting that the contour s determines the outer shape of the model, which should be kept unchanged, while the hollowing contour s^H is the variable that affects the infill material usage. Therefore, the determination of a hollowed slice can be divided into outer contour and hollowing generation, respectively.

2.2. Outer contour generation

The foundation of the contour generation is to compute all the intersection points of each mesh edge with the planar slicer and connect them in one or multiple close contours. Let $e = \widehat{e_1 e_2}$ be a line segment with two endpoints e_1 and e_2, and $P = (p, \mathbf{n})$ be a slicer passing point p with normal $\mathbf{n}$. Any point on the line segment can be represented as $q = e_1(1-t) + e_2 t$, where $t \in [0,1]$. The parameter t^* of the intersection is computed as,

$$t^* = \frac{(p - e_1)\cdot\mathbf{n}}{(e_2 - e_1)\cdot\mathbf{n}} \quad (1)$$

The intersection exists only when t falls into the range between 0 and 1. Let $E = \{e_i e_j\}$ be the edge set of mesh model Ω, the intersection computation is executed for every edge in E resulting in a group of discrete intersection points. To eventually cluster them into close contours, we mark all the valid edges participating in the intersection operation and build an undirected graph G. Then, we gather the connected components to generate point clusters $\{p_i\}$ each of which belongs to one close contour. Note that the weight of each edge in G is assigned to 1, and the directed contours $c = \{c_i\}$ are retrieved by the depth-first search algorithm.

2.3. Hollowing definition

Given contour $c = (V^c, E^c)$ as a close polygon, where V^c and E^c are the vertex and edge set, respectively, the hollowing contour s^H can be modeled as the shrinks of s by shifting each vertex $v_i \in V$ in the polygon along the direction $\mathbf{d}_i$, with a distance l,

$$v_i' = v_i + \mathbf{d}_i l, \quad i = 1,2,\dots,n \tag{2}$$

where n is the number of vertices in the contour. The offset distance l is a global design variable that controls the shape of the hollowing contour s^H. The shifting direction d_i is determined with the help of the Voronoi diagram's poles constructed from a dense sampling on the original contour s. We also impose the lower and upper bound on the vertex displacement to avoid potential self-intersection, i.e., $l \in [l^L, l^U]$. The lower bound l^L is assigned to the minimum thickness that a predefined nozzle can be printed ($0.1mm$ for a $0.4mm$ nozzle) to ensure the manufacturability. The upper bound l^U is assigned as the minimum distance of all the contour vertices to the Voronoi boundary along the corresponding direction, i.e.,

$$l^U = \min ||v_i - \mathcal{B}(v_i)|| \tag{3}$$

where $\mathcal{B}(v_i)$ is the first Voronoi boundary that intersects with the ray starting at v_i along direction $\mathbf{d}_i$, as shown in Figure 1. The minimum distance is the most conservative value to avoid self-intersection of the hollowing contour. For details, please refer to Ref. [11].

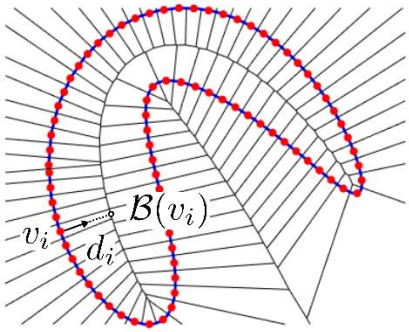


Figure 1 Voronoi diagram of close contour.

2.4. Problem formulation

Due to the layer-wised material additive paradigm, the current build layer should always lay upon the previous layer. Let S and S_0 be the current and the previous build layer, respectively. Given a built layer S_0, the objective function for the next layer S can be defined with the notation and variable introduced above as,

$$\max_{P,l} \mathcal{F}(S, S_0) \tag{4}$$

Note that the variables encapsulated in Eq. (4) can be categorized into two parts: the planar slicer P determines the outer contour generated by the plane and mesh intersection (Section 2.1); the offset distance (Eq. (1)) controls the shape of the hollowing contour (Section 3.2). In this paper, the objective $\mathcal{F}$ is defined as an indicator that reflects the print efficiency between two adjacent layers as our objective (Section 4.2). The variables are also subject to several constraints relating to the physical limitations of the print nozzle and other manufacturing concerns, which will be further investigated in the later sections.

3. Fabrication concerns

The variable direction planar slicer (VPS) is the extension of the conventional slicer by adding the freedom to alter each layer's build direction. In this section, we mainly focus on two fabrication constraints in the printing process:

(1) Build thickness. Due to the physical limitation of the nozzle geometry in the FDM-base printing, the deposition thickness of the material should be constrained in a specific range.

(2) Support-free. The overhang features may increase the material used for support structures, which drags down the print efficiency. Another side effect is that the removal of the support structure may jeopardize the print quality.

3.1. Build thickness limitation

Given a specified nozzle size, the maximally and minimally allowed building thickness is introduced to ensure the print quality. Let k_{min} and k_{max} be the minimally allowed and maximally allowed building thickness, respectively. k_{min} ensures sufficient bonding between adjacent layers while k_{max} should always smaller than the nozzle size R_0 in case of over-extruding, which may deteriorate the commonly assumed ellipse cross-section shape. In this paper, k_{min} is set to $0.1R_0$ and k_{max} is set to $0.9R_0$. Therefore, the build thickness constraint can be expressed as,

$$k \in [k_{min}, k_{max}] \tag{5}$$

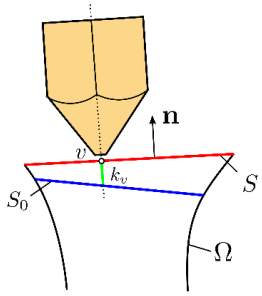


Figure 2 Illustration of the build thickness

For a multi-axis printer in which the build plate can tilt and rotate, we assume that the nozzle axis is always aligned with the slicer normal to avoid the collision between the slice and the nozzle. Therefore, the build thickness should be measured along the build direction. Generally, when the slicer normal of the two adjacent layers are different, the build thickness varies at each print location. Let $v \in P(p, \mathbf{n})$ be a print location to be built, the build thickness is computed as the signed distance to the previous layer $P_0(p_0, \mathbf{n}_0)$, that is

$$k_v = (v - p_0) \cdot \mathbf{n} \tag{6}$$

The build thickness for all the possible print locations should be tested again with the build thickness constraint (Eq. (5)). The build thickness between two neighboring layers is illustrated in Figure 2.

3.2. Support-free condition

Previous researches have confirmed the unrivaled advantage of multi-axis printing over the conventional 2.5-axis one on the capability of printing objects without (or with reduced) support structures [12]. The inevitable support structure usage has two major problems. On the one hand, the excessive usage of support structures may lead to substantially increased print time. On the other hand, residue defects left on the print part after the cleanup may jeopardize the surface quality. The multi-axis printing with the introduced variable direction planar slicer may resolve the problem. In this section, the condition for support-free printing of the VPS is studied. Then, the support-free region of a close contour is defined, which plays a crucial role in the support-free slicing process of the next spherical slicer.

In our previous work [13], we utilized the spherical duality to reveal the relationship between a point and its appropriate support-free build directions. Suppose we have a triangular surface patch with surface normal **n**, the facet is buildable (self-supported by the built geometry) once the following condition is satisfied,

$$\mathbf{n} \cdot \mathbf{b} \geq \cos(\alpha + \frac{\pi}{2}) \tag{7}$$

where **b** is the normalized build direction, and α is the maximal self-supporting angle. For the FDM process adopted in this paper, $\alpha = 45°$ is used. In the context of the fixed build direction paradigm, the whole facet can be built support-freely as long as the build direction satisfies Eq. (7). To ensure that the self-support property is always held for any point on the facet with variable build direction, we need to find the support-free region in which any facet that belongs to the layer can be built support-freely.

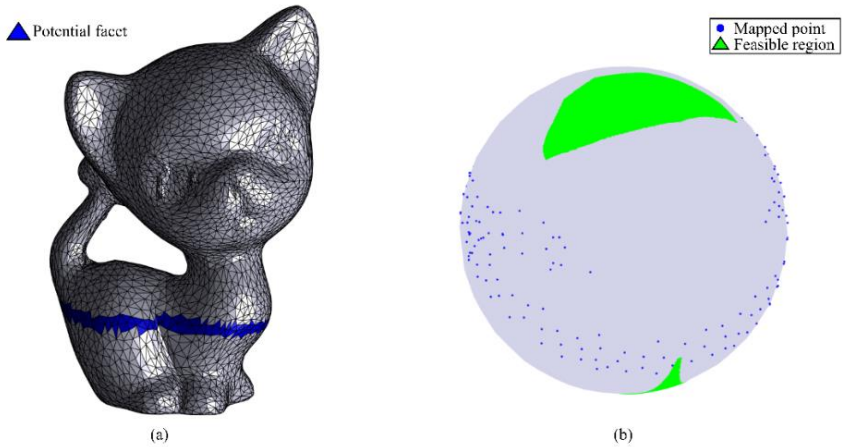


Figure 3 Support-free build direction region determination. (a) A set of candidate facets are selected for computation; (b) Illustration of the facet normal and the support-free region on a unified Gaussian sphere

Given a unified Gaussian sphere, any point on it indicates a normalized vector. Let p_n denote the mapped point of normal vector **n**, the support-free build directions satisfying Eq. (7) essentially form a region $\Phi(p_n)$ on the Gaussian sphere. In fact, the fabrication of a single layer

covers multiple facets, each of which has a distinguish surface normal. Therefore, for a set of mapped points $\{p_n^i\}$, it is easy to find that the support-free build direction lays in the region formed by the Boolean summation of the correspondent region of each mapped point, that is

$$\Phi = \Phi(p_1) \cap \Phi(p_2) \cap \cdots \cap \Phi(p_n) \quad (8)$$

Figure 3(b) shows an example of the computation of the support-free build direction region for the fabrication of a cluster of potential facets given in Figure 3(a).

4. Process planning

Process planning is the crucial step in FDM, which includes both slicing and print path generation. The slicing strategy is first utilized to divide the target object into multiple layers, on which the print paths are then generated. The VPS proposed in Section 2 provides us a more general tool for the model decomposition. Instead of planar layers of fixed build direction generated from the conventional planar slicing, the target object can be segmented into a stacking of planar layers of variable build directions. The most challenging parts are the determination of the build layer sequence and the print path planning on each generated layer. The two issues will be alleviated in this section.

4.1. Curved skeleton extraction

The key to the fabrication sequence determination is to find a sequence of VPS, i.e., $\{P(p_i, \mathbf{n}_i)\}$, each of which corresponds to a cut plane. Conceivably, there will be an infinite number of possible planes that intersect with the target mesh. The curved skeleton is a powerful feature embedded in a 3D object to reflect the shape topology, which is widely adopted in the realm of computer graphics, like shape matching, animation. In this paper, we suppose that the target object is a genus-zero geometry with a single skeleton to reduce the problem difficulty.

The mean curvature-based [14] and the Laplacian smoothing [15][16] are the two most popular skeleton extraction methods in defining the shape topology. As pointed out by Ref. [17], the former may result in a 2D surface that can not be directly applied to describe the tendency of the topology changes. In this paper, we resort to the Laplacian smoothing to extract the skeleton of the given geometry. Referring to Ref. [16], the curved skeleton is extracted by iteratively contracting on the discrete geometry data. The method converges fast and is robust for multi-genus geometries. The discrete skeleton generated from Ref. [16] cannot be directly utilized in the process planning. Moreover, the existence of potential sharp corners in the original skeleton is another issue. To address these problems, we first apply the Laplacian smoothing and utilize the B-spline fitting to convert the discrete vertices into a parametric curve.

Let $\mathcal{D}(t)$ denote as the parametric skeleton generated from the Laplacian-based contraction, where $t \in [0,1]$ is the curve parameter. Starting from $\mathcal{D}(0)$ to $\mathcal{D}(1)$, we find that the growth of the curve roughly matches the additive direction of the geometry from base to top. In other words, point $\mathcal{D}(t)$ reflects the sequential state of the manufacturing process. Therefore, the position variable in the VPS, i.e., p, is replaced by the skeleton with a single parameter t. Figure 4 illustrates two examples of curved skeleton extraction and skeleton refinement.

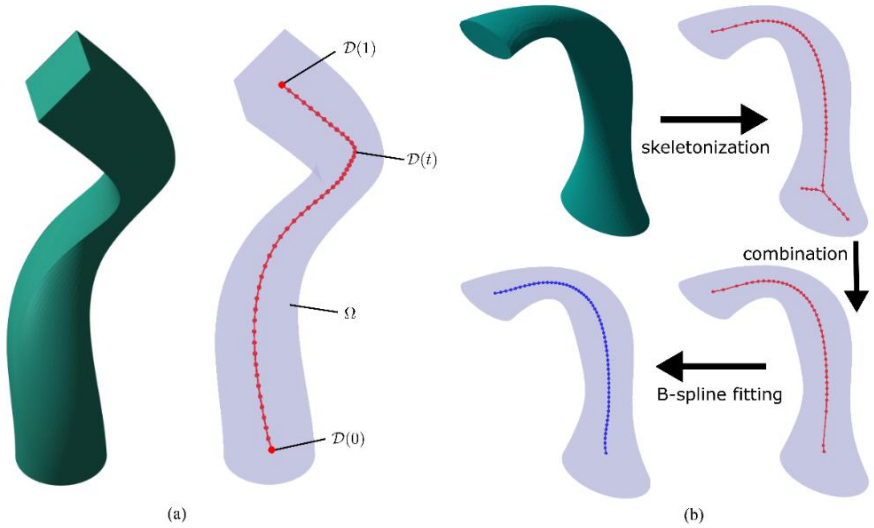


Figure 4 Illustration of the curved skeleton extraction and refinement of two test models.

4.2. Print sequence optimization

Layered manufacturing often suffers from long print time with limited print depth for each layer. Compared with the fabrication of objects with overhang features, support-free printing saves time for the fabrication of support structures. Even for support-free printing, different build directions may result in different layer shapes (both outer and hollowing contour), affecting the print time. Since the previously built layer is treated as the given condition for the optimization of the next layer, the local optimization of the next layer may not be the global optimization for the total print time. Based on the observation that the total volume of the target geometry is a fixed value, we introduce a two-stage optimization scheme to increase print efficiency. In the first stage, a sequence of VPSs is computed by maximizing the volume between two consecutive layers without hollowing, while the offset distance is optimized for the mesh hollowing.

The introduction of the parametric curved skeleton reduces the variables in the objective function (Eq. (4)) to one curve parameter and the normal of the slicer. The optimization problem defined in Section 2.4 can be written as,

$$\max_{t,\mathbf{n}} Vol(S, S_0) \qquad (9)$$

$$\text{s.t.} \quad p_n \in \Phi\big(\mathcal{N}(S_0)\big), \;\; t \in (t(S_0), 1], \;\; k(S, S_0) \in [0.1R_0, 0.9R_0]$$

where: p_n is the mapped point of normal $\mathbf{n}$ on the unified Gaussian sphere, $\mathcal{N}(S_0)$ is the potential facets on the target geometry Ω whose signed geodesic distances to the facets that intersect with S_0 are below a threshold (maximally allowed print depth is adopted in the paper), $t(S_0)$ is the skeleton parameter that determines the slicer of S_0. The layer thickness should always satisfy the build thickness constraint. To estimate the volume between the two layers, we first adopt the strip triangulation to stitch the two layers into a watertight triangulated mesh. The tetrahedral meshing [18] is then applied to convert the triangulated surface patches into tetrahedrons. The total volume is the summation of the volume of each tetrahedron.

The optimization starts with the initial slicer $P_0 = (\mathcal{D}(0), [0,0,1])$ and the corresponding contour S_0, the optimized VPSs are found in sequence until the skeleton parameter reaches one.

The target geometry decomposes into a stacking of outer contours by the mesh and plane intersection. The next task is to generate the support-free hollowing.

4.3. Support-free hollowing

The condition for successful filament deposition requires that the current layer should always be supported by the previously built geometry. In other words, given two neighboring layers, S and S_0, layer S is supported by layer S_0 as long as the projection of S onto S_0 always inside the region bounded by the layer contour of S_0. Let $\Gamma(S)$ denote as the projection of S along the negative direction of the cut plane normal onto the previous layer S_0, the inclusion relation between the projection $\Gamma(S)$ and S_0 can be represented as

$$\Pi = \Gamma(S) - \Gamma(S) \bigcap S_0 \tag{10}$$

As shown in Figure 5(a), the deposited filament can be supported when $\Pi = \emptyset$. Due to the self-support property of the material, a relaxation distance r is added to the projection to normal shrinking the bounded region (see Figure 5(b)). A conservative value is chosen to ensure the self-support property of the hollowing contour, that is

$$r = \min_{p \in S} k_p \sin \alpha \tag{11}$$

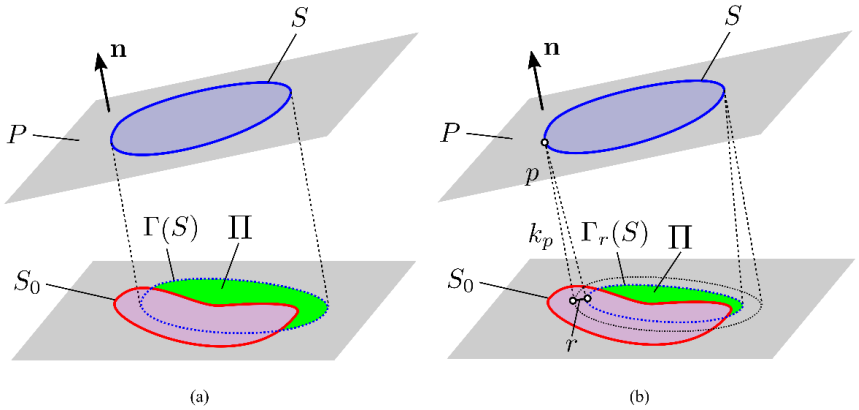


Figure 5 Illustration of the layer projection (a) without relaxation; (b) with relaxation.

The outer contours of the neighboring two layers have already known, when the hollowing contour of layer S is given, the offset distance l_0 that determine the hollowing contour of layer S_0 can be optimized when the overlap region of S_0 and the relaxed projection $\Gamma_r(S)$ is minimized while Eq. (10) is preserved, that is

$$l_0^* = \arg \min_{\Pi=0, l_0 \in [l^L, l^U]} ||S_0(l_0) - \Gamma_r(S)|| \tag{12}$$

The offset distance optimization should be applied to each layer. According to Eq. (12), the offset distance computation starts from the very top layer and follows the top-down manner sequentially.

4.4. Print path planning

The print path planning is also known as the path filling process that generates the print path to fill the region between the outer contour and hollowing contour. Since the hollowing contour is defined as the normal offset of the outer contour, the simple but effective contour parallel print path is adopted in the paper.

Referring to Figure 6(a), the nozzle deposits the melted filament onto the printed part during the printing, which fills up the gap between the nozzle and the previous layer. In our previous work [5], the cross-section of the re-solidified filament is modeled as an ellipse with w_1 and w_2 being the major axis length the minor axis length, respectively. The major axis length is roughly equal to the nozzle diameter, i.e., $w_1 = 2r_0$. The minor axis length is equal to the build thickness k. A proper overlapping between the two neighboring paths should be enforced to obtain enough lateral bonding, which is known as the path *interval*. According to our ellipse model, in the conventional 2.5-axis printing, since the build thickness is a fixed value, the constant lateral bonding is preserved with constant path interval. However, for print locations with different build thicknesses, constant path interval may result in overlapping difference, which corresponds to different lateral bonding. In this work, to simplify the problem, we set a conservative path interval, which is equal to the major axis length, to control the minimum lateral bonding.

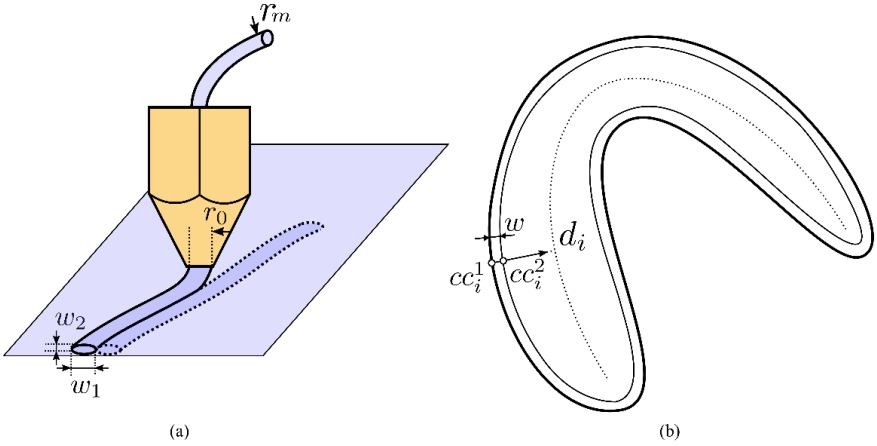


Figure 6 (a) Side step determination; (b) Contour parallel path.

The generation of the contour parallel paths begins with the outer contour S. The next contour is obtained by the curve offsetting such that the interval between the two paths equals to w_1. Borrowing the concept of cutter contact (CC) point in the subtractive machining process [19], let cc_i^j denote the i-th contour point on the j-th path, the corresponding contour point on the next contour is computed as,

$$cc_i^{j+1} = cc_i^j + \mathbf{d}_i w_1 \tag{13}$$

The process iterates until the region between the outer contour, and the hollowing contour is filled with print paths. It is worth noting that the feed step should also be considered for print path with variable build thickness. The feed step should be set to comply with the volume equilibrium between the deposited material and the filament. The feed step l_e sending to the feed extruder for depositing the material between two neighboring print locations is computed as,

$$l_e = \frac{w_1 w_2 \left|\left|cc_{i+1} - cc_i\right|\right|}{4r_m^2} \tag{14}$$

where r_m is the filament radius. For details, please refer to our previous work [5].

So far, we have presented the process planning containing both the layer decomposition and print path planning. The target geometry is discretized into a sequence of planar layers, each of

which is filled with contour-parallel paths. The paths are then converted into G-code for the printer to execute.

5. Results and discussion

A suite of algorithms proposed in this paper has been implemented by us in MATLAB and C++. The remeshing that generates the triangulated mesh in a close region is achieved by invoking the isotropic remeshing related function derived from CGAL, and the well-known Mitchell-Mount-Papadimitriou (MMP) algorithm is applied to find the geodesic-based potential facets. The multi-axis FDM hardware system, which integrates an FDM extruder and a 6-DOF UR5 robotic arm, is developed to conduct the physical experiment. For the configuration details, please refer to Ref. [5].

The single-branch model shown in Figure 4(a) is selected as our test example. The nozzle size is fixed to $R_0 = 0.4\,mm$, which limits the feasible build thickness in the range between 0.04 mm and 0.3 mm. Before optimizing the print sequence, the curved skeletons of the two models are extracted using the method adopted in Section 4.1. The results are shown in Figure 4. For comparison, the top layer of the first model is removed, resulting in an open mesh. Since the skeleton extraction method adopted takes the discrete point cloud as input, the result remains the same for the first example.

Next, we call the print sequence subroutine to generate an optimized sequence of VPS. In total, 471 VPSs are found. Once the VPSs are obtained, the outer contour of each layer is computed by the plane and mesh intersection. The point order is sorted by considering the mesh topology (see Figure 7(a)). Then, the optimization for the hollowing contour of each layer is conducted in a top-down manner. Since the given model is an open mesh, the minimally allowed offset distance is enough for support-free printing while maintaining manufacturability. The offset distance begins to increase after some layer to satisfy the self-support condition.

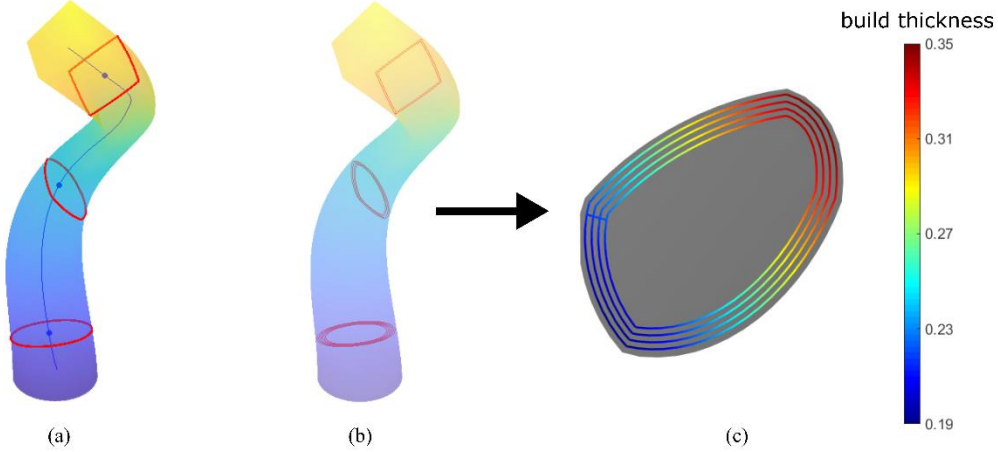
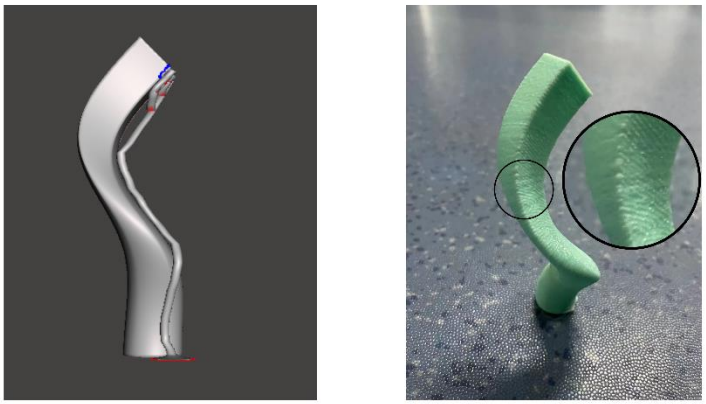


Figure 7 Process planning.

The contour-parallel path is generated for each layer with a fixed path interval. Before converting to the machine-aware G-code, the print path on each layer is connected in a head-tail manner, while a traversing path is added between the paths of the neighboring layers. To avoid the potential collision between the nozzle and the already built geometry, we set a clearance distance for the traversing path. The rotation operation for the build direction change can only be

executed after the nozzle reaches the clearance height. The print paths on three different layers are given in Figure 7(b). The print path with build thickness visualization for layer# 200 is also depicted in Figure 7(c).

The physical test is conducted on our multi-axis 3D printer [5], and the result is shown in Figure 8(b). Clearly, the inevitable support structure (generated from Meshmixer with default settings) is needed to fabricate the model on a conventional 3-axis printer as shown in Figure 8(a), which takes up to around 8% of the total volume.



(a) (b)

Figure 8 (a) The model with overhang features; (b) 3+2-axis support-free printing.

6. Conclusion

We have presented a framework of the support-free hollowing of 3+2-axis printing for column-like geometry with a single skeleton. To overcome the tangling of the print sequence determination and the hollowing, we have developed a two-stage strategy. With the introduction of the 3+2-axis printing scheme, it is expected that the print efficiency will improve while the residue artifacts caused by the support structure on the contact surface can be ultimately eradicated. The limitation of the current strategy is the flexibility in handling geometries with multiple skeletons or multi-genus. The segmentation strategy developed in the field of machining [20] may be adopted to resolve the problem.

Acknowledgement

The authors acknowledge support from the National Natural Science Foundation of China (52005082).

References

[1] N. Guo and M. C. Leu, "Additive manufacturing: Technology, applications and research needs," *Front. Mech. Eng.*, vol. 8, no. 3, pp. 215–243, 2013.

[2] M. Livesu, S. Ellero, J. Martínez, S. Lefebvre, and M. Attene, "From 3D models to 3D prints: an overview of the processing pipeline," *Comput. Graph. Forum*, vol. 36, no. 2, pp. 537–564, 2017.

[3] X. Wang, L. Chen, T. Y. Lau, and K. Tang, "A skeleton-based process planning framework for support-free 3+2-axis printing of multi-branch freeform parts," *Int. J. Adv.*

Manuf. Technol., vol. 110, no. 1–2, pp. 327–350, 2020.
[4] J. Etienne *et al.*, "CurviSlicer : Slightly curved slicing for 3-axis printers," *ACM Trans. Graph.*, vol. 38, no. 4, p. Article 81, 2019.
[5] L. Chen, M.-F. M.-F. Chung, Y. Tian, A. Joneja, and K. Tang, "Variable-depth curved layer fused deposition modeling of thin-shells," *Robot. Comput. Integr. Manuf.*, vol. 57, pp. 422–434, 2019.
[6] C. Dai, C. C. L. Wang, C. Wu, S. Lefebvre, G. Fang, and Y.-J. Liu, "Support-free volume printing by multi-axis motion," *ACM Trans. Graph.*, vol. 37, no. 4, p. Article 1, 2018.
[7] O. Diegel, S. Singamneni, B. Huang, and I. Gibson, "Curved Layer Fused Deposition Modeling in conductive polymer additive manufacturing," *Adv. Mater. Res.*, vol. 199–200, pp. 1984–1987, 2011.
[8] Y. Huang *et al.*, "FrameFab: robotic fabrication of frame shapes," *ACM Trans. Graph.*, vol. 35, no. 6, p. Article 224, 2016.
[9] M. A. Isa and I. Lazoglu, "Five-axis additive manufacturing of freeform models through buildup of transition layers," *J. Manuf. Syst.*, vol. 50, pp. 69–80, 2019.
[10] K. K. Xu, Y. Li, L. Chen, and K. Tang, "Curved layer based process planning for multi-axis volume printing of freeform parts," *Comput. Des.*, vol. 114, pp. 51–63, 2019.
[11] W. Wang *et al.*, "Support-free hollowing," *IEEE Trans. Vis. Comput. Graph.*, pp. 1–13, 2017.
[12] T. H. Kwok, X. Li, Y.-S. Leung, Y. Chen, C. C. L. Wang, and Y. Yang, "Challenges and Status on Design and Computation for Emerging Additive Manufacturing Technologies," *J. Comput. Inf. Sci. Eng.*, vol. 19, no. 2, p. 021013, 2019.
[13] K. Xu, L. Chen, and K. Tang, "Support-free layered process planning towards 3+2-axis additive manufacturing," *IEEE Trans. Autom. Sci. Eng.*, vol. 16, no. 2, pp. 838–850, 2019.
[14] A. Tagliasacchi, I. Alhashim, M. Olson, and H. Zhang, "Mean curvature skeletons," in *Eurographics Symposium on Geometry Processing*, 2012, vol. 31, no. 5, pp. 1735–1744.
[15] O. K.-C. Au, C.-L. Tai, H.-K. Chu, D. Cohen-Or, and T.-Y. Lee, "Skeleton extraction by mesh contraction," *ACM Trans. Graph.*, vol. 27, no. 3, pp. 1–10, 2008.
[16] J. Cao *et al.*, "Point cloud skeletons via Laplacian-based contraction," in *International Conference on Shape Modeling and Applications*, 2010, pp. 187–197.
[17] P. K. Saha, G. Borgefors, and G. Sanniti di Baja, "A survey on skeletonization algorithms and their applications," *Pattern Recognit. Lett.*, vol. 76, pp. 3–12, 2016.
[18] Y. Hu, Q. Zhou, X. Gao, A. Jacobson, D. Zorin, and D. Panozzo, "Tetrahedral Meshing in the Wild," *ACM Trans. Graph.*, vol. 37, no. 4, 2018.
[19] Y. Sun, F. Ren, X. Zhu, and D. Guo, "Contour-parallel offset machining for trimmed surfaces based on conformal mapping with free boundary," *Int. J. Adv. Manuf. Technol.*, vol. 60, no. 1–4, pp. 261–271, Jan. 2012.
[20] İ. Demir, D. G. Aliaga, and B. Benes, "Near-convex decomposition and layering for efficient 3D printing," *Addit. Manuf.*, vol. 21, pp. 383–394, 2018.