Computer-Aided Design and Applications
© 2008 CAD Solutions, LLC
http://www.cadanda.com

Machining Triangular Mesh Surfaces via Mesh Offset Based Tool Paths

Daoshan OuYang[1] and Hsi-Yung Feng[2]

[1]Husky Injection Molding Systems Ltd., daoshan.ouyang@gmail.com
[2]The University of British Columbia, feng@mech.ubc.ca

ABSTRACT

This paper presents a new method to generate iso-planar numerical control (NC) tool paths for the finishing machining of triangular mesh surfaces. One main concern in generating the piecewise linear NC tool paths is to ensure that their resulting machining errors are within the specified tolerance. To this end, it is proposed that the cutter location (CL) tool paths be within the 3D tolerance zone defined by two offset surfaces of the triangular mesh surface. One offset surface corresponds to the under-cut limit and the other offset surface corresponds to the over-cut limit. Also, the scallop-height offset surface is used to facilitate the determination of side steps between adjacent iso-planar tool paths. The common self-intersections in the offset triangles are eliminated using the Z-map approach. The applicability and effectiveness of the presented method was validated through implementations on typical triangular mesh surfaces.

Keywords: NC machining, tool path generation, triangular mesh surface, mesh offset, Z-map.
DOI: 10.3722/cadaps.2008.254-265

1. INTRODUCTION

Freeform surfaces are frequently incorporated in the design of complex-shaped parts such as automobile body panels, toys, and medical replicas. The die/molds for these geometrically complex parts are manufactured by computer numerical control (CNC) machine tools in two stages: roughing and finishing. The majority of the workpiece material is removed in the roughing stage while a smaller amount material is removed in the finishing stage. In general, flat-end milling cutters are utilized for roughing and ball-end milling cutters are used for finishing. This common machining practice has driven extensive research studies in the area of NC tool path generation [5].

This paper focuses on NC tool path generation for the finishing machining of freeform surface geometry using ball-end milling. The most important requirement in finishing machining is that the quality of the machined surface has to meet the specified tolerance requirement. Hence, it is widely acknowledged that the cutter locations (CL) of the generated ball-end milling tool path ideally have to be on an offset surface of the design surface. This offset surface is at a surface normal offset distance of the cutter radius with respect to the design surface [2][13] or can be defined by the Mincowski sum of the design surface and the inverse cutter [3]. Due to the popular linearly interpolated NC tool motions, NC tool paths are often piecewise linear and consist of a series of straight line segments. In other words, an actual CL tool path is a set of polygonal segments that approximate the desired curved tool path on the CL offset surface. Because of the approximation, chordal deviations between the curved and piecewise linear CL tool path are employed to control the length of each linear CL tool path segment.

It is evident that chordal deviations of the CL tool path segments are not the same as the resulting machining errors due to the tool path approximation. In order to conform to the machined surface quality requirement, specified as the form deviation or profile tolerance, it is proposed in this work that the CL tool path be confined within the tolerance zone defined by two offset surfaces of the design surface. One offset surface corresponds to the under-cut limit and the other corresponds to the over-cut limit (Fig. 1). In CNC machining, executing the piecewise linear tool path requires frequent acceleration and deceleration motion cycles. Too many small tool path segments will evidently downgrade the machining operation due to the many acceleration and deceleration cycles. Hence in this work, each CL tool path segment is to be generated as long as possible between the under-cut and over-cut limit surface.

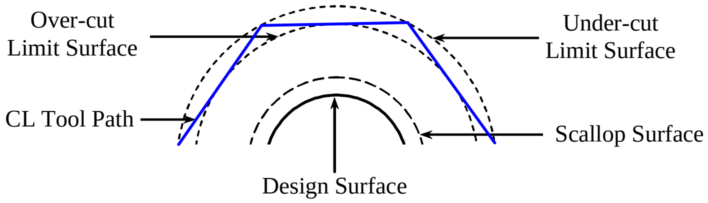


Fig. 1: CL tool path conforming to the specified tolerance limits.

The distance between two adjacent tool paths is commonly referred as side step or tool path interval. Generally the scallop-height requirement is employed to control the side steps of CNC tool paths. For example, in the iso-parametric machining method, the uniform tool path interval in the parametric space between adjacent tool paths is constrained by the scallop-height requirement [7],[17]. In the iso-planar machining method, the uniform tool path interval in the Euclidean space is also determined according to the scallop-height requirement [1],[4],[9],[10]. More recently, the tool path generation methods to achieve constant scallop height [8],[15],[16],[24-26], attempted to reduce tool path redundancy in the iso-parametric and iso-planar method.

It should be noted that constant scallop-height tool paths does not guarantee the best machining efficiency. This is because the total CL tool path length is not necessarily proportional to the total machining time. A CL tool path composed of small line segments is less efficient than a CL tool path with the same total length but composed of longer line segments. Kim and Choi [12] demonstrated that the zig-zag iso-planar tool path is the most efficient tool path in die/mold machining. In this research, iso-planar machining method was employed to generate tool paths from a triangular mesh, due to its robustness and simplicity. And the scallop surface [8], an offset surface to the design surface corresponding to the scallop height, was used to control the interval between adjacent cutting planes.

The scallop surface and the over-cut and under-cut limit surfaces, are all offset surfaces of the design surface (Fig. 1). Offsetting surfaces have been an active research topic for many years due to its importance in many fields [18],[23]. An offset surface can be defined as the collection of points that are at constant surface normal offset distance from the generator surface. In general, an offset surface is functionally more complex than its progenitor because of the square root involved in the expression of the surface normal vector. Thus a number of researchers have developed approximation algorithms for offsetting surfaces. In the context of 3-axis iso-planar tool path generation, Park [22] proposed a method to calculate the intersections between the cutting planes and the offset mesh surface from the design surface. In this research, the offset mesh surface was approximated by a Z-map model, which was obtained from the polygon rendering hardware due to its efficiency and simplicity [11].

Fig. 2 illustrates our proposed tool path generation procedure for the 3-axis CNC machining of a triangular mesh surface. At first, the offset surfaces for the under-cut, over-cut and scallop-height limit are approximated by Z-map models. An initial piecewise linear tool path is determined via the user specified cutting plane and the under-cut and over-cut limit Z-map models. The side step is then determined from the current tool path swept surface and the scallop-height limit Z-map model. In Section 2, triangular mesh surface offset and Z-map sampling details are described. Tool path forward step and side step determination methods based on the offset Z-map models are addressed in Sections 3 and 4, respectively. Implementation results and conclusions are given in Sections 5 and 6.

2. TRIANGULAR MESH SURFACE OFFSET

Surfaces of an object are most commonly represented by parametric surfaces in the Computer-Aided Design (CAD) environment. As stated previously, offsetting these surfaces in the parametric format is extremely difficult, if not impossible. To deal with this mathematical challenge, researchers have been defining these surfaces in different representation formats, such as point clouds or triangular meshes. In fact, the targeted surface in this work is not a parametric surface but a triangular mesh surface. This surface representation format is becoming popular due to recent advances in the 3D laser scanning technologies. The advantage is the ability to generate the machining process plan directly from the laser scanned point cloud data without the need for the demanding reverse engineering tasks (to reconstruct a smooth continuous surface from the scanned point cloud). This development is particularly important for developing the machining process plan for a physical object when its CAD model does not exist.

It is straightforward to construct a correct triangular mesh from a point cloud (a dense set of data points) if the underlying surface geometry is known. Much effort has been devoted to triangular mesh reconstruction from points when the underlying surface geometry is unknown [6],[19]. In this work, the method based on the clustering of Delaunay spheres is used to reconstruct the triangular mesh from points due to its simplicity and efficiency. This algorithm can reliably reconstruct a triangular mesh from randomly sampled points even in sharp edge/corner areas, without any prior knowledge and user-defined parameters.

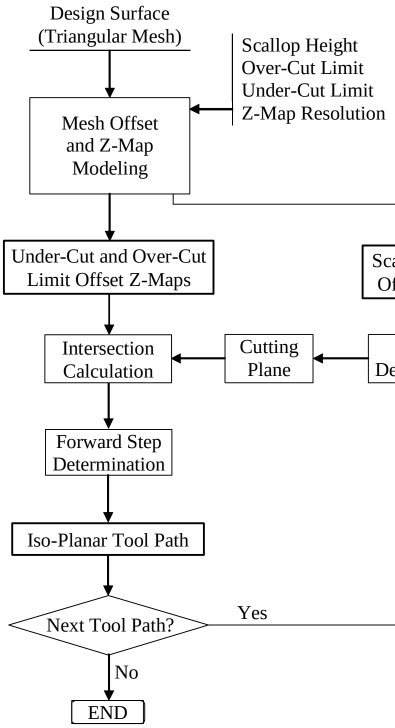


Fig. 2: The proposed tool path generation procedure.

With a triangular mesh surface, it is relatively straightforward to establish its offset by respectively offsetting its triangles, edges and vertices. It is evident that the offset of a triangle remains a triangle. Suppose one triangle's normal vector is $\mathbf{n}$ and its three vertices are P_i, $(i = 1,2,3)$. The normal vector of its offset will be $\mathbf{n}$ and the three vertices of its offset be $P_i* = P_i + R\mathbf{n}$, $(i = 1,2,3)$, where R is the offset distance. Fig. 3 shows the offsets of three triangles in a convex corner region. In this figure, triangle $P_1^*P_2^*P_3^*$ is the offset of triangle $P_1P_2P_3$.

Any edge, $\mathbf{e}$, in the triangular mesh of a 2-manifold surface model has at most two incident triangles. If $\mathbf{e}$ has two incident triangles, the offset corresponding to $\mathbf{e}$ will be a partial cylindrical surface with $\mathbf{e}$ as the axis and $\mathbf{e}$'s corresponding edges on the two offset incident triangles as the boundaries. For instance, in Fig. 3, the partial cylindrical surface $P_1^*P_2^*P_2'P_1'$ corresponds to the offset of edge P_1P_2. It is not a challenging task to tessellate such a partial cylindrical surface into a triangular mesh [11]. It should, however, be pointed out that in a concave corner region, the partial cylindrical surface does not exist.

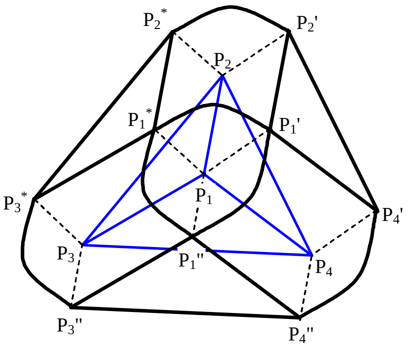


Fig. 3: Offset of P_1 and its incident edges and triangles.

The offset corresponding to a vertex $\mathbf{v}$ is a partial spherical surface with $\mathbf{v}$ as the center and the offset distance R as the radius. Most parts of the complete spherical surface can be excluded from the tessellation process since they are under the offset triangles and cylindrical surfaces of $\mathbf{v}$'s neighboring triangles and edges. This can be done by culling the spherical surface patches outside of any constrain plane pertaining to $\mathbf{v}$ (a constrain plane is defined as the plane passing through $\mathbf{v}$ and perpendicular to one of its mesh edges). In Fig. 3, the partial spherical surface $P_1^* P_1' P_1''$ corresponds to the offset of the vertex P_1. In this work, the spherical surface for each vertex is tessellated by the method of Hierarchical Triangular Mesh proposed by Kunszt et al. [14], due to its efficiency in indexing objects on the spherical surface. Each tessellated triangle outside of a constrain plane will be discarded.

It is difficult to detect self-intersections in the offset triangular mesh. For the purpose of detecting and eliminating self-intersections, the offset mesh is to be approximated by a Z-map model. In the Z-map representation, a surface is approximated by a set of points, which are the intersections between the surface and the vectors parallel to the Z-axis and passing through regular square grid points on the XY-plane. A Z-map model is in effect generated via orthographic projections and can be effectively used for surfaces that are always visible towards the negative Z direction. Thus, Z-map is quite suitable to eliminate self-intersections in the offset surface and to correctly represent the offset meshes, in particular for 3-axis CNC ball-end milling tool path generation.

The Z-map for the offset mesh surface can be efficiently obtained by employing polygon-rendering (or graphics) hardware. After the rendering process of the offset triangular mesh, the Z-value of the top point on the offset surface is automatically sampled at every pixel location. The Z-values of the Z-map model are recorded in the hardware depth buffer with 24 or 32 bit precision, which is sufficient for most 3-axis CNC machining operations. The X and Y values of the Z-map model, which represent the Z-map grid resolution, are limited only by the computer storage capacity, not limited by the hardware frame buffer size because a large surface can be divided into small regions to be rendered and recorded in the computer storage system. This ensures that the entire offset surface (for either the under-cut or over-cut limit) can be well represented by the Z-map model. Fig. 4 shows three typical triangular meshes (X × Y dimensions: 70 mm × 100 mm) examined in this work and their offset Z-maps (offset distance: 3 mm). Fig. 4(a) demonstrates that the (outward) offset loses some curved region on the original surface when the offset distance is relatively large. Fig. 4(b) shows that the (outward) offset may have no self-intersections if the original surface is relatively flat and the offset distance is small. Fig. 4(c) displays the self-intersections in an (inward) offset and they leave sharp corners on the offset surface.

It should be noted that the above proposed Z-map approximation method for establishing the triangular mesh surface offset can also be used as an indirect method to approximate the offset of any 2-manifold object surfaces. The object surface first needs to be approximated by a triangular mesh. The proposed Z-map method can then be used to establish the corresponding triangular mesh surface offset.

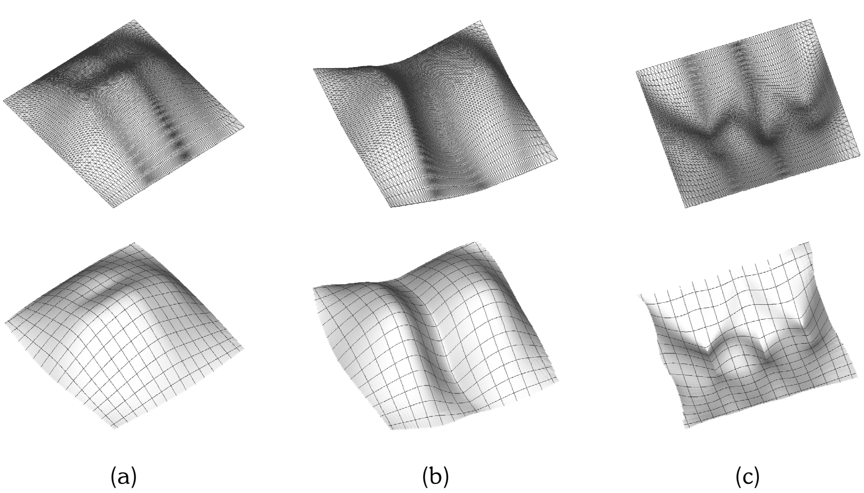


Fig. 4: Typical triangular meshes (upper) and their offsets (lower).

3. FORWARD STEP DETERMINATION

As stated in Section 1, three offset mesh surfaces (scallop, under-cut and over-cut limit surfaces) are established and represented with Z-map models in this work in order to generate applicable tool paths. For simplicity, iso-planar tool paths are to be generated. After determining the offset Z-map models, intersections between the initial cutting plane and the under-cut and over-cut limit offsets are calculated first. Suppose the initial cutting plane passes through the lower-left boundary point (p_x, p_y) in the XY-plane and the normal direction of the cutting plane is (n_x, n_y, n_z). It is evident that for iso-planar tool paths, $n_z = 0$ and the side step direction is (n_x, n_y). Let the projected tool path direction on the XY-plane be (d_x, d_y). As (d_x, d_y) is always perpendicular to (n_x, n_y), $(d_x, d_y) \cdot (n_x, n_y) = 0$. One such case is shown in Fig. 5. Determining the optimal direction of the projected iso-planar tool paths is by itself an active research subject. A valid method has been proposed for generating this projected direction from point cloud data [9] and is directly applicable to the present case of triangular mesh surfaces. Without loss of generality, it is assumed that $\left|d_x\right| \geq \left|d_y\right|$. It is then straightforward to calculate the intersection between the cutting plane and the Z-map model for every X grid line. Let one X grid line be $g_{x,i}$. The intersection point between this grid line and the cutting plane is then I$(g_{x,i}, g_y)$ on the XY-plane (Fig. 5), where

$$g_y = \left(g_{x,i} - p_x\right)\frac{d_y}{d_x} + p_y \tag{3.1}$$

It is straightforward to locate the two Y grid lines $g_{y,j}$ and $g_{y,j+1}$ enclosing g_y: $g_{y,j} \leq g_y \leq g_{y,j+1}$. Suppose the Z-values of the two grid points A$(g_{x,i}, g_{y,j+1})$ and B$(g_{x,i}, g_{y,j})$ are $Z_{i,j+1}$ and $Z_{i,j}$. The Z-value of the intersection point I can be found by linear interpolation as:

$$g_z = Z_{i,j} + \frac{(g_y - g_{y,j})(Z_{i,j+1} - Z_{i,j})}{g_{y,j+1} - g_{y,j}} \tag{3.2}$$

All the intersection points between the cutting plane and the over-cut and under-cut limit surface Z-maps along all the X grid lines can be determined by Eqn. (3.1) and (3.2). Let the intersection points on the over-cut limit surface Z-map be O_i and the intersection points on the under-cut limit surface Z-map be U_i. The applicable tool path on this cutting plane can be established by constructing consecutive longest line segments between the over-cut and under-cut piecewise linear curve connecting these intersection points.

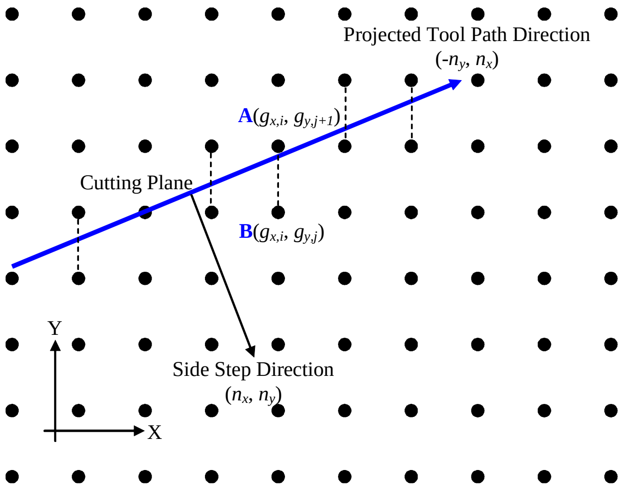


Fig. 5: Intersection between a cutting plane and a Z-map.

In Fig. 6, the starting point S of the first linear tool path segment is taken as the mid-point of the two intersection points on the over-cut and under-cut curve along the corresponding grid line $g_{x,s}$. From point S, the limit point U_m on the under-cut curve is defined as the point such that the line SU_m is below all the points on the under-cut curve and above all the points on the over-cut curve in-between grid lines $g_{x,s}$ and $g_{x,i}$. This condition, however, does not hold in-between grid lines $g_{x,s}$ and $g_{x,i+1}$. The limit point O_m on the over-cut curve is defined in a similar way in-between grid lines $g_{x,s}$ and $g_{x,i}$. The intersection point between line SU_m and the grid line $g_{x,i}$ is named as U, and the intersection point between line SO_m and the grid line $g_{x,i}$ is named as O. T is the mid-point of U and O. The line segment ST is then the first segment of the piecewise linear tool path on the current cutting plane. From point T, the next forward step (linear tool path segment) can be found in the same way.

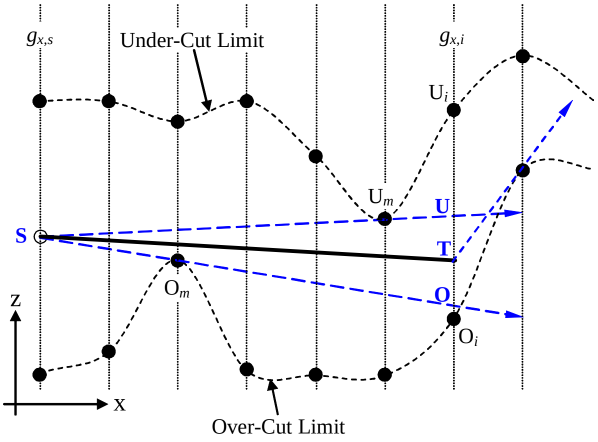


Fig. 6: Forward step determination.

The starting and ending points of a linear tool path segment are taken as the mid-points of the corresponding intersection points at a specific X grid line. This is because CNC machines cannot follow the commanded tool path trajectory exactly when two linear tool path segments are connected, especially in high-speed machining operations where high feed rate is employed. After all the consecutive linear tool path segments are determined on one cutting plane, the next cutting plane is established according to the side step. The side step is obtained through the current tool path swept surface and the scallop-height limit offset Z-map.

4. SIDE STEP DETERMINATION

Side step in the present work is determined via the following three steps:

Step 1. For each point P_i on the current tool path, its corresponding point S_i in the side step direction on the scallop surface Z-map is identified. To facilitate constant scallop height machining, these scallop points represent the intersection of tool swept surfaces of the current and the next tool path [8].

Step 2. For a scallop point S_i, its corresponding tool center point N_i on the next tool path is identified on the under-cut limit surface Z-map. The distance between N_i and the current cutting plane is denoted as d_i.

Step 3. The minimum of all the d_i's is then taken as the side step.

Step 3 in the above procedure is trivial. Steps 1 and 2 are to be described in more detail. As shown in Fig. 7, let point P_i is on the tool path segment T_iT_{i+1} (tool is moving from T_i to T_{i+1}). P_i's corresponding point S_i in the side step direction on the scallop surface is then on the normal plane of T_iT_{i+1} at P_i [8]. To determine S_i, the intersection curve between the ball-end mill's spherical surface centered at P_i and the scallop surface Z-map is established first. This intersection curve will intersect with the normal plane of T_iT_{i+1} at P_i, which gives the scallop point S_i. Note that the Z-map grid point closest to S_i will be taken as the solution. To reduce the computational load, only the Z-map grid points in area A (or B) need to be checked if the Z-value of T_{i+1} is less (or larger) than the Z-value of T_i.

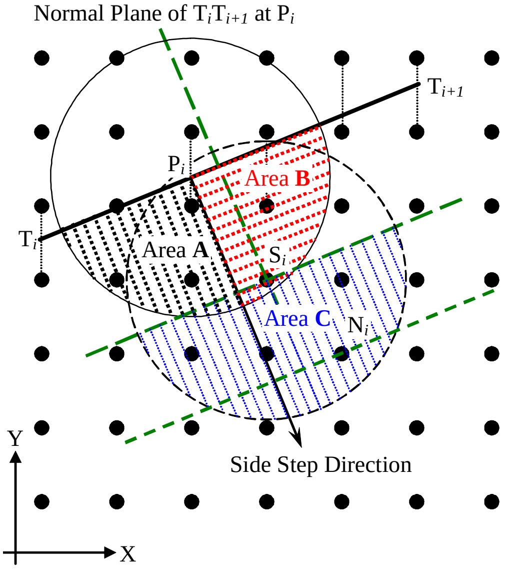


Fig. 7: Side step determination.

To determine the corresponding tool center point N_i on the next tool path for a scallop point S_i, the intersection curve between the ball-end mill's spherical surface centered at S_i and the under-cut limit surface Z-map is established first. The point N_i is then selected as the point on the intersection curve that is farthest from S_i in the side step direction. Again, to reduce the computational load, only the Z-map grid points in area C (Fig. 7) need to be checked.

In fact, it is not really necessary to establish the intersection curve from the Z-map grid points in area C. The point N_i can be selected as the point satisfying the following three conditions: (1) it is an under-cut limit Z-map grid point; (2) it is inside the ball-end mill's spherical surface centered at S_i; and (3) it is farthest to the current cutting plane. The above procedure is summarized as the pseudo-code below for improved clarification.

```
Find the Z-map grid point $\mathrm{S}_i$ on the XY-plane;
for(r=ToolRadius; r>0; r−=ZMapGridLength)
{
    for(d=–ToolRadius; d<=ToolRadius; d+=ZMapGridLength)
    {
        Find the Z-map grid point G: $\mathbf{G} = \mathbf{S}_i + r(n_x, n_y) + d(d_x, d_y)$;
        Find the corresponding under-cut limit Z-map point N for G;
        if(Distance between N and $\mathrm{S}_i$ <=ToolRadius)
        {
            Define N as $\mathrm{N}_i$;
            return;
        }
    }
}
```

After the side step interval ΔS is determined, the cutting plane for the next tool path is defined. The next cutting plane's normal direction is still (n_x, n_y) and will pass through the associated "lower-left boundary point" (p_x, p_y) with updated coordinates:

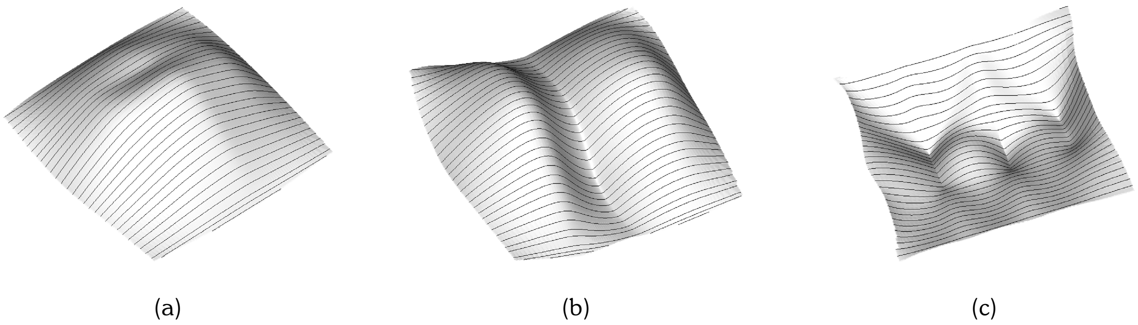
$$(p_x, p_y) = (p_x, p_y) + \Delta S(n_x, n_y) \tag{4.1}$$

The next tool path can then be established through the procedure presented in Section 3. This repetitive process continues until the cutting plane passes the entire region of the offset Z-maps.

5. IMPLEMENTATION RESULTS AND DISCUSSION

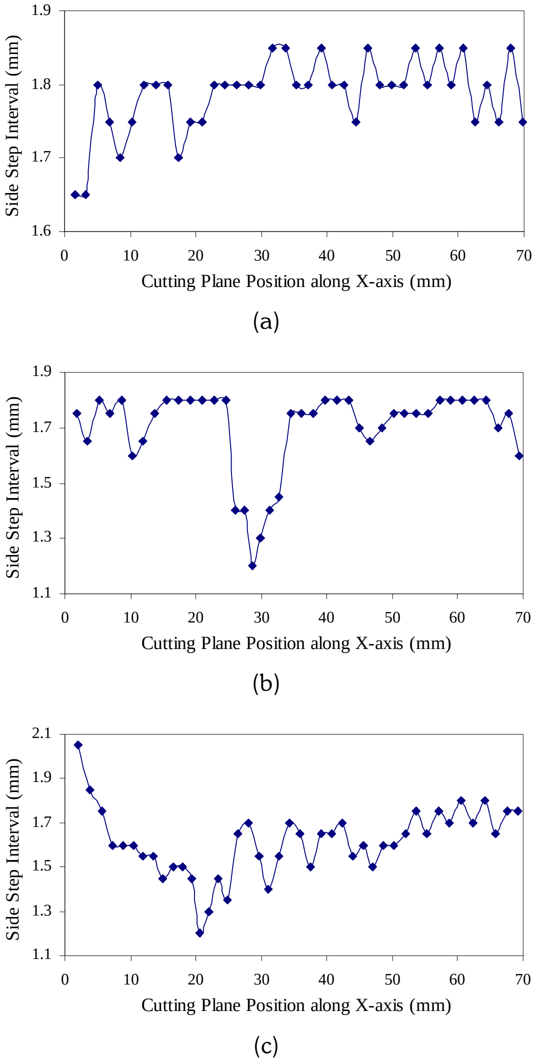
The presented tool path generation method has been implemented using Visual C++ .NET 2003 and OpenGL under Windows. To provide some guidelines regarding the computational efficiency of our method, the computational complexity of the algorithm for offsetting a triangular mesh in Section 2 is $\mathbf{O}(N_p + N_e + N_t)$, where N_p, N_e and N_t denote the number of points, edges and triangles in the mesh, respectively. The computational complexity for obtaining the Z-map of an offset triangular mesh in Section 2 is $\mathbf{O}(N_t^o)$, where N_t^o denotes the number of triangles in the offset mesh and is generally in the same order of magnitude of N_t. The computational complexity for forward step determination on one cutting plane in Section 3 is $\mathbf{O}(N_g)$, where N_g denotes the number of Z-map grid lines along the X-axis (the number of Z-map grid lines along the Y-axis is generally in the same order of magnitude of N_g). The computational complexity for side step determination (for one tool path) in Section 4 is $\mathbf{O}(kN_g)$, where k denotes the number of Z-map grid points under the ball-end mill's spherical surface.

Fig. 8 illustrates the generated tool paths (as solid lines on the shaded surface image) for the triangular mesh surfaces shown in Fig. 4. The relevant cutting conditions are: cutter diameter = 6 mm, Z-map grid length = 0.05 mm, under-cut limit surface offset = 3.1 mm, over-cut limit surface offset = 2.9 mm, scallop-height = 0.25 mm. In practice, the optimal selection of the cutter size is not a simple task. For machining triangular mesh surfaces using 3-axis ball-end milling, the maximum non-gouging cutter size can be determined by the minimum Delaunay sphere at the machining side of the surface [21]. The under-cut and over-cut limits directly affect the machined surface geometry. They should



(a) (b) (c)

Fig. 8: Generated tool paths for the triangular mesh surfaces shown in Fig. 4.



(a)

(b)

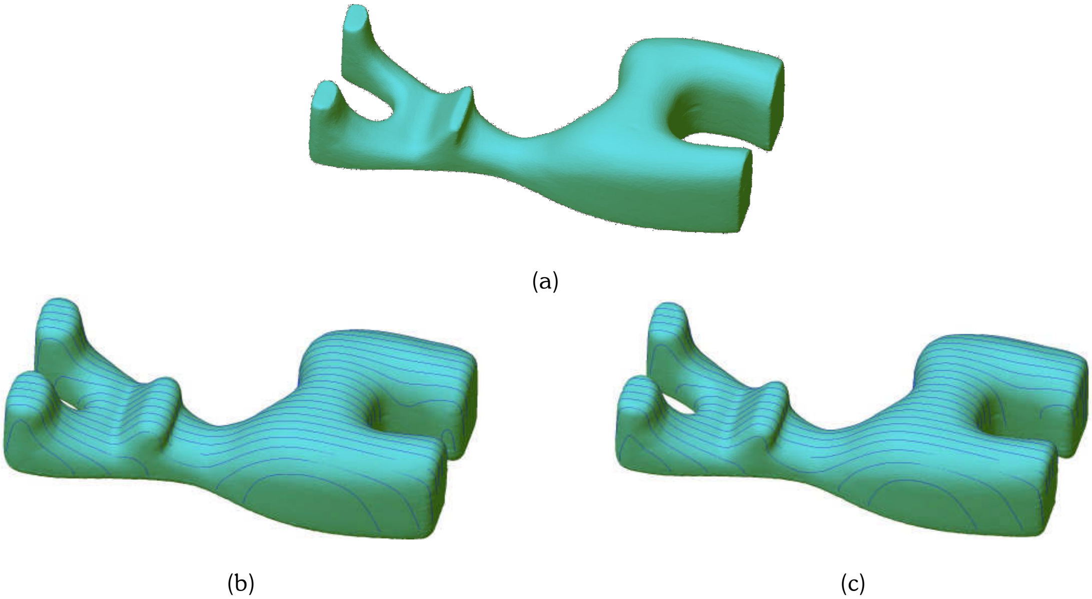
(c)

Fig. 9: Side step intervals for the examined triangular surface meshes.

be set to meet the given functional requirements of and/or to preserve specific geometric feature shapes on the machined surface.

The resulting side step intervals are shown in Fig. 9. It can be seen from Fig. 9 that the side step interval varies with the maximum surface slope in the side step direction along the current tool path. In Fig. 9(a), the side step interval is the smallest for the initial tool paths, where the surface of Fig. 4(a) has the largest slope in the side step direction. In Fig. 9(b), the side step interval is down dramatically around the center of the surface of Fig. 4(b), indicating smaller surface slopes around the top of the two ridges. In Fig. 9(c), the large side step interval for the initial tool paths clearly corresponds to the associated flatter surface geometry.

Fig. 10(a) shows the triangular mesh surface of a Reindeer model file downloaded from the Internet [27]. The approximate dimension of this model is: 136 mm (width) × 63 mm (depth) × 33 mm (height). Fig. 10(b) illustrates its under-cut limit offset surface for an offset value of 2.5 mm and the under-cut CL tool path limit curves produced by intersecting the offset surface with the cutting planes. Fig. 10(c) illustrates the over-cut limit offset surface for an offset value of 3.5 mm and the over-cut CL tool path limit curves. For a cutter diameter of 6 mm and the scallop height constraint of 0.5 mm, the generated tool paths for the Reindeer model are depicted in Fig. 11. The generated CL tool paths (in red) are always kept within the corresponding under-cut (in green) and over-cut (in blue) limit curves. To further highlight the main feature of this work, a typical CL tool path as seen in the cutting plane is shown in Fig. 12. It can be clearly observed in this figure that the generated piecewise linear CL tool path does not change direction until it comes in contact with the under-cut or over-cut limit curve. Note that the results are for many cases globally optimized with respect to the number of tool path segments. However, the globally optimal solution cannot be guaranteed as the starting point S of the first linear tool path segment is in fact a free parameter (which is taken in this work as the mid-point of the two corresponding points on the over-cut and under-cut curve). A straightforward but computationally expensive marching search could be applied to identify the optimal solution. Another possibility is that the global optimization of the CL tool path in-between the under-cut and over-cut limits appears to be quite similar to the Klee's Art Gallery problem [20]. The related optimization problem will be addressed in more detail and reported later.



(a)

(b)

(c)

Fig. 10: Reindeer model and its under-cut and over-cut limit offset surfaces.

6. CONCLUSIONS

A new method has been presented in this paper to generate applicable iso-planar piecewise linear NC tool paths for the finishing machining of triangular mesh surfaces with ball-end mills. The main feature of the presented method is that the machining errors due to the geometric deviations between the ideal and actual tool paths are explicitly considered and constrained with respect to the specified tolerance. This is achieved by generating the CL tool paths within the tolerance zone defined by two offset surfaces of the design surface (a triangular mesh surface). One offset surface corresponds to the under-cut limit and the other corresponds to the over-cut limit. The triangular mesh surface offset is done by respectively offsetting the triangles, edges and vertices of the mesh, and then approximated by a Z-map model to eliminate self-intersections among the offset triangles. As the hardware-based Z-map method samples the offset surface along the Z-axis direction, it is not able to handle surfaces that have more than one intersection with any vector parallel to the Z-axis (or simply surfaces that cannot be represented by a standard height map). Fortunately, a surface, which can be machined by 3-axis ball-end milling, can always be represented by a height map. In fact, this method can also be employed as an offset surface approximation method for general sculptured surfaces. The basic requirement would be that the surface to be offset can be sampled and correctly represented by a triangular mesh.

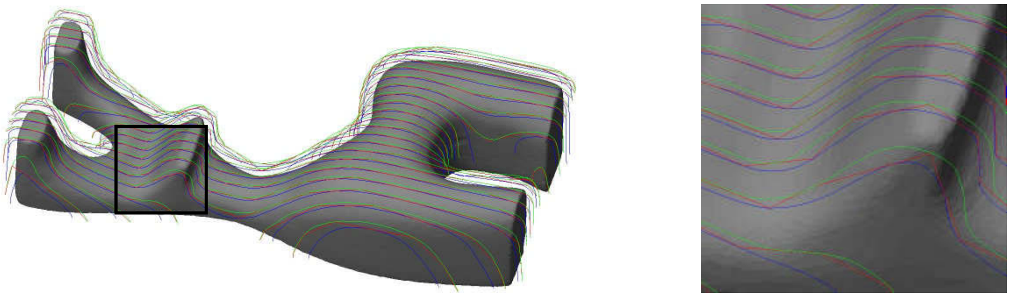


Fig. 11: Generated tool paths for the Reindeer model.

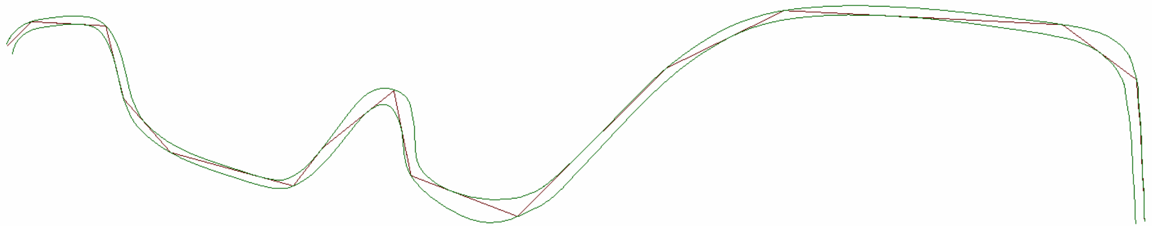


Fig. 12: A typical CL tool path in the cutting plane within the under-cut and over-cut limit curves.

7. ACKNOWLEDGEMENTS

The bulk of this work was completed when the authors were with the Department of Mechanical and Materials Engineering at the University of Western Ontario. The financial support was in part provided by Natural Sciences and Engineering Research Council of Canada (NSERC).

8. REFERENCES

[1] Bobrow, J. E.: NC machine tool path generation from CSG part representations, Computer-Aided Design, 17, 1985, 69-76.

[2] Chen, Y. J.; Ravani, B.: Offset surface generation and contouring in computer-aided design, ASME Journal of Mechanisms, Transmissions, and Automation in Design, 109, 1987, 133-142.

[3] Choi, B. K.; Jerard R. B.: Sculptured Surface Machining: Theory and Applications, Kluwer Academic, London, 1998.

[4] Ding, S.; Mannan, M. A.; Poo, A. N.; Yang, D. C. H.; Han Z.: Adaptive iso-planar tool path generation for machining of freeform surfaces, Computer-Aided Design, 35, 2003, 141-153.

[5] Dragomatz, D.; Mann, S.: A classified bibliography of literature on NC milling path generation, Computer-Aided Design, 29, 1997, 239-247.

[6] Edelsbrunner, H.: Shape reconstruction with the Delaunay complex, Proceedings of LATIN'98: Theoretical Informatics, 1998, 119-132.

[7] Elber, G.; Cohen, E.: Tool path generation for freeform surfaces, Computer-Aided Design, 26, 1994, 490-496.

[8] Feng, H. Y.; Li, H.: Constant scallop-height tool path generation for three-axis sculptured surface machining, Computer-Aided Design, 34, 2002, 647-654.

[9] Feng, H. Y.; Teng, Z.: Iso-planar piecewise linear NC tool path generation from discrete measured data points, Computer-Aided Design, 37, 2005, 55-64.

[10] Huang, Y.; Oliver, J. H.: Non-constant parameter NC tool path generation on sculptured surfaces, International Journal of Advanced Manufacturing Technology, 9, 1994, 281-290.

[11] Inui, M.: Fast inverse offset computation using polygon rendering hardware, Computer-Aided Design, 35, 2003, 191-201.

[12] Kim, B. H.; Choi, B. K.: Machining efficiency comparison direction-parallel tool path with contour-parallel tool path, Computer-Aided Design, 34, 2002, 89-95.

[13] Kim, K. I.; Kim, K.: A new machine strategy for sculptured surfaces using offset surface, International Journal of Production Research, 33, 1995, 1683-1697.

[14] Kunszt, P. Z.; Szalay, A. S.; Thakar, A. R.: The hierarchical triangular mesh, Mining the Sky: Proceedings of the MPA/ESO/MPE workshop, 2001, 631-637.

[15] Lee, E.: Contour offset approach to spiral tool path generation with constant scallop height, Computer-Aided Design, 35, 2003, 511-518.

[16] Lin, R. S.; Koren, Y.: Efficient tool-path planning for machining freeform surfaces, ASME Journal of Engineering for Industry, 118, 1996, 20-28.

[17] Loney, G. C.; Ozsoy, T. M.: NC machining of free form surfaces, Computer-Aided Design, 19, 1987, 85-90.

[18] Maekawa, T.: An overview of offset curves and surfaces, Computer-Aided Design, 31, 1999, 165-173.

[19] Mencl, R.; Müller, H.: Interpolation and approximation of surfaces from three-dimensional scattered data points, Proceedings of the Dagstuhl Conference on Scientific Visualization, 1997, 223-232.

[20] O'Rourke, J.: Computational Geometry in C, 2nd edition, Cambridge University Press, New York, NY, 1998.

[21] OuYang, D.; Van Nest, B. A.; Feng, H. Y.: Determining gouge-free ball-end mills for 3D surface machining from point cloud data, Robotics and Computer-Integrated Manufacturing, 21, 2005, 338-345.

[22] Park, S. C.: Sculptured surface machining using triangular mesh slicing, Computer-Aided Design, 36, 2004, 279-288.

[23] Pham, B.: Offset curves and surfaces: a brief survey, Computer-Aided Design, 24, 1992, 223-229.

[24] Sarma, R.; Dutta, D.: The geometry and generation of NC tool paths, ASME Journal of Mechanical Design, 119, 1997, 253-258.

[25] Suresh, K.; Yang, D. C. H.: Constant scallop-height machining of freeform surfaces, ASME Journal of Engineering for Industry, 116, 1994, 253-259.

[26] Tournier, C.; Duc, E.: A surface based approach for constant scallop-height tool-path generation, International Journal of Advanced Manufacturing Technology, 19, 2002, 318-324.

[27] http://www.b3dscann.com/samples.htm, Berding 3D Scanning.