Accessibility in 5-Axis Milling Environment *

Gershon Elber †
Department of Computer Science
Technion, Israel Institute of Technology ‡
Haifa 32000, ISRAEL

February 28, 1997

Abstract

Using algorithms from computer graphics, namely hidden line and surface removal, techniques have been constructed to derive the 3-axis visible or millable set of a computer model from a given orientation or simulate and verify machining toolpaths. In this paper, an approach that reduces the accessibility problem of 5-axis milling using a flat end tool into a 3-axis accessibility problem is discussed. Using this approach, global detection of possible gouging into other parts, while a 5-axis flat end mill is in use, is made feasible and collisions can be avoided.

Key Words: NURBs, accessibility, 5-axis machining.

1 Introduction

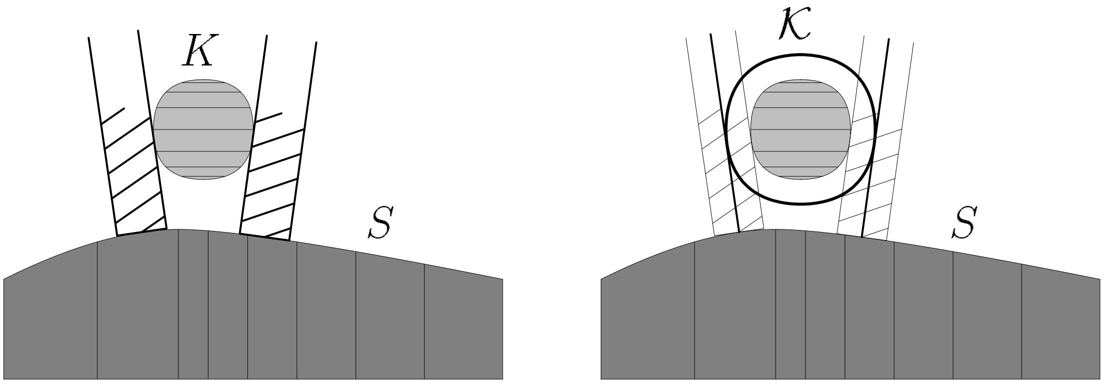
Several methods are in use, when freeform computer models are realized using milling machines. Possibly the most common approach utilizes a ball end tool with three degrees of freedom, namely translation in x, y, and z [1, 2, 4, 5, 6, 7]. The ball end center of the tool follows an offset [8] to the model by the tool radius while the ball end surface is tangent to the model. This method is known as 3-axis milling.

One way to generalized the use of 3-axis ball end milling is to add rotational degrees of freedom to the tool. The tool can now not only be positioned at a specific location, but can also be arbitrarily oriented. Two extra rotational degrees of freedom

*This work was supported in part by DARPA (N00014-91-J-4123). All opinions, findings, conclusions or recommendations expressed in this document are those of the authors and do not necessarily reflect the views of the sponsoring agencies.

†Appreciation is expressed to IBM for partial fellowship support of the author.

‡This work was done while at the Department of Computer Science, University of Utah, Salt Lake City, UT 84112 USA



(a) (b)

Figure 1: Using 5-axis milling, the flat end tool axis is aligned with the surface normal (a). The flat end tool will be tangent to the check surface, K, if and only if the tool central axis is tangent to the offset of K by the tool radius, $\mathcal{K}$ (b).

are usually provided, as in the spherical coordinate system, spanning any directional orientation on the unit sphere. This approach permits the use of a flat end mill, if it is positioned with its central axis aligned with the normal of the surface (see Figure 1(a)) because then the tool will not gouge into the surface, provided the surface is convex. Symbolic approach to isolate the convex regions of a surface was developed in [9]. This milling mode in known as 5-axis milling, for the five degrees of freedom required. Not only that offset approximation to surfaces are unnecessary for the generation of toolpaths, but the surface finish that result from the use of a flat end mill is superior to the surface finish by a ball end mill resulting in a smaller scallop height (compare Figures 5 and 6). Furthermore, a flat end tool can mill faster than the same radius ball end tool because of the vanishing cutting speed at the bottom of the ball end tool.

Other uses for 5-axis milling are with the side of the tool tangent to the surface, a milling operation known as side or peripheral milling. A different need for 5-axis milling rises when the accessibility using 3-axis approach is limited. Cavities, like the inside of the teapot in Figures 5 and 6, cannot be milled using 3-axis milling and a single fixture. Using 5-axis milling, a single fixture might be sufficient.

With all the advantages of adding rotational degrees of freedom, there is the question why is it that 5-axis milling so rarely used? The answer is not obvious and several factors have to be considered. Clearly 3-axis operations are easier to perceive than 5-axis. The complex intermixed motion of a 5-axis operations is difficult to comprehend, and therefore to compute, simulate and verify.

Several computer graphics algorithms where developed for removing hidden surfaces [10, 11]. It is clear that 3-axis milling tool cannot have access to the regions that are found hidden. The hidden surface removal algorithms can almost directly be employed in deciding what (visible) portion of the model can be milled using 3-axis and a given fixture or tool directional orientation. Unlike the light ray, the milling tool has a cylindrical shape of a finite radius. However, by offsetting the model by the tool radius, and applying the computer graphics hidden surface removal algorithms

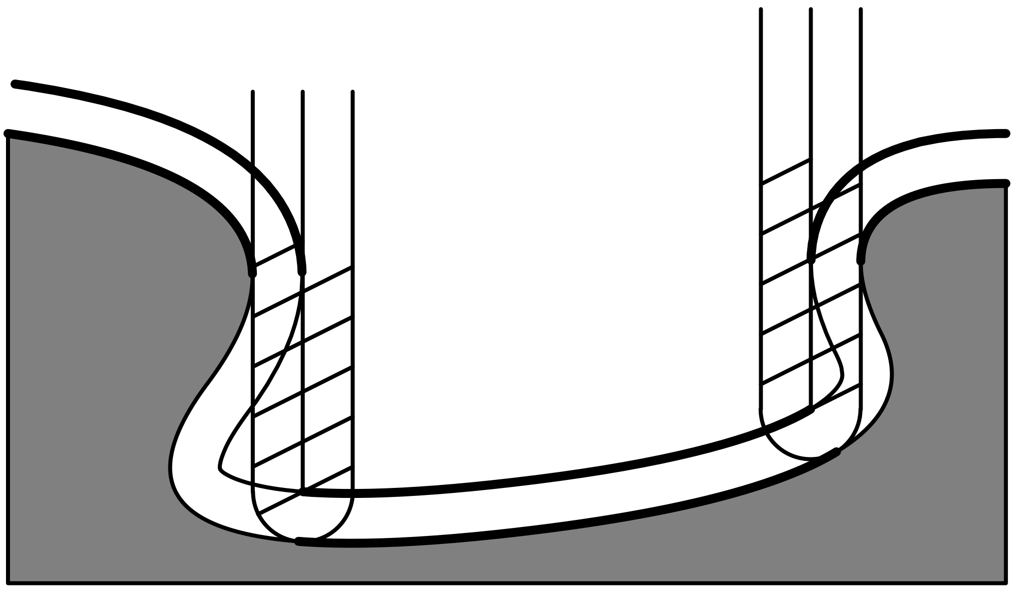


Figure 2: Using 3-axis milling, solving the problem of tool access to the original surface is equivalent to solving the computer graphics hidden line removal problem on the offset to the surface. Accessible regions are drawn using wide lines.

to the offset of the model, the accessible and therefore millable regions of the model immediately result (see Figure 2). A similar approach can be utilized for 5-axis flat end milling in which the check surface [12] is offset by the flat end tool radius as in Figure 1. Then again, the accessibility problem is transformed into a hidden surface removal problem. This time, however, the variable tool direction, following the surface normals, immensely increase the level of difficulty in solving this problem.

By considering a discrete approximation to the hidden surface removal, rendering techniques can be applied. An image is rendered viewed from the 3-axis tool direction, and which holds a "z-depth" image instead of shaded colors. The "z-depth" image is used for generating or verifying machining toolpaths in [6, 7], and for shadow generation in [13]. This technique is becoming popular in simulating and verifying 3-axis toolpaths.

Unfortunately, all these hidden surface removal methods are relevant for the use of 3-axis milling only. If the tool orientation is not fixed, as is the case for 5-axis milling, none of these methods can be effectively used. In this paper, we describe an approach that allows one to map the 5-axis accessibility problem into a 3-axis accessibility problem which can then be solved using hidden surface removal algorithms.

Section 2 provides the required background and discusses a possible theoretical approach. In section 3, an algorithm to compute the mapping that reduces the 5-axis accessibility problem into a 3-axis visibility or accessibility problem is described. Finally, in section 4 we discuss an example in which this approach was used. The left side of the body of the Utah teapot was milled using 5-axis flat end tool, while global gouging prevention was computed using the introduced algorithm.

2 Background

In the ensuing discussion we assume surfaces are regular and C^1 continuous. Let S be a convex surfaces. That is, for any point $p \in S$, the two principal curvatures [14] are positive. The sphere is an example of a convex object. In the ensuing discussion it is also assumed that the normals of a convex surface span an angular domain of less than 180 degrees, and that the surface is C^1 continuous. In other words, the Gauss map [14] of the surface can be fit into a open hemi-sphere. The Gauss map is a mapping from surface S to the unit sphere, S^2, and is injective or one-to-one if S is a completely convex region with the above angular bounds. That is, for each point $p \in S$, there exists only one and only one point on the Gauss map of S and visa versa. An arbitrary convex surface can always be subdivided until this condition hold.

Definition 1 *Let q be a point* above S. *That is, there exists a point $p \in S$ such that $q = p + \lambda_q n_p$, where n_p is the unit normal of surface S at p pointing outside, and $\lambda_q \geq 0$.*

Let $K(s,t)$ be a check surface [12] with some points above $S(u,v)$. Let $\mathcal{K}(s,t)$ be the offset of K by the flat end tool radius. One can derive the conditions in which the flat end tool is tangent to the check surface K while it is normal to S. Because this condition is equivalent to the condition that the tool central axis is tangent to $\mathcal{K}$ (Figure 1), one can derive (see Figure 3(a)) the following constraints. The orthogonality constraint of surface normals at the tangent point can be expressed as,

$$\left\langle n^{\mathcal{K}}(s,t), n^s(u,v) \right\rangle = 0, \tag{1}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product. In order to satisfy a positional constraint we can have,

$$\left\langle n^{\mathcal{K}}(s,t), (S(u,v) - \mathcal{K}(s,t)) \right\rangle = 0. \tag{2}$$

Both equation are necessary conditions that are satisfied when the tool is tangent to K and aligned along the normal of S. Because both equations need to satisfy a zero set condition, the surface normals in these equations need not necessarily be unit normals. This allow one to solve these equations and find the zero sets of four-variate Bézier or NURBs scalar fields, by symbolically representing the unnormalized normals as $\hat{n}^s = \frac{\partial S(u,v)}{\partial u} \times \frac{\partial S(u,v)}{\partial v}$ [15]. One can consider applying a numeric or a subdivision based methods to solve for the simultaneous zero set of these nonlinear equations, an approach that needs further investigation. However, equations 1 and 2 are not sufficient, since they do not coerce the vector $(S(u,v) - \mathcal{K}(s,t))$ to be collinear with n^s, nor make sure that $\lambda \geq 0$. Alternatively, one can try and map this problem into a simpler one with a known solution, an approach taken herein.

Lemma 1 *Let q be a point above convex surface S. There is one and only one point $p \in S$ such $q = p + \lambda_q n_p$, where n_p is the unit normal of surface S at p pointing out, and $\lambda_q \geq 0$.*

Proof: Assume there exists another point $r \in S$, such that $q = r + \delta_q n_r$, where n_r is the unit normal of surface S at r pointing outside, and $\delta_q \geq 0$. For a convex

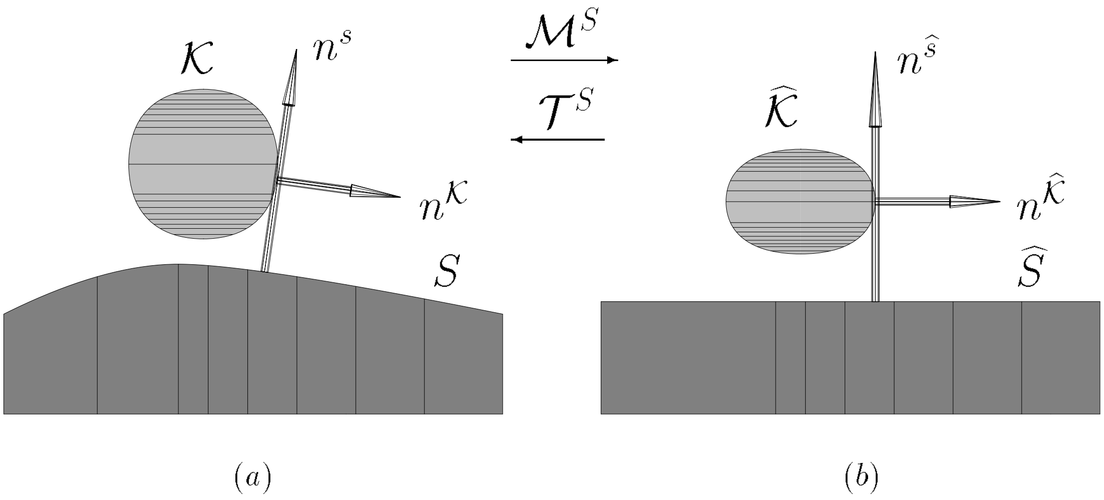


Figure 3: Using the $\mathcal{M}^S = \mathcal{T}^{S^{-1}}$ mapping, the 5-axis accessibility problem (a) can be reduced to a 3-axis visibility problem (b). See also Figure 1.

surface S, and for any tangent plane, T_p, to a point $p \in S$, S must lay on one side of T_p, while p being the only surface point on T_p. Without loss of generality, we assume $\|q - p\| \geq \|q - r\|$, where $\| \cdot \|$ denotes the L^2 norm. But then $r \in S$ is either on T_p (equality), or r and S are on the opposite sides of T_p (inequality), both contradicts the condition that S lies completely on one side of T_p. ■

Using lemma 1, one can construct the following injective mappings onto $\mathcal{W}$.

Definition 2 *Define the map $\mathcal{T}^S : \mathcal{D} \rightarrow \mathcal{W} \in \mathcal{R}^3$, $\mathcal{D} \in \mathcal{R}^3$ imposed by S, as $\mathcal{T}^S(u, v, \lambda) = S(u, v) + \lambda n^s(u, v)$, where $n^s(u, v)$ is the unit normal of S pointing outside. The map $\mathcal{M}^S : \mathcal{W} \rightarrow \mathcal{D}$ is define as the inverse map, $\mathcal{T}^{S^{-1}}$.*

$\mathcal{M}^S$ exists and has the same continuity as $\mathcal{T}^S$ by the inverse function theorem since by lemma 1 $\mathcal{T}^S$ is one-to-one and $\mathcal{T}^S$ is onto $\mathcal{W}$.

In other words, the maps prescribed by definition 2 uniquely identify any point q_i above S by a location (u_i, v_i) on S and elevation in the surface normal direction above this location, λ. $\mathcal{T}^S(u, v, \lambda)$ is a trivariate function that maps an orthogonal subspace of $\mathcal{R}^3$, $\mathcal{D}$, to a warped subspace of $\mathcal{R}^3$, $\mathcal{W}$. The trivariate representation was used in [16] to embed bivariate surfaces and deform them using the $\mathcal{T}^S$ mapping. Given surface $\widehat{\mathcal{K}}(s, t) = (u(s, t), v(s, t), \lambda(s, t))$ embedded in $\mathcal{T}^S$ domain, the image of the composition $\mathcal{K}(s, t) = \mathcal{T}^S \circ \widehat{\mathcal{K}}(s, t)$ is the deformed surface. This composition can be computed symbolically [15] or approximated as described in [16]. Figure 4 shows an example of this mapping.

If one can transform the offset of the check surface, $\mathcal{K}$, using the inverse mapping of $\mathcal{T}^S$, $\mathcal{M}^S$, imposed by S, then the set of (u, v) values on the surface S for which the flat end tool is tangent to K while being perpendicular to S can be resolved using hidden surface removal algorithms as in 3-axis milling accessibility testing. Unfortunately, the $\mathcal{M}^S$ mapping is nonlinear and has no closed form, in general. Although $\mathcal{T}^S$ can be computed symbolically as a composition operation, its inverse cannot. Aside from lines along the normals of S or the λ parameter, straight lines are not mapped by $\mathcal{M}^S$ to straight line, nor angles are preserved.

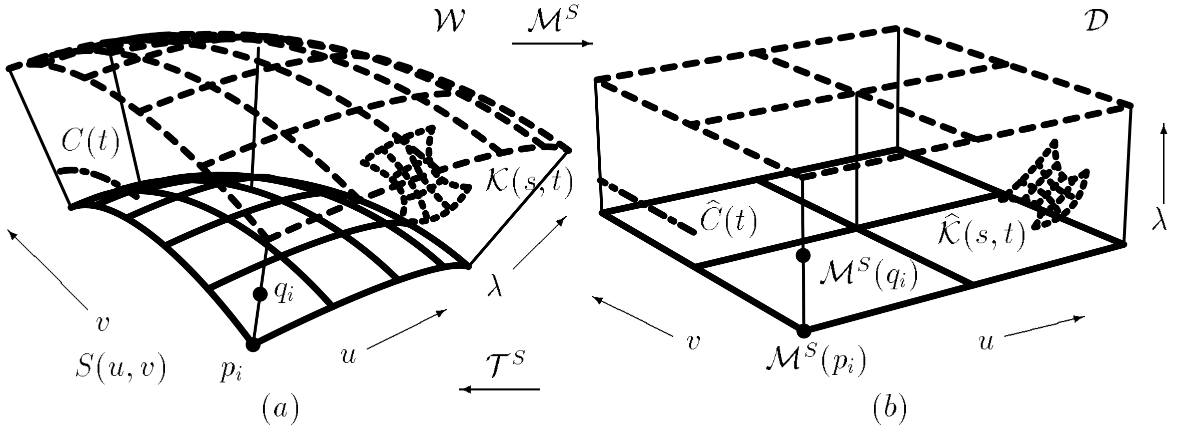


Figure 4: The $\mathcal{M}^S$ mapping. For each point q_i above surface S, there exists one and only one $p_i = S(u_i, v_i)$, such that $q_i = p_i + \lambda_i n_{p_i}$, where $\lambda_i \geq 0$ and n_{p_i} is the normal of surface S at p_i pointing outside. Then $\mathcal{M}^S(q_i) = (u_i, v_i, \lambda)$

Let $C(t)$ be a curve above S and let $\widehat{C}(t) = \mathcal{M}^S \circ C(t)$. The error between a C^2 continuous function and its Schoenberg variation diminishing spline approximation [17] over a knot vector $\{t_i\}$ is $O(|\{t_i\}|^2)$, where $|\{t_i\}| = max_i\{t_{i+1} - t_i\}$. By using a sequence of Schoenberg variation diminishing spline approximations to $C(t)$, each one based on a knot vector that is a refinement of the previous one, and a sequence, $\{C^i(t)\}$, of refined representations of $C(t)$, based on the same sequence of knot vectors, we form a convergent sequence of approximations to $C(t)$. Therefore, the sequence $\{\widehat{C}^i(t)\}$, where $\widehat{C}^i(t)$ is formed by transforming each of the control points of $C^i(t)$ using the $\mathcal{M}^S$ mapping, converges to the curve $\widehat{C}(t)$.

A tensor product surface is a convex blend of curves and therefore the above clearly holds for Bézier and NURBs tensor product surfaces as well. In other words, one can approximate $\widehat{\mathcal{K}}$, the $\mathcal{M}^S$ mapping of $\mathcal{K}$, by simply applying the mapping to all the control points of $\mathcal{K}$, and using refinement, converge to $\widehat{\mathcal{K}}$.

The $\mathcal{M}^S$ mapping of a single point above S, q, can be computed numerically. From lemma 1 it is clear that only one such solution exists. Because the surface S is convex, any point on S that is selected as a starting point to a numerical surface marching that minimizes the distance to q would converge to the required solution $p \in S$ such that $q = p + \lambda_q n_p$, resulting in a point-surface distance function with one minimum, the minimum of interest and which therefore has a very stable behavior. In practice, the node value associated with the control point of S that is closest to q was used as a starting point. Only two or three iterations were necessary, in general, to achieve a convergence of a six digits accuracy, using a steepest decent approach that minimized the distance to q.

Although refinement can improve this approximation, it is neither clear how good such an approximation is nor what locations are the optimal candidates for refinement. Because $\widehat{\mathcal{K}}$ is embedded in the domain of $\mathcal{T}^S$, it immediately follows that $\mathcal{K}$ is a result of the following composition,

$$\mathcal{K} = \mathcal{T}^S \circ \widehat{\mathcal{K}} \tag{3}$$

as done numerically in [16], or symbolically in [15].

Let $\widehat{\mathcal{K}}_1(s,t)$ be the first approximation to $\widehat{\mathcal{K}}(s,t)$ formed by numerically transforming all the control points of $\mathcal{K}(s,t)$ using the $\mathcal{M}^S$ mapping. Let $\overline{\mathcal{K}_1}(s,t) =$

$\mathcal{T}^S(u,v,\lambda) \circ \widehat{\mathcal{K}}_1(s,t)$ and let $\mathcal{E}(s,t) = \|\mathcal{K}(s,t) - \overline{\mathcal{K}_1}(s,t)\|^2$ be a scalar field measuring the square of the Euclidean distance between the corresponding parametric locations. If $\widehat{\mathcal{K}}_1(s,t)$ is exact, then $\overline{\mathcal{K}_1} = \mathcal{T}^S \circ \widehat{\mathcal{K}}_1 = \mathcal{T}^S \circ \widehat{\mathcal{K}} = \mathcal{K}$ or $\mathcal{E}(s,t) \equiv 0$. Otherwise, $\mathcal{E}(s,t)$ can directly be used not only as a measure to the error of the approximation, but also to specify where $\mathcal{K}$ should be further refined. By using $\mathcal{E}(s,t)$ to measure the approximation accuracy, a bound is established on the error of the approximation of $\widehat{\mathcal{K}}$ in the image space of $\mathcal{T}^S$, the space above $S(u,v)$. Said differently, since we cannot, in general, exactly compute $\mathcal{M}^S \circ \mathcal{K}$, it is approximated numerically, while the composition with $\mathcal{T}^S$, $\mathcal{T}^S \circ (\mathcal{M}^S \circ \mathcal{K})$, is computed symbolically. Then, the error in the numeric approximation of $\mathcal{M}^S \circ \mathcal{K}$ can be measured by inspecting the magnitude of the difference $\mathcal{T}^S \circ (\mathcal{M}^S \circ \mathcal{K}) - \mathcal{K}$.

Section 3 derives an iterative algorithm to compute an error bounded approximation to the transformation of a surface through the $\mathcal{M}^S$ mapping, and use it to map the 5-axis accessibility problem of a flat end tool into a 3-axis visibility problem that can be solved using hidden surface removal algorithms.

3 Algorithm

Using the approach described in section 2 we can now derive an iterative algorithm to compute an error bounded approximation to the transformation of an offset of a check surface, $\mathcal{K}$, by the $\mathcal{M}^S$ mapping:

Algorithm 1

Input:
- Convex surface, $S(u,v)$.
- Offset surface of a check surface, $\mathcal{K}(s,t)$.
- An error bound, τ.

Output:
- $\widehat{\mathcal{K}}$, $\mathcal{K}$ transformed by $\mathcal{M}^S$ with error bounded by τ.

Algorithm:

$i \Leftarrow 1$.
$\mathcal{K}_1 \Leftarrow \mathcal{K}$.
do
  $\widehat{\mathcal{K}_i} \Leftarrow \mathcal{M}^S \circ \mathcal{K}_i$, computed numerically over all control points of $\mathcal{K}_i$.
  $\overline{\mathcal{K}_i} \Leftarrow \mathcal{T}^S \circ \widehat{\mathcal{K}_i}$, computed symbolically.
  $\mathcal{E} \Leftarrow \|\mathcal{K} - \overline{\mathcal{K}_i}\|^2$.
  if (Maximum($|\mathcal{E}|$) $\geq \tau$)
  begin
    $\mathcal{K}_{i+1} \Leftarrow \mathcal{K}_i$ refined in regions where $|\mathcal{E}| \geq \tau$
    i $\Leftarrow$ i + 1.
  end
while (Maximum($|\mathcal{E}|$) $\geq \tau$).

Both the metric $\|\mathcal{K} - \overline{\mathcal{K}_i}\|^2$ and the composition $\mathcal{T}^S \circ \widehat{\mathcal{K}_i}$, in algorithm 1 can be computed symbolically for polynomial (Bézier) and piecewise polynomial and rational (NURBs) domains, resulting is a representation that is accurate to within the machine precision [15]. $\|\mathcal{K} - \overline{\mathcal{K}_i}\|$ is not representable in the Bézier and NURBs domains, in general, due to the computation of the square root that is necessary. Fortunately, computing the square of this L^2 norm is sufficient for our purposes. The Maximum($|\mathcal{E}|$) in algorithm 1 can be computed by simply inspecting the coefficients of the control mesh of $\mathcal{E}$ and using its maximum, which by the convex hull property of the Bézier and NURBs representation, cannot be less than the maximum of $|\mathcal{E}|$.

With the ability to approximate the transformation of a surface using $\mathcal{M}^S$ mapping and globally bound the error of the transformation, one can now reduce the accessibility problem of a 5-axis milling using flat end tool to a 3-axis motion which can be resolved using hidden surface removal algorithms.

Algorithm 2

Input:
- Convex surface, $S(u, v)$.
- Check surface offset, $\mathcal{K}(s, t)$.
- Flat end tool radius, r.
- An error bound, τ.

Output:
- $\mathcal{C}$, a set of curves describing the boundary of the accessible domain of S using 5-axis flat end mill.

Algorithm:
- $\mathcal{K} \Leftarrow$ offset approximation of K by distance r.
- $\widehat{\mathcal{K}} \Leftarrow \mathcal{M}^S \circ \mathcal{K}$ approximation within τ, using algorithm 1.
- $\mathcal{C} \Leftarrow$ boundaries of the visible domains of S with respect to K computed using hidden surface removal algorithm in $\mathcal{D}$ space, and viewing point $(0, 0, \infty)$.

4 Case study

The algorithm discussed in section 3 was used to help and prevent gouging into either the spout or the handle of the Utah teapot while milling the left side of the body of the Utah teapot, using 5-axis flat end mill. The toolpaths that were used to mill both parts in Figure 5 and Figure 6 were generated used the adaptive isocurve algorithm developed in [18]. Offsets for the different stages of this algorithm were computed to within a global tolerance using the algorithm discussed in [8].

The teapot in figure 5 is a result of a 3-axis ball end milling using two fixtures, one for the left and one for the right side of the teapot. The teapot in figure 6 is a result of the use of the 5-axis accessibility algorithm that is introduced here, for

Figure 5: The Utah teapot model, milled using a ball end 3-axis milling.

Figure 6: The Utah teapot model, partially milled using a flat end 5-axis milling.

the left side of the teapot body. Note that the inaccessible regions of the left side of the body of the teapot, due to the spout and the handle, are correctly detected and milled using traditional 3-axis milling approach. Figure 7 shows a close-up of these two regions for the 5-axis milled teapot in Figure 6

Figures 8, 9, and 10 demonstrates this process. In Figure 8, the left side of the body of the Utah teapot is used as the convex surface to be milled in 5-axis flat end mode. The left side of the spout and the left side of the handle are both the check surfaces and are therefore offset by the flat end tool radius. In Figure 9, the $\mathcal{M}^S$ mapping was applied to the left side of both the spout and the handle and the region of the body of the teapot that is hidden by them is computed (bounded by the dotted lines in Figure 9). Finally, the domain of the teapot body that can be milled using

(*a*) (*b*)

Figure 7: The inaccessible regions of the left side of the body using 5-axis flat end milling, near the spout (a) and near the handle (b) were machined with 3-axis ball end tool.

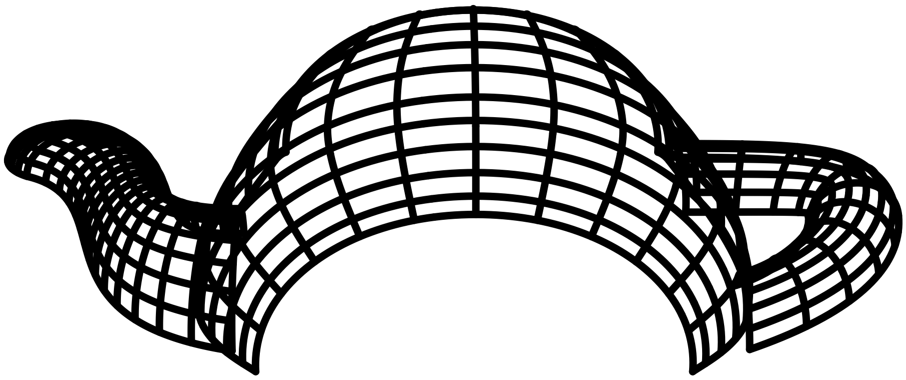


Figure 8: The left side of the body of the Utah teapot is to be milled using a 5-axis flat end mill. The two check surfaces, the spout and the handle, were offset by the flat end tool radius, before $\mathcal{M}^S$ can be applied to them.

5-axis flat end mill is calculated by trimming away the regions that are hidden by the spout and the handle, in Figure 10.

5 conclusion

An approach to map the 5-axis accessibility problem of flat end milling on a convex surface to 3-axis accessibility problem is described. This method can be easily extend to support trimmed surfaces. Because $\mathcal{M}^S$ was required to be injective, it can only be applied to regions that are completely convex. It is reasonable to assume that flat end milling will be used in convex regions only. However, side milling or ball end 5-axis milling can be used in saddle-like or concave regions as well. Removing the one-to-one constraint on $\mathcal{M}^S$ should be carefully considered, while such extension

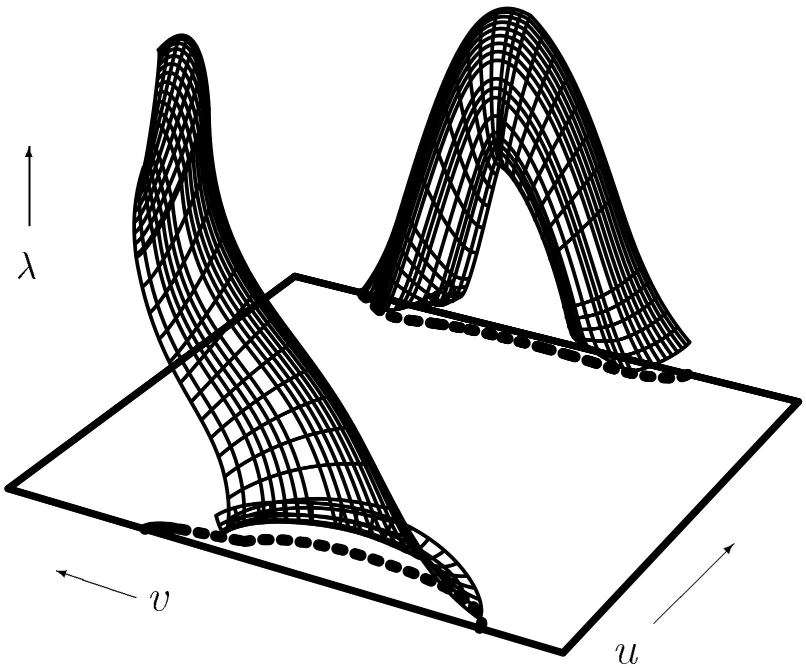


Figure 9: Using the $\mathcal{M}^S$ mapping imposed by the teapot body, the spout and the handle from Figure 8 are transformed to the orthogonal space $\mathcal{D}$ spanned by (u, v, λ). The hidden regions of the body, viewed from $(0, 0, \infty)$, are then computed (dotted lines).

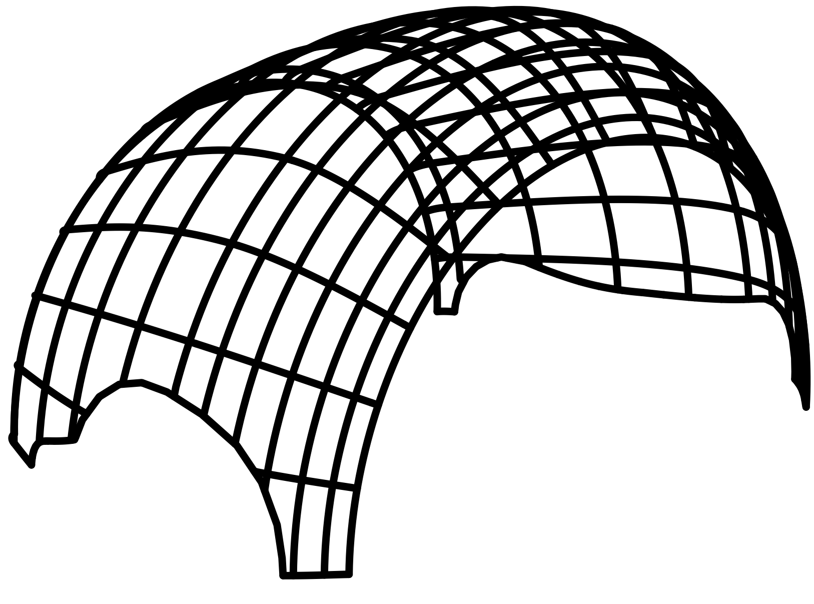


Figure 10: Using hidden surface removal algorithms, the hidden portion of the body of the teapot from Figure 9 is trimmed away. Left is the domain of the body that can be milled using 5-axis flat end mill with gouging to neither the spout nor the handle.

should be further investigated. Alternatively, robust and reasonably fast methods to solve the two simultaneous nonlinear equations 1 and 2, should be explored, yielding a possible solution for the accessibility problem for both the elliptic and the hyperbolic surface regions.

References

[1] **J. E. Bobrow.** NC machine tool path generation from CSG part representations. *Computer Aided Design,* Volume 17, number 2, pp 69-96, March 1985.

[2] **B. K. Choi and C. S. Jun.** Ball-end cutter interference avoidance in NC machining of sculptured surfaces. *Computer Aided Design*, Volume 21, number 6, pp 371-378, July/August 1989.

[3] **B. Cobb.** Design of Sculptured Surfaces Using The B-spline Representation. Ph.D. Thesis, University of Utah, Computer Science Department, June 1984.

[4] **R. B. Jerard, J. M. Angleton and R. L. Drysdale.** Sculptured Surface Tool path Generation with Global Interference Checking. Design Productivity Conference, Feb. 6-8, 1991, Honolulu, Hawaii.

[5] **G. C. Loney and T. M. Ozsoy.** NC machining of free form surfaces. *Computer Aided Design,* Vol. 19, No. 2, pp 85-90, March 1987.

[6] **S. W. Thomas.** Scanline Rendering for 3-Axis NC Toolpath Generation, Simulation, and Verification. Dept. of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109-2122, Technical Report CSE-TR-43-90, January 1990.

[7] **D. Zhang and A. Bowyer.** CSG Set-Theoretical Solid Modelling and NC Machining of Blend Surfaces. The Second Computation Geometry Conference, ACM 1986.

[8] **G. Elber and E. Cohen.** Error Bounded Variable Distance Offset Operator for Free Form Curves and Surfaces. *International Journal of Computational Geometry and Applications,* Vol. 1., No. 1, pp 67-78, March 1991.

[9] **G. Elber and E. Cohen.** Second Order Surface Analysis Using Hybrid Symbolic and Numeric Operators. Transaction on Graphics, Vol. 12, No. 2, pp 160-178, April 1993.

[10] **G. Elber and E. Cohen.** Hidden Curve Removal for Free Form Surfaces. *Computer Graphics*, Vol. 24, pp. 95-104, Siggraph 1990.

[11] **C. Hornung, W. Lellek, P. Pehwald, and W. Strasser.** An Area-Oriented Analytical Visiblity Method for Displaying Parametrically Defined Tensor-Product Surfaces. *Computer Aided Geometric Design,* 2, pp. 197-205, 1985.

[12] **I. D. Faux and M. J. Pratt.** *Computational Geometry for Design and Manufacturing.* John Wiley & Sons, 1979.

[13] **L. Williams.** Casting Curved Shadows on Curved Surfaces. *Computer Graphics*, Vol. 12, pp. 270-274, Siggraph 1978.

[14] **M. P. DoCarmo.** *Differential Geometry of Curves and Surfaces.* Prentice-Hall 1976.

[15] **G. Elber.** Free Form Surface Analysis using a Hybrid of Symbolic and Numeric Computation. Ph.D. thesis, University of Utah, Computer Science Department, 1992.

[16] **T. W. Sederberg and S. R. Barry.** Freeform Deformation of Solid Geometric Models. *Computer Graphics*, Vol. 20, pp. 151-160, Siggraph 1986.

[17] **M. Marsden and I. J. Schoenberg.** On Variation Diminishing Spline Approximation Methods. *Mathematica,* Vol. 8(31), No. 1, pp 61-82, 1966.

[18] **G. Elber and E. Cohen.** Tool Path Generation for Freeform Surface Models. The Second ACM/IEEE Symposium on Solid Modeling and Applications, Montreal Canada, May 1993.

[19] **B. K. Choi, J. W. Park, and C. S. Jun.** Cutter-location data optimization in 5-axis surface machining. *Computer Aided Design*, Volume 25, number 6, pp 377-386, June 1993.