DSCarver: Decompose-and-Spiral-Carve for Subtractive Manufacturing

HAISEN ZHAO, Shandong University
HAO ZHANG, Simon Fraser University
SHIQING XIN, Shandong University
YUANMIN DENG, Shandong University
CHANGHE TU, Shandong University
WENPING WANG, Hong Kong University
DANIEL COHEN-OR, Tel Aviv University
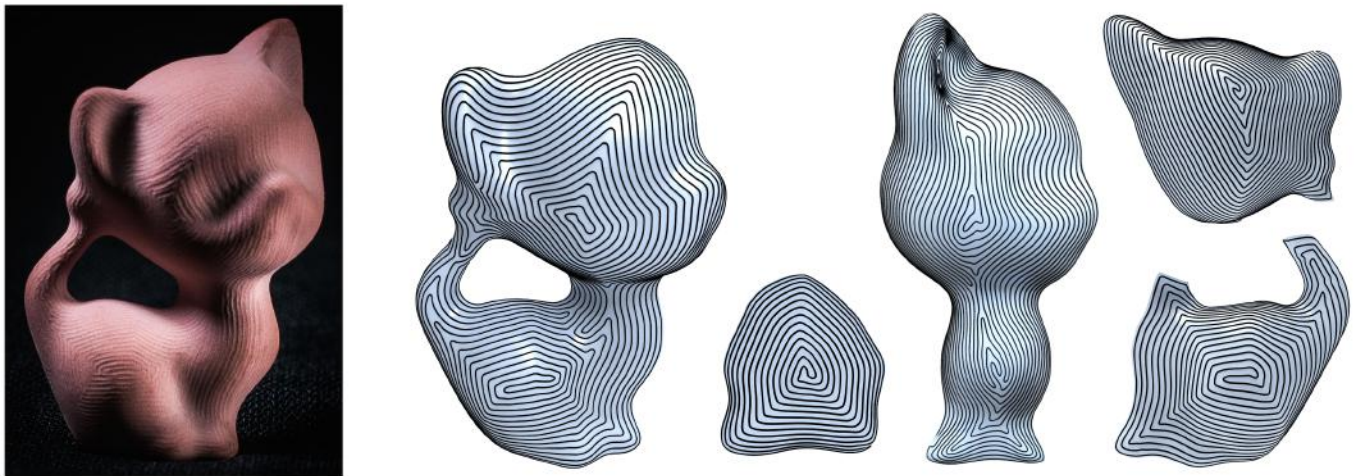BAOQUAN CHEN, Shandong University and Peking University

Fig. 1. A freeform 3D object, the KITTEN, automatically decomposed into five machinable patches, each carved using a CNC machine following a continuous iso-scallop Fermat spiral path. The fully machined object is shown on the left. We deliberately lowered the path resolution to ease path visualization.

We present an automatic algorithm for *subtractive* manufacturing of freeform 3D objects using high-speed machining (HSM) via CNC. A CNC machine operates a cylindrical cutter to carve off material from a 3D shape stock, following a tool path, to "expose" the target object. Our method decomposes the input object's surface into a small number of patches each of which is fully accessible and machinable by the CNC machine, in *continuous* fashion, under a fixed cutter-object setup configuration. This is achieved by covering the input surface with a minimum number of *accessible regions* and then extracting a set of machinable patches from each accessible region. For each patch obtained, we compute a continuous, space-filling, and *iso-scallop* tool path which conforms to the patch boundary, enabling efficient carving with high-quality surface finishing. The tool path is generated in the form of *connected Fermat spirals*, which have been generalized from a 2D fill pattern for layered manufacturing to work for curved surfaces. Furthermore, we develop a novel method to control the spacing of Fermat spirals based on directional surface curvature and adapt the heat method to obtain iso-scallop carving. We demonstrate automatic generation of accessible and machinable surface decompositions and iso-scallop Fermat spiral carving paths for freeform 3D objects. Comparisons are made to tool paths generated by commercial software in terms of real machining time and surface quality.

CCS Concepts: • **Computing methodologies** → **Computer graphics**; *Shape analysis*;

Additional Key Words and Phrases: Subtractive manufacturing, tool path planning, CNC, surface decomposition, set cover, Fermat spirals

ACM Reference Format:
Haisen Zhao, Hao Zhang, Shiqing Xin, Yuanmin Deng, Changhe Tu, Wenping Wang, Daniel Cohen-Or, and Baoquan Chen. 2018. DSCarver: Decompose-and-Spiral-Carve for Subtractive Manufacturing. *ACM Trans. Graph.* 37, 4, Article 137 (August 2018), 14 pages. https://doi.org/10.1145/3197517.3201338

Authors' addresses: Haisen Zhao, Shandong University; Hao Zhang, Simon Fraser University; Shiqing Xin, Shandong University; Yuanmin Deng, Shandong University; Changhe Tu, Shandong University; Wenping Wang, Hong Kong University; Daniel Cohen-Or, Tel Aviv University; Baoquan Chen, Shandong University and Peking University.

1 INTRODUCTION

With the increasing popularity of additive manufacturing and 3D printing in computer graphics, one should not overlook the fact that *subtractive* processes remain the core and dominant technology in manufacturing today[1]. Compared to additive manufacturing in most cases, subtractive manufacturing is faster and more cost-effective for the same level of product precision, accommodates a vastly wider range of materials, and is capable of superior surface finishes. Moreover, while a 3D printer builds a 3D shape one 2D layer at a time, subtractive manufacturing of 3D objects is intrinsically a three-dimensional problem, involving direct manipulation of 3D objects and operation over curved surfaces. The ensuing geometry processing appears to possess more compelling twists and technical challenges than those arising from additive fabrication.

Subtractive manufacturing is primarily realized by computer numerical control (CNC) machining tools. A CNC machine operates a cylindrical cutter to *carve* off material from a shape stock in 3D space to "expose" the target 3D object; see Figure 3. The cutter head traces out a space curve, called a *tool path*, which must completely fill the object surface. Desirable properties of tool paths for efficient CNC machining, in particular high speed machining (HSM), include *fairness* (i.e., low-curvature) and *continuity* (i.e., less on/off switchings or tool retractions), similar to additive manufacturing, but all considerations must be shifted from 2D regions to curved surfaces. One added twist for subtractive manufacturing is *accessibility*: in general, the CNC cutter may not be able to access all regions of the object no matter how it is oriented. Another new issue is controlling the amount of residual material, called *scallop*, after carving to ensure a quality surface finishing; see Figure 3. The typical goal for scallop optimization is to *maximize uniformity* of the scallops while minimizing their height without over-carving the surface.

Existing methods for CNC tool path planning from the computer-aided design (CAD) and manufacturing (CAM) domains are primarily designed to machine relatively simple geometric primitives, e.g., planar and quadric surfaces, swept volumes, and CAD models composed of planes and other parameterizable patches. Conventional tool path patterns such as zigzags work quite effectively for such surfaces with simple boundaries and interior geometries. In terms of *setup planning*, where a machinist decides how to orient and stabilize the shape stock with fixtures for carving, it is highly critical to minimize the number of setups, i.e., to avoid re-fixturing and re-orientation of the object or the CNC machine cutters. In practice, setup planning is predominantly a manual process, where machinists rely heavily on their domain knowledge and experience.

In this paper, we are interested in efficient subtractive manufacturing of 3D objects formed by *freeform* or sculpted surfaces [Lasemi et al. 2010], such as the example shown in Figure 1. In general, a 3D object cannot be fully machined under a single setup. Thus during the setup planning phase, there is an inherent surface *decomposition* problem which seeks to segment the object's surface into a minimum number of patches each of which can be fully machined

under one setup. For a freeform object with moderate complexity, regardless of how it is decomposed, the machinable patches are likely to exhibit irregular interior undulations and wavy boundaries, which would pose various challenges to tool path planning, especially when taking scallop optimization into account.

Our goal is automatic optimization of setup and tool path planning for *finish-stage* machining of free-form 3D objects using *3+2* machines, where at this finishing stage of the carving process, the current object is already geometrically close to the final product. In the CAD/CAM industries, freeform surfaces are typically carved by 5-axis machining and 3+2 machining represents a special but dominant configuration for 5-axis CNC machines. Specifically, the cutter of a 3+2 machine has a fixed orientation during carving, but can move with three degrees of freedom. The cutter orientation can be adjusted with two degrees of freedom for the next carving[2].

Given an input 3D object represented by a closed two-manifold surface, we develop an algorithm to tackle two key technical problems in setup and tool path planning:

(1) *Surface decomposition.* During setup, the core problem is to minimize the number of object or cutter setups (i.e., re-fixturing or re-orientation of the CNC cutter) to ensure accessibility of the entire input surface by the CNC cutter. To this end, we cover the input surface with a minimum number of *accessible regions* by posing and solving a SET-COVER problem; see Figure 2(b). Then from each accessible region, we extract a set of patches each of which can be fully machined by 3+2 machining, in a single fixed cutter-object setup. We obtain these patches by integrating the accessible regions with a pre-segmentation of the input surface into a small set of *height fields*; see Figure 2(a). Together, these patches, which we refer to as *machinable patches*, form a decomposition of the input surface; see Figure 2(c).

(2) *Tool path planning.* In the carving phase, for each machinable patch obtained from the decomposition step, we compute a *continuous*, space-filling, and *iso-scallop* tool path which conforms to the patch boundary, where iso-scollop paths seek to maximize uniformity of the scallop height over the patch. To this end, we advocate the use of *connected Fermat spirals* [Zhao et al. 2016] as the preferred tool path pattern. This is justified by the observation that Fermat spirals tend to outperform zigzag, the dominant tool path patterns for CNC machining, as the patch boundary and interior geometry become more complex.

To compute iso-scallop Fermat spirals, we first generalize the original Fermat spirals designed for layered manufacturing to work for curved surfaces. Then we develop a novel method to control the spacing of Fermat spirals based on *directional surface curvature* and adapt the heat method [Crane et al. 2013a] to obtain iso-scallop carving; see Figure 2(d).

We call our setup and tool path planning algorithm *decompose-and-spiral-carve*, or *DSCarver*, for short. We demonstrate automatic generation of accessible and machinable surface decompositions and iso-scallop Fermat spiral carving paths for 3+2 machining of

---

[2]Equivalently, the additional two degrees of freedom are provided by adjusting the tilt-rotary table that support and hold the carved object, as shown in Figure 3
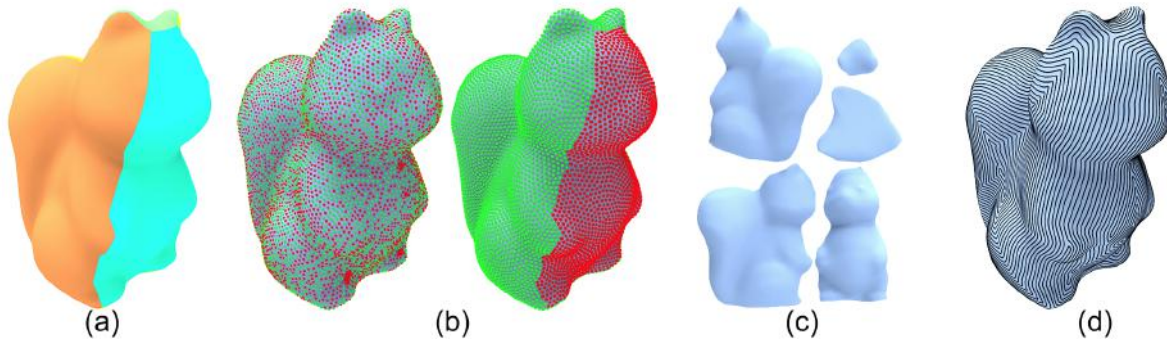
Fig. 2. Overview of DSCarver, our decompose-and-carve algorithm for 3+2-axis CNC machining of freeform 3D objects. (a) Input 3D shape with pre-segmentation into few height fields. (b) Decomposition into accessible regions (left: with overlaps; right: after boundary extraction). (c) Integration of accessibility decomposition (b) and height fields (a) into machinable patches. (d) Connected iso-scallop Fermat spiral paths computed for a few patches.

Fig. 3. 3-axis pocket milling (left) with a square cutter vs. 3+2-axis machining (cutter on top) with a ball cutter. Scallop (red) is the material residual left between adjacent ball cutter (green) paths.

freeform 3D objects. Comparisons are made to conventional tool paths generated by high-end CAD/CAM systems, both over real CNC machining time and surface quality.

2 BACKGROUND AND RELATED WORK

The core geometry problems for both additive and subtractive manufacturing can be classified into setup and tool path planning. In the setup stage for 3D printing, a 3D object may be hollowed [Lu et al. 2014], decomposed [Hu et al. 2014; Luo et al. 2012], or reconfigured in other ways [Bermano et al. 2017; Chen et al. 2015; Prévost et al. 2013; Stava et al. 2012] to improve print quality and/or save material consumption. Frequently adopted and commercially available tool path patterns for 3D printing include zigzag [Ding et al. 2014] and contour parallel paths [Yang et al. 2002]. Zhao et al. [2016] introduce connected Fermat spirals as an alternative and demonstrate their advantages over conventional tool paths for layered manufacturing.

CNC machining basics. CNC machining operates a cylindrical cutter with a prescribed length and size (measured on the cutter's horizontal profile) and goes around in 3D space with its head spinning at high speed to carve off material from a shape stock. During rough-stage machining, larger chunks of material are carved off along the path by a thicker cutter often with a flat end. In the surface finishing stage, a rounded or ball end cutter is often employed. The fine lines of residuals left between adjacent tool paths after surface finishing are referred to as scallop; see Figure 3. The height and width of the scallop should be properly controlled and they depend on path spacing, cutter orientation, and surface curvature. For example, over convex regions (compared to concave regions

and assuming that the angle between the cutter orientations stay fixed), path spacing needs to be denser to reduce scallop height.

Full 5-axis vs. 3+2 machining. 3-axis machining or pocket milling is similar to layered manufacturing as it also traverses a 2D domain, but removes instead of injects material. Curved freeform or sculpted surfaces [Lasemi et al. 2010] are typically carved by full 5-axis or 3+2 machining. The cutter of a full 5-axis machine can move with five degrees of freedom. In contrast, the cutter of a 3+2 machine has a fixed orientation and moves in x, y, z directions only. In both cases, the cutters typically only point downward at an oblique angle, not upward. As well, it is desirable that the cutter orientation does not deviate from the surface normal too much to bound the scallop height [Farouki 2016; Farouki and Li 2013; Zhao et al. 2013].

In our current work, we choose 3+2 machining over full 5-axis CNCs due to several factors. 3+2 machining represents the dominant 5-axis CNC technology in the industry and 3+2 machines are a lot more accessible and easier to work with compared to full 5-axis machines. Conventional tool paths such as zigzag work much more naturally with 3+2 machining. Given the same surface patch to carve, 3+2 machining almost always beats full 5-axis CNC in speed and accuracy. Overall, improving the state of the art in 3+2 machining offers more value to the domain of subtractive manufacturing.

Setup planning for CNC machining. Generally, setup planning involves the preparation of instructions for setting up parts for CNC machining [Hazarika et al. 2015; Xu et al. 2007]. The key issue is how to orient the parts, perhaps in multiple configurations, to attain a high level of efficiency and surface quality. Setup planning methods from the CAD and manufacturing literature have mainly focused on CAD models, where it is widely assumed that the input consists of feature-based designs or outputs from a feature recognition system [Xu et al. 2007]. Typically, the machined parts are assumed to be composed of prismatic or rotational primitives [Amaitik and Kiliç 2007], or geometric features that possess certain manufacturing or functional significance [Tseng and Joshi 1998], e.g., k-sided pockets, through, semi-blind, or compound slots, etc.

Fig. 4. Frequently applied CNC tool path patterns.

Tool paths for CNC machining. Existing methods for carving curved surfaces can be roughly classified into parameterization-based and drive surface methods [Choi and Jerrard 1998]. By parameterizing a curved patch onto the plane, a tool path can be planned on the plane and then mapped back to the surface patch, e.g., [Ren et al. 2009]. Methods based on drive surfaces intersect the input surface patch with a set of planes to obtain the tool paths. The most frequently adopted drive surfaces are equidistant parallel planes, resulting in *iso-planar* tool paths. The orientations of the drive planes can be optimized, resembling the slicing problem for additive manufacturing [Hildebrand et al. 2013]. Iso-planar curves resulting from plane-surface interactions can be turned into zigzag [Misra et al. 2005], contour-parallel offset, or spiral patterns [Hauth and Linsen 2012; Held and Spielberger 2014; Zhou et al. 2016]; see Figure 4.

Desirable properties of CNC tool paths include fairness, continuity, and good spacing for quality scallop and surface finishing. Improper spacing between adjacent, parallel tool paths can lead to over- or under-fill for layered manufacturing. For CNC machining, under-fill translates to uncut strips over the surface, which is caused by widely spaced tool paths. Such artifacts on machined surfaces are more problematic than inefficiencies caused by paths spaced too closely. The latter, an "over-fill" in the context of CNC machining, leads to overlapping between cutter trajectories over the surface and thus inefficiency; it may also over-cut the desired parts.

For high-speed machining, path continuity and fairness are even more critical than in the case of layered manufacturing since for CNC cutters operating at high speed, cutter lifting, retraction, and deacceleration, as the results of path discontinuities and sharp turns, are especially counter-productive [Park et al. 2003; Zhou et al. 2016]. Therefore, continuous tool paths based on Fermat or double spirals are preferred [Hauth and Linsen 2012; Wang et al. 2015; Zhao et al. 2016; Zhou et al. 2016]. In the CAD and manufacturing domains, many methods have been presented for iso-scallop tool path planning, e.g., [Agrawal et al. 2006; Can and Ünüvar 2010; Zoua et al. 2014]. However, most of these methods were designed to work with zigzag tool paths and our work is the first attempt at generating iso-scallop Fermat spiral paths for curved surfaces.

Surface decomposition. There have been many works in computer graphics on shape decomposition [Shamir 2008]. The central criterion for our decomposition analysis is accessibility. The fairness criterion for region boundaries is sought since conventional tool paths typically conform to the boundaries [Hauth and Linsen 2012; Lasemi et al. 2010; Zhou et al. 2016]. Existing works from the subtractive manufacturing domain, e.g., [Hauth and Linsen 2012; Held and Spielberger 2014; Zhao et al. 2016], mainly considered the problem of decomposing a *planar* region into simple geometric segments

for efficient pocket machining. Such simple segments often admit continuous and fairer tool paths, e.g., using Fermat spirals, and the per-segment tool paths can be linked to attain global continuity [Zhou et al. 2016]. Our work follows a similar approach but must deal with free-form surfaces and accessible regions with complex boundaries and interior undulation.

With its connection to the SET COVER problem, it is known that generally, finding the minimum number of orientations to ensure full accessibility, i.e., the accessibility-based decomposition problem, is NP-hard [Frank et al. 2006; Gupta et al. 1996]. Early work by Gupta et al. [1996] takes a greedy approach which iteratively identifies accessible regions of maximal surface areas. Frank et al. [2006] analyze accessibility in one planar slice and solve a set cover problem. Various solution mechanisms including decision trees [Keeney and Raiffa 1993], swarm intelligence [Guo et al. 2009], and genetic algorithms [Bo et al. 2006] have been proposed to solve the difficult optimization problem. Also related is the problem of decomposing a surface into a set of approximate height fields [Herholz et al. 2015], which we adopt for pre-segmentation of the input surface.

3 OVERVIEW

The input to our algorithm is a freeform 3D object represented as a 2-manifold triangle mesh. During preprocessing, the input mesh surface is first segmented into a small number of height fields. We compute height fields since each such surface region can be fully machined by a 3+2 machine with a fixed cutter orientation and fixed cutter-object setup. Then we cover the input surface by a minimum number of accessible regions and integrate the resulting regions with the pre-segmentation to obtain a small number of machinable surface patches, which form a decomposition of the input surface. Tool path planning is carried out for each patch to obtain a continuous space-filling curve attaining maximal scallop uniformity. Figure 2 illustrates the algorithm pipeline.

Surface decomposition. Our accessibility analysis involves finding a minimum number of object setups to ensure accessibility of the entire input surface by the CNC cutter. Each object orientation induces an accessible region, which is the set of all points on the input surface that are accessible by the cutter in *some* valid orientation. When machining the same accessible region, the fixture setup for the CNC machine remains unchanged. Switching from one accessible region to another, the fixtures need to be adjusted to re-stabilize the shape stock, which is a delicate and time-consuming endeavor.

We proceed by sampling a set of object orientations, so that the union of their induced accessible regions completely covers the input surface. With this cover as input, we find the minimum number of orientations by solving a SET-COVER problem [Cormen et al. 2001]. Typically, the optimal solution incurs significant overlaps between the accessible regions. We resolve these overlaps and arrive at a surface decomposition by integrating the accessible regions with the pre-segmented height fields. This is followed by boundary optimization to obtain the set of machinable patches.

Tool path planning. Given a machinable patch, we produce a single continuous space-filling curve for that patch using connected Fermat spirals [Zhao et al. 2016]. The main innovation is to ensure that the patch finishing using the spiral carving path is optimized
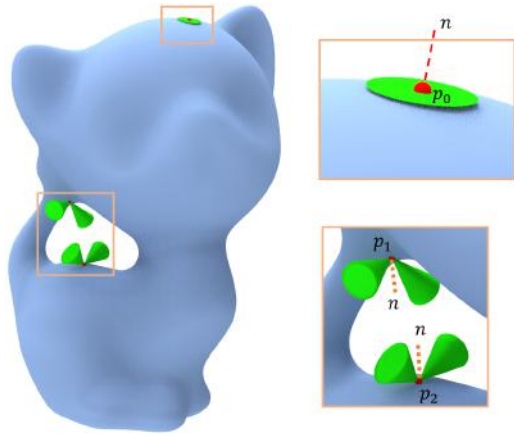
Fig. 5. Accessibility cones on the KITTEN model. For the object orientation shown, p_0 has a full accessibility cone; p_2's cone is split into two sub-cones due to cutter collision with the KITTEN's tail. p_1 is inaccessible.

for scallop quality, i.e., to compute an iso-scallop Fermat spiral. To this end, we adjust the path spacing based on directional curvature over the input surface path, to optimize uniformity of the resulting scallops. We show that an *nonhomogeneous* version of the heat method [Crane et al. 2013a] for geodesic computations can be adapted to compute iso-scallop level-set contours over the surface patch, from which we can extract the connected Fermat spiral paths. Each machinable patch is machined separately following the iso-scallop Fermat spiral paths under a fixed 3+2 machining setup.

4 SURFACE DECOMPOSITION

In this section, we detail the surface decomposition step. It produces a small set of connected and machinable surface patches, each of which is fully accessible by the CNC cutter with respect to one of few machining setups. The patches possess fair or low-curvature boundaries to facilitate efficient tool path planning therein.

Height field decomposition. During preprocessing, we decompose the input object's surface into a small number of height fields by implementing a scheme that is very similar to [Herholz et al. 2015]. Our height field decomposition problem is almost exactly the same as the one addressed by the said approach except for two minor differences. First, we do not deform the input surface to lower the number of height fields produced. Second, the height fields in their work were defined using ideal lines/rays with infinitesimal thickness. In our implementation, we replace the rays with cylinders with non-negligible radius which reflect the physical girth of the CNC cutter, assuming that the cutter is sufficient long during carving.

Accessibility cones. In CNC machining, there is typically a bound on the angle φ between cutter orientation and surface normal at a point on the object surface. This bound defines an accessibility cone around each surface normal. If the surface is oriented in such a way that the cutter can orient itself to fall inside the accessibility cone at a point $p \in S$, then p is accessible with respect to that surface orientation. In our work, we liberally set the angle at $\varphi = \pi/2$ but account for the cutter's physical girth and possible global collision with parts of the 3D object. Such a collision leads to a splitting of the
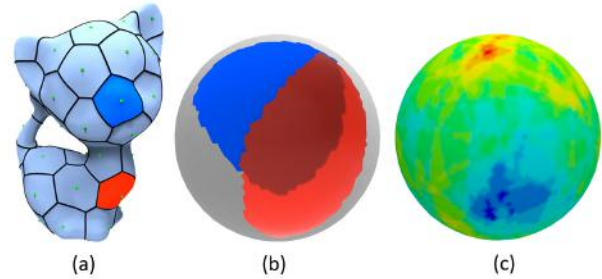

Fig. 6. Cell accessibility and illustrations on the Gaussian sphere. (a) Intrinsic Voronoi cells over object surface. (b) Corresponding regions on the Gaussian sphere representing object orientations which would allow the blue and red cells (a) to be accessible. (c) Color coding of the number of accessible Voronoi cells for each object orientation (red = higher count).
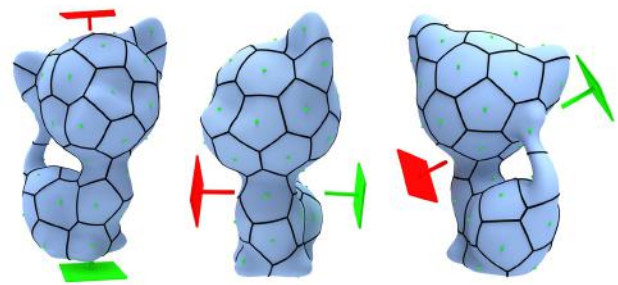

Fig. 7. Three different MINORI solutions computed by SCP, with the same object orientations count (three).

full cone into sub-cones conservatively to ensure full accessibility of the sub-cones; see point p_2 in Figure 5.

Point accessibility. Most 5-axis CNC machines can only point the cutter downward inside an oblique angle between $0°$ and $90°$ [Apro 2008]. For the orientation of the KITTEN model shown in Figure 5, point p_0 is accessible from any direction in its accessibility cone, p_1 is inaccessible since its cone is entirely pointing downward, and p_2 is partially accessible due to potential cutter collision.

Cell accessibility. We start by uniformly sampling N points over the input surface mesh and computing an intrinsic Voronoi tessellation with the sample points as sites. For each Voronoi cell c_i, $1 \le i \le N$, based on point accessibility and accessibility cones, we estimate a set R_i of *object* orientations each of which would allow all points in c_i to be accessible by the CNC cutter. Mapping R_i onto the Gaussian sphere defines a region $\mathscr{R}_i$ on the sphere; see Figure 6. The color coding indicates the number of accessible Voronoi cells for a number of orientation candidates. Red colors are associated with orientations for which many cells are accessible. The set of candidate object orientations can be sampled uniformly or randomly.

Accessibility cover. We compute the accessible regions by formulating the problem as an instance of the SET-COVER problem (SCP). Then we resolve overlapping between the obtained covers to arrive at a surface decomposition. Given a set $\mathcal{U} = \{1, 2, \ldots, n\}$, called the universe, and a collection $\mathcal{S}$ of subsets of the universe whose
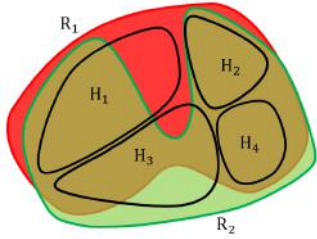
Fig. 8. Orientation label assignment and propagation. (a) Height field H_1 is assigned region $\mathcal{R}_1$'s label since it covers some non-overlapping parts (red) of $\mathcal{R}_1$ and of $\mathcal{R}_1$ only. H_3 is not assigned any label since it covers non-overlapping parts from both $\mathcal{R}_1$ and $\mathcal{R}_2$. (b) H_1 propagates its label to H_2 since H_2 is entirely covered by an overlap (yellow) involving $\mathcal{R}_1$. (c) Unassigned height fields H_3 and H_4 may be split by graph cut.

union equals the universe, the set cover problem is to identify the *smallest* sub-collection (one with the fewest subsets) of $\mathcal{S}$ whose union equals to $\mathcal{U}$. SCP is one of most classic NP-hard problems in combinatorics and computer science [Cormen et al. 2001].

For our accessibility region problem, we consider all the Voronoi cells $c_i, i = \{1, 2, \ldots, n\}$ as the elements of the universe $\mathcal{U}$, and for each sampled orientation P, all of its accessible cells consist of a subset $\mathcal{S}$ of universe $\mathcal{U}$. Then given all sampled orientation P_i and their corresponding subsets $\mathcal{S}_i, i = \{1, 2, \ldots, m\}$, using SCP we could get a minimal number of orientations $P_i, i = 1, 2, .., k$, the union of their corresponding subsets $\mathcal{S}_i, i = 1, 2, \ldots, k$ equals to $\mathcal{U}$. We call such a set of minimal number of orientations MINORI.

To solve the MINORI problem, which is an instance of the *overlapping* SCP, we resort to the greedy scheme from Chvatal [1979]. For a typical freeform 3D object with the prescribed set of sampled orientations, our MINORI solution often include only a handful of orientations. Moreover, solutions with the same orientation count are often not unique. Figure 7 shows three possible MINORI solutions for the KITTEN model. After overlap resolution, the MINIORI solution that leads to the smallest number of machinable patches is selected for tool path planning, after boundary refinement.

Overlap resolution. A MINORI solution typically contains many cells that are accessible from more than one object orientation. Thus, the accessible regions in a MINORI are expected to overlap significantly; see Figure 7. On the other hand, the number of accessible regions in a MINORI gives us the minimum number of object orientations, or fixture setups, for CNC machining. By definition, an accessible region thus obtained can be fully accessed by the CNC, assuming that the CNC cutter can be oriented differently. However, when machining a surface piece, the cutter orientation is fixed in a 3+2-axis setup. For the piece to be fully machinable with that fixed orientation, the piece must be a height field. Hence, to resolve the overlap and obtain a surface decomposition into 3+2-axis machinable patches, we must *integrate* the accessible regions from a MINORI with the height fields computed from pre-segmentation.

Integrating accessible regions and height fields. The integration step produces a surface decomposition for a given MINORI solution by assigning object orientation labels associated with the MINORI to height fields from the pre-segmentation. The label assignment starts

away from the overlaps between active regions and progresses towards the overlaps via *label propagation*. Specifically, we first identify any height field that contains surface points which belong to some non-overlapping part of one and *only one* accessible region; we assign the orientation label associated with this accessible region to the height field; see Figure 8(a). Then we propagate, recursively, orientation labels from assigned height fields to adjacent unassigned ones only when they are entirely covered by an overlap between appropriate accessible regions. For example, height field H_1, with label from accessible region $\mathcal{R}_1$, propagates its label to H_2 only when H_2 is entirely covered by an overlap involving $\mathcal{R}_1$; see Figure 8(b). After the propagation, any remaining unassigned height field may be split by the boundary extraction process we outline next.

The integration step keeps the number of accessible regions or fixture setups fixed, but can split a height field. The above assignment procedure aims to keep such splits to a minimum since the final number of height field pieces (i.e., the machinable patches for tool path planning) corresponds to how many times the 3+2-axis CNC machine needs to be re-oriented.

Boundary extraction and refinement. The orientation label assignments to the height fields already provide a partial set of patch boundaries. To "close the loop" and complete the remaining boundaries, we apply graph cut [Boykov et al. 2001] to split regions corresponding to unassigned height fields along low curvature paths. The graph cut is formulated as an energy minimization defined over the cells $c_i, 1 \leq i \leq m$ in the region of interests. Specifically, we seek a cell assignment r that minimizes the following energy function:

$$E(r) = \sum_{i=1}^{m} D(r(c_i)) + \alpha \sum_{(ij)} S(r(c_i), r(c_j)),$$

where D is the unary data term, S is the pairwise smoothness term, and α provides a trade-off (we set $\alpha = 100$ in our experiments). The data term D estimates the likelihood of c_i to belong an orientation $r(c_i)$ by measuring its distance to a cell with a definite orientation. The smoothness term S measures the curvature of the object surface along the border between c_i and c_j and penalizes high-curvature edges. To simplify the computation, we define S as follows:

$$S(r(c_i), r(c_j)) = \begin{cases} |\mathcal{K}_{ij}| & \text{if } r(c_i) \neq r(c_j), \\ 0 & \text{otherwise,} \end{cases}$$

where $\mathcal{K}_{ij}$ is the normal curvature of the surface at the midpoint of the edge (c_i, c_j) in the tangent direction perpendicular to the edge (c_i, c_j). Finally, the combined patch boundaries are smoothed by geometric snakes [Lee and Lee 2002].

Final selection of surface decomposition. As described above, each MINORI solution would lead to a surface decomposition into machinable patches. We select one result which contains the least number of patches, and if there are ties, boundary quality is the tie breaker. Tool paths are computed for the final set of machinable patches.

5 TOOL PATH PLANNING

Once the surface is decomposed into a collection of surface patches, a tool path plan is designed for each patch. This tool path planning is not as simple as generating an equally-spaced filling curve, and
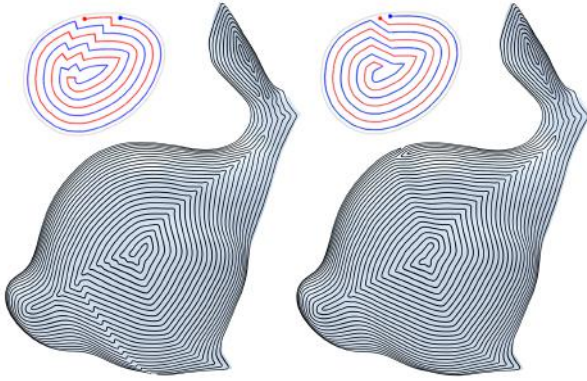
Fig. 9. Replacing zigzag re-routing connections (see top-left inset, which is taken from Figure 5(c) of [Zhao et al. 2016]) with "oblique" lines that conform better to tangential directions of the iso-contours, as shown in the top-right inset. This can effectively reduce the number of sharp turns in the final tool paths (after tool path optimization); see the before (left) and after (right) pictures of the bunny patch and focus on the lower left region.

commonly it is designed by skillful professionals in an ad hoc manner. For carving the surface, one needs to plan the path of a ball-end cutter, which has some physical prescribed radius, and hence implies two key requirements:

- *Consideration of scallop.* The problem is that equally-spaced curves may not necessarily lead to a uniform scallop distribution. To obtain uniform scallop on a surface, the gap between two neighboring paths needs to be adaptive to the directional curvatures of the points along two nearby paths. This requirement is the most distinct feature of this tool path planning problem.
- *Smoothness.* Generally speaking, a smooth tool path is preferred in practice due to the upper limit of velocity and the acceleration of the cutter. A zigzag path not only slows down the cutter, but also tends to cause damage to the cutter.

Based on these two requirements, we design a three-step algorithm for generating the final tool path: (1) compute a shape aware scalar field whose isolines meet the gap requirement, (2) connect the isolines into a continuous tool path using the Fermat spiral generation technique, and (3) smooth the tool path while keeping the gap varying as small as possible. In the following, we shall elaborate the details of Step 1 and Step 3. For details on Step 2, we refer the reader to the recent work by Zhao et al. [2016], in particular, Figures 4 and 5 in that paper. In this work, we apply a simple trick to alter the way the iso-contours are re-routed to obtain the Fermat spirals, effectively reducing the number of sharp turns in the tool paths (as shown in Figure 9) and improving machining speed. Specifically, instead of making zigzag connections which would result in many right-angle turns, we replace them with short "oblique" curves which conform better to the *tangential* directions of the iso-contour at the re-routing points; see insets in Figure 9 for an illustration.

5.1 Shape-aware tool path generation

As mentioned above, the tool path design is constrained by geometry variations of the points along the path. More precisely, the scallop h is deemed to have close relationship with the directional curvature [Kim et al. 2006]. Let p be a point on a surface S and path Π that goes through p. Following [Kim et al. 2006], the dependency between the scallop h and the gap g between adjacent sections of the path is empirically formulated as

$$g(p, \Pi) = \sqrt{\frac{8hR_{\text{cutter}}}{1 + R_{\text{cutter}}\mathbf{G}(p, \Pi)}}, \quad R_{\text{cutter}} \gg h, \quad (1)$$

where $\mathbf{G}(p, \Pi)$ is the curvature of p at the direction perpendicular to the forward direction of tool path Π at point p. It is difficult to generate the tool path directly from this formula because one cannot determine the gaps before the continuation of the generated path is defined.

Our key idea when computing the tool path is to obtain a shape aware metric tensor field $\mathbf{g}$ on the surface from the directional curvature tensor field $\mathbf{G}$, and use its isolines as the tool paths with the required uniform scallop. Once the metric field is defined, the boundary ∂S is set to be the zero-level isoline, and then the other isolines are iteratively defined, with respect to $\mathbf{g}$, by increasing the geodesic distance to the boundary ∂S by g during each step. We recall that a fast marching method can be used for this purpose. After the isoline L_i has been extracted, it can be used to generate L_{i+1} by considering the projected metric tensor $\mathbf{g}|_{L_i}$ at L_i. Note that $\mathbf{g}|_{L_i}$ should be orthogonal to L_i. However, directly setting it is not easy since L_{i+1} has to be used to estimate $\mathbf{g}|_{L_i}$ but L_{i+1} is still undetermined before $\mathbf{g}|_{L_i}$ is known. Alternatively, we solve this problem by the following iterative optimization approach.

As the output of the above PDE, the isolines of the resulting geodesic distance field is helpful in setting the desired tool paths. Moreover, we use the gap g defined in Eq. 1 at each point of the surface as a modifier to adjust the distance value in the field. This makes the gap between two adjacent paths vary from the directional curvature of the path points. The gradient direction at each point of the distance field is used to compute directional curvature. Recall that the heat method [Crane et al. 2013b] uses a PDE to solve the geodesic problem on a mesh surface S, and the heat, after diffusion for a short period t, provides a good approximation to the gradients of the real geodesic distance field. Inspired by the approximation power of the heat based method, we apply a similar approach to the PDE:

$$(A - tL_C \otimes \mathrm{H})u = \delta_\gamma,$$

where A is the diagonal matrix given by the triangle areas, $A^{-1}L_C$ defines the Laplacian matrix, the Dirac function δ_γ provides an initial heat distribution rooted at the boundary, and the symmetric matrix H associates each mesh edge with the average of the gaps at the two end points of the edge. Note that H is absorbed into this PDE by an element-wise product.

To make the gap between isolines of the geodesic distances more consistent with the metric tensor $\mathbf{g}$, we iteratively update the direction-related gap matrix H based on the values obtained in the previous iteration. As shown in Figure 10, our experiments confirm that the iteration converges extremely fast and in practice, only
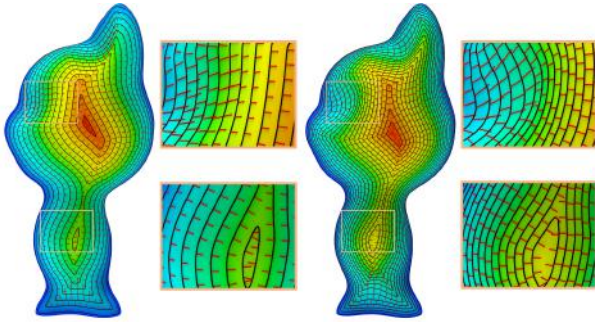
Fig. 10. The iterations of tool paths optimization: the black curves are the tool paths generated from the isolines of the surface geodesic distance field. The short red lines represent the desired gaps computed at the points sampled along the tool paths according to Eq. 1. From left to right, two time steps are depicted, showing that the gaps between two adjacent tool paths are quickly optimized to be consistent with the desired gaps and stable.
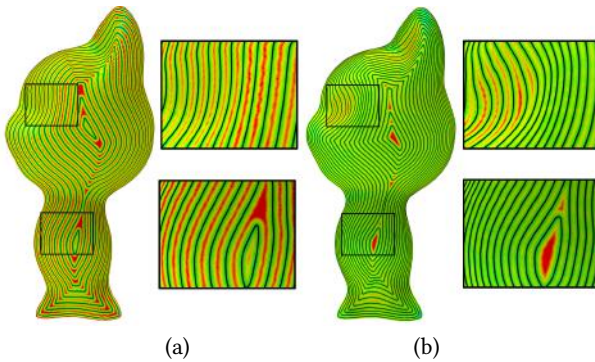

(a)　　　　　　　　(b)

Fig. 11. Scallop (shown in red color on the machined surface) with uniform gap of tool paths (a) and with adaptive gap of tool paths (b).

$2\sim3$ iterations are sufficient to obtain a scalar field whose isolines meet the direction-related gap requirements. Figure 11 shows the resulting scallop reduction by adaptive gaps between two adjacent tool paths.

Note that while our tool path generation problem is directly related to geodesic distances, it is a bit different from a general distance field problem. We have to iteratively solve the problem, i.e., to update the existing distance field by considering the gap requirements of its iso-contours. However, during each iteration, we can use either the heat equation or other, possible better alternatives, such as Short-Term Vector Dijkstra (STVD) of Campen et al. [2013].

5.2 Tool path refinement

The initial tool path must be refined by considering two important spacing constraints: (i) the widest space at p, between adjacent sections of the path Π, cannot exceed the gap constraint $g(p, \Pi)$ (written as g for simplicity), or equivalently, the largest empty circle has a radius not larger than $g/2$, and (ii) the narrowest space at p is as close as possible to g. Let $\{\mathbf{x}_i\}_{i=1}^n$ be the point sequence that represents the tool path. The idea is to evolve the initial tool path

by optimizing the following objective function:

$$\frac{d\mathbf{x}_i}{dt} = \lambda_1 \times \mathbf{T}_{\text{Smooth}} + \lambda_2 \times \mathbf{T}_{\text{Attraction}} + \lambda_3 \times \mathbf{T}_{\text{Repulsion}},$$

where $\mathbf{T}_{\text{Smooth}}, \mathbf{T}_{\text{Attraction}}, \mathbf{T}_{\text{Repulsion}}$ characterize the smoothness requirement, the attraction of $\mathbf{x}_i$ to the centers of nearby large empty circles, and the repulsion between $\mathbf{x}_i$ and its nearby points on the path, respectively. Figure 12 shows a sequence of intermediate results (note that the time t can be understood as the number of iterations). On the right, plot shows the widest/narrowest spaces as function of time. We use a conventional Laplacian smoothing technique to express the smoothness requirement.

For the attraction term, we first find a collection of empty circles such that (i) each circle has a radius larger than $g/2$ and (ii) any two circle centers have a distance of at least g. We call such points $\{\mathbf{q}_j\}_{j=1}^{K_1}$ *anchors*; See the red points in Figure 12. The term $\mathbf{T}_{\text{Attraction}}$ attracts the path to the nearby anchors, where each anchor has an influence geodesic disk with a radius of $3g/2$. The repulsion term is applied to $\mathbf{x}_i$ if there are other points $\{\mathbf{x}_j\}_{j=1}^{K_2}$ along the path for which the geodesic in between, i.e. $\|d_{i,j}\|_g$, is less than g. See Appendix for more details on the definitions of $\mathbf{T}_{\text{Smooth}}, \mathbf{T}_{\text{Attraction}}, \mathbf{T}_{\text{Repulsion}}$.

The above optimization is applied on a finite set of points sampled over the surface. In our current experiments for CNC machining, the size of the real models is about $50 \times 60 \times 70 mm^3$. For such models, we found that a sampling of $80K$ points is sufficient. Hence, we precompute $80K$ blue noise points on the surface to serve as the anchor point candidates. For larger models, we can adaptively adjust the number of sample points according to the model size, treating it as a scaling issue.

During each iteration, for each candidate point q_j, we compute the largest empty circle centered at q_j and keep its radius r_j for q_j. The candidate points are then selected to form an anchor set in a decreasing order of the radius to define a largest empty circle. Two criteria for candidate selection include (i) the radius r_j is larger than $g/2$, and (ii) the newly added anchor point have a distance, of at least g, to the selected anchor points.

It should be noted that the optimization is performed directly on the mesh surface, rather than in the parameter space. During each step, the point $\mathbf{x}_i$ on the tool path moves along the tangent plane and is then projected onto the surface using the proximity query package (PQP) [Gottschalk et al. 1999]. The geodesic distance between $\mathbf{x}_i$ and $\mathbf{x}_j$, i.e. $\|d_{i,j}\|_g$, is approximated by following formula [Bowers et al. 2010]:

$$\|d_{i,j}\|_g = \frac{\arcsin(\mathbf{n}_i \cdot \bar{d}_{i,j}) - \arcsin(\mathbf{n}_j \cdot \bar{d}_{i,j})}{\mathbf{n}_i \cdot \bar{d}_{i,j} - \mathbf{n}_j \cdot \bar{d}_{i,j}} \|d_{i,j}\|,$$

where $\mathbf{n}_i$ and $\mathbf{n}_j$ are respectively the normal vectors at $\mathbf{x}_i$ and $\mathbf{x}_j$, $d_{i,j} = \mathbf{x}_i - \mathbf{x}_j$, $\bar{d}_{i,j}$ is the normalization of $d_{i,j}$. Then, the estimation formula reduces to

$$\|d_{i,j}\|_g \approx \frac{1}{\sqrt{1 - (\mathbf{n}_i \cdot \bar{d}_{i,j})^2}} \|d_{i,j}\|$$

if $\mathbf{n}_i$ and $\mathbf{n}_j$ are very close to each other.

Note that currently, our tool path refinement scheme cannot add or remove spirals, which could potentially be beneficial. We leave this investigation for future considerations.

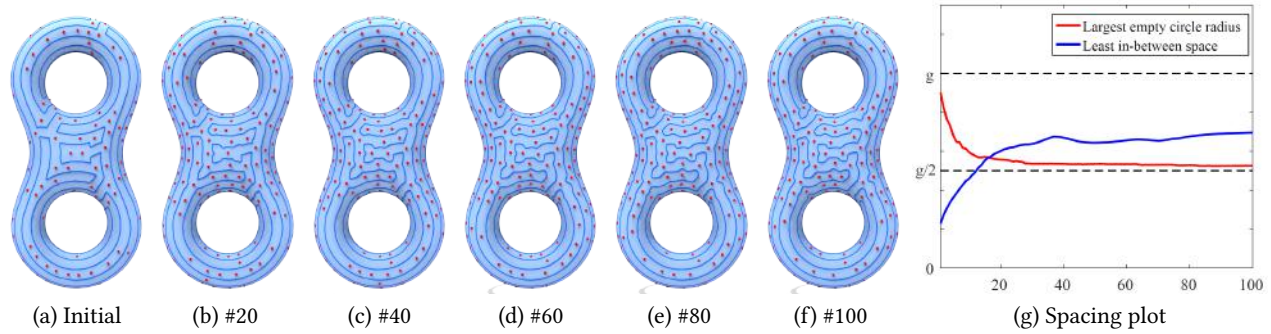| (a) Initial | (b) #20 | (c) #40 | (d) #60 | (e) #80 | (f) #100 | (g) Spacing plot |

Fig. 12. Tool path optimization. Guided by the smoothness/spacing constraints, the initial tool path is evolved into a smooth pseudo-geodesic spiral with at most $(g/2 + \epsilon)$-wide space on both sides after 100 iterations, where $g/2$ is the sweeping radius of the cutter. That is to say, when the program terminates, the largest empty circle has a radius that is very close to $g/2$ (the red points are the centers of the typical empty circles). The plot (g) shows the change of the largest empty circle radius and the least in-between space of the tool path. In other words, the cutter is able to cover the entire surface if following the optimized tool path. (Note that g varies on the surface in practice due to shape variation. We normalize g for the visualization purpose in the plot (g).
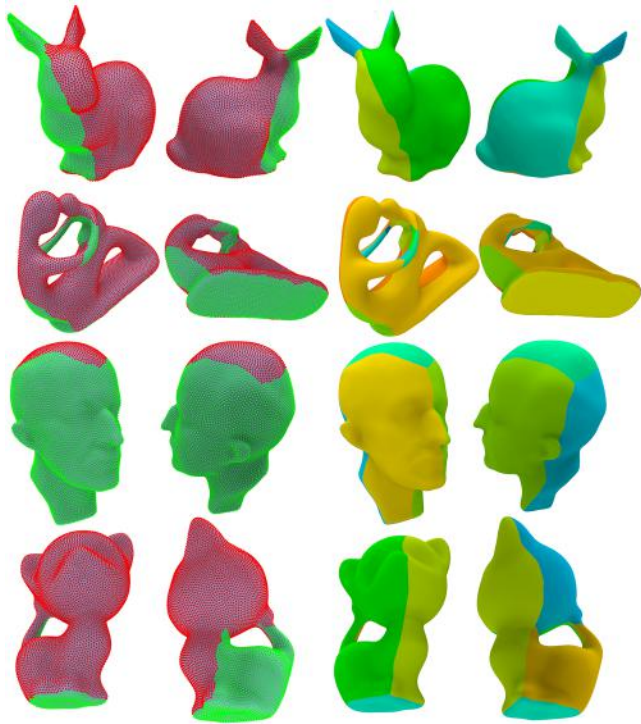
6 RESULTS AND EVALUATION

Fig. 13. A gallery of surface decomposition results for 3+2-axis machining. For each model in each row, the first two images show the accessible regions obtained after overlap resolution in two different views; the next two images show the final machinable patches obtained in two views.

In this section, we show surface decomposition and tool path generation results for freeform 3D shapes with varying degrees of geometric complexity. Comparisons to conventional tool paths, i.e., zigzag and contour-parallel, for CNC machining are provided to evaluate our iso-scallop space filling curves using Fermat spirals.

We also report real machining times and show fully machined 3D objects using a 3+2 machine with machining setups and tool paths planned by our fully automatic method.

Implementation and parameters. Our surface decomposition and tool path generation methods have both been implemented in C++. We set the cutter diameter at 4.0mm during height field decomposition and for defining the accessibility cones. We produce physical machining of full 3D objects with high-quality surface finishing, setting the scallop height at 0.02mm. To make the carving paths more visible for visualization purposes only, we relax the scallop height to 0.045mm when machining some surface patches. All the results shown in this section were obtained with the same parameter setting: four iterations of the heat method and 40 iterations of tool path generation optimization.

Decomposition. Figure 13 shows results of our accessible surface decompositions as well as the final machinable patches obtained after integrating the accessible regions with the pre-segmentation into height fields suitable for 3+2 machining. Table 1 shows some statistics related to the surface decomposition step, including running times of the sub-steps. Running times were measured on an
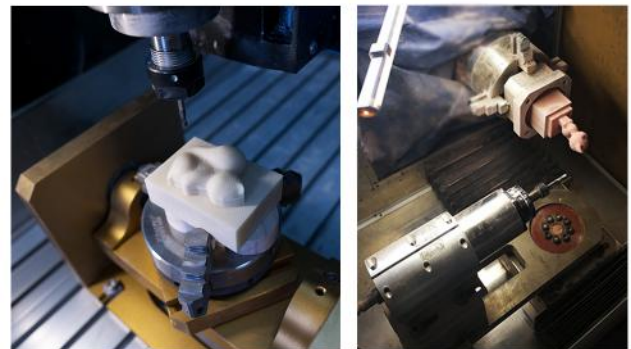

Fig. 14. Photographs capturing a couple of fixture configurations applied to stabilize the 3D objects during CNC machining.

Table 1. Some statistics and running times for our surface decomposition and tool path generation. We report the number of accessible regions (#A) and the number of machinable patches (#P) after integrating the height fields and accessible regions and boundary optimization. For running times (all in seconds), we report time needed for computing accessibility covers (t_A), the total time for the surface decomposition phase (t_P), and average time for computing tool paths for each machinable patch ($\bar{t}_P$).

3D Model	#A	#P	t_A(s)	t_P(s)	$\bar{t}_P$(s)
RABBIT	2	4	14.2	17.5	5.6
SQUIRREL	2	5	17.5	21.0	5.7
BUNNY	2	5	18.3	21.1	4.3
KITTEN	2	5	24.2	28.4	6.2
MAXPLANCK	2	4	27.1	30.5	6.0
FERTILITY	2	11	48.9	57.2	4.8

Intel® Core™ i7-7700 CPU 4.2GHz with 16GB RAM. For geometrically complex models such as the fertility which also has a non-zero genus, our algorithm is able to obtain a small number of accessible regions and for each region, a small number of patches that can be machined by a 3+2 machine without changing the fixture. Figure 14 provides two photographs demonstrating the fixtures applied to stabilize the shape blocks for real machining.

Tool path generation. Figures 1 and 15 show continuous iso-scallop Fermat spiral paths generated by our algorithm for several curved surface patches obtained from the decomposition step. The patches exhibit varying geometric characteristics in their boundaries and interiors to demonstrate the generality of our tool path planning method. To make the carving paths more visible, we deliberately chose a low resolution setting. Scallop heights for some of the patches in Figure 15 can be visualized in Figure 16, left column, and compared to zigzag (middle) and contour-parallel (right). Real machining results can be found in Figure 17. Average running times for computing the spiral tool paths for each patch are reported in the last column of Table 1. As we can see, our tool path planning scheme is quite efficient.

Real machining. Our real machining experiments have been conducted on a CNC 6040 2200W 5-axis machine, with machinable resin board as the testing material to form solid 3D objects. CNC cutting results and analyses are based on the default machine setting: cutter diameter at 4.0mm, maximal feed rate at 500mm/min, chord error at 0.001mm, and spindle speed at 15,000r/min. G-code is used to transfer the tool paths.

Figure 17 shows several photographs taken of the real machining results obtained for several freeform 3D objects of varying geometric complexity. Close-ups are provided to show the carving paths and scallops from our iso-scallop Fermat spirals. Please also check out the supplementary video to see the Fermat spirals in action.

Comparison to conventional tool paths. The two most frequently adopted tool path patterns for CNC machining are zigzag and contour parallel (also referred to as iso-contour) paths. In Figure 16, we compare visually the scallop height distributions for zigzag paths, contour-parallel paths, and Fermat spiral paths generated by our method. All the zigzag and contour-parallel paths were generated with the NX package from Siemens PLM Software [2016].
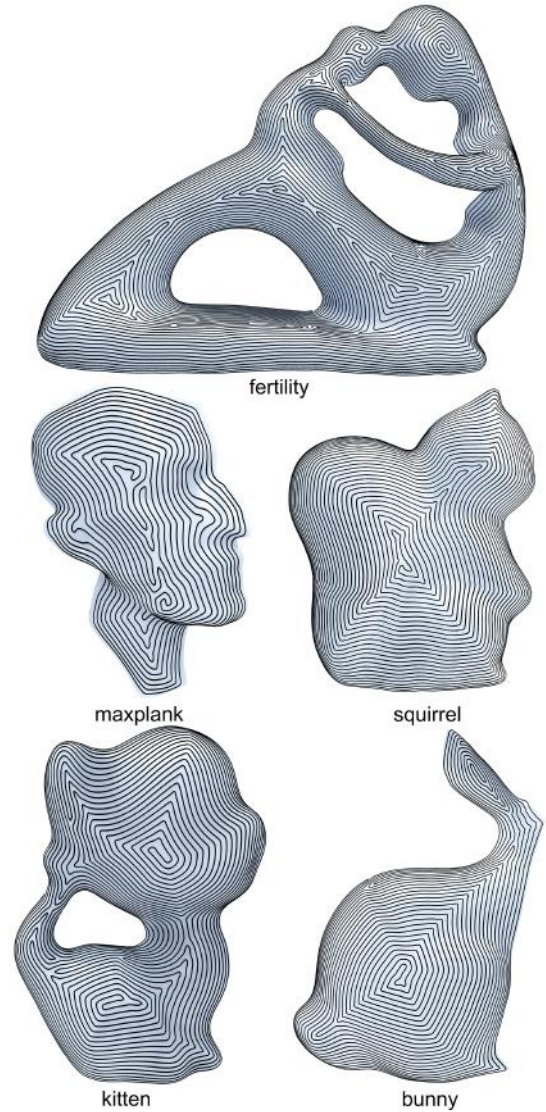

Fig. 15. Continuous iso-callop Fermat spirals generated by our method, over patches with diverse geometric characteristics. To ease visualization, we show carving paths obtained at a low resolution.

Siemens NX, formerly known as NX Unigraphics, is a high-end CAD/CAM/CAE software package. Overall, our results exhibit a higher degree of height uniformity with the same path spacing.

Table 2 shows statistics collected for the three kinds of tool paths, including real machining time using the CNC 6040 2200W 5-axis machine. The tested patches are shown in Figure 15 and Figure 2. Unlike the Fermat spiral paths we produce, zigzag and contour-parallel paths are not always able to cover an entire patch using a single traversal, resulting more than one tool path segments. The segment counts are expected to increase when the patch boundary is wavier or contain more thin structures or multiple components, as in the case of the FERTILITY and KITTEN patches with holes. In terms of
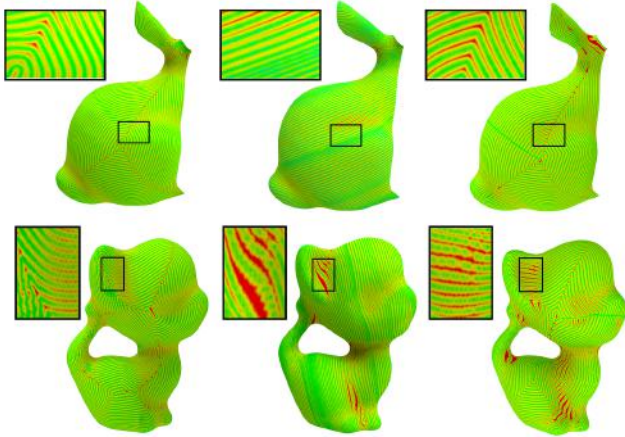
Fig. 16. Visualizing scallop heights over several machined patches using our Fermat spiral paths (left), conventional zigzag paths (middle), and contour-parallel paths (right). Redish regions indicate higher residual marks.

Table 2. Comparing zigzag (Z) and contour-parallel (C) tool paths generated by commericial software packages to iso-scallop Fermat spirals (F) generated by our method. We report results on patches shown in Figure 15 using the following statistics: number of tool path segments (#sgZ, #sgC, and #sgF); percentage of sharp turn points (%tnZ, %tnC, and %tnF), and real machining time in seconds (t_Z, t_C, and t_F), using the CNC 6040 2200W machine.

Patch	#sgZ	#sC	#sgF	%tnZ	%tnC	%tnF	t_Z	t_C	t_F
#1 (BUNNY)	9	4	**1**	7.1%	4.7%	**1.5%**	450	368	**342**
#2 (FERTILITY)	18	6	**1**	6.6%	4.0%	**3.8%**	1908	1054	**1034**
#3 (MAXPLANK)	5	**1**	**1**	7.6%	6.0%	**2.5%**	245	232	**205**
#4 (SQUIRREL)	6	**1**	**1**	6.0%	2.8%	**1.9%**	539	428	**416**
#5 (KITTEN)	11	2	**1**	7.4%	3.7%	**2.8%**	469	381	**370**

real machining time, our iso-scallop Fermat spirals generally outperform zigzag and contour-parallel paths, while the improvement over contour-parallel paths is more marginal.

To count the number of sharp turns along a tool path π, we uniformly sample 50, 000 points along π and at each point, we estimate its integral curvature [Pottmann et al. 2009] with a circle of radius 0.2mm, which is appropriate for the size of fabricated layers and the default fill with in our experiments. A point is deemed to represent a sharp turn if the smaller of its associated area coverage for curvature estimation is less than 30% of the circle area. In Table 2, we report the percentages of points deemed as sharp turns. It is quite evident that the number of sharp turns produced by connected Fermat spirals is much lower than that of zigzags.

7 DISCUSSION, LIMITATION, AND FUTURE WORK

Subtractive processes via CNC machining still dominate the manufacturing industry today. The CNC setup and tool path planning problem is rather complex in practice due to the multitude of factors that are at play. Most, if not all, of the factors involve geometry optimization spanning a diversity of forms and search spaces. The method we present focuses on two particular aspects of the problem: accessibility decomposition and efficient iso-scallop tool path generation, aiming for a fully automatic optimization. Results and

comparisons to conventional tool paths demonstrate the effectiveness of our decompose-and-spiral-carve (DSCarver) approach.

The original Fermat spirals of Zhao et al. [2016] were developed for layered manufacturing. Our extension to curved surfaces shares all the desirable properties offered by this class of space-filling curves. In fact, Fermat spirals appear to be even more suited to CNC machining than to 3D printing via fused deposition modeling. The particular characteristics of Fermat spirals imply that gaps between tool paths may take relative long to fill. As a result, FDM using these tool paths may suffer from reduced material cohesion between adjacent and parallel tool paths since the long delay can cause material to cool down. Such issues are not encountered during CNC machining. In our work, we show that Fermat spirals can be adapted to produce iso-scallop carving paths for CNC.

Engineering/CAD parts. We reiterate that the goal of our current work is to automate CNC machining of free-form 3D objects, not typical CAD/engineering parts. Also, Fermat spirals excel as tool paths for surface patches with wavy boundaries and irregular interior undulations, which are not characteristic of CAD models. Decompositions of CAD models with sharp features should respect these features, but neither the height field decomposition nor our accessibility analysis is feature-sensitive. In Figure 18, we show surface decompositions and tool paths obtained by our method on a CAD part. Clearly, the sharp features of the part would have been better accentuated if the tool paths were to conform to these features; this would require a feature-sensitive decomposition. One possibility would be to replace the height field decomposition with a feature-sensitive one, and we leave this for future work.

Practical CNC machining issues. DSCarver overlooks several such issues pertinent to CNC machining, including fixture design, cutter switching, and rough- vs. finish-stage machining. We also do not address situations of inaccessibility, e.g., tunnels or hollow parts, that are due to the physical (e.g., size) limitation of the cutter. In our work, we focus on finish-stage machining with a fixed cutter width and without factoring in constraints arising from fixture design. How a 3D object is clamped on object orientation. In other words, fixture placement is a factor that should be incorporated into tht MINORI problem. At the same time, fixtures affect cutter accessibility, namely, any surface regions that are attached to or covered by the fixture would be inaccessible. In general, fixture design is a highly non-trivial geometry optimization problem [Hazarika et al. 2015] and deserves separate investigation; it does seem to share some commonality with connector designs [Koyama et al. 2015]. Last but not the least, we only consider 3+2 machining where the CNC cutter holds a fixed orientation. Tool planning for full 5-axis CNC would involve a full five-dimensional search.

Rough surfaces. We experimented our machining algorithm mostly on locally smooth surfaces. For a 3D object whose surfaces are filled with many small concavities, small areas of the surface may not be reachable by the cutter. In such situations, CNC can benefit from approximating the surface with few height fields to reduce the number of 3+2-axis setups for machining. This is a strong merit of the work by Herholz et al. [2015] and is applicable here.
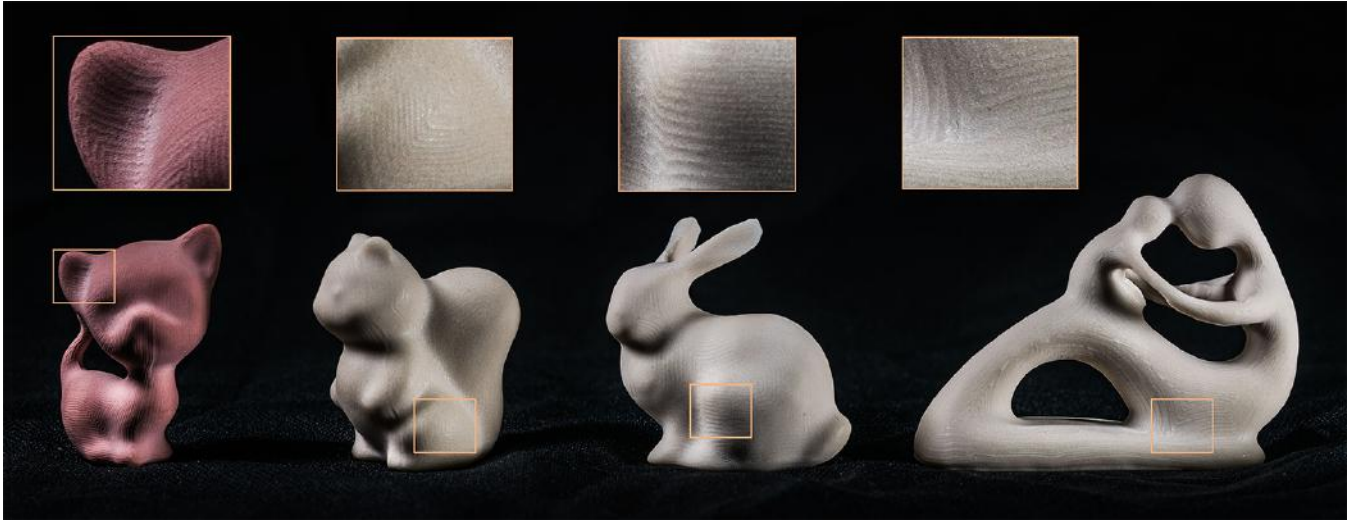
Fig. 17. Photographs and close-ups of real machining results for full 3D objects, following our fully automatic method for surface decomposition and tool path planning. The results were obtained using a CNC 6040 2200W 5-axis machine, with machinable resin board as the testing material.
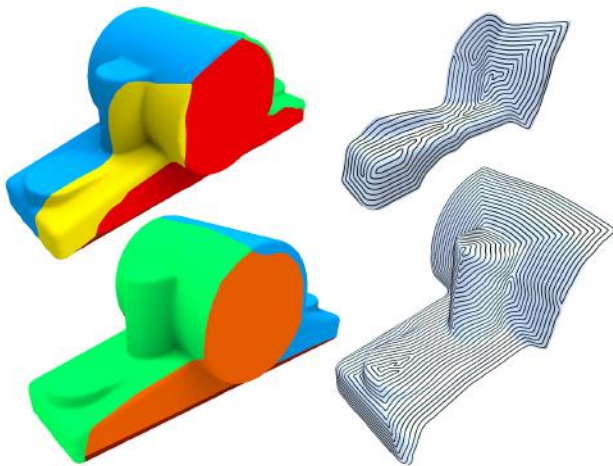

Fig. 18. Surface decomposition (two views are shown) and tool path planning results on an engineering/CAD part obtained by our method.

Global continuity of carving path. Our current method does not produce a globally continuous carving path over the whole accessibility region; the cutter needs to retracted to adjust to a different 3+2-axis setup when switching between different machinable patches. Global continuity may be possible using a full 5-axis CNC machine as its cutter head can move in five degrees of freedom, but this is a new tool planning problem. Another possibility is to only apply the 5-axis pass during the transitioning phase between machinable patches. Both problems are worth investigating in future work.

Future work. An obvious next step is to investigate the applicability of DSCarver for rough-stage machining, where the distinction, as well as challenge, is that the shape to be carved changes after each cutter pass over the surface. Along the same lines, it would

be interesting to integrate our algorithm to existing practices from state-of-the-art CAD/CAM systems. Finally, automatic fixture design which combines accessibility and machinability analyses and globally continuous iso-scallop Fermat spirals under full 5-axis CNC machining are both intriguing problems to explore.

ACKNOWLEDGMENTS

REFERENCES

Rajneesh Kumar Agrawal, D. K. Pratihar, and Roy Choudhury. 2006. Optimization of CNC isoscallop free form surface machining using a genetic algorithm. *The International Journal of Advanced Manufacturing Technology* 46, 7-8 (2006), 811–819.

Saleh M Amaitik and S Engin Kiliç. 2007. An intelligent process planning system for prismatic parts using STEP features. *The International Journal of Advanced Manufacturing Technology* 31, 9-10 (2007), 978–993.

Karlo Apro. 2008. *Secrets of 5-axis machining.* Industrial Press Inc.

Amit H Bermano, Thomas Funkhouser, and Szymon Rusinkiewicz. 2017. State of the Art in Methods and Representations for FabricationâĂŘAware Design. *Computer Graphics Forum* 36, 2 (2017), 509–535.

Zhang Wei Bo, Lu Zhen Hua, and Zhu Guang Yu. 2006. Optimization of process route by genetic algorithms. *Robotics and Computer-Integrated Manufacturing* 22, 2 (2006), 180–188.

John Bowers, Rui Wang, Li Yi Wei, and David Maletz. 2010. Parallel Poisson disk sampling with spectrum analysis on surfaces. *ACM Transactions on Graphics* 29, 6 (2010), 166.

Yuri Boykov, Olga Veksler, and Ramin Zabih. 2001. Fast Approximate Energy Minimization via Graph Cuts. *IEEE Trans. Pat. Ana. & Mach. Int.* 23, 11 (2001), 1222–1239.

Marcel Campen, Martin Heistermann, and Leif Kobbelt. 2013. Practical anisotropic geodesy. In *Computer Graphics Forum*, Vol. 32. Wiley Online Library, 63–71.

Ahmet Can and Ali Ünüvar. 2010. A novel iso-scallop tool-path generation for efficient five-axis machining of free-form surfaces. *The International Journal of Advanced Manufacturing Technology* 51, 9-12 (2010), 1083–1098.

Xuelin Chen, Hao Zhang, Jinjie Lin, Ruizhen Hu, Lin Lu, Qixing Huang, Bedrich Benes, Daniel Cohen-Or, and Baoquan Chen. 2015. Dapper: Decompose-and-pack for 3D Printing. *ACM Trans. on Graph* 34, 6 (2015), 213:1–213:12.

B. Choi and R. Jerrard. 1998. *Sculptured Surface Machining: Theory and Applications.* Kluwer Academic Publishers.

V. Chvatal. 1979. A Greedy Heuristic for the Set-Covering Problem. *Mathematics of Operations Research* 4, 3 (1979), pp. 233–235. http://www.jstor.org/stable/3689577

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2001. *Introduction to Algorithms.* MIT Press and McGraw-Hill.

Keenan Crane, Clarisse Weischedel, and Max Wardetzky. 2013a. Geodesics in Heat: A New Approach to Computing Distance Based on Heat Flow. *ACM Trans. on Graph* 32, 5 (2013), 152:1–152:11.

Keenan Crane, Clarisse Weischedel, and Max Wardetzky. 2013b. Geodesics in heat: A new approach to computing distance based on heat flow. *ACM Transactions on Graphics* 32, 5 (2013), 13–15.

Donghong Ding, Zengxi Stephen Pan, Dominic Cuiuri, and Huijun Li. 2014. A tool-path generation strategy for wire and arc additive manufacturing. *Int. J. of Adv. Manufact. Tech.* 73, 1-4 (2014), 173–183.

R. T. Farouki. 2016. Rational rotation-minimizing frames — Recent advances and open problems. *Appl. Math. Comput.* 272 (2016), 80–91.

R. T. Farouki and S. Li. 2013. Optimal tool orientation control for 5-axis CNC milling with ball-end cutters. *Computer Aided Geometric Design* 30, 2 (2013), 226–239.

Matthew C Frank, Richard A Wysk, and Sanjay B Joshi. 2006. Determining setup orientations from the visibility of slice geometry for rapid computer numerically controlled machining. *Journal of manufacturing science and engineering* 128, 1 (2006), 228–238.

Stefan Gottschalk, Ming Lin, Dinesh Manocha, and Eric Larsen. 1999. PQP–the proximity query package. *Accessed October* 4 (1999), 2011.

YW Guo, Antony R Mileham, Geraint W Owen, Paul G Maropoulos, and WD Li. 2009. Operation sequencing optimization for five-axis prismatic parts using a particle swarm optimization approach. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 223, 5 (2009), 485–497.

Prosenjit Gupta, Ravi Janardan, Jayanth Majhi, and Tony Woo. 1996. Efficient geometric algorithms for workpiece orientation in 4-and 5-axis NC machining. *Computer-Aided Design* 28, 8 (1996), 577–587.

Steffen Hauth and Lars Linsen. 2012. Double-spiral tool path in configuration space. *The International Journal of Advanced Manufacturing Technology* 54, 9 (2012), 1011–1022.

Manjuri Hazarika, Uday Shanker Dixit, et al. 2015. *Setup planning for machining.* Springer.

Martin Held and Christian Spielberger. 2014. Improved Spiral High-Speed Machining of Multiply-Connected Pockets. *Computer-Aided Design and Applications* 11, 3 (2014), 346–357.

Philipp Herholz, Wojciech Matusik, and Marc Alexa. 2015. Approximating Free-form Geometry with Height Fields for Manufacturing. *Computer Graphics Forum (Eurographics)* 34, 2 (2015), 239–251.

Kristian Hildebrand, Bernd Bickel, and Marc Alexa. 2013. Orthogonal slicing for additive manufacturing. *Computers & Graphics* 37, 6 (2013), 669 – 675. Shape Model. Intl. (SMI) Conf.

Ruizhen Hu, Honghua Li, Hao Zhang, and Daniel Cohen-Or. 2014. Approximate Pyramidal Shape Decomposition. *ACM Trans. on Graph* 33, 6 (2014), 213:1–213:12.

Ralph L Keeney and Howard Raiffa. 1993. *Decisions with multiple objectives: preferences and value trade-offs.* Cambridge university press.

Su Jin Kim, Dong Yoon Lee, Hyun Chul Kim, Sung Gun Lee, and Min Yang Yang. 2006. CL surface deformation approach for a 5-axis tool path generation. *International Journal of Advanced Manufacturing Technology* 28, 5-6 (2006), 509–517.

Yuki Koyama, Shinjiro Sueda, Emma Steinhardt, Takeo Igarashi, Ariel Shamir, and Wojciech Matusik. 2015. AutoConnect: Computational Design of 3D-printable Connectors. *ACM Trans. on Graph* 34, 6, Article 231 (2015), 231:1–231:11 pages.

Ali Lasemi, Deyi Xue, and Peihua Gu. 2010. Recent development in CNC machining of freeform surfaces: A state-of-the-art review. *Computer-Aided Design* 42, 7 (2010), 641–654.

Yunjin Lee and Seungyong Lee. 2002. Geometric Snakes for Triangular Meshes. *Computer Graphics Forum* 21, 3 (2002), 229–238.

Lin Lu, Andrei Sharf, Haisen Zhao, Yuan Wei, Qingnan Fan, Xuelin Chen, Yann Savoye, Changhe Tu, Daniel Cohen-Or, and Baoquan Chen. 2014. Build-to-Last: Strength to Weight 3D Printed Objects. *ACM Trans. on Graph* 33, 4 (2014), 97:1–97:10.

Linjie Luo, Ilya Baran, Szymon Rusinkiewicz, and Wojciech Matusik. 2012. Chopper: Partitioning Models into 3D-Printable Parts. *ACM Trans. on Graph* 31, 6 (2012), 129:1–129:9.

Debananda Misra, V Sundararajan, and Paul K Wright. 2005. Zig-zag tool path generation for sculptured surface finishing. In *Geometric and algorithmic aspects of computer-aided design and manufacturing: DIMACS workshop computer aided design and manufacturing*, Vol. 67. 265.

Sang C Park, Chang Y C, and Byoung Kyu Choi. 2003. Contour-parallel offset machining without tool-retractions. *Computer-Aided Design* 35, 9 (2003), 841–849.

Helmut Pottmann, Johannes Wallner, Qi-Xing Huang, and Yong-Liang Yang. 2009. Integral invariants for robust geometry processing. *Comp. Aided Geom. Design* 26, 1 (2009), 37–60.

Romain Prévost, Emily Whiting, Sylvain Lefebvre, and Olga Sorkine-Hornung. 2013. Make It Stand: Balancing Shapes for 3D Fabrication. *ACM Trans. on Graph* 32, 4 (2013), 81:1–81:10.

Fei Ren, Yuwen Sun, and Dongming Guo. 2009. Combined reparameterization-based spiral toolpath generation for five-axis sculptured surface machining. *Int. J. of Adv. Manufac. Tech.* 40, 7 (2009), 760–768.

Ariel Shamir. 2008. A survey on Mesh Segmentation Techniques. *Computer Graphics Forum* 27, 6 (2008), 1539–1556.

Siemens. 2016. NX Software. http://www.plm.automation.siemens.com/en_us/products/nx/index.shtml. (2016).

Ondrej Stava, Juraj Vanek, Bedrich Benes, Nathan Carr, and Radomír Měch. 2012. Stress relief: improving structural strength of 3D printable objects. *ACM Trans. on Graph* 31, 4 (2012), 48:1–48:11.

Y-J Tseng and SB Joshi. 1998. Recognition of interacting rotational and prismatic machining features from 3D mill-turn parts. *International Journal of Production Research* 36, 11 (1998), 3147–3165.

Yu Wang, Kai-Min Yu, and Charlie CL Wang. 2015. Spiral and conformal cooling in plastic injection molding. *Computer-Aided Design* 63 (2015), 1–11.

Nuo Xu, Samuel H Huang, and Y Kevin Rong. 2007. Automatic setup planning: current state-of-the-art and future perspective. *International journal of manufacturing technology and management* 11, 2 (2007), 193–208.

Y Yang, HT Loh, JYH Fuh, and YG Wang. 2002. Equidistant path generation for improving scanning efficiency in layered manufacturing. *Rapid Prototyping Journal* 8, 1 (2002), 30–37.

Haisen Zhao, Fanglin Gu, Qi-Xing Huang, Jorge Garcia, Hao Zhang, Daniel Cohen-Or, Yong Chen, Changhe Tu, and Baoquan Chen. 2016. Connected Fermat Spirals for Layered Fabrication. *ACM Trans. on Graph* 35, 4 (2016).

J. Zhao, B. Zhong, and Q. Zou. 2013. Tool orientation planning for five-axis CNC machining of open free-form surfaces. *Journal of Systems Science and Complexity* 26, 5 (2013), 667–675.

Bo Zhou, Jibin Zhao, Lun Li, and Renbo Xia. 2016. A smooth double spiral tool path generation and linking method for high-speed machining of multiply-connected pockets. *Precision Engineering* (2016).

Qiang Zoua, Juyong Zhanga, Bailin Deng, and Jibin Zhao. 2014. Iso-level tool path planning for free-form surfaces. *Computer-Aided Design* 53 (2014), 217–225.

APPENDIX A: TOOL PATH OPTIMIZATION

Suppose that $\{\mathbf{x}_i\}_{i=1}^n$ be the point sequence to denote the tool path. We evolve the initial tool path by the following equation:

$$\frac{d\mathbf{x}_i}{dt} = \lambda_1 \times \mathbf{T}_{\text{Smooth}} + \lambda_2 \times \mathbf{T}_{\text{Attraction}} + \lambda_3 \times \mathbf{T}_{\text{Repulsion}},$$

where $\mathbf{T}_{\text{Smooth}}, \mathbf{T}_{\text{Attraction}}, \mathbf{T}_{\text{Repulsion}}$ respectively characterize the smoothness requirement, the attraction of $\mathbf{x}_i$ to the nearby anchor points (the centers of nearby large empty circles) and the repulsion between $\mathbf{x}_i$ and other path points close to $\mathbf{x}_i$. Note that the time t can be understood as the number of iterations.

First, it is natural to use the Laplacian smoothing technique to express the smoothness requirement, i.e.,

$$\mathbf{T}_{\text{Smooth}}\,|_{\mathbf{x}_i} = \frac{\mathbf{x}_{i-1} + \mathbf{x}_{i+1}}{2} - \mathbf{x}_i.$$

Second, to guarantee that there is no large space, we capture the anchor points $\{\mathbf{q}_j\}_{i=1}^{K_1}$ whose distance to the path is larger than $g/2$, and then attract the tool path where it is close to one of $\{\mathbf{q}_j\}_{i=1}^{K_1}$.
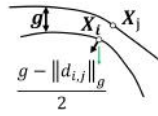
Let $r_j (> \frac{g}{2})$ be the distance between $\mathbf{q}_j$ and the tool path. The attraction term, for pulling $\mathbf{x}_i$ to the nearby anchor point $\mathbf{q}_j$ ($\|\mathbf{x}_i - \mathbf{q}_j\|_g \leq \frac{3g}{2}$) is defined as follows.
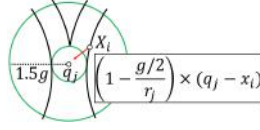
$$\mathbf{T}_{\text{Attraction}} \mid_{\mathbf{x}_i} = \frac{\sum_{\|\mathbf{x}_i - \mathbf{q}_j\|_g \leq \frac{3g}{2}} \frac{r_j - \frac{g}{2}}{\|\mathbf{x}_i - \mathbf{q}_j\|_g - r_j + \epsilon} \times (1 - \frac{g/2}{r_j}) \times (\mathbf{q}_j - \mathbf{x}_i)}{\sum_{\|\mathbf{x}_i - \mathbf{q}_j\|_g \leq \frac{3g}{2}} \frac{r_j - \frac{g}{2}}{\|\mathbf{x}_i - \mathbf{q}_j\|_g - r_j + \epsilon}},$$

where $(1 - \frac{g/2}{r_j}) \times (\mathbf{q}_j - \mathbf{x}_i)$ is able to shrink the distance between $\mathbf{x}_i$ and $\mathbf{q}_j$ to $\frac{g}{2}$ if $\mathbf{x}_i$ is exactly the point that gives the nearest distance r_j to $\mathbf{q}_j$. The weighting scheme $\frac{r_j - \frac{g}{2}}{\|\mathbf{x}_i - \mathbf{q}_j\|_g - r_j + \epsilon}$ is to emphasize the influence of $\mathbf{q}_j$ if $\mathbf{q}_j$ defines a large empty circle but deemphasize this term if $\mathbf{x}_i$ is not the point that gives the nearest distance r_j.

Finally, we introduce repulsion to $\mathbf{x}_i$ if there are other points $\{\mathbf{x}_j\}_{j=1}^{K_2}$ on the tool path meeting $\|d_{i,j}\|_g < g$, where $\|d_{i,j}\|_g$ is the geodesic between the two points. Considering that there may be multiple points that are close to $\mathbf{x}_i$, we compute the averaged repulsion as follows.

$$\mathbf{T}_{\text{Repulsion}} \mid_{\mathbf{x}_i} = \frac{\sum_{\|d_{i,j}\|_g < h} \frac{1}{\|d_{i,j}\|_g + \epsilon} \times \frac{g - \|d_{i,j}\|_g}{2} \times \frac{\mathbf{x}_i - \mathbf{x}_j}{\|d_{i,j}\|_g}}{\sum_{\|d_{i,j}\|_g < h} \frac{1}{\|d_{i,j}\|_g + \epsilon}}.$$

The rational behind

$$\frac{g - \|d_{i,j}\|_g}{2} \times \frac{\mathbf{x}_i - \mathbf{x}_j}{\|d_{i,j}\|_g}$$

is that if $\mathbf{x}_i$ and $\mathbf{x}_j$ move away from each other at the same time, the new distance between them would become exactly g, i.e, the diameter of the cutter head. The weighting scheme $\frac{1}{\|d_{i,j}\|_g + \epsilon}$ is to emphasize the influence between $\mathbf{x}_i$ and $\mathbf{x}_j$, if they are too close to each other.

As for the parameter configuration, we choose $\lambda_1 = 0.6, \lambda_2 = \lambda_3 = 0.2, \epsilon = 10^{-4}$ in our experiments. The termination condition is that the difference between the widest space and the narrowest is negligible, i.e., less than 5% of the parameter g.