Intersecting a freeform surface with a general swept surface

Joon-Kyung Seong[a], Ku-Jin Kim[b], Myung-Soo Kim[a,c,*], Gershon Elber[d,1], Ralph R. Martin[e]

[a]*School of Computer Science and Engineering, Seoul National University, Seoul 151-742, South Korea*
[b]*Department of Computer Engineering, Kyungpook National University, Kyungpook, South Korea*
[c]*Institute of Computer Technology, Seoul National University, Seoul, South Korea*
[d]*Department of Computer Science, Technion, Haifa 32000, Israel*
[e]*School of Computer Science, Cardiff University, South Wales, UK*

Abstract

We present efficient and robust algorithms for intersecting a rational parametric freeform surface with a general swept surface. A swept surface is given as a one-parameter family of cross-sectional curves. By computing the intersection between a freeform surface and each cross-sectional curve in the family, we can solve the intersection problem. We propose two approaches, which are closely related to each other. The first approach detects certain critical points on the intersection curve, and then connects them in a correct topology. The second approach converts the intersection problem to that of finding the zero-set of polynomial equations in the parameter space. We first present these algorithms for the special case of intersecting a freeform surface with a ruled surface or a ringed surface. We then consider the intersection with a general swept surface, where each cross-sectional curve may be defined as a rational parametric curve or as an implicit algebraic curve.

© 2004 Elsevier Ltd. All rights reserved.

1. Introduction

Surface–surface intersection (SSI) is an important problem in geometric modeling and processing, particularly for applications in CAD/CAM and solid modeling. Many different approaches have been proposed. However, it is still a difficult problem to solve with sufficient accuracy, efficiency and robustness. The main difficulty lies in analyzing the topological structure of the intersection curve: it is not easy to determine the exact number of connected components and the correct topological arrangement of these components.

Because of the difficulty in dealing with general freeform surfaces, considerable research has been devoted to intersecting surfaces of special types [13]. Martin et al. [14] and de Pont [3] considered intersection of a cyclide with a quadric, and with another cyclide. Johnstone [11] proposed an algorithm to intersect a cyclide with a ringed surface. (The term *ringed* surface was coined by Johnstone [11]; it is a surface generated by sweeping a circle while changing its size and orientation). Heo et al. [8] presented an algorithm for intersecting two ruled surfaces. They reduced the intersection problem to a search for the zero-set of the function: $f(u, v)=0$, where $f(u, v)$ is a bivariate polynomial of relatively low degree. This approach was later applied to the case of intersecting two ringed surfaces [7]. In the present paper, we extend this result to the intersection of a rational parametric freeform surface with a general swept surface. A swept surface is a one-parameter family of cross-sectional curves. In this paper, we focus on the case where each curve in the family is obtained by applying an affine transformation to a template curve. By taking an affine spline motion applied to a template curve [10], we can

* Corresponding author. Address: School of Computer Science and Engineering, Seoul National University, Seoul 151-742, South Korea. Tel.: +82 2 880 1838; fax: +82 2 889 1838.

E-mail addresses: swallow@3map.snu.ac.kr (J.-K. Seong), kujinkim@knu.ac.kr (K.-J. Kim), mskim@cse.snu.ac.kr (M.-S. Kim), gershon@cs.technion.ac.il (G. Elber), ralph@cs.cf.ac.uk (R.R. Martin).
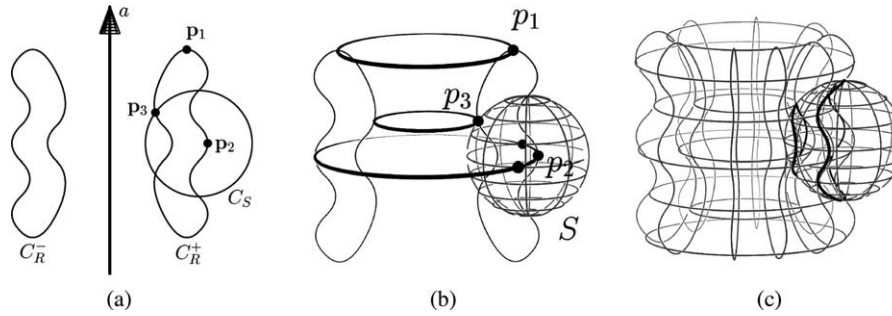
[1] Fax: +972-4-829-5538.

Fig. 1. Three different types of intersection between S and the circles of R.

generate a swept surface. Ruled surfaces and ringed surfaces are special types of swept surfaces, which are constructed by sweeping a line or a circle.

We present two methods for intersecting a freeform surface with a swept surface. The first approach detects certain critical points on the intersection curve, and connects them in a correct topology; the curve segments connecting each pair of adjacent critical points is generated by a numerical curve tracing technique [1,2,5]. This procedure gives the entire intersection curve. The second approach transforms the intersection problem into a simple problem of solving a system of $(n-1)$ polynomial equations in n variables. (See Elber and Kim [6] or Patrikalakis and Maekawa [15] for details of solving a system of m polynomial equations in n variables).

Kim et al. [12] previously proposed a related topology-construction approach for intersecting a sphere with a surface of revolution. A surface of revolution is a special kind of swept surface which is generated by sweeping a circle of varying radius along a line. Fig. 1(a) shows the silhouette C_S of a sphere S and the generating curve C_R^+ for a surface of revolution R. Each point $\mathbf{p}_1(\in C_R^+)$ outside the silhouette C_S generates a circle on the surface R that has no intersection with the sphere S. Thus we can completely ignore the curve segments of C_R^+ that lie outside C_S. On the other hand, each point $\mathbf{p}_2(\in C_R^+)$ inside C_S generates a circle that intersects the sphere S transversally in two distinct points. When we follow the curve C_R^+ inside the circle C_S, we can trace two branches of the intersection curve $S \cap R$, one on the front hemisphere and the other on the back hemisphere. The two branches are first created (starting at the same point) when the curve C_R^+ enters the circle C_S, and they merge to form a closed loop when the curve C_R^+ leaves C_S. Finally, each point $\mathbf{p}_3$ lying *on* C_R^+ and $C_S(\in C_R^+ \cap C_S)$ generates a circle that intersects S tangentially. All tangential intersections between the sphere S and the cross-sectional circles of R can be detected by computing $C_R^+ \cap C_S$, which can be reduced to solving a univariate polynomial equation. We will use similar principles in the first approach presented in this paper.

Here, we consider the intersection between a rational parametric freeform surface and a general swept surface. The topology of the intersection curve is in this case

determined by certain critical points where cross-sectional curves of a swept surface intersects a freeform surface tangentially. The geometric condition of tangential intersection can be formulated as a system of n polynomial equations in n variables, which in general produces discrete solutions. The tangential condition is formulated as one of these equations. By deleting this equation, we obtain a system of $(n-1)$ equations in n variables, the solution of which generates a 1-manifold in the parameter space of the given freeform surface and the swept surface. The intersection curve is then constructed by projecting the 1-manifold into the uv-parameter plane of the freeform surface $S(u, v)$.

We first consider the problem of intersecting a freeform surface with a ruled surface or a ringed surface. We then consider the case of intersecting a freeform surface with a general swept surface. Each cross-sectional curve of the swept surface may be defined as a rational parametric curve or as an implicit algebraic curve that moves under a rational affine spline motion [10].

The rest of this paper is organized as follows. In Section 2, we discuss how to construct the correct topology of an intersection curve based on detecting certain critical points on the curve. In Section 3, we give a simple technique for reducing the intersection problem to that of computing the simultaneous zero-set of $(n-1)$ polynomial equations in n variables. Section 4 presents some experimental results. Finally, in Section 5, we conclude this paper.

2. Topology construction

We will first present an algorithm that detects critical points on the intersection curve and constructs the correct topological structure of the curve. The critical points are tangential intersection points between the freeform surface and the cross-sectional curves of the swept surface. The basic idea of this approach is first explained using an illustrative example for a simple case where a freeform surface is intersected with a general cylindrical surface (i.e. a linear extrusion surface). Then we proceed to the cases where a freeform surface is intersected with a ruled surface, or a ringed surface. Finally, we show algorithms for
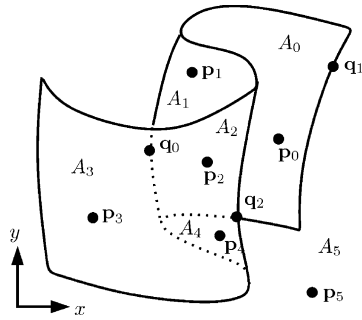
Fig. 2. Regions on the xy-plane delimited by the silhouette curves and the boundary curves of $S(u,v)$.

intersecting a rational freeform surface with a general swept surface.

2.1. Intersection with a cylindrical surface

For the sake of simplicity, we first consider the case of intersecting a freeform surface with a cylindrical surface. We may assume that the cylindrical surface is generated by extruding a plane curve $C(t)=(x(t), y(t), 0)$ along the z-direction. Let $L(\mathbf{p})$ denote a line passing through a point $\mathbf{p}$ in the xy-plane and parallel to the z-axis. The line $L(C(t))$ intersects a freeform surface $S(u,v)=(x(u,v), y(u,v), z(u,v))$ tangentially if and only if the point $C(t)=(x(t), y(t), 0)$ is located on the silhouette of $S(u,v)$ when viewed along the z-direction. The silhouette curves and the boundary curves of $S(u,v)$ subdivide the xy-plane into several regions. Fig. 2 shows some such surface $S(u,v)$ and its silhouette and boundary curves projected on to the xy-plane. Also shown are the corresponding six regions $A_0,\dots,A_5$ on the plane.

The location of a point $\mathbf{p}$ (in the xy-plane) leads to the following relation between a line $L(\mathbf{p})$ and the surface $S(u,v)$:

(1) If $\mathbf{p}$ is on the silhouette curve of $S(u,v)$, the line $L(\mathbf{p})$ intersects $S(u,v)$ tangentially at some location.
(2) If $\mathbf{p}$ is inside a region A_i, then $L(\mathbf{p})$ intersects $S(u, v)$ transversally at each intersection point.

(3) If $\mathbf{p}$ and $\mathbf{q}$ are in the same region A_i, then $L(\mathbf{p})$ and $L(\mathbf{q})$ have the same number of intersections with $S(u,v)$.

For points $\mathbf{p}_i$, $(0 \leq i \leq 5)$, located in the region A_i (Fig. 2), the intersection between $L(\mathbf{p}_i)$ and $S(u,v)$ is classified as follows:

(1) $L(\mathbf{p}_0) \cap S(u, v)$ consists of one transversal intersection point.
(2) $L(\mathbf{p}_1) \cap S(u, v)$ consists of two transversal intersection points.
(3) $L(\mathbf{p}_2) \cap S(u, v)$ consists of three transversal intersection points.
(4) $L(\mathbf{p}_3) \cap S(u, v)$ consists of one transversal intersection point.
(5) $L(\mathbf{p}_4) \cap S(u, v)$ consists of two transversal intersection points.
(6) $L(\mathbf{p}_5) \cap S(u, v)$ has no intersection point.

For points $\mathbf{q}_j$, $(0 \leq i \leq 2)$, on the silhouette or boundary curves, the intersection between $L(\mathbf{q}_j)$ and $S(u,v)$ is classified as follows:

(1) $L(\mathbf{q}_0) \cap S(u, v)$ consists of one tangential intersection point and one transversal intersection point.
(2) $L(\mathbf{q}_1) \cap S(u, v)$ consists of one boundary point of $S(u,v)$.
(3) $L(\mathbf{q}_2) \cap S(u, v)$ consists of one tangential intersection point and one boundary point of $S(u,v)$.

As one can notice from the above example, the number of intersections between the line $L(\mathbf{p})$ and the surface $S(u,v)$ changes when the point $\mathbf{p}$ passes across the silhouette or boundary curves of $S(u,v)$. When the point $\mathbf{p}$ traces a continuous curve $C(t)$ in the xy-plane, this observation leads to a classification of the topology of the intersection curve between a surface $S(u,v)$ and a cylindrical surface. Let $R(s,t)$ denote the cylindrical surface generated by extruding $C(t)=(x(t), y(t), 0)$ along the z-direction, where we take $s=z$. That is, the curve $C(t)$ in Fig. 3(a) is the projection of the cylindrical surface $R(s,t)$ on to the xy-plane. For a fixed t_0, each ruling line $R(s,t_0)$ is projected on to a curve point $C(t_0)$.
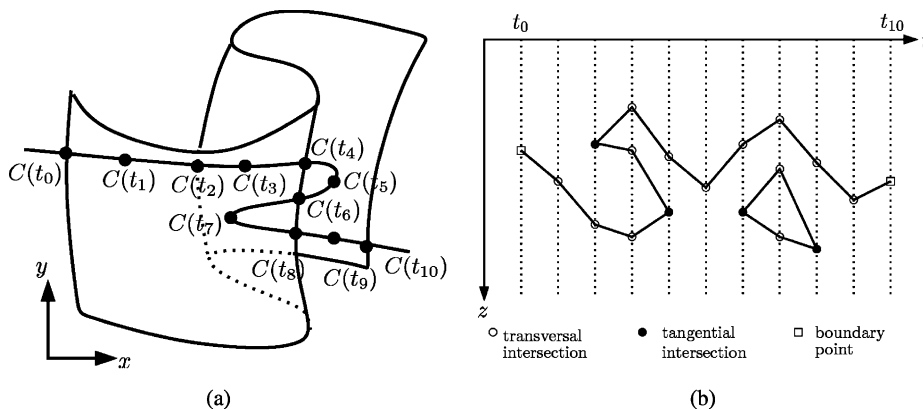

(a)   (b)

Fig. 3. Topological structure of the intersection curve between $R(s,t)$ and $S(u,v)$.

The topological structure of the intersection curve between $S(u,v)$ and $R(s,t)$ is completely determined by the intersection between $C(t)$ and the silhouette and boundary curves of $S(u,v)$. By subdividing $S(u,v)$ and $C(t)$ if necessary, we may assume that the curve $C(t)$ does not pass through any of the self-intersection points of the silhouette or boundary curves of $S(u,v)$; that is, it does not pass through points such as q_2 in Fig. 2. Moreover, we assume that the surface $S(u,v)$ and the curve $C(t)$ have no self-intersection.

Fig. 3(a) shows a sequence of points $C(t_i)$, for $i=0,\dots,10$, where the even-indexed curve points $C(t_0),C(t_2),\dots,C(t_{10})$ are located *on* the silhouette or boundary curves of $S(u,v)$. On the other hand, the odd-indexed curve points $C(t_1),C(t_3),\dots,C(t_9)$ lie *inside* the regions A_i; the t_i satisfy $t_{i-2} < t_{i-1} < t_i$. Now, in Fig. 3(b), the curve $C(t)$ is stretched out along its parameter line, which is the t-axis. Each dotted line (parallel to the z-axis) corresponds to a ruling line $L(C(t_i))$ of the cylindrical surface $R(s,t)$. The correct topology of the intersection curve is completely determined by the discrete set of intersection points $L(C(t_i)) \cap S(u,v)$, for $i=0,\dots,10$. The intersection points on each line $L(C(t_i))$ are sorted along the z-direction. We connect discrete points on two adjacent vertical lines according to their z-order. Each tangential intersection point is connected to two transversal intersection points on an adjacent vertical line; or it is an isolated point.

Three cases where the number of points on $L(C(t_{i+1}))$ is larger than on $L(C(t_i))$ are shown in detail in Fig. 4. In Fig. 4(a) there is a tangential intersection point (marked with a filled circle) on $L(t_{i+1})$, whereas in Fig. 4(b) there is a boundary point (marked by a square) on $L(t_{i+1})$. In these cases, a new component starts at the point, and the other remaining transversal intersection points are connected as usual. Fig. 4(c) shows the case where $L(t_i)$ contains a tangential intersection point, which may be considered as a double point and is thus connected to two transversal intersection points. In the opposite case, where the number of points on $L(t_i)$ is larger than the number on $L(t_{i+1})$, we apply a similar rule. In this case, two branches may meet at a tangential intersection point, where the corresponding loop is closed.

In the above discussion, we considered the case where the curve $C(t)$ intersects the silhouette or boundary curves of $S(u,v)$ transversally. The case of tangential intersection is
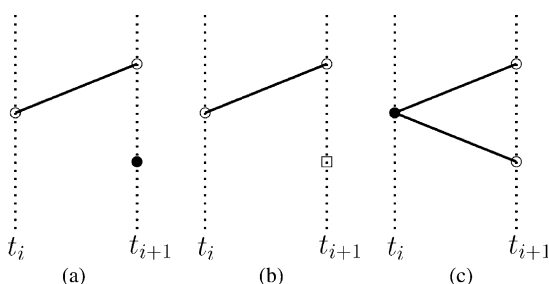

Fig. 4. Three cases where there are more points on $L(t_{i+1})$ than on $L(t_i)$.

more involved. When the curve $C(t)$ intersects the silhouette curve of $S(u,v)$ tangentially at **p**, we may consider it as a double intersection. The tangential intersection point at $L(\mathbf{p}) \cap S(u,v)$ becomes a singular point where two different loops meet and the intersection curve self-intersects. In the case where the curve $C(t)$ touches the silhouette curve of $S(u,v)$ externally, the tangential intersection point may be an isolated intersection point.

Once the topological structure has been determined, the intersection curve itself can be constructed by numerically tracing the intersection curve using a conventional technique [1,2,5].

In some degenerate cases, the curve $C(t)$ may also intersect the silhouette curves of $S(u,v)$ along some curve segments $C(t)$, ($t_a \le t \le t_b$), not just at discrete points. The surface–surface intersection then includes some tangential intersection curves, or even some cylindrical surface patches. Such degenerate singular intersections are extremely difficult to deal with using conventional techniques. Nevertheless, in our approach, the problem is reduced to a simpler problem of extracting the silhouette curve segments or the silhouette surface patches of $S(u,v)$ that project on to the corresponding curve segments $C(t)$, ($t_a \le t \le t_b$). Let $N(u,v) = (n_x(u,v), n_y(u,v), n_z(u,v))$ denote a normal vector field of the surface $S(u,v)$. We can construct the silhouette curve or the silhouette surface patch of $S(u,v)$ along the z-direction by solving: $n_z(u,v) = 0$. When the function $n_z(u,v)$ vanishes over certain open regions in the uv-plane, the surface $S(u,v)$ will intersect the cylindrical surface $R(s,t)$ in some surface patches, not just in 1-manifold curves.

2.2. Intersection with a ruled surface

Now we consider the more general problem of intersecting a freeform surface $S(u,v)$ with a ruled surface $R(s,t)$. A ruled surface is defined by connecting two space curves $C_1(t)$ and $C_2(t)$ by a line:

$$R(s,t) = C_1(t) + s(C_2(t) - C_1(t)).$$

The ruling direction is no longer fixed, but is a function of t, namely $C_2(t) - C_1(t)$. Nevertheless, we can apply a similar argument to the one used in the previous section to classify the critical points on an intersection curve. So, we need to detect the values of t that correspond to the ruling lines (in $R(s,t)$) that intersect $S(u,v)$ tangentially or at its boundary curves $S(u_0, v)$ or $S(u_0, v)$. Fig. 5 shows a loop in the intersection curve. The loop is delimited by two tangential intersections of the moving line with the surface $S(u,v)$.

When a ruling line is tangent to the surface $S(u,v)$, the surface point $S(u,v)$ is on the line. Moreover, the ruling line is contained in the tangent plane of $S(u,v)$ and thus it is orthogonal to the normal vector $N(u,v)$. (See Fig. 6(a) for a configuration where a ruling line touches the surface $S(u,v)$.) From these conditions, we obtain the following system of three constraint equations:
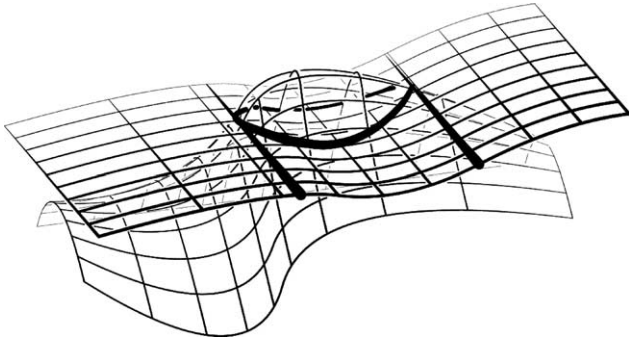
Fig. 5. An intersection curve. Two ruling lines intersect tangentially with a freeform surface and delimit a loop on the intersection curve.

$$f_1(u,v,t) = \langle S(u,v) - C_1(t), N_1(t) \rangle = 0, \tag{1}$$

$$g_1(u,v,t) = \langle S(u,v) - C_1(t), N_2(t) \rangle = 0, \tag{2}$$

$$h_1(u,v,t) = \langle C_2(t) - C_1(t), N(u,v) \rangle = 0, \tag{3}$$

where $N_1(t)$ and $N_2(t)$ are two non-parallel vectors which are orthogonal to the ruling direction $C_2(t) - C_1(t)$. Hughes and Möller [9] proposed an elegant method for the construction of such vectors $N_1(t)$ and $N_2(t)$, which have degree no higher than that of $C_2(t) - C_1(t)$. When the ruling direction $C_2(t) - C_1(t) = (d_x(t), d_y(t), d_z(t))$ has its first component as the one with largest magnitude: $|d_x(t)| \geq |d_y(t)|, |d_z(t)|$, we can take $N_1(t) = (d_y(t), -d_x(t), 0)$ and $N_2(t) = (d_z(t), 0, -d_x(t))$. Other cases can be handled in a similar way.

The intersection between a boundary curve $S(u,v_0)$ and a ruling line can be computed by solving the following system of two equations:

$$f_2(u,t) = \langle S(u,v_0) - C_1(t), N_1(t) \rangle = 0,$$

$$g_2(u,t) = \langle S(u,v_0) - C_1(t), N_2(t) \rangle = 0.$$

The intersection between $S(u_0, v)$ and a ruling line can be computed in a similar way.

After all critical points have been detected by solving the above systems of polynomial equations, their t-values are sorted in ascending order and given even indices: $t_0 < t_2 < \cdots < t_{2m}$. Now take $t_{2k-1} = (t_{2k-2} + t_{2k})/2$. For each t_i, $(0 \leq i \leq 2m)$, we intersect the freeform surface

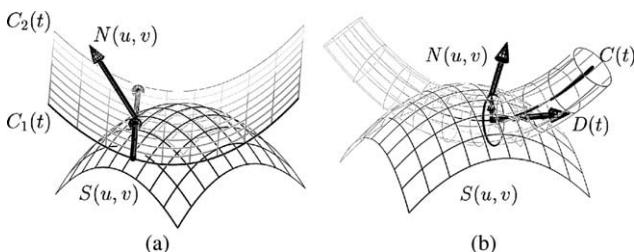

Fig. 6. A line (a) or a circle (b) touches the surface $S(u,v)$.

$S(u,v)$ with a ruling line

$$R(s,t_i) : C_1(t_i) + s(C_2(t_i) - C_1(t_i)).$$

Let $R(s_j,t_i)$, $(1 \leq j \leq n_i)$, denote the intersection points sorted along the ruling's s-direction. We can now apply the topology construction scheme previously illustrated in Fig. 3; in this case the s-axis is used instead of the z-axis. Once the topology has been determined, each segment of the intersection curve is again generated by numerical curve tracing along the intersection between two surfaces [1,2,5].

In some degenerate cases, the ruled surface may intersect the surface $S(u,v)$ tangentially along curve segments or even on surface patches. Eqs. (1)–(3) will then produce a set of 1-manifold curve segments or 2-manifold surface patches in the uvt-space as their solution set. We can distinguish the two cases by checking the dimensionality of the solution set. Whenever $S(u,v)$ and $R(s,t)$ intersect tangentially, the implicit surfaces $f_1(u,v,t)=0$ and $g_1(u,v,t)=0$ also intersect tangentially on their common solution set since in this case the equation $f_1(u,v,t) - g_1(u,v,t) = 0$ has multiple roots. Moreover, the implicit surface $h_1(u,v,t)=0$ becomes singular on the common zero-set since the tangential condition is almost satisfied in the region close to tangential intersections. When the zero-set includes 2-manifold surface patches, the surfaces $S(u,v)$ and $R(s,t)$ also intersect tangentially over surface patches. This observation also applies to other types of surface–surface intersections to be discussed in later sections.

2.3. Intersection with a ringed surface

A ringed surface is a one-parameter family of circles, where the generator circle moves in space while continuously changing its position, size and orientation. Given a freeform surface $S(u,v)$ and a ringed surface $R(s,t) = \cup O^t$, their intersection curve can be computed by characterizing the intersection points between each circle O^t and the surface $S(u,v)$.

Let $C(t)$ denote the center of the circle O^t; and assume that the circle has radius $r(t)$ and is contained in a plane with normal $D(t)$. When the circle O^t intersects the surface $S(u,v)$, the surface point $S(u,v)$ is contained in the circle O^t:

$$f_3(u,v,t) = \langle S(u,v) - C(t), D(t) \rangle = 0, \tag{4}$$

$$g_3(u,v,t) = \|S(u,v) - C(t)\|^2 - r^2(t) = 0. \tag{5}$$

Eq. (4) implies that $S(u,v)$ is contained in the plane of the circle O^t; and Eq. (5) means that the circle O^t has radius $r(t)$ Moreover, when the circle O^t intersects the surface $S(u,v)$ tangentially, the three vectors $S(u,v) - C(t)$, $D(t)$ and $N(u,v)$ become coplanar. (See Fig. 6(b) for a configuration where a circle touches the surface $S(u,v)$.) Consequently, we have

$$h_3(u,v,t) = \langle (S(u,v) - C(t)) \times D(t), N(u,v) \rangle = 0. \tag{6}$$

The intersection between a boundary curve $S(u,v_0)$ and a circle O^t can be computed by solving the following system of two equations:

$$f_4(u,t) = \langle S(u,v_0) - C(t), D(t) \rangle = 0,$$

$$g_4(u,t) = \|S(u,v_0) - C(t)\|^2 - r^2(t) = 0.$$

The case of $S(u_0,v) \cap O^t$ can be computed similarly.

The intersection curve itself is constructed by a numerical curve tracing. However, in the case of intersection with a ringed surface, the intersection points along the s-axis repeat themselves with a period of 2π. Thus, matching between two adjacent t_i-circles becomes slightly more complicated. Given a set of discrete intersection points $R(s_j,t_i)$, $(0 \leq i \leq 2m; 1 \leq j \leq n_i)$, on the ringed surface, we start a numerical curve tracing from a point $R(s_j,t_i)$, with an even index i, and continue the tracing until we reach an adjacent circle $R(s,t_{i\pm1})$, where we find a mate $R(s_j,t_{i\pm1})$ from $n_{i\pm1}$ possible candidates of transversal intersection points. Once this connection has been made, the matching becomes straightforward for other intersection points on the two adjacent circles $R(s,t_i)$ and $R(s,t_{i\pm1})$.

2.4. Intersection with a swept surface

Now we consider the intersection of a freeform surface with a swept surface. A swept surface may be generated by sweeping a rational parametric curve in space or by sweeping an implicit algebraic curve. In the previous two cases of intersection with a ruled surface or a ringed surface, we represented the moving line or the moving circle as the intersection of two planes or as the intersection of a plane and a sphere. The same approach can be applied to a general cross-sectional curve that is defined as the intersection of two implicit surfaces. The curve can move under an affine spline motion [10] by making the two implicit surfaces move under the same motion. The cross-sectional curves may also be defined as the result of a rational affine spline motion applied to a template rational spline curve $K(s)$. In this case, the curve parameter s also appears in the characterizing equations for tangential intersections.

2.4.1. Sweeping an implicit algebraic curve

We consider a general swept surface where each cross-sectional curve is defined as an intersection of two time-dependent implicit algebraic surfaces: $F^t(x,y,z)=0$ and $G^t(x,y,z)=0$. The intersection curve between a rational freeform surface $S(u,v)$ and the swept surface can be computed by solving

$$f_5(u,v,t) = F^t(S(u,v)) = 0, \tag{7}$$

$$g_5(u,v,t) = G^t(S(u,v)) = 0. \tag{8}$$

Moreover, when the cross-sectional curve intersects the surface $S(u,v)$ tangentially, we have

$$h_5(u,v,t) = \langle \nabla F^t(S(u,v)) \times \nabla G^t(S(u,v)), N(u,v) \rangle = 0, \tag{9}$$

where ∇F^t is the gradient of F^t and ∇G^t is the gradient of G^t.

When two algebraic surfaces $F(x,y,z)=0$ and $G(x,y,z)=0$ are under the same affine spline motion, the two time-dependent algebraic surfaces are given as follows

$$F^t(x,y,z) = F([(x,y,z) - T(t)]L^{-1}(t)) = 0,$$

$$G^t(x,y,z) = G([(x,y,z) - T(t)]L^{-1}(t)) = 0,$$

where $T(t)$ represents a rational translational motion and $L(t)$ is a non-singular 3×3 matrix with rational spline functions as its entries. Note that the inverse matrix $L^{-1}(t)$ also has rational spline functions as its entries (as can be seen from Cramer's rule). Fig. 7 shows an example of a swept surface, where each cross-sectional curve is defined as the intersection of two quadrics.

The intersection between a boundary curve $S(u,v_0)$ and the swept surface can be computed by solving

$$f_6(u,t) = F^t(S(u,v_0)) = 0, \quad g_6(u,t) = G^t(S(u,v_0)) = 0.$$

The case of intersecting $S(u_0,v)$ and the swept surface can be handled in a similar way.

2.4.2. Sweeping a rational parametric curve

Let $K(s) = (x(s), y(s), z(s))$ denote a rational spline curve. Now we apply a time-dependent non-singular linear transformation $L(t)$ to $K(s)$ and then a translation $T(t) = (\alpha(t), \beta(t), \gamma(t))$ to the result $K(s)L(t)$, where $L(t)$ is a 3×3 matrix with each entry as a rational spline function of t, and $\alpha(t)$, $\beta(t)$, and $\gamma(t)$ are also rational functions of t. Then a rational swept surface $R(s,t)$ is defined as $R(s,t) = \cup[K(s)L(t) + T(t)]$. The intersection between $R(s,t)$ and a freeform surface $S(u,v)$ is characterized by the following vector equation:

$$\mathbf{f}_7(u,v,s,t) = S(u,v) - K(s)L(t) - T(t) = \mathbf{0}. \tag{10}$$

At a tangential intersection point, the tangent vector $K'(s)L(t)$ of a cross-sectional curve $K(s)L(t) + T(t)$ should be orthogonal to the normal vector $N(u,v)$ of the freeform
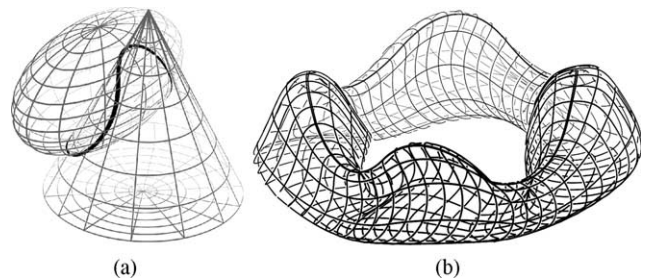

Fig. 7. (a) A cross-sectional curve is represented as an intersection of two quadrics; and (b) a swept surface is generated by sweeping the intersection of two quadrics.
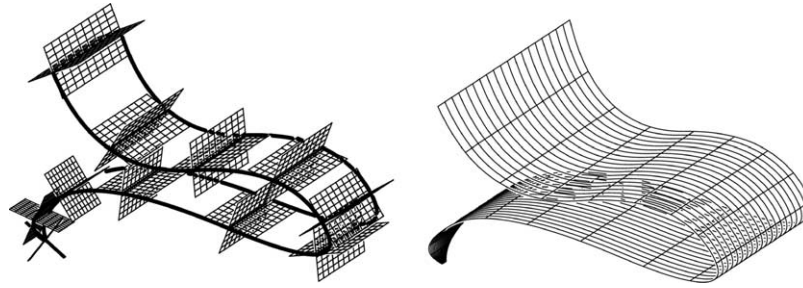
Fig. 8. A ruled surface as a one-parameter family of lines, where each line is defined as the intersection of two non-parallel planes.

surface $S(u,v)$:

$$h_7(u, v, s, t) = \langle K'(s)L(t), N(u, v) \rangle = 0. \tag{11}$$

The intersection between a boundary curve $S(u,v_0)$ and the swept surface $R(s,t)$ can be computed by solving

$$\mathbf{f}_8(u, s, t) = S(u, v_0) - K(s)L(t) - T(t) = \mathbf{0}.$$

The case of intersecting $S(u_0, v)$ and $R(s,t)$ can be handled in a similar way.

3. Reduction to parameter space

We now go on to present an alternative approach that reduces the intersection problem to that of computing the zero-set of $(n-1)$ polynomial equations in n variables. When the cross-sectional curves are defined as the intersection of two implicit surfaces, the problem is formulated with $n = 3$, i.e. we need to solve two equations in three variables. On the other hand, when the cross-sectional curves are rational parametric curves, the problem is formulated with $n = 4$. In either case, the result is a 1-manifold in the parameter space. (In some degenerate cases, we may also produce some 2-manifold surface patches in the solution set.) By projecting the 1-manifold on to the uv-parameter plane of $S(u,v)$, the intersection curve can be constructed.

In the case of intersection with a ruled surface, we may represent each ruling line

$$R(s, t) = C_1(t) + s(C_2(t) - C_1(t))$$

as the intersection of two non-parallel planes with normal vectors $N_1(t)$ and $N_2(t)$ (Fig. 8). The intersection between

a freeform surface $S(u,v)$ and a ruling line is characterized as follows:

$$f_1(u, v, t) = \langle S(u, v) - C_1(t), N_1(t) \rangle = 0,$$

$$g_1(u, v, t) = \langle S(u, v) - C_1(t), N_2(t) \rangle = 0,$$

which are two polynomial equations in three variables u, v, t.

Fig. 9 shows a ringed surface as a one-parameter family of circles, where each circle is defined as the intersection between a sphere and a plane. The intersection condition between a freeform surface $S(u,v)$ and a circle O^t is given as follows:

$$f_3(u, v, t) = \langle S(u, v) - C(t), D(t) \rangle = 0,$$

$$g_3(u, v, t) = \|(u, v) - C(t)\|^2 - r^2(t) = 0,$$

which are two polynomial equations in three variables u, v, t.

We can represent a swept surface as a one-parameter family of cross-sectional curves. For a swept surface generated by a one-parameter family of implicit algebraic curves, each defined by two algebraic surfaces, the intersection curve is constructed by solving

$$f_5(u, v, t) = F^t(S(u, v)) = 0, \quad g_5(u, v, t) = G^t(S(u, v)) = 0,$$

which are two polynomial equations in three variables u, v, t. When a swept surface is generated by a rational parametric curve $K(s)$ under a rational affine spline motion, the intersection curve can be constructed by solving

$$\mathbf{f}_7(u, v, s, t) = S(u, v) - K(s)L(t) - T(t) = \mathbf{0},$$

which is a vector equation and represents three polynomial equations in four variables u, v, s, t.
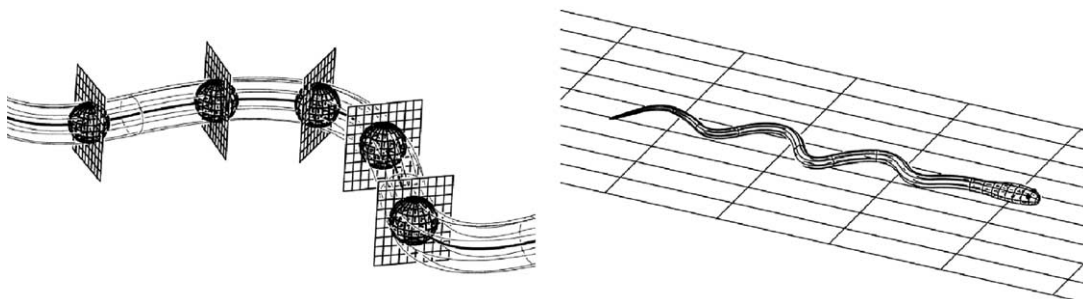

Fig. 9. A ringed surface as a one-parameter family of circles, where each circle is defined as the intersection of a sphere and a plane.

Now we have reduced the intersection problem to that of computing the zero-set of $(n-1)$ polynomial equations in n variables. The result is a 1-manifold. By projecting this zero-set on to the uv-plane, the intersection is generated as a curve embedded in the freeform surface $S(u, v)$.

4. Discussion and experimental results

The polynomial equations for constructing topology have one more equation (for the tangential condition) than those for the problem reduction approach. Moreover, we have to solve n polynomial equations for topology construction, but one less equation for the reduction approach. However, the solutions of n equations are discrete

points in general, whereas the zero-sets of $(n-1)$ equations in the reduction approach are 1-manifolds. It is computationally more efficient to compute discrete solutions than to construct 1-manifolds. Therefore, it is worthwhile to solve the n polynomial equations in n variables needed for topology construction.

Topology construction requires a second stage of numerical tracing that generates segments of the intersection curve. This can be achieved using conventional techniques for numerical tracing along intersection curves. Alternatively, we can solve $(n-1)$ polynomial equations in n variables, which have been formulated for the problem reduction approach. These equations define $(n-1)$ implicit surfaces in an n-dimensional space, which is yet another surface–surface intersection problem when $n=3$. The two
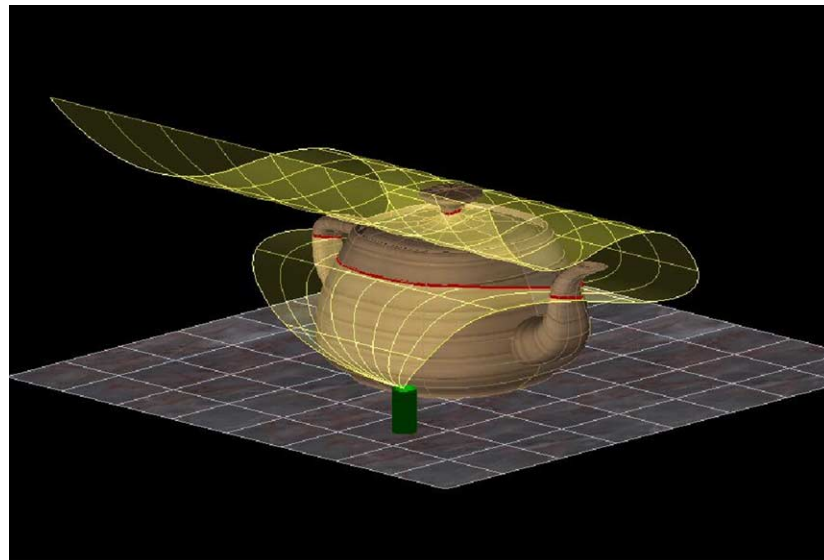
(a)

(b)

Fig. 10. Intersection between the Utah teapot and (a) a ruled surface or (b) a ringed surface.
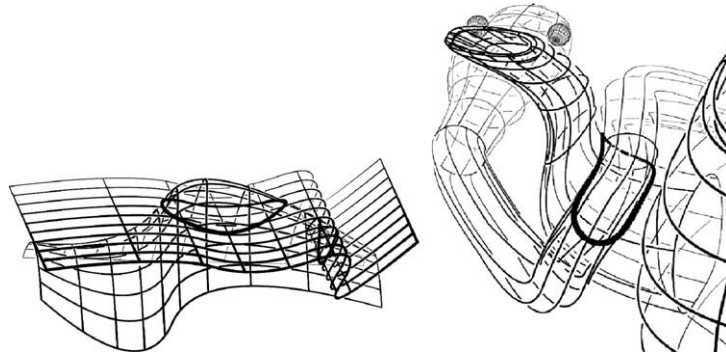
Fig. 11. Intersection based on the problem reduction scheme.

approaches proposed in this paper are thus closely related to each other. The user may choose either one of them or combine both depending on applications and tools available.

Fig. 10(a) shows the result of intersecting the Utah teapot with a ruled surface using the topology construction scheme. Fig. 10(b) shows the same teapot intersected with a ringed surface. Results based on the simple reduction scheme are shown in Fig. 11. A result for intersection with a general swept surface is shown in Fig. 12, where the intersection curves are shown in bold lines. The computation time for these results are about the same, all within one or two seconds on 2GHz Pentium IV machine.

Fig. 13(a) shows a degenerate case where a cylinder and a torus intersect tangentially along a circle. The cylinder is given as a freeform surface $S(u, v)$ and the torus is represented as a ringed surface. Fig. 13(b) shows the zero-sets of $f_3(u,v,t)=0$, $g_3(u,v,t)=0$, and $h_3(u,v,t)=0$, in red, magenta and yellow, respectively. The common zero-set of these three equations includes two discrete

points (in green) and one line (in cyan). Along the cyan line, the two surfaces, $f_3(u,v,t)=0$, and $g_3(u,v,t)=0$, meet tangentially, and the other surface $h_3(u,v,t)=0$, is singular. On the other hand, at two green isolated intersection points, the three surfaces intersect transvesally. Fig. 14(a) shows another example of degenerate intersection where two surfaces intersect tangentially on a surface patch. In Fig. 14(b), the two surfaces of $f_3(u,v,t)=0$ and $g_3(u,v,t)=0$ also meet tangentially on a surface patch (in cyan). Since the surface $h_3(u,v,t)=0$ is singular over this area, the result of applying a marching cube algorithm generates a large volumetric solution set for $-\varepsilon < h_3(u, v, t) < \varepsilon$, even for a small value of $\varepsilon > 0$. Since $h_3(u,v,t)=0$ is singular around this region, it is numerically unstable to compute its zero-set or to evaluate its normal direction. Thus we have constructed the common zero-set of $f_3(u, v, t) = g_3(u, v, t) = 0$ on a tangential surface patch by checking whether their gradients $\nabla f_3(u,v,t)$ and $\nabla g_3(u,v,t)$ are also collinear.
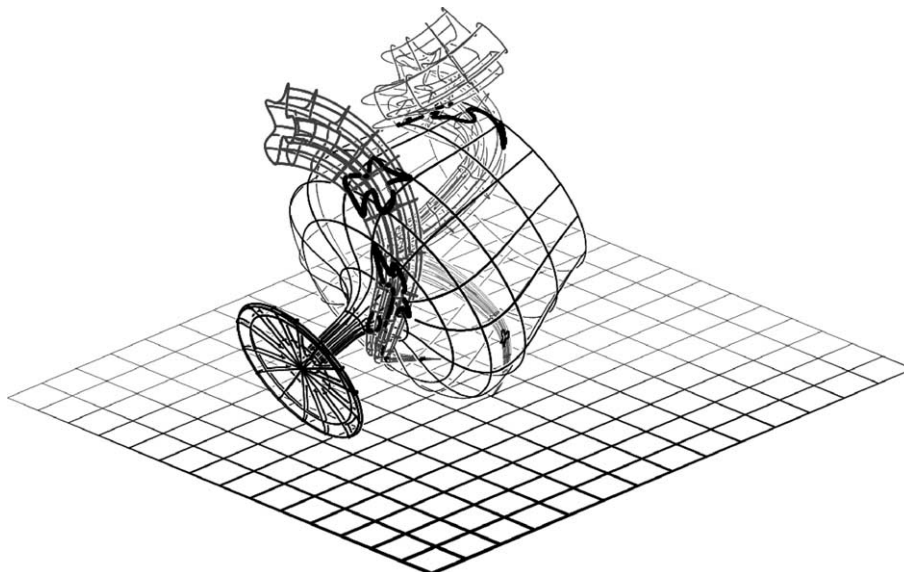

Fig. 12. Intersection between a freeform surface and a swept surface.
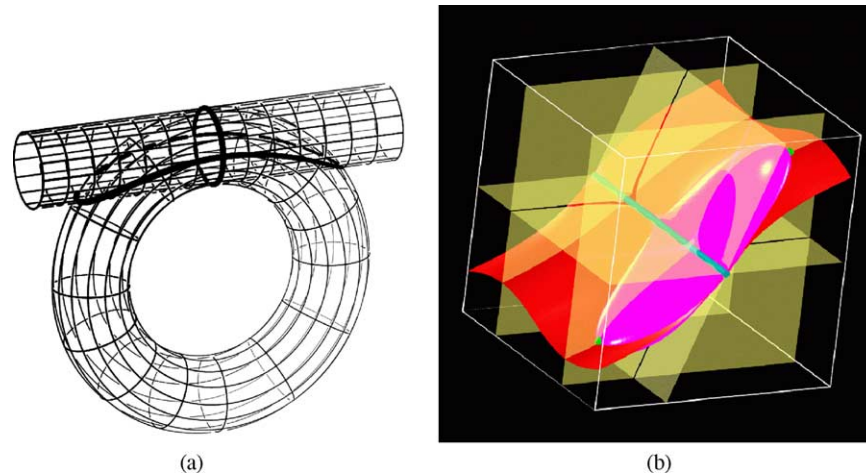
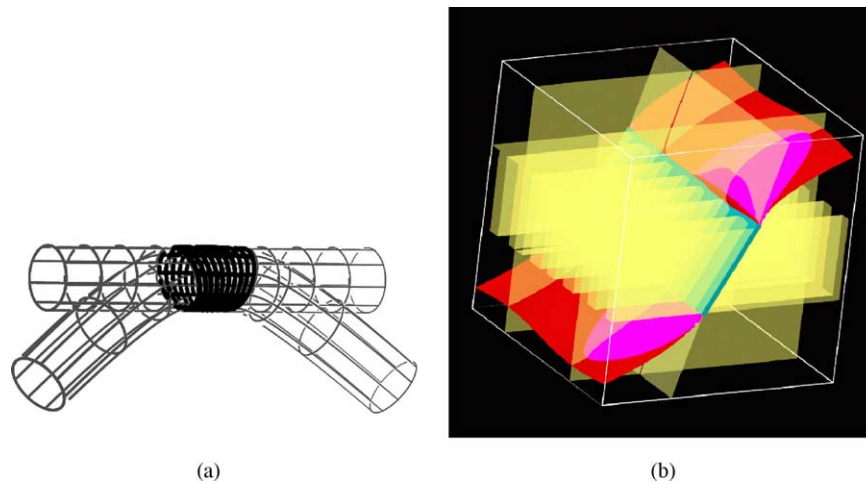Fig. 13. Degenerate tangential intersection along a curve.

Fig. 14. Degenerate tangential intersection on a surface patch.

5. Conclusion

We have introduced two approaches to the problem of intersecting a freeform surface with a swept surface. Our first scheme was based on detecting critical points and constructing the intersection curve with a correct topology based on the critical points. Our second scheme reduces the surface–surface intersection problem to a zero-set finding in the parameter space. These two schemes are closely related to each other. Working in the parameter space, the proposed algorithm can also deal with degenerate singular intersections such as tangential intersection along a curve or overlap on a surface patch.

Acknowledgements

The authors would like to thank the anonymous reviewers for their useful comments. All the algorithms

References

[1] Bajaj CL, Hoffmann CM, Lynch RE, Hopcroft JEH. Tracing surface intersections. Comput Aided Geom Des 1988;5:285–307.
[2] Bajaj CL, Xu G. NURBS approximation of surface–surface intersection curves. Adv Comput Math 1994;2(1):1–21.

[3] de Pont J. Essays on the cyclide patch. PhD Thesis. Cambridge: Cambridge University Engineering Department; 1984.

[4] Elber G. IRIT 9.0 User's Manual. The Technion-IIT, Haifa, Israel; 1997. Available at http://www.cs.technion.ac.il/irit

[5] Elber G, Choi J-J, Kim M-S. Ruled tracing. Vis Comput 1997;13(2): 78–94.

[6] Elber G, Kim M-S. Geometric constraint solver using multivariate rational spline functions. Proceedings of ACM symposium on solid modeling and applications, Ann Arbor, MI, June 4–8; 2001.

[7] Heo H-S, Hong SJ, Seong J-K, Kim M-S, Elber G. The intersection of two ringed surfaces and some related problems. Graph Models 2001; 63:228–44.

[8] Heo H-S, Kim M-S, Elber G. Ruled/ruled surface intersection. Comput Aided Des 1999;31(1):33–50.

[9] Hughes J, Möller T. Building an orthonormal basis from a unit vector. J Graphics Tools 1999;4(4):33–5.

[10] Hyun D-E, Jüttler B, Kim M-S. Minimizing the distortion of affine spline motion. Graph Models 2002;64(2):128–44.

[11] Johnstone J. A new intersection algorithm for cyclides and swept surfaces using circle decomposition. Comput Aided Geom Des 1993; 10(1):1–24.

[12] Kim K-J, Kim M-S, Martin R. Intersecting a translationally or rotationally swept surface with a plane or a sphere. Proceedings of Korea Israel Bi-National Conference on Geometrical Modeling and Computer Graphics in the World Wide Web Era; 1999. p. 165–93.

[13] Kim M-S. Intersecting surfaces of special types. Proceedings of shape modeling and processing 99, March 1–4. Japan: University of Aizu; 1999. p. 122–8.

[14] Martin R, de Pont J, Sharrock T. Cyclide surfaces in computer aided design. Math Surf 1986;14(4):349–75.

[15] Patrikalakis N, Maekawa T. Shape interrogation for computer aided design and manufacturing. Berlin: Springer; 2002.

Myung-Soo Kim is a Professor in the School of Computer Science and Engineering and the Director of the Institute of Computer Technology, Seoul National University. His research interests are in computer graphics and geometric modeling. Prof. Kim received BS and MS degrees from Seoul National University in 1980 and 1982, respectively. He continued his graduate study at Purdue University, where he received an MS degree in applied mathematics in 1985 and MS and PhD degrees in computer science in 1987 and 1988, respectively. Since then until 1998, he was with the Department of Computer Science, POSTECH, Korea. Prof. Kim serves on the editorial boards of Computer-Aided Design, Computer Aided Geometric Design, Computer Graphics Forum, Computer Graphics and Geometry, and Int J of Shape Modeling. He also edited several special issues of journals such as Computer-Aided Design, Graphical Models, J of Visualization and Computer Animation, The Visual Computer, and Int J of Shape Modeling. Recently, with two other editors, G Farin and J Hoschek, he edited Handbook of Computer Aided Geometric Design, North-Holland, 2002.

Gershon Elber is an associate professor in the Computer Science Department, Technion, Israel. His research interests span computer aided geometric designs and computer graphics. Prof. Elber received a BS in computer engineering and an MS in computer science from the Technion, Israel in 1986 and 1987, respectively, and a PhD in computer science from the University of Utah, USA, in 1992. He is a member of the ACM and IEEE. Prof. Elber serves on the editorial board of the Computer Aided Design, Computer Graphics Forum, and the International Journal of Computational Geometry and Applications and has served in many conference program committees including Solid Modeling, Pacific Graphics, Computer Graphics International, and Siggraph. Prof. Elber was one of the paper chairs of Solid Modeling 2003 and Solid Modeling 2004. Elber can be reached at the Technion, Israel Institute of Technology, Department of Computer Science, Haifa 32000, ISRAEL.

Joon-Kyung Seong is a PhD student in the School of Computer Science and Engineering, Seoul National University. His research interests are in computer graphics and geometric modeling/processing. Mr. Seong received BS and MS degrees from Seoul National University in 2000 and 2002, respectively.

Ku-Jin Kim is an Assistant Professor in the Department of Computer Engineering at Kyungpook National University, Korea. Her research interests include computer graphics, non-photorealistic rendering, and geometric/surface modeling. Prof. Kim received a BS degree from Ewha Womans University in 1990, an MS degree from KAIST in 1992, and a PhD degree from POSTECH in 1998, all in Computer Science. She was a PostDoctoral fellow at Purdue University in 1998–2000. Prof. Kim also held faculty positions at Ajou University, Korea, and at University of Missouri, St Louis, USA.

Ralph Martin has been working in the field of CADCAM since 1979. He obtained his PhD in 1983 from Cambridge University for a dissertation on 'Principal Patches', and since then has held posts from Lecturer to Professor at Cardiff University. He has had published over 150 papers and 9 books covering such topics as solid modelling, surface modelling, intelligent sketch input, vision based geometric inspection, geometric reasoning and reverse engineering. He is a Fellow of the Institute of Mathematics and its Applications, and a Member of the British Computer Society. He is on the editorial boards of 'Computer Aided Design' and the 'International Journal of Shape Modelling', and has been active in the organisation of many conferences.