About the geometry of milling paths

Márta Szilvási-Nagy[1][a]**, Szilvia Béla**[a]**, Gyula Mátyási**[b]

[a] Department of Geometry
Budapest University of Technology and Economics

[b] Department of Manufacturing Science and Technology
Budapest University of Technology and Economics

Abstract

In computer-aided manufacturing systems a number of methods have been published for milling path generation considering different geometric requirements and conditions determined by specific environments. In this paper we propose a method for the computation of the moving direction of the cutting tool in 3-axis milling considering the local features of the surface. Our method combines two geometric approaches. Computations are presented on analytical surfaces and on triangle meshes.

Keywords: 3-axis milling, tool path generation, isophotic lines, triangle mesh

MSC: 68U05, 68U07, 65D17, 65D18

1. Introduction

The most frequently used toolpath generation methods in CNC machining of free-form surfaces use planar curves, where the surface is intersected with parallel planes, and the intersection curves are taken as tool paths. The distance between two adjacent intersecting planes determines the distance between two tool paths (called tool path side step). Between the tool paths a scallop (rib) arises, the height of which measures the machining error (Figures 1, 2). This error depends on the shape of the cutting tool (flat end, ball end or other shapes) and also on the shape of the surface. Several computations have been published for optimizing the total length of the tool paths which is longer, if the number of cutting planes are larger, while the error is within a prescribed tolerance.
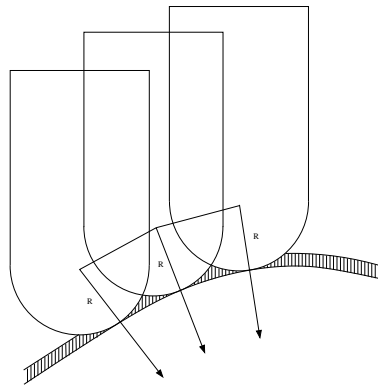
Figure 1: Machining error arising with ball end tool

The change of the angle between the tool axis and the surface normal leads to unevenly distributed side steps and fluctuating errors. A solution of this problem is the segmentation of the surface into regions, where this angle does not change too much. Such a segmentation with isophotic lines is proposed in [2]. Then the distances of the intersecting planes are computed in each region according to the prescribed tolerance (Figure 3). Within such a region the contact surface of the tool end with the material to be removed has a constant size. Consequently, the abrasion of the tool is even. An isophote interpolation method is proposed in [10], which can be used in isophote based tool path generations.

In an other strategy the tools are moving on the surface along isoparametric curves instead of plane sections [4]. Based on curvature values, independently from the parametrization of the surface, local and global millability with a given tool are investigated in [8] and [9]. This strategy takes into account that the width of the machined stripe depends on the curvature of the surface, and proposes a moving direction of the tool in which the stripe is the widest.

Fig 4 shows which part of the material is removed by the milling tool. The common part of the offset surface determined by the prescribed machining tolerance ε and the ball end of the tool is the so called contact surface. The projection of its boundary curve onto the surface determines the width s of the machining stripe, which is wider on flatter surfaces and narrower on more curved surfaces.

In this paper we propose a method for tool path generation considering the following requirements: (i) the change of the angle between the surface normal and the tool axis along a tool path is minimal, and (ii) the side step length is maximal, while the machining error is smaller than a given tolerance. Of course, these requirements cannot be fulfilled at the same time, we try to find a compromise.

Our investigations are restricted to 3-axis milling with ball end tools. We present our computations on analytical and on discrete surfaces.
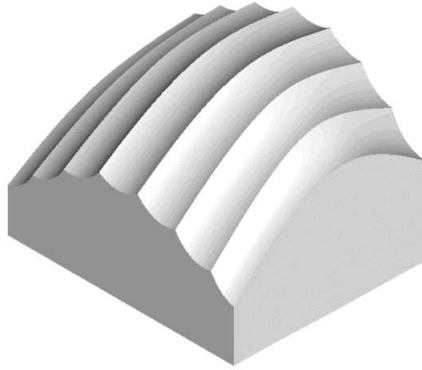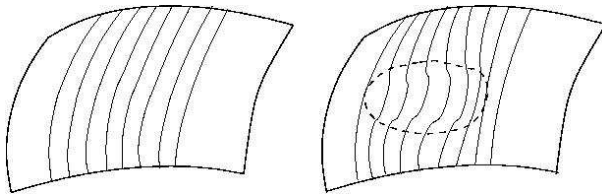
Figure 2: Processed surface with scallops

Figure 3: Segmentation of a surface with varying curvature

2. Analytic surfaces

We assume that the analytic surface is represented by a function $f(x, y)$ over a region in the xy plane, and the axis of the cutter is parallel to the z-axis. Our task is to determine the moving direction of the tool from every point of the surface considering the requirements of tool path generation and the geometrical features of the surface. In our investigations two geometric requirements will influence the shape of milling paths.

The first geometrical requirement is to ensure even abrasion of the tool end. This means that we want to keep the angle between the tool axis and surface normal constant during the cutting motion. A curve on the surface in the points of which the surface normal and a reference direction (here the tool axis) form a constant angle is called isophote or isophotic curve. On a smooth surface an isophotic curve assigned to an angle between zero and 90 degrees is a continuous curve, or a point. On the other hand, each point of the surface belongs to an isophotic curve, or the surface normal is parallel to the reference direction at this point. Properties and computation methods of isophota are described in [5]. In Fig 5 isophotic curves are shown on a quadratic surface by sequences of points which are computed on
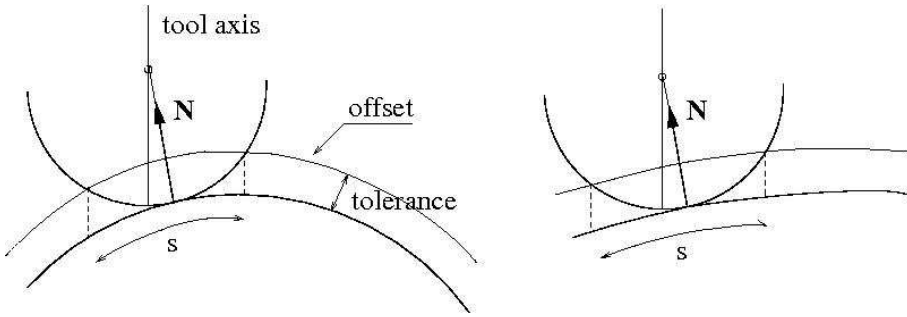
Figure 4: Simplified axial intersection of the milling tool and the
surface

the base of the definition with a given step size.

The distances between adjacent isophota on a generic surface are varying, there-
fore the fluctuation of the machining error will be out of control, when taking
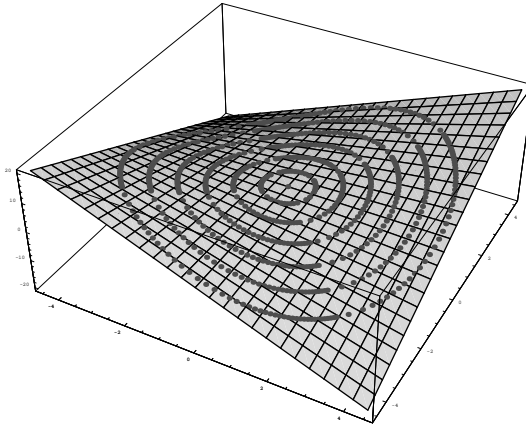isophota for tool paths.

Figure 5: Isophota on the saddle surface

We have to keep the machining error within a given tolerance. Therefore, we
first compute the processed part of the surface around a contact point of the ball
end and the surface. The error is less than a given tolerance ε, if the points of
the processed part are between the task surface $f(x, y)$ and its offset $Off(f, \varepsilon, x, y)$
consisting of the points which have the distance ε to the surface $f(x, y)$. If the
coordinates of the contact point are $(x_0, y_0, f(x_0, y_0))$, then we obtain the equation

$$|Off(f, \varepsilon, x, y) - Off(f, R, x_0, y_0)| = R \qquad (2.1)$$

for the boundary curve (called contact curve) of the processed part which is the curve of intersection of the ball end and the offset surface, where *Off* (f, ε, x, y) is the offset of the surface $f(x, y)$ with distance ε, and the second term gives the center of the ball end. The points of the contact curve are determined by the solutions (x, y) of this equation. (Our numerical method will result in a given number of points.) The normal projection of the contact curve on the surface determines the surface patch of our interest. This is the processed surface patch, and the optimal moving direction of the tool at the actual contact point is perpendicular to the largest diameter of this patch. In this way we get the widest machined stripe.
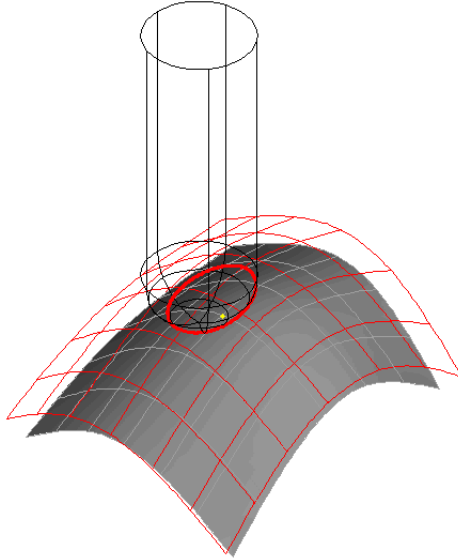

Figure 6: The boundary curve on the offset surface of the processed patch

We can calculate this direction in different ways. One method given in [9] calculates with the difference of the surface of the cutting tool and the task surface $f(x, y)$. This difference surface is approximated in second order, then the boundary curve of the processed patch, for the points of which the approximated difference is less than the tolerance, is projected onto the xy plane. The obtained curve is an ellipse, and its major axis determines the largest width of the machined stripe, while its minor axis determines the proposed moving direction.

In our geometric approach we approximate the moving direction from the equation (2.1). We estimate the diameters of the processed patch in n different directions (n is a given number in our algorithm). In this calculation we use in the above equation (2.1) the Taylor polynomial of degree 8 of the surface $f(x, y)$. First, we

set up n directions around the contact point and a vertical plane through each direction at the contact point. Then for each plane we solve the system of equations formed by (2.1) and the actual plane numerically. The solution gives the (x, y) coordinates of the end points of the patch diameter in this plane. Finally, we choose the direction of the largest diameter, and the proposed moving direction is perpendicular to it.

Now we want to consider the two requirements at the same time. That is, the tool should process a wide stripe, while the abrasion of the tool is even. Our compromise is the following. We modify the moving direction computed from the first requirement in the following way. We compute the isophote passing through the actual contact point. Then we move the tool end neither along this isophote, nor in the computed moving direction, but along a bisector direction of them. According to the two possible orientations of the isophote two bisectors exist. One is in "forward direction", the other one "backwards". The isophote passing through the actual contact point intersects the boundary curve of the processed surface patch in two points (Fig 7). The appropriate direction can be selected with the help of the two points of intersection and the moving direction computed in the former step. We have chosen the next contact point in this corrected moving direction by a specified constant step length.
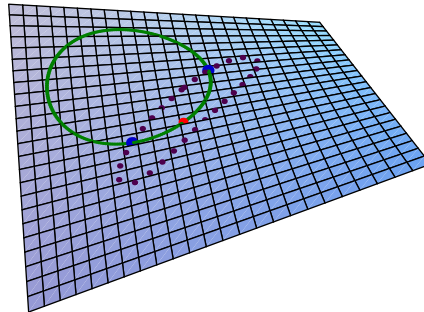

Figure 7: Isophote (the left side curve) and the boundary of the
processed patch at the contact point

In our further investigations we'll try to compute the correction of the moving direction by the isophotic curves and also the step distance considering the local shape of the surface. The overlapping of the processed patches along the adjacent tool paths require further investigations too.
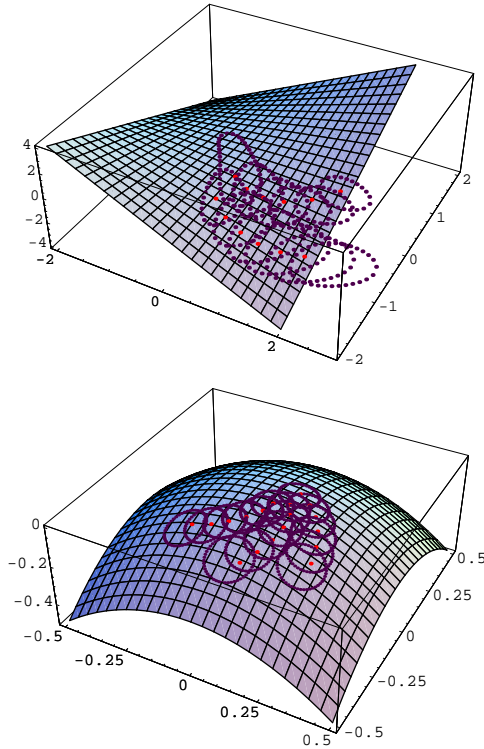
Figure 8: Processed patches along tool paths on the saddle surface
and on the sphere

3. Discrete surfaces

Computation of characteristic values of a free-form surface approximated by a triangular mesh requires quite a different technique from that in the analytic case. Namely, the surface data can be only estimated from the mesh data. Standard representations of triangle meshes are generated by the most CAD systems in STL (stereo lithography) format developed for rapid prototyping. Such an STL data structure contains the set of the mesh triangles, which complemented with adjacency informations becomes a polyhedral data structure. A polyhedral data structure makes possible to compute the line of intersection of the mesh with a plane which is the base of different algorithms, e.g. slicing in layered manufacturing [6] or tool path generation in milling. Though discrete counterparts of differential operators have been developed and published, there are no unique or best methods for estimating the surface normal or the curvature values at the points of the mesh.

For the characterization of the local shape of a surface presented by a triangle

mesh the estimation of principal directions is crucial. We apply in our computations the method of geodesic disk described in [7] (Fig 9). In this method the normal curvature values are estimated at the center point of a given triangle of the mesh in the following way. The mesh is intersected by a set of normal planes passing through the barycentric center of the triangle, and in each normal plane a fixed geodesic radius is measured along the polygonal line of intersection in both directions from the center point. The chord length of such a geodesic diameter characterizes the normal curvature in this intersecting plane. The normal curvature approximated from the geodesic radius r_g and the chord length d is (Fig 10)

$$\kappa_n \approx \frac{1}{r_g} \sqrt{6 \left(1 - \frac{d}{2r_g} \right)}. \tag{3.1}$$

Selecting the maximal normal curvature, i.e. the minimal chord length at the given point determines one of the two principal directions. In Fig 11 a geodesic circle is shown on the mesh of the duck. On the right hand side only feature and silhouette edges are drawn. The principal direction of the biggest normal curvature is indicated by a straight line segment.

We note that the method of the geodesic circle is suitable also for detecting planar and spherical regions on the mesh.
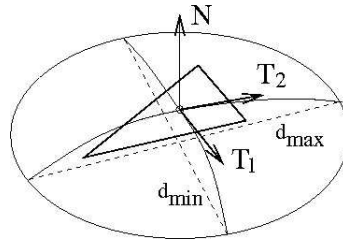

Figure 9: Geodesic disk for estimating normal curvatures and principal directions

In generating the offset of a triangle mesh several problems arise. After moving the facets in their normal directions by a given distance, gaps and overlappings occur in convex and concave regions, respectively. One method for avoiding gaps is offsetting also edges and vertices in averaged normal vector directions, then trimming the adjacent surfaces to each other [3]. Trimming and removing the overlapping portions are made in complicated processes. In an other approach, where tool paths are generated in parallel driving planes, filling of gaps is made in plane sections of the offsetted facets with the driving planes, then arc and line segments are used. The trimming problem is solved also in two dimensions in order to generate a smooth tool path in the actual plane [1].

In our method we solve the offsetting problem in normal sections. In order to determine the processed patch with a ball end on the mesh, the contact curve, i.e.
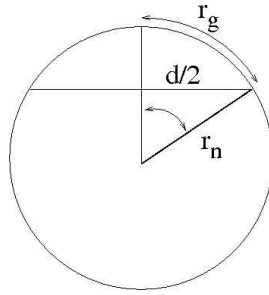
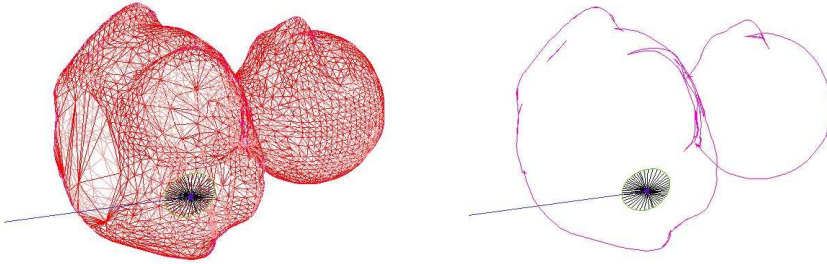Figure 10: Normal curvature estimated in a normal section

Figure 11: Triangle mesh of the duck and a principle direction

the curve of intersection of the surface of the ball end with the offset mesh has to be computed, then this curve has to be projected onto the mesh. We compute the points of the contact curve in a set of normal planes in the following way. 1. We set up n normal planes through the contact point. 2. We intersect the triangle face of the contact point and its two neighbours with the actual normal plane. 3. We move the obtained segments in the normal direction of the intersected triangles by the distance of the prescribed tolerance. 4. We fill the gap between the offset segments, or we remove the overlapping parts (Fig 12). 5. Along the polygonal line obtained in this way we measure the distance of the moving point from the ball end center. If this distance is equal to the ball end radius, then the point is on the contact curve. If all such distances are smaller than the radius, we intersect the neighbouring triangles with the normal plane in both directions, and we repeat the last three steps. 6. We project the two points of the contact curve computed in the actual normal plane onto the mesh. Finally, we get $2n$ boundary points of the processed patch around the contact point. We remark that our local offsetting method works with two dimensional algorithms.

The result of this computation is shown on a "real" triangle mesh of a sphere. The floating surface patch shown with 24 diameters is the part of the offset mesh

intersected with the ball end (Fig 13). Its projection on the mesh is the processed patch with the given tolerance. The perpendicular direction to the direction of the largest diameter of this patch gives a moving direction in which the widest machined stripe arises. This moving direction is to be corrected by minimizing the change of the surface normal direction within a prescribed angular neighborhood, if also the requirement of even abrasion of the tool is considered.
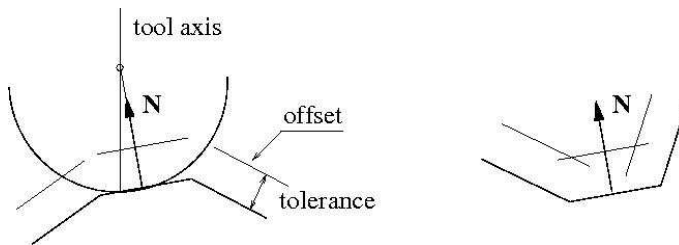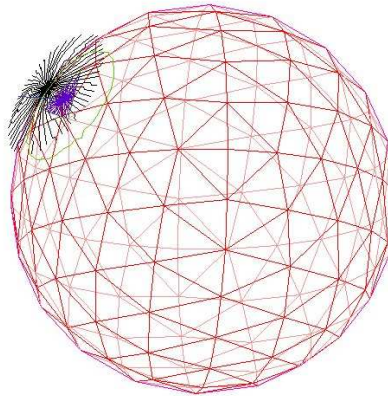

Figure 12: Offsetting in a normal plane

Figure 13: Intersection of a ball end with the offset of the sphere

4. Conclusions

In this paper a method is represented for the computation of the moving direction of a ball end tool in 3-axis milling. In this method two geometric requirements are considered at the same time, and a compromizing solution is proposed to meet both of them.

The computations and the figures are made with the algebraic symbolic program package Mathematica in the case of analytical description of the surface. The algorithms are implemented in the program language Java in the case of triangle meshes.

References

[1] Chuang, C.-M., Yau, H.-T., A new approach to z-level contour machining of triangulated surface models using fillet endmills, *Computer-Aided Design*, Vol. 37 (2005) 1039–1051.

[2] Ding, S., Mannan, M.A., Poo, A.N., Yang, D.C.H., Han, Z., Adaptive isoplanar tool path generation for machining of free-form surfaces, *Computer-Aided Design*, Vol. 35 (2003) 141–153.

[3] Jun, C.S., Kim, D.S., Park, S., A new curve-based approach to polyhedral machining, *Compuer-Aided Design*, Vol. 34 (2002) 379–389.

[4] Elber, G., Cohen, E., Tool path generation for free form surface models, *Compuer-Aided Design*, Vol. 26 (1994) 490–496.

[5] Lang, J., Zur Konstruktion von Isophoten im Computer Aided Design, *CAD-Computergraphik und Konstruktion*, Vol. 33., Wien 1984.

[6] Szilvasi-Nagy, M., Removing errors from triangle meshes by slicing, *Third Hungarian Conference on Computer Graphics and Geometry*, (17–18. Nov. 2005 Budapest, Hungary) 125–127.

[7] Szilvasi-Nagy, M., About curvatures on triangle meshes, *KoG*, 10 (2007) 13-18.

[8] Glaeser, G., Wallner, J., Pottmann, H., Collision-Free 3-Axis Milling and Selection of Cutting-Tools, *Computer-Aided Design*, Vol. 31 (1999) 225–232.

[9] Wallner, J., Glaeser, G., Pottmann, H., Geometric contributions to 3-axis milling of sculptured surfaces, In Machining Impossible Shapes (G. Olling, B. Choi and R. Jerard, eds.), pp. 33–41, Boston: Kluwer Academic Publ., (1999)

[10] Xu, H.Y., Tam, H.Y., Zang, J.J., Isophote interpolation, *Compuer-Aided Design*, Vol. 35 (2003) 1337–1344.

Márta Szilvási-Nagy
Szilvia Béla
Department of Geometry
Budapest University of Technology and Economics
H-1111 Budapest
Egry József u. 1. H. 22.

e-mail:
szilvasi@math.bme.hu
belus@math.bme.hu

Gyula Mátyási
Department of Manufacturing Engineering

Budapest University of Technology and Economics
H-1111 Budapest
Egry József u. 1. E. 2. 11.

e-mail:
`matyasi@manuf.bme.hu`