Geometric Computations in Parameter Space

Myung-Soo Kim* Seoul National University
Gershon Elber† Technion, Israel
Joon-Kyung Seong‡ Seoul National University

Abstract

We review a family of related techniques for geometric computations in the parameter space of freeform curves and surfaces. Geometric constraint equations for freeform curves and surfaces have low degrees (often linear or quadratic) in x, y, z and considerably higher degrees in the curve or surface parameters. We eliminate x, y, and z, so that the constraints are expressed in terms of the curve or surface parameters, while making the variables x, y, z the functions of these parameters under those same constraints. It is relatively straightforward to compute the differential geometric properties of many constructs using this representation. We have successfully addressed the following classes of computation for freeform curves and surfaces: Minkowski sums, bisectors and α-sectors, surface-surface intersections, collision detection, offset trimming, swept volume computation, constructing Voronoi diagrams, convex hulls and kernels, silhouette, and visibility computations. We provide a few simple examples to demonstrate how to apply this technique to a variety of problems in geometric computation.

Keywords: Freeform curves and surfaces, geometric constraints, system of polynomial equations, parameter space

1 Introduction

Conventional research in computer aided geometric design has focused on the design and representation of freeform curves and surfaces [Cohen et al. 2001; Farin 1997; Hoschek and Lasser. 1993; Piegl and Tiller 1995]. Geometric operations permit these geometric shapes to be employed in various applications, such as solid modeling and NC machining. Unary operations include curve and surface offsets, convex hulls, and silhouette computations. Binary operations take two geometric objects as input and produce a different curve or surface as a result. Examples include surface-surface intersection, bisector computation, and determining the convex hull of two curves or surfaces. More generally, we may think of n-ary geometric operations that take n different objects as input and produce some results which are useful for the application under consideration.

Compared with freeform shape design, geometric operations are usually more difficult to support in a reliable way. For example, offsets and surface-surface intersections are in general extremely difficult to compute in a robust way without user intervention. There are always some intricate degenerate cases where these operations produce incorrect results, in particular in the determination of the topology of the resulting curves or surfaces. This fundamental problem is not limited to these two operations. We experience similar deficiency in many geometric operations which deal with constraints specified in xyz-space, i.e. the workspace. One reason for this deficiency is that in most cases the results of geometric operations are produced as algebraic curves and surfaces, and these are unfortunately non-rational in general. In this paper, we present a reasonable way of handling algebraic constraints.

*e-mail:mskim@cse.snu.ac.kr
†e-mail:gershon@cs.technion.ac.il
‡e-mail:swallow@3map.snu.ac.kr

Rational curves and surfaces are represented in parametric form, whereas curves and surfaces are located in the xyz-workspace. This discrepancy between the workspace and the parameter space often causes problems, not only in terms of computational efficiency but also with topological consistency. To resolve these problems, we show how to reduce constraints into the parameter space and transform the results back to the workspace.

Given two freeform rational surfaces $S_1(u, v)$ and $S_2(s, t)$ in xyz-space, geometric constraints are usually represented as nonlinear equations in the seven variables u, v, s, t, x, y, z. The terms in x, y, z are often linear or quadratic since the constraints include conditions such as orthogonality, predetermined fixed angles, equal distances, etc. On the other hand, the terms in u, v, s, t usually have considerably higher degrees. It is easier to eliminate x, y, z and convert the constraints into others that use only the parameters u, v, s, t. If we try to remain in the xyz-space, it is far more difficult to eliminate the high-degree terms in u, v, s, t. In many cases, elimination takes an inordinate amount of computing time. Even if it works, the resulting equations in x, y, z have very high degrees. Thus it is reasonable to eliminate x, y, z instead of u, v, s, t and, more importantly, it is very efficient.

There is another important advantage in eliminating the low-degree terms in x, y, z instead of the high-degree terms in u, v, s, t, under the same constraints. It is relatively straightforward to compute the differential geometric properties of many constructs using this representation. In recent years, we have successfully addressed the following classes of computation for freeform curves and surfaces: Minkowski sums [Lee et al. 1998], bisectors and α-sectors [Elber and Kim 1998; Elber and Kim 2000; Elber et al. 2001], surface-surface intersections [Heo et al. 1999; Heo et al. 2001; Seong et al. 2005a], collision detection [Wang et al. 2001], offset trimming [Seong et al. 2005b], swept volume computation [Elber and Kim 2000], constructing Voronoi diagrams [Hanniel et al. 2005], convex hulls and kernels [Elber et al. 2001; Seong et al. 2001], silhouette computations [Seong et al. 2005b], and visibility computation [Elber et al. 2005]. A recent book [Patrikalakis and Maekawa 2002] also discusses many important problems of shape interrogation that can be reduced to solving equations in parameter space.

Computational geometry has dealt with some of these problems in the discrete domain of points, lines, polygons, and polyhedra. The parameterization of curves and surfaces provides essential information on their spatial coherence. Therefore we need a different approach to developing algorithms for geometric operations on curves and surfaces. In this paper, we review a set of related techniques, which may suggest a different way of looking at geometric computations for freeform curves and surfaces. Our approach suggests that the rational parameterization of freeform curves and surfaces provides the most natural way of dealing with their geometry in the type of geometric operations common in computational geometry and also in computer graphics. Entire computations boil down to solving a set of polynomial equations. This paper is an updated abbreviated version of recent articles [Elber and Kim 2001; Kim and Elber 2000].

The rest of this paper is organized as follows. Sections 2–4 show how to convert problems of geometric constraint into systems of polynomial equations. The problems discussed include the Minkowski sum, bisector, and silhouette computation for freeform shapes. Section 5 concludes this paper.

2 Envelope Curve and Envelope Surface

We present the basic principle of geometric computation in parameter space using the simple examples of envelope curves and surfaces.

2.1 Envelope Curve

Assume that a curve $C_1(u) = (x_1(u), y_1(u))$ is given in the xy-plane and another curve $C_2(v) = (x_2(v), y_2(v))$ moves (with a fixed orientation) along the trajectory curve $C_1(u)$. The sweep of C_2 generates a planar region. This region is bounded by the *envelope curve* $(x(u,v), y(u,v))$, which is defined by

$$x(u,v) = x_1(u) + x_2(v), \quad (1)$$
$$y(u,v) = y_1(u) + y_2(v), \quad (2)$$

under the parallelism constraint $C_1'(u) \parallel C_2'(v)$, which is in turn represented as an algebraic equation in u and v:

$$F(u,v) = x_1'(u)y_2'(v) - y_1'(u)x_2'(v) = 0. \quad (3)$$

In some applications, it is useful to consider the parallelism constraint in the same direction, i.e., $C_1'(u) \parallel C_2'(v)$ and $\langle C_1'(u), C_2'(v) \rangle > 0$. Figure 1 shows two examples of outer envelope curves generated by considering only the pairs of curve points $(C_1(u), C_2(v))$ for which the two curves have the same tangent direction. In Figure 1(a), the moving curve is a circle and the outer envelope curve is thus the untrimmed offset of the trajectory curve. Figure 2(a) shows a non-convex curve moving along a non-convex trajectory curve. The resulting outer envelope curve is considerably more complicated than the two original curves.

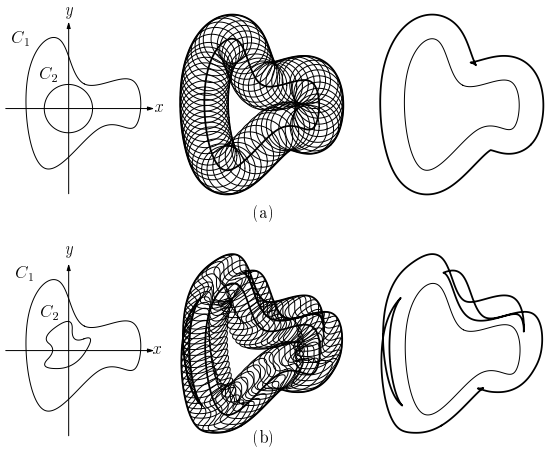


Figure 1: Outer envelope of a curve moving along another curve.

The differential properties of the envelope curve can be computed by evaluating the derivatives of its coordinate functions. Assuming the implicit curve $F(u,v) = 0$ is locally parameterized by u, the parameter v may be interpreted as a function $v(u)$ of u. The first and second derivatives of $v(u)$ can then be computed by differentiating $F(u,v) = 0$ using the chain rule:

$$F_u(u,v) + F_v(u,v)\frac{dv}{du} = 0,$$
$$F_{uu}(u,v) + 2F_{uv}(u,v)\frac{dv}{du} + F_{vv}(u,v)\left(\frac{dv}{du}\right)^2 + F_v(u,v)\frac{d^2v}{du^2} = 0,$$

and more explicitly the derivatives $\frac{dv}{du}$ and $\frac{d^2v}{du^2}$ are given as follows:

$$\frac{dv}{du} = -\frac{F_u}{F_v},$$
$$\frac{d^2v}{du^2} = -\frac{F_{uu} + 2F_{uv}\frac{dv}{du} + F_{vv}\left(\frac{dv}{du}\right)^2}{F_v} = \frac{-F_{uu}F_v^2 + 2F_{uv}F_uF_v - F_{vv}F_u^2}{F_v^3}.$$

Using Equations (1) and (2), the envelope curve is represented as $C(u) = (x(u, v(u)), y(u, v(u)))$. The first and second derivatives of the coordinate functions can be computed as

$$\frac{dx}{du} = x_u(u,v) + x_v(u,v)\frac{dv}{du},$$
$$\frac{dy}{du} = y_u(u,v) + y_v(u,v)\frac{dv}{du},$$
$$\frac{d^2x}{du^2} = x_{uu}(u,v) + 2x_{uv}(u,v)\frac{dv}{du} + x_{vv}(u,v)\left(\frac{dv}{du}\right)^2 + x_v(u,v)\frac{d^2v}{du^2},$$
$$\frac{d^2y}{du^2} = y_{uu}(u,v) + 2y_{uv}(u,v)\frac{dv}{du} + y_{vv}(u,v)\left(\frac{dv}{du}\right)^2 + y_v(u,v)\frac{d^2v}{du^2}.$$

The signed curvature of the envelope curve is then computed by evaluating the following formula [do Carmo 1976]:

$$\kappa(u) = \frac{\frac{dx}{du}\frac{d^2y}{du^2} - \frac{d^2x}{du^2}\frac{dy}{du}}{\left[\left(\frac{dx}{du}\right)^2 + \left(\frac{dy}{du}\right)^2\right]^{3/2}}.$$

Note that the curvature is a non-rational function of u because of the square root function in the denominator.

2.2 Envelope Surface

Given a surface $S_1(u,v) = (x_1(u,v), y_1(u,v), z_1(u,v))$ in xyz-space, another surface $S_2(s,t) = (x_2(s,t), y_2(s,t), z_2(s,t))$ moves (with a fixed orientation) along the surface $S_1(u,v)$ and generates a swept volume. The volume is bounded by *envelope surfaces* $(x(u,v,s,t), y(u,v,s,t), z(u,v,s,t))$, which are defined by

$$x(u,v,s,t) = x_1(u,v) + x_2(s,t), \quad (4)$$
$$y(u,v,s,t) = y_1(u,v) + y_2(s,t), \quad (5)$$
$$z(u,v,s,t) = z_1(u,v) + z_2(s,t), \quad (6)$$

under a parallelism constraint $N_1(u,v) \parallel N_2(s,t)$, where $N_1(u,v) = \frac{\partial S_1}{\partial u}(u,v) \times \frac{\partial S_1}{\partial v}(u,v)$ denotes the normal vector of the surface $S_1(u,v)$, and similarly $N_2(s,t)$ denotes the normal vector of $S_2(s,t)$.

The parallelism constraint $N_1(u,v) \parallel N_2(s,t)$ is formulated as a system of two algebraic equations:

$$F(u,v,s,t) = \left\langle N_1(u,v), \frac{\partial S_2}{\partial s}(s,t) \right\rangle = 0, \quad (7)$$
$$G(u,v,s,t) = \left\langle N_1(u,v), \frac{\partial S_2}{\partial t}(s,t) \right\rangle = 0. \quad (8)$$

Assuming the implicit 2-manifold $F(u,v,s,t) = G(u,v,s,t) = 0$ in $uvst$-space is locally parameterized by u and v, the parameters s and t may be interpreted as functions $s(u,v)$ and $t(u,v)$ of u and v. The first partial derivatives $\frac{\partial s}{\partial u}$ and $\frac{\partial t}{\partial u}$ can be computed by differentiating $F(u,v,s,t) = 0$ and $G(u,v,s,t) = 0$ with respect to u using

the chain rule:

$$F_u(u,v,s,t) + F_s(u,v,s,t)\frac{\partial s}{\partial u} + F_t(u,v,s,t)\frac{\partial t}{\partial u} = 0, \tag{9}$$

$$G_u(u,v,s,t) + G_s(u,v,s,t)\frac{\partial s}{\partial u} + G_t(u,v,s,t)\frac{\partial t}{\partial u} = 0, \tag{10}$$

which produces

$$\frac{\partial s}{\partial u} = -\frac{\begin{vmatrix} F_u & F_t \\ G_u & G_t \end{vmatrix}}{\begin{vmatrix} F_s & F_t \\ G_s & G_t \end{vmatrix}}, \quad \frac{\partial t}{\partial u} = -\frac{\begin{vmatrix} F_s & F_u \\ G_s & G_u \end{vmatrix}}{\begin{vmatrix} F_s & F_t \\ G_s & G_t \end{vmatrix}}. \tag{11}$$

Similarly, the first partial derivatives $\frac{\partial s}{\partial v}$ and $\frac{\partial t}{\partial v}$ can be computed by differentiating $F(u,v,s,t) = 0$ and $G(u,v,s,t) = 0$ with respect to v, using the chain rule:

$$F_v(u,v,s,t) + F_s(u,v,s,t)\frac{ds}{dv} + F_t(u,v,s,t)\frac{dt}{dv} = 0, \tag{12}$$

$$G_v(u,v,s,t) + G_s(u,v,s,t)\frac{ds}{dv} + G_t(u,v,s,t)\frac{dt}{dv} = 0, \tag{13}$$

which produces

$$\frac{ds}{dv} = -\frac{\begin{vmatrix} F_v & F_t \\ G_v & G_t \end{vmatrix}}{\begin{vmatrix} F_s & F_t \\ G_s & G_t \end{vmatrix}}, \quad \frac{dt}{du} = -\frac{\begin{vmatrix} F_s & F_v \\ G_s & G_v \end{vmatrix}}{\begin{vmatrix} F_s & F_t \\ G_s & G_t \end{vmatrix}}. \tag{14}$$

The second partial derivatives $\frac{\partial^2 s}{\partial u^2}$ and $\frac{\partial^2 t}{\partial u^2}$ can be computed by differentiating Equations (9) and (10) with respect to u, again using the chain rule:

$$F_s(u,v,s,t)\frac{\partial^2 s}{\partial u^2} + F_t(u,v,s,t)\frac{\partial^2 t}{\partial u^2} = A(u,v,s,t),$$

$$G_s(u,v,s,t)\frac{\partial^2 s}{\partial u^2} + G_t(u,v,s,t)\frac{\partial^2 t}{\partial u^2} = B(u,v,s,t),$$

where $A(u,v,s,t)$ and $B(u,v,s,t)$ are rational expressions in u,v,s,t, $\frac{\partial s}{\partial u}$, and $\frac{\partial t}{\partial u}$. This produces

$$\frac{\partial^2 s}{\partial u^2} = \frac{\begin{vmatrix} A & F_t \\ B & G_t \end{vmatrix}}{\begin{vmatrix} F_s & F_t \\ G_s & G_t \end{vmatrix}}, \quad \frac{\partial^2 t}{\partial u^2} = \frac{\begin{vmatrix} F_s & A \\ G_s & B \end{vmatrix}}{\begin{vmatrix} F_s & F_t \\ G_s & G_t \end{vmatrix}}. \tag{15}$$

Similarly, the second partial derivatives $\frac{\partial^2 s}{\partial u \partial v}$ and $\frac{\partial^2 t}{\partial u \partial v}$ can be computed by differentiating Equations (9) and (10) with respect to v:

$$F_s(u,v,s,t)\frac{\partial^2 s}{\partial u \partial v} + F_t(u,v,s,t)\frac{\partial^2 t}{\partial u \partial v} = D(u,v,s,t),$$

$$G_s(u,v,s,t)\frac{\partial^2 s}{\partial u \partial v} + G_t(u,v,s,t)\frac{\partial^2 t}{\partial u \partial v} = E(u,v,s,t),$$

where $D(u,v,s,t)$ and $E(u,v,s,t)$ are rational expressions in u,v,s,t, $\frac{\partial s}{\partial u}$, $\frac{\partial s}{\partial v}$, $\frac{\partial t}{\partial u}$, and $\frac{\partial t}{\partial v}$. This produces

$$\frac{\partial^2 s}{\partial u \partial v} = \frac{\begin{vmatrix} D & F_t \\ E & G_t \end{vmatrix}}{\begin{vmatrix} F_s & F_t \\ G_s & G_t \end{vmatrix}}, \quad \frac{\partial^2 t}{\partial u \partial v} = \frac{\begin{vmatrix} F_s & D \\ G_s & E \end{vmatrix}}{\begin{vmatrix} F_s & F_t \\ G_s & G_t \end{vmatrix}}. \tag{16}$$

Finally, using the chain rule one more time, the second partial derivatives $\frac{\partial^2 s}{\partial v^2}$ and $\frac{\partial^2 t}{\partial v^2}$ can be computed by differentiating Equations (12) and (13) with respect to u:

$$F_s(u,v,s,t)\frac{\partial^2 s}{\partial v^2} + F_t(u,v,s,t)\frac{\partial^2 t}{\partial v^2} = I(u,v,s,t),$$

$$G_s(u,v,s,t)\frac{\partial^2 s}{\partial v^2} + G_t(u,v,s,t)\frac{\partial^2 t}{\partial v^2} = J(u,v,s,t),$$

where $I(u,v,s,t)$ and $J(u,v,s,t)$ are rational expressions in u,v,s,t, $\frac{\partial s}{\partial v}$, and $\frac{\partial t}{\partial v}$. This produces

$$\frac{\partial^2 s}{\partial v^2} = \frac{\begin{vmatrix} I & F_t \\ J & G_t \end{vmatrix}}{\begin{vmatrix} F_s & F_t \\ G_s & G_t \end{vmatrix}}, \quad \frac{\partial^2 t}{\partial v^2} = \frac{\begin{vmatrix} F_s & I \\ G_s & J \end{vmatrix}}{\begin{vmatrix} F_s & F_t \\ G_s & G_t \end{vmatrix}}. \tag{17}$$

Using the partial differentials of the bivariate functions $s(u,v)$ and $t(u,v)$, we can compute the first and second partial derivatives of $x(u,v,s(u,v),t(u,v))$, $y(u,v,s(u,v),t(u,v))$, and $z(u,v,s(u,v),t(u,v))$. The first and second fundamental forms and the curvatures can be computed in a straightforward manner for the envelope surface [do Carmo 1976].

3 Bisector Curve and α-Sector Curve

We now apply our problem reduction scheme to the bisector curve of two rational curves in the plane. After that, we will consider the α-sector curve whose minimum distance between the two curves is in the ratio of $\alpha : 1-\alpha$.

3.1 Bisector Curve

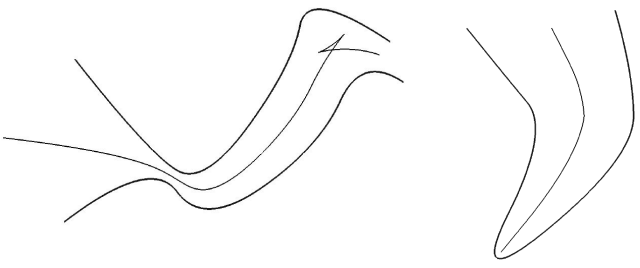
Given two rational curves $C_1(u)$ and $C_2(v)$ in the plane, each point (x,y) on the bisector curve is at an equal minimum distance from $C_1(u)$ and $C_2(v)$; thus each point satisfies

$$\left\langle (x,y) - C_1(u), C_1'(u) \right\rangle = 0, \tag{18}$$

$$\left\langle (x,y) - C_2(v), C_2'(v) \right\rangle = 0, \tag{19}$$

$$\left\langle (x,y) - \frac{C_1(u)+C_2(v)}{2}, C_1(u) - C_2(v) \right\rangle = 0. \tag{20}$$

Figure 2(a) shows an example of a bisector curve between two rational curves in the plane, and Figure 2(b) shows an example of a self-bisector of a rational curve.



(a) (b)

Figure 2: (a) Bisector curve and (b) self-bisector curve.

Conventional approaches would eliminate the parameters u and v to generate an implicit equation $b(x,y) = 0$ for the bisector curve

in the xy-plane [Hoffmann and Vermeer 1991], or trace the curve $b(x,y) = 0$ numerically [Farouki and Johnstone. 1994; Farouki and Ramamurthy 2001]. Unfortunately, the algebraic degree of $b(x,y) = 0$ is very high. (For two cubic curves, the degree of $b(x,y) = 0$ is 46.) Note that Equations (18)–(20) are linear in x and y. Thus it is considerably easier to eliminate the variables x and y than the other curve parameters, u and v. Moreover, the degree of the resulting implicit equation $F(u,v) = 0$ is significantly lower than that of $b(x,y) = 0$; in this case the degree of $F(u,v)$ is 10 when both $C_1(u)$ and $C_2(v)$ are cubic polynomial curves. (For more details, see [Elber and Kim 1998].)

In this context, Equations (18) and (19) mean that the bisector is located at the intersection point of the lines normal to the two curves, which can be computed using Cramer's rule as follows:

$$x(u,v) = \frac{\begin{vmatrix} x_1(u)x_1'(u)+y_1(u)y_1'(u) & y_1'(u) \\ x_2(v)x_2'(v)+y_2(v)y_2'(v) & y_2'(v) \end{vmatrix}}{\begin{vmatrix} x_1'(u) & y_1'(u) \\ x_2'(v) & y_2'(v) \end{vmatrix}}, \tag{21}$$

$$y(u,v) = \frac{\begin{vmatrix} x_1'(u) & x_1(u)x_1'(u)+y_1(u)y_1'(u) \\ x_2'(v) & x_2(v)x_2'(v)+y_2(v)y_2'(v) \end{vmatrix}}{\begin{vmatrix} x_1'(u) & y_1'(u) \\ x_2'(v) & y_2'(v) \end{vmatrix}}. \tag{22}$$

Since this intersection point $(x(u,v),y(u,v))$ is located on the bisector curve, it satisfies Equation (20). Thus we have the following constraint equation for u and v, which represents an implicit curve in the uv-parameter plane:

$$F(u,v) = \left\langle (x(u,v),y(u,v)) - \frac{C_1(u)+C_2(v)}{2}, C_1(u)-C_2(v) \right\rangle = 0.$$

The differential properties of the bisector curve can be computed using the technique introduced in Section 2.1.

3.2 α-Sector Curve

Given two rational curves $C_1(u)$ and $C_2(v)$ in the plane, each point (x,y) on the α-bisector curve is at a relative distance $\alpha : 1-\alpha$ from $C_1(u)$ and $C_2(v)$; thus each point satisfies

$$\left\langle (x,y) - C_1(u), C_1'(u) \right\rangle = 0, \tag{23}$$

$$\left\langle (x,y) - C_2(v), C_2'(v) \right\rangle = 0, \tag{24}$$

$$(1-\alpha)^2 \|(x,y) - C_1(u)\|^2 - \alpha^2 \|(x,y) - C_2(v)\|^2 = 0. \tag{25}$$

The first two equations produce the same solutions as in Equations (21) and (22). However, the third equation produces an implicit curve of higher degree:

$$\begin{aligned} F(u,v) \;=\; & (1-\alpha)^2 \|(x(u,v),y(u,v)) - C_1(u)\|^2 \\ & -\alpha^2 \|(x(u,v),y(u,v)) - C_2(v)\|^2 = 0. \end{aligned}$$

4 Bisector Surface

A similar approach can be applied to computing the bisector surface of two rational freeform surfaces [Elber and Kim 2000]. Figure 3 shows an example of a surface-plane bisector.

Let $S_1(u,v)$ and $S_2(s,t)$ be two rational surfaces. We consider the bisector surface of $S_1(u,v)$ and $S_2(s,t)$, which consists of points (x,y,z) satisfying the following five constraint equations:

$$\left\langle (x,y,z) - S_1(u,v), \frac{\partial S_1(u,v)}{\partial u} \right\rangle = 0, \tag{26}$$

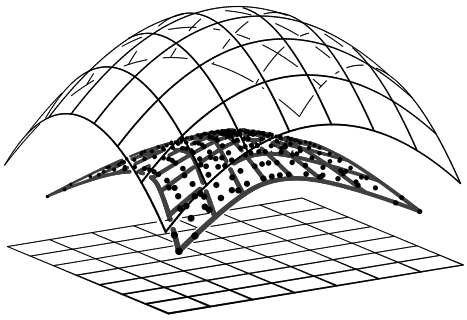


Figure 3: Bisector surface between a freeform surface and a plane.

$$\left\langle (x,y,z) - S_1(u,v), \frac{\partial S_1(u,v)}{\partial v} \right\rangle = 0, \tag{27}$$

$$\left\langle (x,y,z) - S_2(s,t), \frac{\partial S_2(s,t)}{\partial s} \right\rangle = 0, \tag{28}$$

$$\left\langle (x,y,z) - S_2(s,t), \frac{\partial S_2(s,t)}{\partial t} \right\rangle = 0, \tag{29}$$

$$\begin{aligned} & \langle (x,y,z), 2(S_2(s,t) - S_1(u,t)) \rangle \\ & \quad + \|S_1(u,t)\|^2 - \|S_2(s,t)\|^2 = 0. \end{aligned} \tag{30}$$

Equations (26) and (27) mean that the bisector point (x,y,z) is located on the normal to $S_1(u_1,v_1)$, and Equations (28) and (29) imply that the point (x,y,z) is on the normal to $S_2(u_2,v_2)$. Additionally, Equation (30) constrains the point (x,y,z) to the symmetry plane of $S_1(u_1,v_1)$ and $S_2(u_2,v_2)$. Equations (26)–(30) are all linear in (x,y,z).

By applying Cramer's rule to the linear system formed by Equations (26), (27), and (30), we can represent $x(u,v,s,t), y(u,v,s,t), z(u,v,s,t)$ as rational functions of u,v,s,t. Substituting these rational expressions to Equations (28) and (29), we obtain the following two rational constraint equations:

$$F_1(u,v,s,t) = \left\langle B(u,v,s,t) - S_2(s,t), \frac{\partial S_2(s,t)}{\partial s} \right\rangle = 0, \tag{31}$$

$$F_2(u,v,s,t) = \left\langle B(u,v,s,t) - S_2(s,t), \frac{\partial S_2(s,t)}{\partial t} \right\rangle = 0, \tag{32}$$

where $B(u,v,s,t) = (x(u,v,s,t), y(u,v,s,t), z(u,v,s,t))$.

The common zero-set of Equations (31) and (32) satisfies all five constraints of Equations (26)–(30). Hence, we have reduced the surface-surface bisector problem to a common zero-set finding problem for the two four-variate functions of Equations (31) and (32). For each point (u,v,s,t) in the resulting zero-set, the corresponding bisector point can be computed by the rational map $(x(u,v,s,t), y(u,v,s,t), z(u,v,s,t))$. The differential properties of the bisector surface can be computed using the technique introduced in Section 2.2.

5 Perspective Silhouette of Swept Volume

Let O denote a three-dimensional object bounded by a rational freeform surface $S(u,v)$, and let $A(t)$ denote an affine transformation represented by a 4×4 matrix,

$$\begin{bmatrix} a_{11}(t) & a_{12}(t) & a_{13}(t) & t_x(t) \\ a_{21}(t) & a_{22}(t) & a_{23}(t) & t_y(t) \\ a_{31}(t) & a_{32}(t) & a_{33}(t) & t_z(t) \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

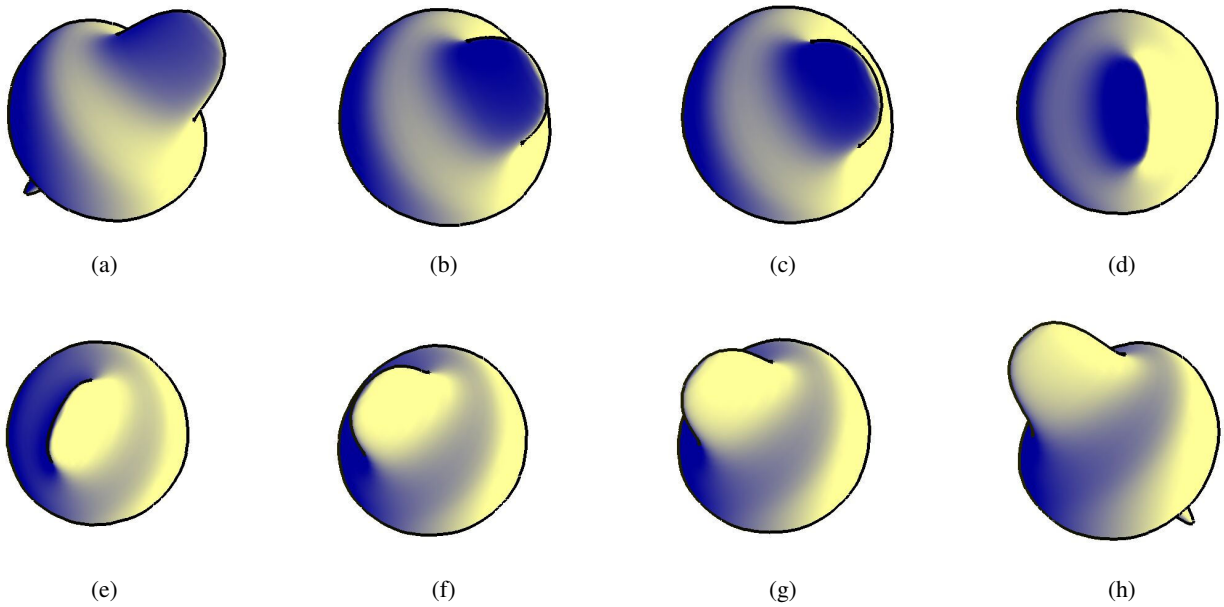


Figure 4: Snapshots of time-varying silhouettes.

where $(a_{ij}(t))_{3\times 3}$ represents a linear transformation (e.g. rotation or shearing) and $(t_x(t), t_y(t), t_z(t))$ denotes a translation of the coordinate system.

The swept volume of the object O under the affine transformation $A(t)$ is given by $\cup_t A(t)[O]$. Assuming $a \le t \le b$, the boundary surface of the swept volume consists of some patches of the surface $A(a)[S(u,v)]$ and some of $A(b)[S(u,v)]$, together with the boundary envelope surface. The set of points on the envelope surface is characterized by the following equation [Martin and Stephenson 1990]:

$$F(u,v,t) = \left| A'(t)[S(u,v)] \;\; A(t)\left[\frac{\partial S}{\partial u}(u,v)\right] \;\; A(t)\left[\frac{\partial S}{\partial v}(u,v)\right] \right| = 0.$$

The silhouette points on the boundary of the swept volume $\cup A(t)[S(u,v)]$, seen from a viewpoint $\mathbf{p}$, satisfy the following implicit equation:

$$G(u,v,t) = \langle A(t)[S(u,v)] - \mathbf{p}, A(t)[N(u,v)] \rangle = 0, \tag{33}$$

where $N(u,v)$ is the normal to $S(u,v)$. Since $N(u,v) = \frac{\partial S}{\partial u} \times \frac{\partial S}{\partial v}$ is rational, the function $G(u,v,t)$ is also rational. The common zero-set of $F(u,v,t) = G(u,v,t) = 0$ produces 1-manifold curves in uvt-space, which correspond to the silhouette curves of the boundary of the swept volume. Figure 4 shows snapshots from an animation of silhouettes in which their topological arrangements change as the eye position moves along a predefined path.

Assuming the implicit 1-manifold curve $F(u,v,t) = G(u,v,t) = 0$ in uvt-space is locally parameterized by t, the parameters u and v may be interpreted as functions $u(t)$ and $v(t)$ of t. The differential properties of the silhouette curve $(x(t), y(t), z(t)) = A(t)[S(u(t), v(t))]$ can be computed by combining the techniques introduced in Sections 2.1 and 2.2 as follows.

The first partial derivatives $\frac{\partial u}{\partial t}$ and $\frac{\partial v}{\partial t}$ can be computed by differentiating $F(u,v,t) = 0$ and $G(u,v,t) = 0$ with respect to t using the chain rule:

$$F_u(u,v,t)\frac{\partial u}{\partial t} + F_v(u,v,t)\frac{\partial v}{\partial t} + F_t(u,v,t) = 0, \tag{34}$$

$$G_u(u,v,t)\frac{\partial u}{\partial t} + G_v(u,v,t)\frac{\partial v}{\partial t} + G_t(u,v,t) = 0, \tag{35}$$

which produces

$$\frac{\partial u}{\partial t} = -\frac{\begin{vmatrix} F_t & F_v \\ G_t & G_v \end{vmatrix}}{\begin{vmatrix} F_u & F_v \\ G_u & G_v \end{vmatrix}}, \quad \frac{\partial v}{\partial t} = -\frac{\begin{vmatrix} F_u & F_t \\ G_u & G_t \end{vmatrix}}{\begin{vmatrix} F_u & F_v \\ G_u & G_v \end{vmatrix}}. \tag{36}$$

Similarly, the second partial derivatives $\frac{\partial^2 u}{\partial t^2}$ and $\frac{\partial^2 v}{\partial t^2}$ can be computed by differentiating Equations (34) and (35) with respect to t, again using the chain rule:

$$F_u(u,v,t)\frac{\partial^2 u}{\partial t^2} + F_v(u,v,t)\frac{\partial^2 v}{\partial t^2} = A(u,v,t),$$
$$G_u(u,v,t)\frac{\partial^2 u}{\partial t^2} + G_v(u,v,t)\frac{\partial^2 v}{\partial t^2} = B(u,v,t),$$

where $A(u,v,t)$ and $B(u,v,t)$ are rational expressions in u, v, t, $\frac{\partial u}{\partial t}$, and $\frac{\partial v}{\partial t}$. This produces

$$\frac{\partial^2 u}{\partial t^2} = \frac{\begin{vmatrix} A & F_v \\ B & G_v \end{vmatrix}}{\begin{vmatrix} F_u & F_v \\ G_u & G_v \end{vmatrix}}, \quad \frac{\partial^2 v}{\partial t^2} = \frac{\begin{vmatrix} F_u & A \\ G_u & B \end{vmatrix}}{\begin{vmatrix} F_u & F_v \\ G_u & G_v \end{vmatrix}}. \tag{37}$$

Using the partial derivatives of the functions $u(t)$ and $v(t)$, we can compute the first and second partial derivatives of the silhouette curve $(x(t), y(t), z(t))$. The curvature of the space curve is then computed as follows [do Carmo 1976]:

$$\kappa(t) = \frac{\|(x''(t), y''(t), z''(t)) \times (x'(t), y'(t), z'(t))\|}{\|(x'(t), y'(t), z'(t))\|^3}.$$

Note that the curvature is a non-rational function of t because of the square root function in the denominator.

6 Conclusions

In this paper we presented a problem reduction scheme that converts geometric constraints in work space to a system of equations in parameter space. The effectiveness of this approach has been demonstrated through a few simple geometric operations. In recent years, we have successfully applied the same approach to a wider variety of geometric computations which are considerably more involved than those discussed in this paper. New techniques have been introduced to deal with these problems. Future work will introduce even more advanced techniques to attack more complex geometric problems.

Acknowledgements

This work was supported in part by the Korean Ministry of Information and Communication (MIC) under the Program of IT Research Center on CGVR, in part by grant No. R01-2002-000-00512-0 from the Basic Research Program of the Korea Science and Engineering Foundation (KOSEF), and in part by the Israeli Ministry of Science Grant No. 01-01-01509.

References

COHEN, E., RIESENFELD, R., AND ELBER, G. 2001. *Geometric Modeling with Splines: An Introduction*, A.K. Peters, Natick, USA.

DO CARMO, M. 1976. *Differential Geometry of Curves and Surfaces*, Prentice-Hall, Upper Saddle River, USA. http://www.cs.technion. ac.il/~irit.

ELBER, G. 1996. *IRIT 7.0 User's Manual*, Technion, http://www.cs.technion. ac.il/~irit.

ELBER, G., BAREQUET, G., KIM, M.-S. 2001. Bisectors and α-sectors of rational varieties. *Computing*, **65**.

ELBER, G., KIM, M.-S. 1998. Bisector curves of planar rational curves. *Computer-Aided Design* **30**, 1089–1096.

ELBER, G., KIM, M.-S. 2000. A computational model for nonrational bisector surfaces: curve-surface and surface-surface bisectors. *Proc. of Geometric Modeling and Processing 2000*, Hong Kong, April 10-12, 2000.

ELBER, G., KIM, M.-S. 2001. Geometric constraint solver using multivariate rational spline functions. *Proc. of ACM Symposium on Solid Modeling and Applications*, Ann Arbor, MI, June 4–8, 2001.

ELBER, G., KIM, M.-S., HEO, H.-S. 2001. The convex hull of rational plane curves. *Graphical Models*, 2001.

ELBER, G., SAYEGH, R., BAREQUET, G., MARTIN, R. 2005. Two-dimensional visibility charts for continuous curves. *Proc. of Shape Modeling International*, 2005.

FARIN, G. 1997. *Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide*, Fourth Ed., Academic Press, San Diego.

FAROUKI, R., JOHNSTONE, J. 1994. Computing point/curve and curve/curve bisectors. *Design and Application of Curves and Surfaces: Mathematics of Surfaces V*, R.B. Fisher (Ed.), Oxford Univ. Press, New York, 327–354.

FAROUKI, R., RAMAMURTHY, R. 1998. Specified-precision computation of curve/curve bisectors. *Int'l J. of Computational Geometry & Applications* **8**, 599–617.

HANNIEL, I., MUTHUGANAPATHY, R., ELBER, G., KIM, M.-S. 2005. Precise Voronoi cell extraction of free-form rational planar closed curves. *Proc. of ACM Symposium on Solid and Physical Modeling*, Cambridge, USA.

HEO, H.-S., KIM, M.-S., ELBER, G. 1999. The intersection of two ruled surfaces. *Computer-Aided Design* **31**, 33–50.

HEO, H.-S., HONG, S., SEONG, J.-K., KIM, M.-S., ELBER, G. 1999. The intersection of two ringed surfaces and some related problems, *Graphical Models*, **63**(4). 228–244.

HOFFMANN, C., VERMEER, P. 1991. Eliminating extraneous solutions in curve and surface operation. *Int'l J. of Comp. Geom. and Appl.* **1**, 47–66.

HOSCHEK, J., LASSER, D. 1993. *Fundamentals of Computer Aided Geometric Design*, A.K. Peters, Wellesley, USA.

KIM, M.-S., ELBER, G. 2000. Problem reduction to parameter space. *The Mathematics of Surfaces IX (Proc. of the Ninth IMA Conference)*, R. Cipolla and R. Martin (Eds), Springer, London, 2000, pp 82–98.

LEE, I.-K., KIM, M.-S., ELBER, G. 1998. Polynomial/rational approximation of Minkowski sum boundary curves. *Graphical Models and Image Processing* **60**, 136–165

MARTIN, R., STEPHENSON, P. 1990. Sweeping of three-dimensional objects. *Computer-Aided Design* **22**, 223–234.

PATRIKALAKIS, N., MAEKAWA, T. 2002. *Shape Interrogation for Computer Aided Design and Manufacturing*, Springer-Verlag, Berlin.

PIEGL, L., TILLER, W. 1995. *The NURBS Book*, Springer, Berlin.

SEONG, J.-K., KIM, M.-S., AND ELBER, G. 2001. The convex hull of rational surfaces. *Proc. of Third Israel-Korea Binational Conf. on Geometric Modeling and Computer Graphics*, Seoul, Korea, 171–175.

SEONG, J.-K., KIM, K.-J., KIM, M.-S., ELBER, G., MARTIN, R. 2005. Intersecting a freeform surface with a general swept surface. *Computer-Aided Design*, **37**(5), 473–483.

SEONG, J.-K., KIM, K.-J., KIM, M.-S., ELBER, G. 2005. Perspective silhouette of a general swept volume. To appear in *The Visual Computer*.

SEONG, J.-K., ELBER, G., KIM, M.-S. 2005. Trimming local and global self-intersections in offset curves/surfaces using distance maps Submitted to *Computer-Aided Design*.

SHERBROOKE, E., PATRIKALAKIS, N. 1993. Computation of the solutions of nonlinear polynomial systems. *Computer Aided Geometric Design* **10**, 379–405.

WANG, W., WANG, J., KIM, M.-S. 2001. An algebraic condition for the separation of two ellipsoids. *Computer Aided Geometric Design*, **18**(6), pp. 531–539.