Freeform Surface Region Optimization for Three- and Five-Axis Milling. *

Gershon Elber[†]
Department of Computer Science
Technion, Israel Institute of Technology
Haifa 32000, Israel

February 28, 1997

Abstract

A regional optimization of automatically generated machining toolpath for computer based freeform mechanical models is considered in the framework of multi-axis (three to five) machining operations. Consider a trichotomy of an arbitrary freeform shaped surface into convex, concave and saddle-like regions. Saddle-like and concave regions can be milled using a ball end tool, in a 3- or a 5-axis mode. A subset of the convex regions can be machined faster and with a smaller scallop, using a flat end tool that is aligned with the normal of the surface, in 5-axis mode. A method to robustly detect and eliminate local gouging of the milling tool into the object is developed and demonstrated on an actual part.

1 Introduction

The automatic generation of machining toolpath for computer based mechanical models is a challenging problem. The generation of an optimized machining toolpath is even more strenuous. Consider the evolution of the computer graphics field. Synthetically

*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.

[†]While visiting the Computer Science Department, University of Utah, Salt Lake City, Utah.

generated computer graphics images of computer models have achieved a remarkable level of accuracy creating strikingly realistic pictures. Unfortunately, the problem of automatic generation of machining toolpaths for computer models is more elusive.

Numerous methods were developed in the last few decades to solve the automatic toolpath generation problem. Different object types, such as polyhedra, and different machining modes, such as 3-axis ball end machining, were investigated, occasionally ignoring crucial ingredients such as tool accessibility.

Clearly, there is a great complexity in the need to bring a milling tool close to the model. Two types of tool accessibility conditions are usually considered, local and global. Locally, the tip of the tool should not gouge into the surface. Globally, none of the parts of the milling machine should interfere with the machined object, its fixture or any other part of the milling machine.

The need to prevent from both local and global gouging is only one aspect of the complexity. Testing and verification of the toolpath are two other aspects. Methods to test and verify machining toolpaths in 3-axis modes exist, with some [1, 2], based on a computer graphics Z-buffer discretization of the working space [3]. The lack of computation and verification models for 5-axis machining modes, makes the use of these modes sparse. Very few methods can be found that successfully generate and guarantee a gouge free 5-axis machining toolpath. In [4], attempt was made to classify visibility maps or the directional domains from which the model is visible and use them to derive the directions from which a model is *locally* accessible. In [5], this approach is extended to freeform surfaces. The global tool accessibility solution, the one that can guarantee a completely gouge free toolpath, not just at the tip of the tool, introduces more difficult problems. In [6], a restricted solution that provides a method to reduce the 5-axis global accessibility problem to 3-axis accessibility problem that can be solved using computer graphics' hidden surface removal technique is presented. This solution is suitable for convex regions, regions that can be machined using flat end tool that is aligned with the normal of the surface. Hereafter, we consider machining with a flat end tool only when the tool is aligned with the normal of the surface.

An important consideration, when the same type of object is to be machined thousands of times, is the optimality of the generated milling toolpath. In [7], a local tool orientation optimization method is described for 5-axis freeform surface machining. Constraints on the scallop (cusp) height and constraints preventing surface gouging are used to derive two angles that specify the orientation of the tool. In [8], a method to adaptively extract isoparametric curves from a parametric surface $S(u, v)$ and automatically bound the resulting scallop height is discussed. This method relieves the user from the need to interact with the internal parametric representation of the surface in order to achieve a small enough scallop height. Moreover, this method generates an almost optimal toolpath in its total length.

In this paper, we investigate a different optimality consideration. In general, flat end tools are superior to ball end tools because of an improved surface finish and a faster feed rate. The latter is due to the fact that the machining speed of a ball end tool vanishes at the bottom of the tip of the tool, a location that might be in contact with the machined surface. One would like to detect the regions of a freeform surface that can be machined using a flat end tool to optimize the manufacturing process in both time and quality.

This paper is organized as follows. In section 2, we develop the necessary background and investigate how one can compute the regions of a freeform surface that may be machined using a flat end tool. A method to extract the toolpath that is millable using a flat end tool that is normal to the surface, is presented in section 3. In section 4, we demonstrate the use of the developed technique on an actual part. Conclusions and possible future work are the topics of section 5.

The proposed approach was tested using the Alpha_1 solid modeling system, that is being developed at the University of Utah.

2 Background

In [9], a symbolic approach to robustly compute a trichotomy of an arbitrary freeform parametric surface, $S(u,v)$, into convex, concave, and saddle-like regions was presented (Figure 1). The convex and concave regions are the elliptic surface regions with a positive Gaussian curvature, $K > 0$ [10]. In a concave region, the surface bends toward the normal of the surface, that is assumed to be pointing outside. In a convex region, the surface bends away from the normal of the surface. The saddle-like regions are the hyperbolic regions with $K < 0$. Let $\mathcal{X}, \mathcal{Y}, \mathcal{Z} \subseteq S$ be the convex, concave, and saddle-like regions of S, respectively (See Figure 1). Furthermore, denote by $\mathcal{R} \subseteq S$ the parabolic

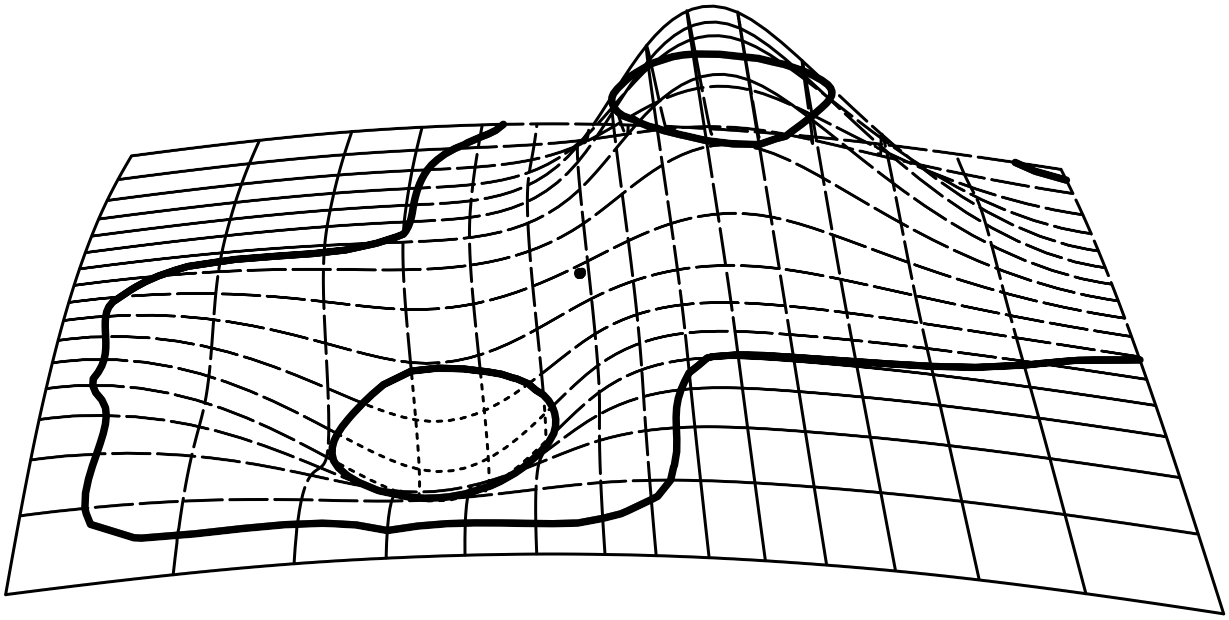

Figure 1: A trichotomy of a bicubic NURBs freeform surface into convex ($\mathcal{X}$, full lines), concave($\mathcal{Y}$, dotted lines) and saddle-like or hyperbolic ($\mathcal{Z}$, dashed lines). The parabolic set ($\mathcal{R}$) is also shown in thick lines.

regions of S, for which $K = 0$. $\mathcal{R}$ is usually unidimensional, that is $\mathcal{R}$ is combined of a set of curves, in general. If a two dimensional region of S is parabolic, that area is said to be developable [10]. A flat-end tool may have access to a developable region depending upon the sign of the second principal curvature. Developable regions can be isolated using techniques described in [11] and specifically handled herein as a simple special case.

Clearly, a flat end tool will gouge into the surface if positioned in a concave ($\mathcal{Y}$) or in a saddle-like ($\mathcal{Z}$) surface region. However, the regions that can be machined using a flat end tool are not the same as the convex ($\mathcal{X}$) regions of the surface as one might expect, but are only a subset of the convex regions. The $\mathcal{X}$ regions, along their boundary with the $\mathcal{Y}$ or $\mathcal{Z}$ regions, must also be machined using a ball end, or else the tool will gouge

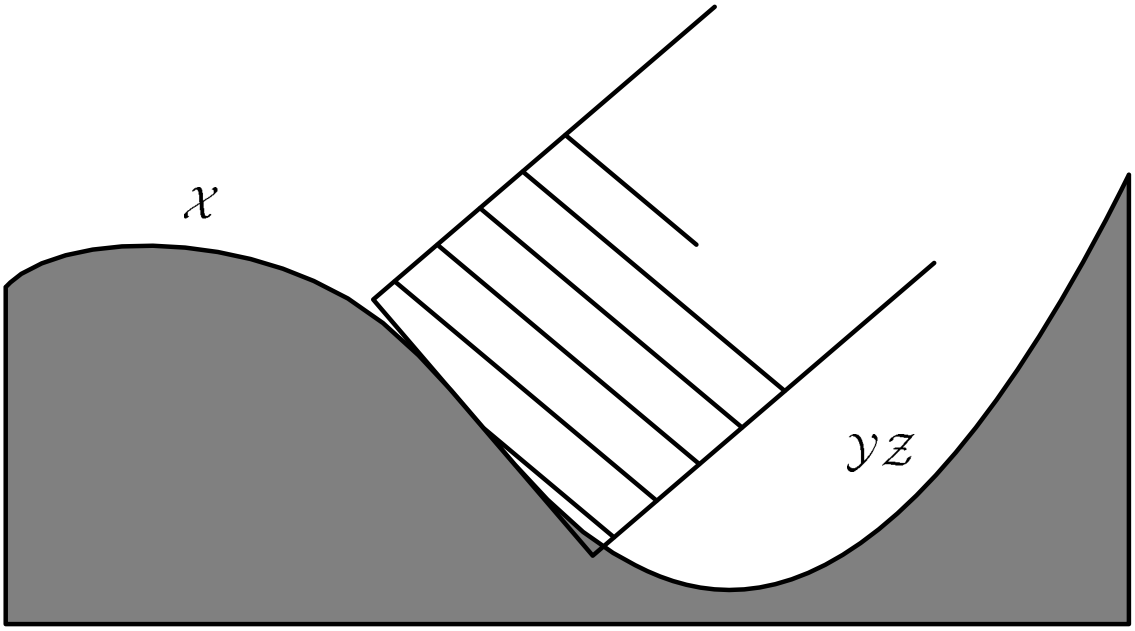

Figure 2: A flat end tool oriented along the normal of the surface will always gouge into a saddle-like or a concave region while moving along the parabolic boundary of a convex ($\mathcal{X}$) and a saddle-like/concave $\mathcal{YZ}$ region.

into the surface. A flat end tool moving in an $\mathcal{X}$ region close to the boundary with a $\mathcal{Y}$ or a $\mathcal{Z}$ region can gouge into the surface in the $\mathcal{Y}$ or the $\mathcal{Z}$ regions (See Figure 2). The boundary between an $\mathcal{X}$ and a $\mathcal{Y}$ or a $\mathcal{Z}$ region is parabolic ($\mathcal{R}$) (See Figure 1).

Let $\mathcal{X}_s \subseteq \mathcal{X}$ be the region in $\mathcal{X}$ that is gouge free millable with a flat end tool. That is, for each point $P \in \mathcal{X}_s$, a flat end tool positioned at P and aligned along the normal of S at P will *not* gouge into S. Let $\mathcal{P}$ be a toolpath that covers the entire surface S. Driving a tool along the curves of $\mathcal{P}$ will machine S with a prescribed precision and with a bounded scallop height. In [8], a valid toolpath coverage is derived using isoparametric curves that are extracted adaptively. This method of coverage generation is further optimized herein. Given $\mathcal{P}$, one would like to compute $\mathcal{F} = \mathcal{P} \cap \mathcal{X}_s$, the subset of $\mathcal{P}$ that can be machined using a flat end tool without gouging.

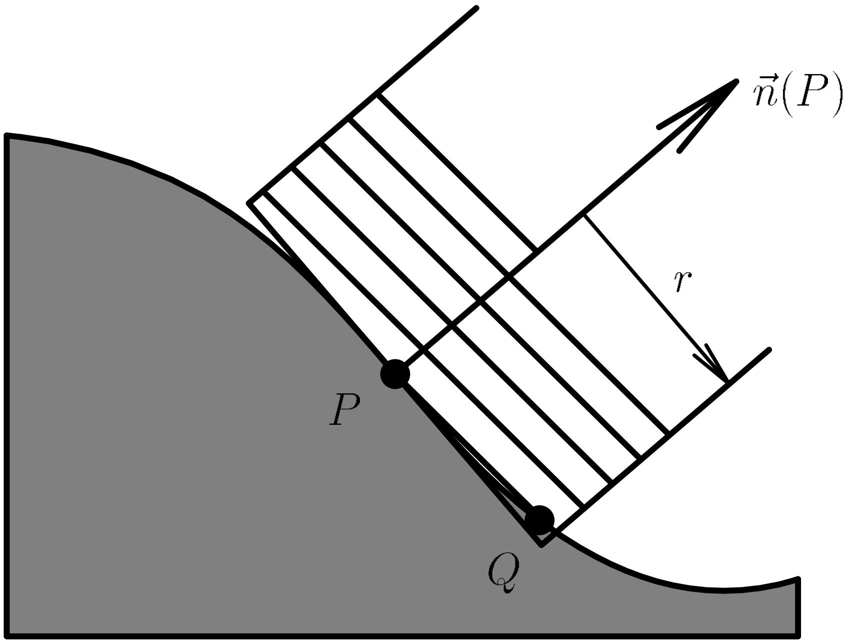

Figure 3: A flat end tool positioned at P will gouge into surface S if there exist point $Q \in S$ such that $\|P - Q\| < r$ and $\langle (P - Q), \vec{n}(P) \rangle < 0$.

The flat end machining tool, $\mathcal{T}$, can be represented as a half cylinder in space. Clearly, $\mathcal{T}$ will gouge into the surface S iff $\mathcal{T} \cap S \neq \emptyset$. This constraint can be reformulated as the condition of existence of a point $Q \in S$ that is closer than r to the center line of $\mathcal{T}$, where r is the radius of $\mathcal{T}$ (See Figure 3).

Denote by $\mathcal{L}[P](t)$ the center line of $\mathcal{T}$ at $P \in S(u, v)$. $\mathcal{L}[P](t)$ is aligned with the normal of the surface at $P = S(u_0, v_0)$, $\vec{n}(P)$:

$$\mathcal{L}[P](t) = P + t\vec{n}(P) = S(u_0, v_0) + t \frac{\partial S(u,v)}{\partial u} \times \left. \frac{\partial S(u,v)}{\partial v} \right|_{u=u_0, v=v_0}, \qquad 0 \leq t \leq t_0. \quad (1)$$

$\mathcal{T}$ positioned in $P \in \mathcal{X}$ will gouge into S at a point $Q \in \mathcal{YZ} = \mathcal{Y} \cup \mathcal{Z}$ (or possibly gouge into a different surface), iff there exist $0 \leq t \leq t_0$ such that,

$$\begin{aligned} \|\mathcal{L}[P](t) - Q\| &< r, \\ \langle (P - Q), \vec{n}(P) \rangle &< 0, \end{aligned} \qquad (2)$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product.

Unfortunately, not only the two non linear inequalities (2) are difficult to solve, but inequalities (2) need to be solved for all points $P \in \mathcal{P} \cap \mathcal{X}$. Hence, we limit the discussion for the possibility of *local gouging* of the tool, considering only the neighborhood of the tip of the tool. Setting $t = 0$ in inequalities (2) yields a new set of inequalities that need to be solved for all points $P \in \mathcal{P} \cap \mathcal{X}$,

$$\|\mathcal{L}[P](0) - Q\| = \|P - Q\| < r,$$
$$\langle (P - Q), \vec{n}(P) \rangle < 0. \tag{3}$$

The first inequality constraint in (3) is crucial for a locally gouge free flat end toolpath. $\mathcal{T}$, positioned at $P \in \mathcal{X}$, might locally gouge into S if there exist a point $Q \in \mathcal{YZ} \subseteq S$ such that $\|P - Q\| < r$.

The second inequality constraint in (3) guarantees that we consider only one side of the line $\mathcal{L}[P](t)$ with respect to the tangent plane of S at (u, v), for which t is positive. Relaxing the second inequality over-constrains the *local gouging* condition, possibly with false prevention of flat end milling. More importantly, the elimination of the second constraint in 3 will not allow the flat end tool to gouge into the model.

Considering only the first inequality constraint of (3), it is a nonlinear equation in several unknowns, the locations of both P and Q. In section 3, we show how this condition can be made into a feasible and robust method to subdivide a freeform surface into regions that can be machined using a flat end tool and regions that can be machined using a ball end tool.

3 Algorithm

The first inequality in (3) should be verified for every point on the *generated toolpath* for the flat end tool. This is a key observation that dramatically simplifies the computation. We need not verify (3) for every point $P \in S$ or even for every point $P \in \mathcal{X}$, but only for points on the generated toolpath in the convex regions of S, $P \in \mathcal{P} \cap \mathcal{X}$. Let $C(u) \subseteq \mathcal{P} \cap \mathcal{X}$, be an isoparametric toolpath curve in the *convex* region of S and consider point $Q \in \mathcal{YZ}$. Using equation (3) define,

$$d(u) = \|C(u) - Q\| - r, \tag{4}$$

to be the distance function (minus r) from $C(u)$ to point $Q \in \mathcal{YZ}$.

If $d(u_0) < 0$, a flat end tool positioned at $C(u_0)$ might (recall we ignored the second constraint of (3)) gouge into $\mathcal{YZ}$ region. Assume $d(u_0) < 0$ for Q. If $C(u)$ is long enough, their exist a location on $C(u)$ such that $d(u_1) = 0$, for Q. Because we consider only local gouging and because every $\mathcal{YZ}$ region is separated from an $\mathcal{X}$ region by a parabolic region $\mathcal{R}$, consider only points $Q \in \mathcal{R}$. Recall that the parabolic set of a freeform surface is univariate, in general. Let $R(w) \subseteq \mathcal{R}$ be a parabolic curve on S. Rewriting (4) as a zero set constraint,

$$\|C(u) - R(w)\| - r = 0. \tag{5}$$

Thus, the problem of finding the domain of $C(u)$ that can be machined using a flat end tool is reduced to finding the zero of a bivariate function, in u and w,

$$0 = \|C(u) - R(w)\|^2 - r^2 = \|C(u,w) - R(u,w)\|^2 - r^2 = \mathcal{Q}(u,w), \quad \forall R(w) \subseteq \mathcal{R}, \tag{6}$$

and eliminating the regions in $C(u)$ that are too close to any $R(w) \subseteq \mathcal{R}$. The solution of equation (6) can be manifested as a variation on a freeform curve-curve intersection problem [12, 13, 14]. Herein, the zero set of the bivariate function $\mathcal{Q}(u, w)$ is computed, and the u domain boundaries of the zero set are used to specify the locations that $C(u)$ must be subdivided at.

$\mathcal{R}$, the parabolic set of surface S is computed as the zero set of the determinant of the second fundamental form [10], a method that is discussed in [9] and used herein. $R(w) \subseteq \mathcal{R}$ are represented, in the approach taken in [9], as piecewise linear approximations of the real parabolic set of S.

Equation (6) is computed as the square of equation (5) because we cannot symbolically compute a square root term and represent it as a (piecewise) polynomial or rational, in general. However, $\mathcal{Q}(u, w)$ in equation (6) is computable and representable as a (piecewise) polynomial or rational surface [11], mapping the problem to a zero set finding on $\mathcal{Q}(u, w)$. Algorithm 1 summarizes this process.

4 Examples

The above algorithmic approach was applied and tested on a freeform model of a finger (Figure 4). The finger is a single bicubic NURBs surface composed of 48 patches (patches are counted as the bivariate polynomials or rationals resulting from subdividing the NURBs surface at all the interior knots). A toolpath consisting of isoparametric curves extracted adaptively using the method described in [8] is derived (Figure 5). The surface of the finger is subdivided into convex and saddle-like regions (Figure 6), uncovering no

Algorithm 1

```
Input:
    S(u,v), surface to dichotomize into flat end 5-axis millable
                    regions and ball end 3-/5-axis millable regions.
    r, radius of tool.
Output:
    F, toolpath for regions of S that can be 5-axis machined
                                        using a flat end mill.
    B, toolpath for regions of S that can be 3-/5-axis machined
                                        using a ball end mill.
Algorithm:
    OptimalRegionSurfaceMill( S, r )
    begin
        X <= convex (elliptic) regions of S;
        Y <= concave (elliptic) regions of S;
        Z <= saddle-like (hyperbolic) regions of S;
        YZ <= Y ∪ Z;
        R <= parabolic regions of S;
        P <= toolpath that covers S;
        F <= P ∩ X;
        B <= P ∩ YZ;
        foreach curve R(w) ⊆ R do
            foreach curve C(u) ⊆ F do
                Q(u,w) <= ||C(u) - R(w)||^2 - r^2;
                if ( zero set of Q(u,w) is not empty ) then
                    F <= F - {C(u)};
                    C_i(u) <= C(u) Split at the u zero
                                            set boundary of Q;
                    foreach curve C_i(u) do
                        if ( ||C_i(u) - R(w)||^2 <= r^2 ) then
                            B <= B ∪ {C_i(u)}
                        else
                            F <= F ∪ {C_i(u)}
                        endif
                    endfor
                endif
            endfor
        endfor
    end
```

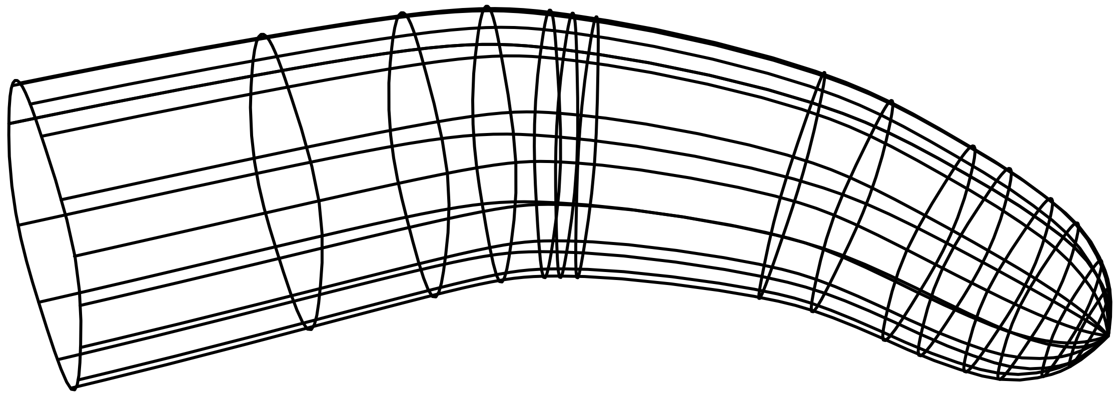

Figure 4: A model of a finger was used to test the proposed algorithm.

cavities. The toolpath is split according to the different regions that result, assigning a ball end tool to the toolpath in the saddle-like region. Then, the toolpath of the convex region is further trimmed according to the parabolic set, $\mathcal{R}$, of the surface to be at least radius of the tool, r, away from $\mathcal{R}$. The last stage divides the toolpath in the convex region into the toolpath that can and cannot be machined using a flat end tool. Finally, and since a toolpath for a ball end mill should specify the center of the ball end motion, the ball end toolpath is offset by r. The final toolpath is shown in Figure 7, and a machined part, from aluminum, is shown in Figure 8.

5 Conclusions

We presented an algorithm to optimize the toolpath to machine a freeform surface using a combination of a flat end and a ball end tool. The flat end tool can machine faster and can generate smaller scallop, making it a superior selection. However, a flat end tool must be used in 5-axis machining mode and is usually aligned with the normal to the surface. The combination presented exploits flat end 5-axis machining on regions

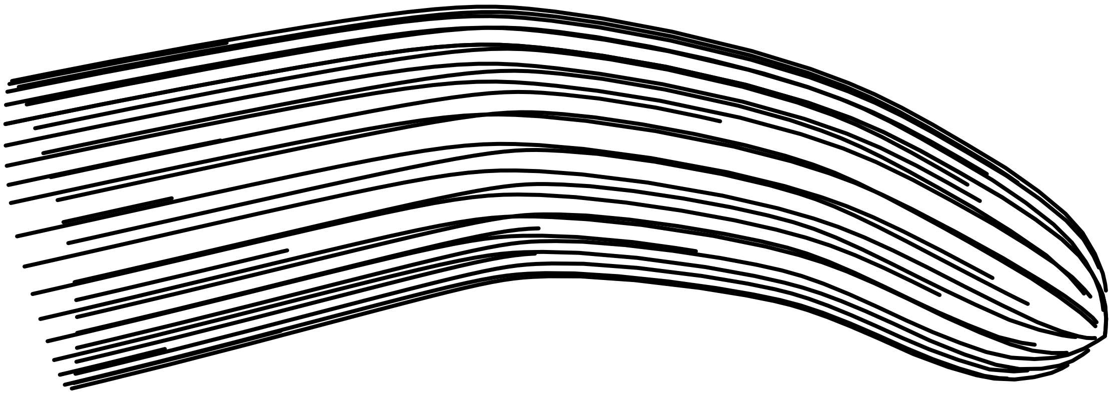

Figure 5: Toolpath derived for the finger model in Figure 4 using adaptive isoparametric curves.

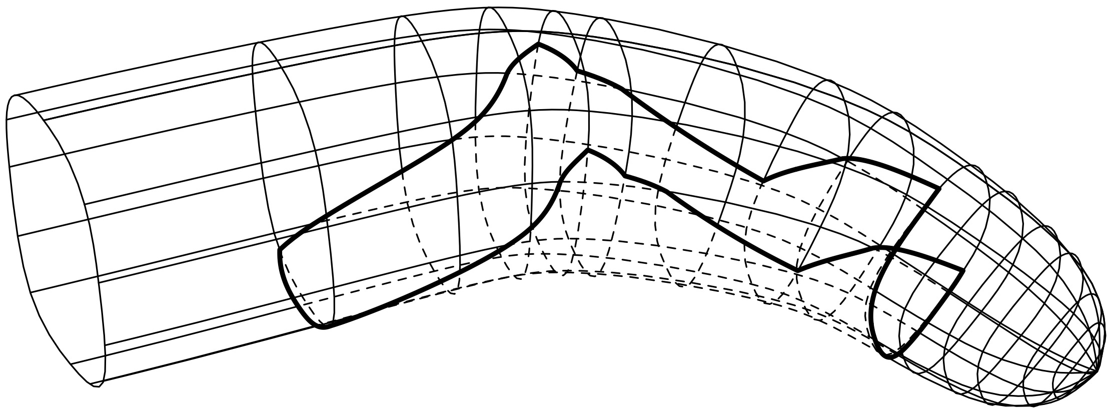

Figure 6: The bicubic surface of the finger in Figure 4 is dichotomized into a convex region (full lines), and a saddle-like region (dashed lines). No concave regions exists in this surface. The parabolic set is also shown in thick lines.

that can be accessed in this mode and drives a ball end tool on the rest of the surface area, yielding a potentially faster machining operation with a better finish quality. The method proposed herein can only improve the machining stage with no possible penalty. The larger the percentage of the convex region in the surface, the larger the percentage of the nice finish region and faster the overwhole operation are going to be.

The algorithm we presented subdivides a freeform surface into two types of regions

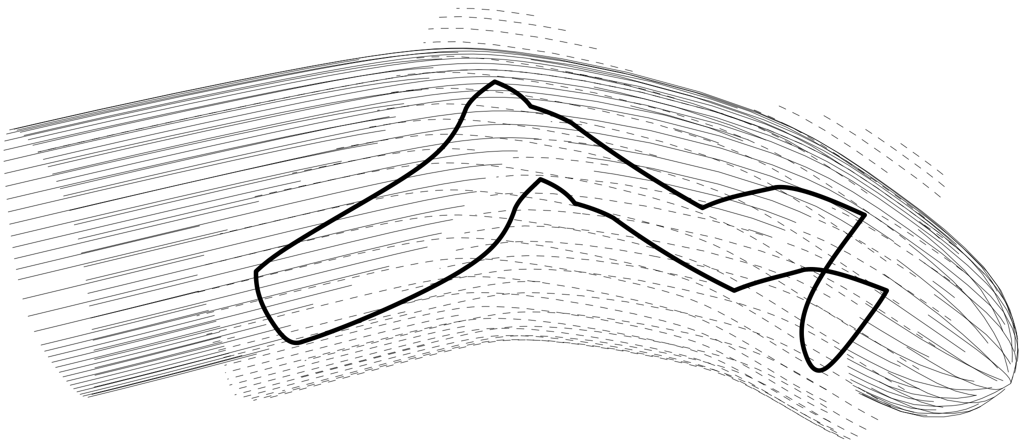

Figure 7: The toolpath of the finger in Figure 5 is split with the aid of the surface shape dichotomy in Figure 6 into regions that can be machined using a flat end tool (full lines) and using a ball end tool (dashed lines).

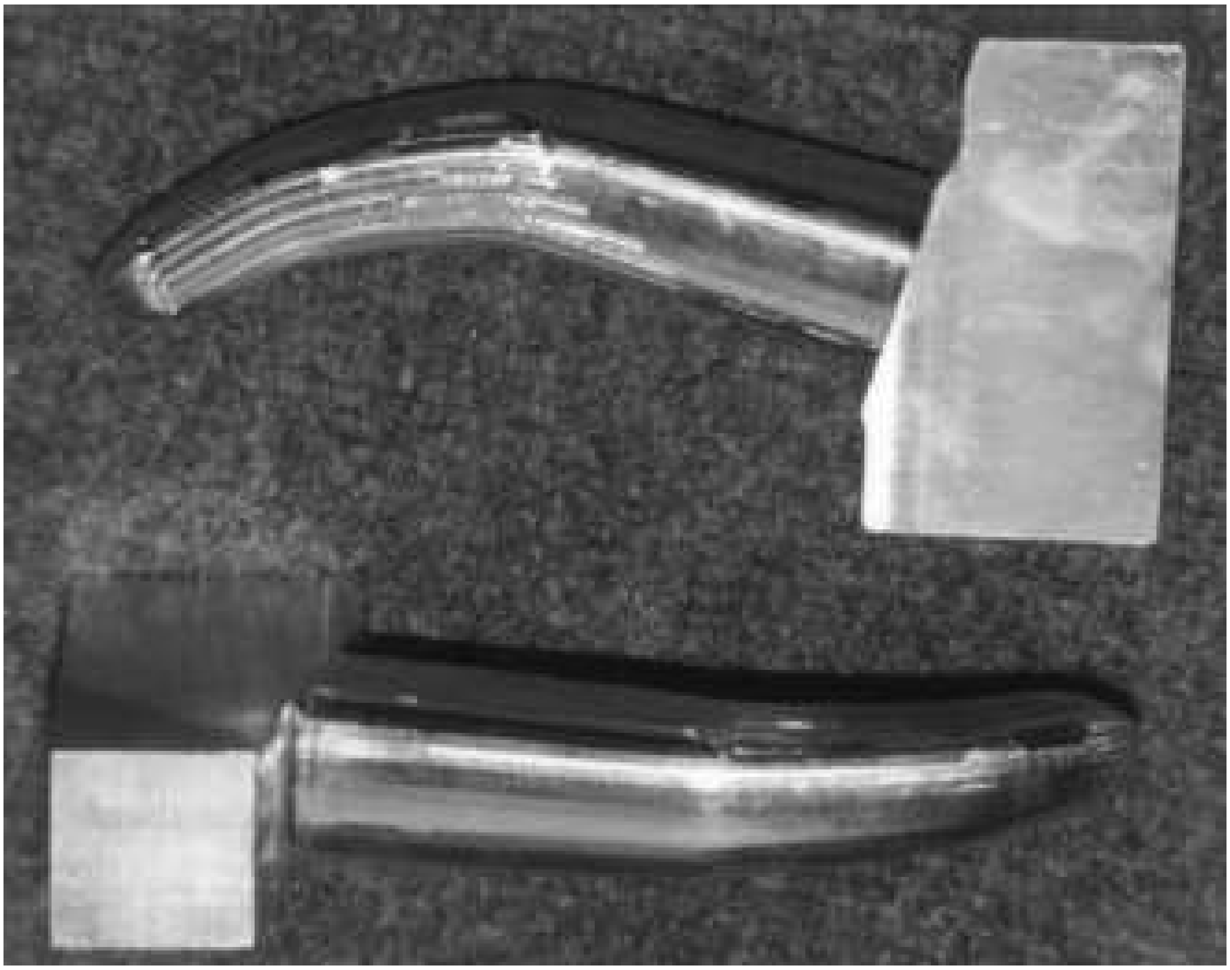

Figure 8: Finger parts machined using the toolpath in Figure 7.

that can be machined using either a flat end or a ball end tool. One can generalize this surface subdivision scheme for regions that can be machined using a flat end tool, regions that can be machined using a large ball end tool, and regions that can be machined using a small ball end tool. This extension can be arbitrarily refined to several sizes of ball end tools. In [9], a method to subdivide a surface into regions with different curvature bounds is discussed. This subdivision can be directly exploited for further refinements of the algorithm proposed herein, for several levels of different sizes of ball end tools, a flat end tool, and possibly even concave ball end tools.

6 Acknowledgments

The author is grateful to the anonymous reviewers of this paper for their valuable remarks.

References

[1] **T. Saito and T. Takahashi.** NC machining with G-buffer method. Computer Graphics, Vol. 25, No. 4, pp 207-216, July 1991 (Siggraph 91).

[2] **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.

[3] **J. D. Foley, A. V. Dam, S. K. Feiner, and J. F. Hughes.** Computer Graphics, Principles and Practice. Addison-Wesley, Second Edition.

[4] **L. L. Chen, S. Y. Chou, and T. C. Woo.** Separating and Intersecting Spherical Polygons: Computing Machinability of Three-, Four- and Five-Axis Numerically Controlled Machines. Transaction on Graphics, Vol. 12, No. 4, pp 305-326, October 1993.

[5] **G. Elber and E. Cohen.** Exact Computation of Gauss Maps and Visibility Sets for Freefrom Surfaces. In preparation.

[6] **G. Elber.** Accessibility in 5-Axis Milling Environment. To appear in CAD.

[7] **B. K. Choi, J. W. Park, and C. S. Jun.** Cutter-Location Data Optimization in 5-axis Surface Machining. Computer Aided Design, Vol. 25, No. 6, pp 377-386, June 1993.

[8] **G. Elber and E. Cohen.** Tool Path Generation for Freeform Surface Models. Second ACM/IEEE Symposium on Solid Modeling and Applications, Montreal Canada, May 1993. Also to appear in CAD.

[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] **M. D. Carmo.** Differential Geometry of Curves and Surfaces. Prentice-Hall 1976.

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

[12] **M. Markot and R. L. Magenson.** Solutions of Tangential Surface and Curve Intersections. *Computer Aided Design*, Vol. 21, No. 7, pp 421-429, September 1989.

[13] **T. W. Sederberg and S. Parry.** Comparison of Three Curve Intersection Algorithms. *Computer Aided Design*, Vol. 18, No. 1, pp 58-63, January/February 1986.

[14] **T. W. Sederberg and T. Nishita.** Curve Intersection Using Bezier Clipping. *Computer Aided Design*, Vol. 22, No. 9, pp 538-549, November 1990.