GRAPHICAL MODELS AND IMAGE PROCESSING
Vol. 57, No. 1, January, pp. 55–66, 1995

Geometric Algorithms for Detecting and Calculating All Conic Sections in the Intersection of Any Two Natural Quadric Surfaces

JAMES R. MILLER
University of Kansas, Lawrence, Kansas 66045-2192

AND

RONALD N. GOLDMAN
Rice University, Houston, Texas 77001

Received March 17, 1994; revised September 19, 1994; accepted September 26, 1994

One of the most challenging aspects of the surface–surface intersection problem is the proper disposition of degenerate configurations. Even in the domain of quadric surfaces, this problem has proven to be quite difficult. The topology of the intersection as well as the basic geometric representation of the curve itself is often at stake. By Bezout's Theorem, two quadric surfaces always intersect in a degree four curve in complex projective space. This degree four curve is degenerate if it splits into two (possibly degenerate) conic sections. In theory the presence of such degeneracies can be detected using classical algebraic geometry. Unfortunately in practice it has proven to be extremely difficult to make computer implementations of such methods reliable numerically. Here, we present geometric algorithms that detect the presence of these degeneracies and compute the resulting planar intersections. The theoretical basis of these algorithms—in particular, proofs of correctness and completeness—are extremely long and tedious. We briefly outline the approach, but present only the results of the analysis as embodied in the geometric algorithms. Interested readers are referred to R. N. Goldman and J. R. Miller (Detecting and calculating conic sections in the intersection of two natural quadric surfaces, part I: Theoretical analysis; and Detecting and calculating conic sections in the intersection of two natural quadric surfaces, part II: Geometric constructions for detection and calculation, Technical Reports TR-93-1 and TR-93-2, Department of Computer Science, University of Kansas, January 1993) for details. © 1995 Academic Press, Inc.

1.0. INTRODUCTION

The surface–surface intersection problem in geometric modeling continues to be a challenging one for system developers. Especially problematic are those situations where the curve of intersection between two surfaces is in some fashion degenerate. This may occur when the two surfaces are tangent—either along a curve or at a finite number of points—or, as we will see, in a variety of other situations, not all of which are entirely intuitive. Such problems are challenging even when the domain of surfaces is restricted to the quadrics; witness the considerable attention this subject continues to draw [2, 11, 17, 20, 21], even though most of the basic results have been known in computer-based implementations for quite some time [3, 10, 19].

The importance of detecting the presence of conic sections in quadric surface intersections is well established [2, 11, 15, 19, 20, 21]. Often cited advantages include more efficient and precise data base representations, more reliable tests for common curves in the boundary evaluation algorithm of solid modeling, and more accurate analytical computations of geometric properties such as intersections, arc length, and tangent direction. Another benefit has recently been discovered in the construction of blending surfaces. In general, an algebraic blend surface between two arbitrary quadric surfaces will have degree four. Warren [22, 23] has shown that quadric surfaces that intersect in planar curves can be blended with surfaces of degree three or less. In addition, two cones can be blended by a Dupin cyclide if and only if they have a planar intersection [18].

The majority of the published results dealing with quadric surface intersections are based either on the methods of classical algebraic geometry [2, 10, 19] or on case-by-case geometric analysis [11, 17]. Algebraic methods are valuable for ensuring completeness and generality. That is, one can prove rigorously that all possible geometric configurations have been properly taken into account. It has, unfortunately, proven to be quite difficult to make algorithms based solely on algebraic geometry sufficiently reliable numerically for production use in solid modeling systems. Geometric methods, on the other hand, have proven to be faster and more reliable numerically, but they are based on a case-by-case analy-

1077-3169/95 $6.00

sis in which it is much more difficult, if not impossible, to prove that all the cases are enumerated and handled properly.

Our approach is to tap the best of both worlds by limiting the scope of surfaces covered to those most commonly used (namely the natural quadrics [7]), applying algebraic geometry to characterize with certainty all situations under which a pair of natural quadrics have planar intersections, and finally reinterpreting the algebraic conditions in geometrically invariant terms so that robust geometric algorithms can be implemented to detect and calculate the resulting planar intersections. The complete analysis is extremely long and tedious. Here we only sketch the approach, summarize the results in tables, and illustrate some of the more interesting cases by describing the algorithms derived from the analysis. Interested readers can study the complete details in [6] and [14].

Many of the results summarized here are not new. For example, it is well known that axial quadrics have a planar intersection if they have a common inscribed sphere. Moreover, there are certainly alternative strategies one could employ to derive some of these results. For example, simple geometric proofs can be used to establish some of the cases for which our derivations are more complex. What is unique and important about our approach is that we have a *single* method that is guaranteed to identify *all possible situations* where degenerate intersections arise. By comparison, the common inscribed sphere condition is a sufficient but not a necessary condition. Similarly, other geometric arguments can establish additional sets of sufficient conditions in which planar intersections appear. With the possible exception of Shene and Johnstone [20, 21], none of these other schemes guarantee the identification of conditions that are both sufficient *and necessary*. Therefore the point is not that our methods reproduce some well-known results, but rather than they are guaranteed to reveal all possible configurations where degenerate intersections appear. This analysis is of value to system developers since they can use this list of conditions and be confident that degeneracies will arise in these and only these situations.

It is also important to note that the complicated portion of our analysis [6] is *not* implemented in code. It is purely a "paper analysis" that generates a short but exhaustive list of all possible relative geometric configurations where a pair of natural quadrics have a planar intersection.

We proceed in the following fashion. In Section 2 we survey previous work in the field and we provide most of the theory necessary for understanding our own approach. In Section 3 we give a brief overview of the theoretical approach used in [6] to characterize all degenerate intersections, and then we review the basic geometric descriptions of conic curves and natural quadric surfaces. We also introduce the notation and fundamental geometric tools that we will use throughout the paper. In Section 4 we begin the calculation of degenerate intersections by discussing how to find isolated tangent points. Section 5 is the bulk of this paper. Here we present the list of configurations yielding planar intersection curves and discuss some of the more interesting cases. In Section 6 we describe the conditions under which the intersection consists of a line and a space cubic. Finally, in Section 7 we summarize our results and make some concluding observations.

2.0. BACKGROUND THEORY AND DISCUSSION

A general quadric in arbitrary position is represented algebraically as

$$Ax^2 + By^2 + Cz^2 + 2Dxy + 2Eyz + 2Fxz + 2Gx + 2Hy + 2Jz + K = 0. \tag{1}$$

This equation can be written in matrix form:

$$p\mathbf{Q}p^{\mathrm{T}} = 0,$$

where

$$\mathbf{Q} = \begin{pmatrix} A & D & F & G \\ D & B & E & H \\ F & E & C & J \\ G & H & J & K \end{pmatrix}$$

$$p = (x, y, z, 1).$$

The 4×4 symmetric matrix **Q** completely characterizes a given quadric; therefore we will refer to particular quadrics in terms of their corresponding matrices. For simplicity of expression, however, we will say "given a quadric **Q** . . ." rather than the precise but more awkward "given a quadric whose 4×4 symmetric matrix is **Q**. . . ." Note that the equation of a quadric surface, and therefore the matrix **Q**, is unique only up to constant multiples.

The natural quadrics are the sphere, the right circular cylinder, and the right circular cone. The general quadric surface is either a cylinder or cone lying over a conic section, or it is an ellipsoid, paraboloid, or hyperboloid. However a quadric may be degenerate and/or consist of one or two lower degree or lower dimensional shapes in real affine space. The possibilities are:

- *a single plane:* if G, H, J, and K are the only nonzero terms in (1),
- *a pair of identical, parallel, or intersecting planes:* if

(1) can be factored into two terms, each of which is linear in x, y, and z,

- *a single line:* if, for example, the only nonzero elements of $\mathbf{Q}$ are $A = B = 1$,
- *a single point:* if, for example, the only nonzero elements of $\mathbf{Q}$ are $A = B = C = 1$.

In this paper, when we speak of "planar surfaces" we mean any of these possibilities *except* the single point [6].

Given two quadrics $\mathbf{Q}_1$ and $\mathbf{Q}_2$, we can describe parametrically a family of quadrics (called the pencil of $\mathbf{Q}_1$ and $\mathbf{Q}_2$) as

$$\mathbf{Q}(\lambda) = \mathbf{Q}_1 - \lambda\mathbf{Q}_2.$$

In order to establish the conditions under which the intersection of a pair of quadric surfaces is degenerate, we need the following two theorems. Similar statements of and proofs for these theorems are given in [6, 19]. We include the statements here for completeness.

THEOREM 1. *The intersection of two quadric surfaces is a planar curve (or a pair of planar curves) if and only if there is a planar surface in the pencil of the two quadric surfaces.*

THEOREM 2. *A quadric surface represents a planar surface if and only if the rank of the* 4×4 *symmetric matrix that represents the surface is less than or equal to* 2.

Degenerate intersections of quadric surfaces are generally represented by conic sections, but tangencies at one or two points can also occur. The proof of Theorem 1 can easily be extended to the case where the conic section is just two points and the planar surface is a straight line. This observation will be used in Section 4. The proof fails, however, when the quadrics are tangent at a single point, because in this case it is not generally possible to conclude that there is a single point surface in the pencil.

The upshot is that the analysis presented in [6] produces a characterization of all two-point tangencies, but does not provide a characterization for all one-point tangencies. This limitation on the theoretical analysis driven by Theorem 1 has not resulted in any limitations in the implementation, however. As will be described in Section 4, we use a different approach for detecting isolated tangent points that not only detects all the two-point tangencies predicted in [6], but also detects all the configurations involving single point tangencies.

Whether it is ultimately of value to detect all the situations where two quadrics intersect in one or two isolated tangent points depends on the application. In solid modeling, such tangent points may indicate the presence of nonmanifold topologies on the boundary of the solid and may therefore be of some interest.

2.1. *Previous Work*

The first reported technique for detecting conic sections in a computer-based implementation of quadric surface intersections was described by Levin [10]. This method was later extended by Sarraga [19]. As we do in the analysis described here, Sarraga searches the pencil looking for a planar surface. There are, however, fundamental differences in the two approaches. Most notably, Sarraga applies purely computer-based numerical methods to each pair of quadrics in the data base using sensitive numerical tests to detect planar intersections. On the other hand, our method is based on a symbolic analysis of the six possible combinations of natural quadric surfaces, done once and for all and by hand. The results of this symbolic analysis are a simple set of robust geometric tests which are implemented on a computer to detect planar intersections between a given pair of natural quadrics [14].

Piegl [17] presents an approach based on a geometric construction for computing quadric surface intersection curves. Given a pair of natural quadrics, each represented as a trimmed tensor product rational B-spline surface, he first numerically extracts from the B-spline description the geometric data describing the quadric (e.g., the center and radius if it is a sphere) and then analyzes this data geometrically to determine the type of intersection. No claim is made that all planar intersections involving natural quadrics are detected.

Farouki *et al.* [2] address the problem of automatically determining all degenerate intersections involving quadric surfaces of all types. The presence of degenerate intersection branches (lines, conics, cubics, or nodal or cuspidal quartics) is signaled by the vanishing of various polynomial expressions involving the quadric coefficients. When such a configuration is detected, a multivariate polynomial factorization algorithm is invoked to isolate the various reducible components of the intersection.

Ocken *et al.* [15] use algebraic representations to parameterize intersections between general pairs of quadric surfaces. Their analysis is numerical and must be performed at execution time for each pair of surfaces in the model. Furthermore, while some mention is made of configurations in which the intersection is planar, there is no comprehensive treatment of these situations.

O'Connor [16] determines parameterizations for the curve of intersection between an arbitrary pair of natural quadric surfaces using various geometric constructions and projections. As with Ocken *et al.* [15], there is no systematic attempt to detect degeneracies in order to represent the result explicitly as one or two (possibly degenerate) conic sections.

Finally, Shene and Johnstone [20, 21] describe a geometric approach for detecting planar intersections between pairs of natural quadric surfaces, and they justify their work with purely geometric reasoning. They begin by demonstrating that the intersection of two axial natural quadrics can have a conic component only if their axes are coplanar. In such situations, they develop necessary and sufficient geometric conditions for the intersection to contain a planar component. Their method is markedly different from ours. Theirs is a procedural approach for the detection of conic components, while ours is driven from an exhaustive (but surprisingly short) list of specific relative geometric configurations. The primary advantage of their procedural method is that it leads to a common geometric algorithm for axial quadrics. On the other hand, testing our conditions seems to be more computationally efficient, and our approach is not tailored to axial quadrics.

3.0. OVERVIEW OF THE METHOD, GEOMETRIC NOTATION, AND PRIMITIVE UTILITIES

We treat only the natural quadrics (i.e., sphere, right circular cylinder, and right circular cone) in our work. For simplicity of expression, we shall use the term "cylinder" for "right circular cylinder" and "cone" for "right circular cone". A complete but extremely tedious analysis leading to a characterization of all possible configurations in which a pair of natural quadrics has a planar intersection is presented in [6]. The cylinder–cone case is analyzed in [5]. In this section we merely sketch the approach employed there to give the reader some assurance that our results are correct. While the analysis is certainly applicable to the other quadrics as well, the details get increasingly complex. Fully half of [6] is devoted to analyzing the intersection between two cones. It is significant to note that characterizing the degenerate intersections is relatively easy. The majority of the analysis is devoted to showing that there are no other cases; i.e., that these conditions are necessary as well as sufficient.

For each of the six possible combinations of pairs of natural quadric surface types, we establish all possible sets of necessary and sufficient conditions so that their pencil contains a planar surface. The approach to this analysis is derived from Theorem 2. That is, we characterize all those conditions that force all 3×3 subdeterminants of the pencil matrix to be zero. These conditions can then be interpreted as sets of constraints on the relative position and orientation of the two types of surfaces which, if satisfied, indicate the presence of a planar intersection. Then, because of Theorem 1, we will know that conic sections will arise only when one of these constraints is satisfied.

Geometric representations of conics and quadrics are generally characterized by a local coordinate system with

TABLE 1
Geometric Descriptions of Lines and Conics

Curve	Description of geometric parameters	Notation
Line	(Base point, direction vector)	$(B, \mathbf{w})$
Circle	(Center, normal to the plane containing the circle, radius)	$(C, \mathbf{w}, r)$
Ellipse	(Center, major axis, minor axis, major radius, minor radius)	$(C, \mathbf{u}, \mathbf{v}, r_u, r_v)$
Parabola	(Vertex, directrix vector, focus vector, focal length)	$(V, \mathbf{u}, \mathbf{v}, f)$
Hyperbola	(Center, major axis, minor axis, major radius, minor radius)	$(C, \mathbf{u}, \mathbf{v}, r_u, r_v)$

associated scalar parameters. The local coordinate system is defined by (i) three mutually perpendicular unit vectors ($\mathbf{u}$, $\mathbf{v}$, $\mathbf{w}$) which describe the orientation of the conic or quadric and (ii) a base point O which fixes the position of the curve or surface in space. The scalar parameters determine the size of the conic or quadric. Obviously there is redundant information in the complete coordinate system, and only portions of it need be specified to determine uniquely the position and orientation of a particular conic or quadric. A set of geometric parameters which uniquely define lines and conics are summarized in Table 1 and illustrated in Fig. 1. Similarly, a set defining planes and natural quadrics are listed in Table 2 and illustrated in Fig. 2. Throughout this paper, we shall assume that the vectors associated with the geometric representations are unit vectors.

Using vector techniques it is easy to derive both parametric and implicit representations of these second-degree curves and surfaces from their geometric representations [13]. Certain primitive functions creating and manipulating scalars, points, vectors, curves, and surfaces are assumed in the sequel. Some of the main ones are briefly summarized below.

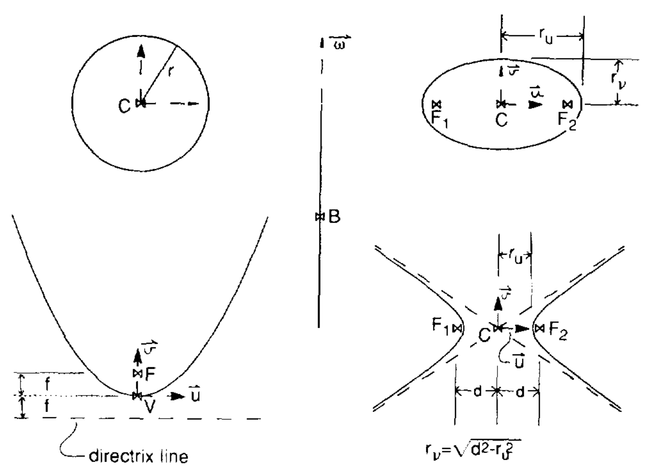

FIG. 1. Geometric definitions of conic sections.

TABLE 2
Geometric Descriptions of Planes and Natural Quadric Surfaces

Surface	Description of geometric parameters	Notation
Plane	(Base point, normal vector)	$(B, \mathbf{w})$
Sphere	(Center, radius)	(C, r)
Cylinder	(Base point, axis vector, radius)	$(B, \mathbf{w}, r)$
Cone	(Vertex, axis vector, half-angle)	$(V, \mathbf{w}, \alpha)$

The function line $(Q, \mathbf{v})$ returns a line whose base point and unit direction vector are as specified. Similarly, the function plane $(Q, \mathbf{n})$ returns a plane whose base point and unit normal vector are given by the parameters. The function normalize $(\mathbf{v})$ returns a unit vector whose direction is the same as that of $\mathbf{v}$.

The function "distance" returns the distance between its two parameters which may be any combination of points, lines, and planes. The function signed_distance_along_line (Q, L) assumes that the point Q is on the line L and calculates the signed distance from the base point of L to Q as

$$(Q - L.B) \cdot L.\mathbf{w}.$$

Based on these functions, we assume that we can test reliably (i.e., to within some reasonable tolerance) whether a point is on a plane or line, or whether two points are identical.

When intersecting cylinders and cones, we will sometimes need to intersect their axis lines. General algorithms for intersecting two 3D lines can be found in a variety of sources (e.g., [4, 12]). It is also necessary to test whether the axis lines are skew. The method described in [4] actually generates the parameter values on each of the two lines for the points of closest approach. If these two points are identical, then they describe the point at which the lines intersect; if they are different, the lines are skew.

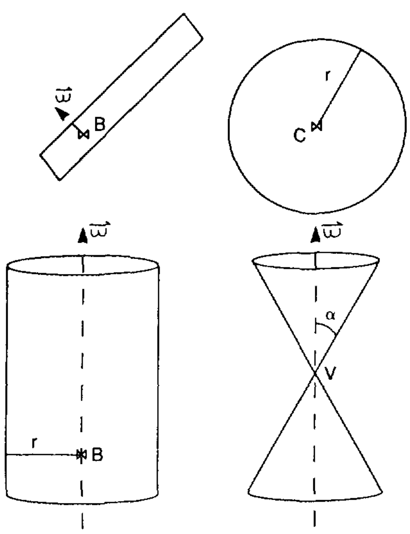

FIG. 2. Geometric definitions of the natural quadric surfaces.

Given a triangle with sides of length $d1$, $d2$, and $d3$, we often use the Law of Cosines to compute the cosine of the angle between the sides of length $d1$ and $d2$ (see Fig. 3). It is common to then compute the projection h of $d1$ on $d2$ by $h = d1{*}\cos\beta$ (Fig. 3). For optimal numerical reliability, we advocate instead the use of the following auxiliary procedure.

```
procedure LawOfCosines (input d1, d2, d3: real;
output cosβ, h: real);
begin
  h := (d1*d1 + d2*d2 − d3*d3) / (2*d2);
  cosβ := h / d1
end;
```

Finally we will need tests to determine if a point lies on a cylinder or a cone. To perform these tests, we can use the geometric form of the implicit surface equations. A point Q lies on a cylinder C if and only if [11]

$$(Q - C.B) \cdot (Q - C.B) - ((Q - C.B) \cdot C \cdot \mathbf{w})^2 - C.r^2 = 0.$$

Similarly, a point Q lies on a cone C if and only if [11]

$$((Q - C.V) \cdot C.\mathbf{w})^2 - \cos^2\alpha(Q - C.V) \cdot (Q - C.V) = 0.$$

4.0. ISOLATED TANGENT POINTS

We show in [6] that two point tangencies can arise between pairs of natural quadrics only under the situations enumerated in Table 3. In our implementation, we do not test explicitly for these conditions, and, if they are satisfied, invoke individual geometric constructions to compute the pairs of tangent points. Instead we have a single common algorithm, briefly outlined below, that finds all isolated tangent points, whether they are single, double, or occur in conjunction with other nonplanar intersection curves. This procedure is incorporated into the general algorithm for computing nonplanar intersections

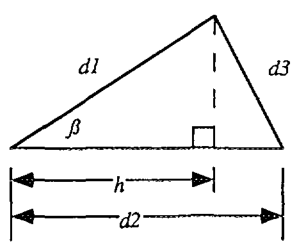

FIGURE 3

TABLE 3
Summary of Conditions Giving Rise to Two Point Tangencies Between Pairs of Natural Quadric Surfaces

Surface pair	Geometric conditions
Sphere/cone	Center of sphere in plane $(V, \mathbf{w})$ at distance $d = r/\cos\alpha$ from vertex V.
Cylinder/cone	Skew axes; distance between axes, $d = r\sin\theta/\sqrt{\sin^2\theta - \sin^2\alpha}$, where θ is the acute angle between the axis vectors.
Cone/cone	Perpendicular axes; V_2 in plane $(V_1, \mathbf{w}_1)$; distance, μ, from V_1 to axis line$(V_2, \mathbf{w}_2)$ and distance, v, from V_1 to plane $(V_2, \mathbf{w}_2)$ are related as $v = \sqrt{1 - \tan^2\alpha_1 \tan^2\alpha_2}\cos\alpha_1 \cot\alpha_2\,\mu$.
	Same as previous with roles of the cones reversed.
	Skew axes and two additional complicated constraints. (See [6].)

between two natural quadrics and is described fully in [11].

Consider two quadrics $\mathbf{Q}_1$ and $\mathbf{Q}_2$ and suppose that $\mathbf{Q}_1$ is a ruled quadric to be used as a parameterization surface for a nonplanar intersection curve as described in [11]. The (s, t) parameters on $\mathbf{Q}_1$ have the following geometric interpretation. The parameter t selects a ruling $(-\pi \leq t \leq +\pi)$, and s is a signed distance along the ruling. Nonplanar intersections between two such quadrics can be described using an implicit equation in the parameter space of $\mathbf{Q}_1$ as [10, 11, 19]

$$a(t)s^2 + b(t)s + c(t) = 0. \quad (2)$$

The functions a, b, and c are rational quadratic polynomials that depend upon the types of quadrics involved. Ordered sets of points can be generated along the intersection curve by selecting successive rulings on $\mathbf{Q}_1$ (i.e., by selecting successive values of t in the range $-\pi$ to $+\pi$) and then solving the resulting quadratic equation (2) in s. In general, only the subsets of the entire $-\pi$ to $+\pi$ range where the discriminant of (2) is nonnegative correspond to real portions of the intersection curve. The ranges of parameter space that satisfy this inequality are delimited by the zeros of the discriminant, i.e., by those values of t satisfying

$$b(t)^2 - 4a(t)c(t) = 0. \quad (3)$$

Equation (3) can be expressed equivalently as a rational quartic polynomial equation [10, 11]; thus there are up to four real roots corresponding to the t values of rulings on $\mathbf{Q}_1$ that are tangent to $\mathbf{Q}_2$. All other values of t correspond to rulings on $\mathbf{Q}_1$ that either have no real intersection with $\mathbf{Q}_2$ (if the left-hand side of (3) is negative) or intersect $\mathbf{Q}_2$ in two distinct real points (if the left-hand side of (3) is positive). Points of tangency between $\mathbf{Q}_1$ and $\mathbf{Q}_2$ determine unique rulings on $\mathbf{Q}_1$ that are tangent to $\mathbf{Q}_2$, and each ruling on $\mathbf{Q}_1$ that is tangent to $\mathbf{Q}_2$ corresponds to a real root of (3). Thus if $\mathbf{Q}_1$ and $\mathbf{Q}_2$ are tangent at one or two points, those points can be computed by solving (2) using the appropriate roots of (3). The appropriate roots of (3) are those where the t values in intervals on either side yield a negative discriminant. In [11] we describe robust geometric methods to find all such roots without explicitly solving any equation of degree greater than two. We will therefore not consider isolated tangent points further in this paper.

5.0. PAIRWISE ANALYSIS OF SURFACE INTERSECTIONS

Table 4 summarizes the configurations of natural quadrics that yield planar intersection curves as derived in [6]. The remainder of this section is devoted to a discussion of how the planar curves in the intersection can be computed for some of the more complex cases. Complete details for all cases are given in [14]. The case-specific pseudocode presented in that report is generally short and was carefully developed in order to maximize speed and numerical reliability.

Our algorithms work with the natural quadrics in general position and orientation, and they do not employ coordinate system transformations of any sort. The algorithms for the sphere–sphere, sphere–cylinder, sphere–cone, and cylinder–cylinder cases operate by directly determining the type of the planar curve(s) and computing the defining parameters shown in Table 1. Since these algorithms are straightforward, we will not present the details here. For a complete analysis, see [6, 14].

As we will see below, some degenerate configurations involving cylinder–cone and cone–cone intersections are more complex. In those cases we determine the plane or planes containing the conics, thereby reducing the problem to a pair of plane–cone or plane–cylinder intersections. Detailed procedures for intersecting planes and natural quadrics based on geometric constructions are described in [9, 13].

We will not present here the derivations of the expressions for the planes containing the intersection as they—like those leading to the characterizations summarized in Table 4—are generally long and somewhat tedious. These derivations can be found in [14]. Our general approach there is to write the quadric equation of the planes containing the conics as determined by the appropriate pencil matrix and the corresponding constraints derived in [6]. We then manipulate the equation into a form that can be easily factored into two linear terms. These terms are the plane equations from which the normal vectors

TABLE 4
Summary of Conditions Giving Rise to Planar Intersection Curves Between Pairs of Natural Quadric Surfaces

Surface pair	Geometric conditions	Results
Sphere/sphere	All	Empty, one tangent point, or one circle
Sphere/cylinder	Center of sphere on axis of cylinder	Empty, one tangent circle, or two circles
Sphere/cone	Center of sphere on axis of cone	Empty, one tangent circle, one circle + vertex, or two circles
Cylinder/cylinder	Parallel axes Intersecting axes and equal radii	Empty, one tangent line, or two lines Two ellipses
Cylinder/cone	Coincident axes	Two circles
	Axes intersect in a point at distance $d = r/\sin\alpha$ from the vertex of the cone	Two ellipses (same or opposite halves of the cone), or one ellipse and one tangent line
Cone/cone	Parallel axes, same half angle	Ellipse, shared tangential ruling, or hyperbola
	Coincident axes	Two circles or single vertex
	Axes intersect at point I such that $d_1 \sin\alpha_1 = d_2 \sin\alpha_2$ where d_i is the distance from vertex i to I. (This includes the case where the vertices coincide; i.e., $d_1 = d_2 = 0$.)	Various combinations of pairs of conics *or* a tangent line plus a conic (1–4 lines if the vertices coincide)

can immediately be found. It is then straightforward to find a point common to the two planes.

5.1. *Cylinder–Cone Intersections*

In [6] we showed that the intersection of a cylinder and a cone is a planar curve if and only if either (i) the axes coincide, or (ii) the axes intersect at a distance $r/\sin\alpha$ from the cone vertex.

5.1.1. *Coincident Axes.* We consider first the case of identical axes. The intersection is two circles lying on opposite cone halves at a distance $d = r/\tan\alpha$ from the vertex (see Fig. 4):

```
input: cyl: cylinder; con: cone
d := cyl.r/tan(con.α)
output: circle 1: C := con.V + d*con.w
                  w := con.w
                  r := cyl.r
        circle 2: C := con.V − d*con.w
                  w := con.w
                  r := cyl.r
```

5.1.2. *Intersecting Axes.* If the axes intersect at a point I whose distance from the vertex is $r/\sin\alpha$, then the intersection is either one ellipse plus a tangent line or a pair of ellipses. The former results if the acute angle between the axis lines is the same as the cone half-angle. Our approach is to find the pair of planes containing the intersection. When the intersection is a tangent line plus an ellipse, one of the planes is tangent to the cylinder and the cone along the shared ruling. While in principle, therefore, we need not distinguish between the line-plus-ellipse and two-ellipse cases, it is desirable to do so since detecting tangentially shared rulings is quite delicate numerically. If we were to compute the plane and then rely on the plane–cylinder algorithm to detect a tangent line of intersection, there would be a much greater probability that, due to small numerical errors, the plane might be judged to intersect the cylinder in two parallel lines, a

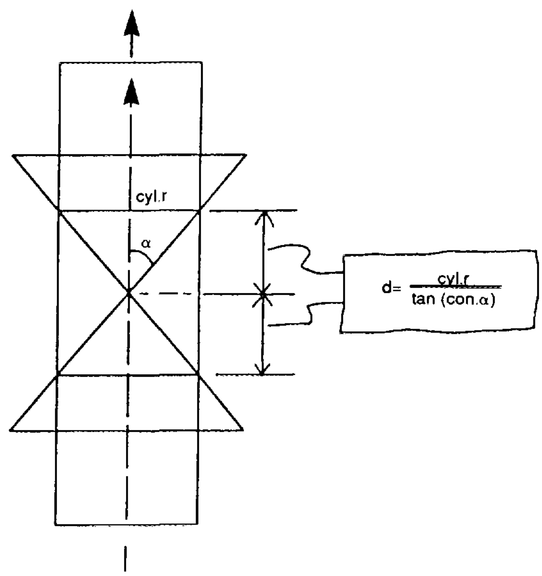

FIGURE 4

long skinny ellipse, or not at all. Fortunately a simple test in terms of the parameters of the given cylinder and cone suffices. We therefore advocate detecting this special case and then computing directly the defining parameters of the tangent line.

It can be shown that the normals to the planes containing the intersection are [14]

$$\mathbf{n}_1 = \text{cyl.}\mathbf{w} + \sec\alpha\ \text{con.}\mathbf{w}$$

and

$$\mathbf{n}_2 = \text{cyl.}\mathbf{w} - \sec\alpha\ \text{con.}\mathbf{w}.$$

To complete the specification of the planes containing the conic curves, we must find a point on each plane. A long but relatively straightforward calculation shows that a point Q common to the two planes can be computed as [14]

$$Q = V + \frac{c\omega\sin^2\alpha}{s^2}\,\text{cyl.}\mathbf{w} + \left(\omega - \frac{\omega\sin^2\alpha}{s^2}\right)\text{con.}\mathbf{w},$$

where

$$c = \text{con.}\mathbf{w}\cdot\text{cyl.}\mathbf{w} = \cos\theta;\ c^2 + s^2 = 1,$$

and ω is the signed distance of I along the cone axis line ($|\omega| = r/\sin\alpha$.) [6].

In the line-plus-ellipse case, $\theta = \alpha$, and it can be shown that $\mathbf{n}_2$ is perpendicular to the cylinder axis vector and that Q is a point on both the cylinder and the cone [14]. When we determine that $\theta = \alpha$, we only intersect the cylinder with plane (Q, $\mathbf{n}_1$); we express the tangent line directly.

We show I—the intersection of the cylinder and cone axes—as a parameter to this routine. This is reasonable since I would have been computed previously while testing to see if the intersection is planar.

We summarize our results in the following pseudocode:

```
input: cyl: cylinder; con: cone; I: point
cos_theta := cyl.w.con.w
sin_sqr_theta := 1 − cos_theta*cos_theta
cos_alpha := cos(con.α)
F := 1 / cos_alpha
sin_sqr_alpha := 1 − cos_alpha*cos_alpha
w := signed_distance_along_line(I , line(con.V,con.w))
n1 := normalize(cyl.w + F*con.w)
t := w*sin_sqr_alpha/sin_sqr_theta
Q := con.V + cos_theta*t*cyl.w + (w-t)con.w
conic_1 := intersect(cyl , plane(Q,n1))
if abs(cos_theta) = cos_alpha then
  conic_2 := line(con.V,cyl.w)
else
  n2 := normalize (cyl.w − F*con.w)
  conic_2 := intersect(cyl , plane(Q,n2))
```

Figure 5 illustrates the three possible results: (a) two intersecting ellipses on the same half of the cone, (b) an ellipse plus a tangentially shared ruling, and (c) two ellipses on opposite halves of the cone.

5.2. *Cone–Cone Intersections*

We showed in [6] that the intersection between two cones is a planar curve if and only if one of the following three conditions is satisfied: (i) the axes are distinct but parallel, and the cones have the same half-angle; (ii) the axes intersect at a point I equidistant from the two cones; or (iii) the axes are coincident. Condition (ii) is equivalent to $r_1\sin\alpha_1 = r_2\sin\alpha_2$, where r_i is the distance from vertex i to I, and α_i is the half-angle of cone i. The treatment of condition (iii) is fairly straightforward [14]. We now consider the other two cases in turn.

5.2.1. *Distinct Parallel Axes, Same Half-Angle.* When condition (i) is satisfied, the pencil contains a single plane, and the intersection is therefore a single (possibly degenerate) conic. The intersection is a hyperbola, an ellipse, or a double line if the vertex of one cone is, respectively, outside, inside, or on the other cone. (See Fig. 6). The intersection cannot be a parabola, nor can it be a circle since the axes are distinct.

We derive an invariant characterization of the plane in terms of the parameters of the two cones which will yield

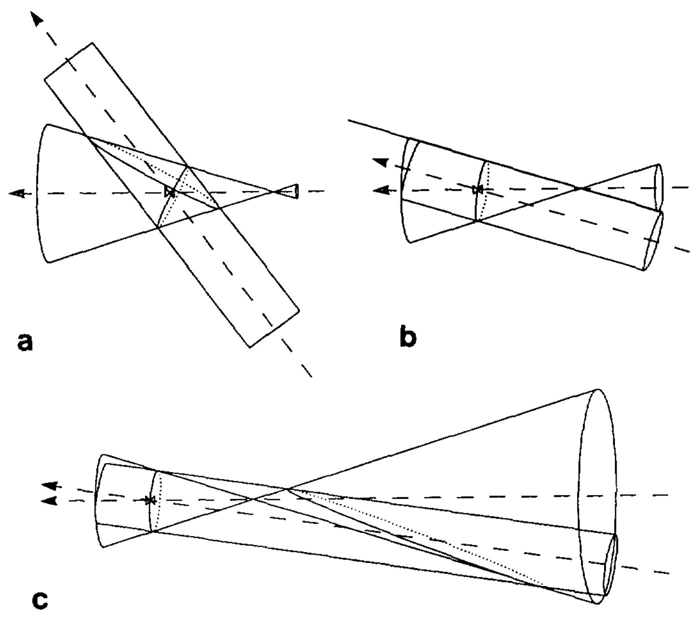

FIGURE 5

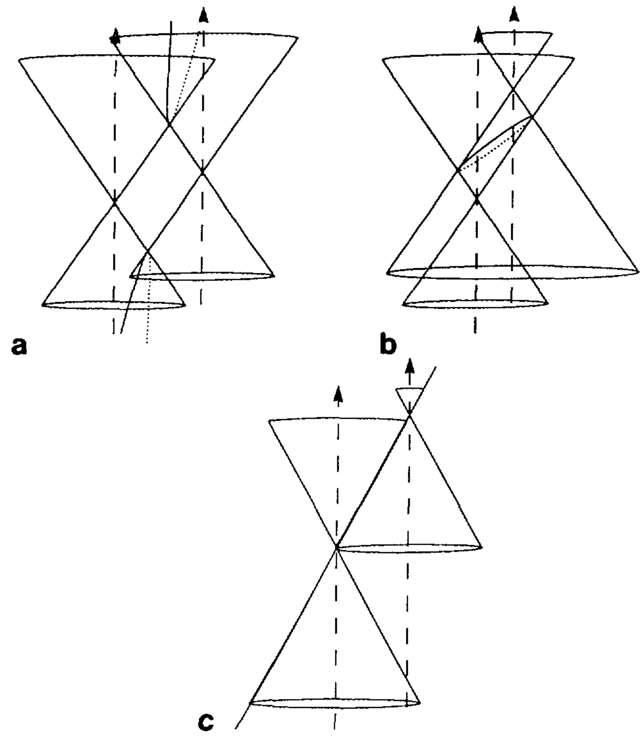

FIGURE 6

the correct plane for any of these relative vertex locations. As we argued in Section 5.1.2, however, it is prudent to detect the tangent line case since it is especially delicate numerically. This configuration is easily discovered by testing to see if the vertex of one cone lies on the surface of the other cone. For the other cases, we use the following results derived in [14]:

- the normal vector to the plane is parallel to $\mathbf{n} = (V_2 - V_1) - \omega \sec^2 \alpha_2 \mathbf{w}_1$, where ω is the signed distance between V_2 and plane $(V_1, \mathbf{w}_1)$,
- the midpoint of the vertices lies on the plane.

We summarize these results in the following algorithm. Figure 6 illustrates the possible results.

```
input: C1, C2: cone
v1_to_v2 := C2.V − C1.V
if C2.V on C1 then
  conic := line(C1.V, normalize(v1_to_v2))
else
  w := v1_to_v2·C1.w
  Q := midpoint(C1.V , C2.V)
  F2 := sec(C2.α)
  n := normalize(v1_to_v2 − w*F2*F2*C1.w)
  conic := intersect(C1 , plane(Q,n))
```

5.2.2. *Intersecting Axes.* Finally we consider the more complex condition (ii). If the vertices of the two cones are identical (i.e., $r_1 = r_2 = 0$), the cones intersect in one to four lines, or they intersect only at their common vertex. If the vertices are distinct (i.e., condition (ii) is satisfied with $r_i \neq 0$), the intersection may consist of various pairs of (possibly degenerate) conics depending on the angle between the axes and on the cone half-angles.

As before, our approach is to compute the pair of planes containing the conics and then to perform two plane–cone intersections. It can be shown that the normals to the two planes are parallel to [14]

$$\mathbf{n}_1 = \sec \alpha_2 \mathbf{w}_2 + \sec \alpha_1 \mathbf{w}_1$$
$$\mathbf{n}_2 = \sec \alpha_2 \mathbf{w}_2 - \sec \alpha_1 \mathbf{w}_1.$$

A long and tedious calculation demonstrates that a point common to the two planes can be computed as [14]

$$Q = C1.V - gC2.\mathbf{w} + \left(\frac{e_1}{d_1} + cg\right) C1.\mathbf{w},$$

where

$$g = \frac{ce_1}{d_2} + \frac{e_2}{d_3}$$
$$d_1 = 2 \sec \alpha_1$$
$$d_2 = s^2 d_1$$
$$d_3 = 2s^2 \sec \alpha_2,$$

and e_1, e_2, s, c, v, and ω can be defined in terms of trigonometric constants and signed distances as summarized in the pseudocode below. It can also be shown that this expression for Q reduces to $C1.V$ if the vertices are coincident. Thus, while in principle this need not be a special case, we handle it as such in the algorithm in the interests of performance and numerical reliability.

```
input: C1, C2: cone
/* Compute trigonometric constants */
c := C1.w·C2.w
c_sqr := c*C
s_sqr := 1.0 − c_sqr
s := sqrt(s_sqr)
F1 := sec(C1.α)
F2 := sec(C2.α)
F2_sqr :1 F2*F2
/* Compute vectors normal to planes containing conics */
n1 := normalize(F2*C2.w + F1*C1.w)
n2 := normalize(F2*C2.w − F1*C1.w)
/* Compute point common to two planes */
if C1.V = C2.V then
```

```
   Q := C1.V
else
   v1_to_v2 := C2.V - C1.V
   w := v1_to_v2·C1.w
   u := (v1_to_v2·C2.w - c*w)/s
   e1 := 2(s*w-c*u)/(s*F1)
   e2 := 2(u(1-s_sqr*F2_sqr)-s*c*F2_sqr*w)/(s*F2)
   d1 := 2*F1
   d2 := s_sqr * d1
   d3 := 2 * s_sqr * F2
   g := c*e1/d2 + e2/d3
   Q := C1.V - g*C2.w + (e1/d1 + c*g)C1.w

/* Compute the two (possibly degenerate) conics by intersecting one of the cones with the two planes. */
conic_1 := intersect(C1 , plane(Q,n1))
conic_2 := intersect(C1 , plane(Q,n2))

/* Finally, ensure that no extraneous vertex intersections are reported. */
if conic_1 is a single point then
   return only conic_2
else if conic_2 is a single point then
   return only conic_1
else
   return both conic_1 and conic_2
```

Figure 7 illustrates the results of applying this single algorithm to a variety of cone pairs. Shown are coincident vertices yielding two real lines of intersection (Fig. 7a), a pair of intersecting ellipses (Fig. 7b), an ellipse plus a tangentially shared ruling (Fig. 7c), and an ellipse plus a hyperbola (Fig. 7d).

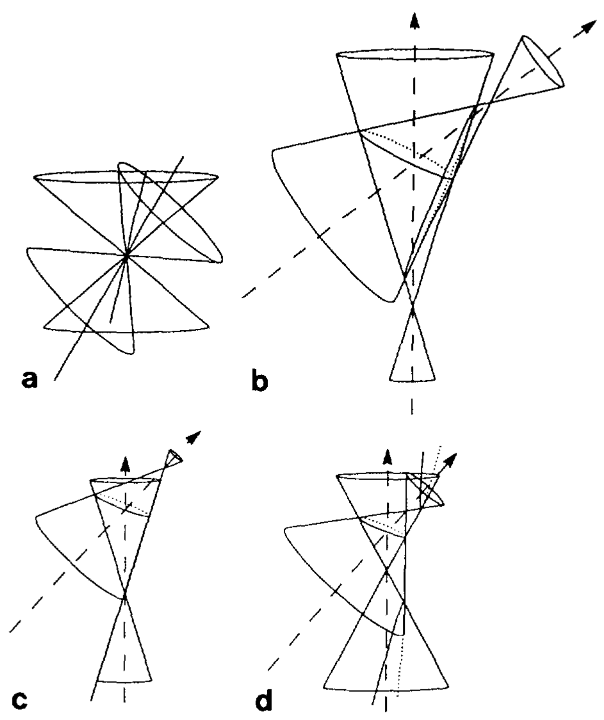

FIGURE 7

6.0. OTHER POSSIBLE DEGENERATE INTERSECTIONS

By Bezout's Theorem, two quadric surfaces always intersect in a degree four curve in complex projective space. Thus far we have been concerned here exclusively with situations where this degree four curve splits into two (possibly degenerate) degree two curves. The other possible way in which the intersection curve can degenerate is into a line and a nondegenerate degree three space curve. This cannot happen when spheres are involved since there are no straight lines on a sphere. Clearly too it cannot happen in cylinder–cylinder intersections since the cylinder axis vectors must be parallel in order for the intersection to contain a straight line. But when their axes are parallel, two cylinders either have no real intersection, or they intersect in one tangent or two parallel lines. Therefore we need only concern ourselves with cylinder–cone and cone–cone intersections. The following two theorems present necessary and sufficient conditions for the line plus space cubic case to arise in cone–cylinder and cone–cone intersections.

Although portions of this material are natural consequences of Bezout's Theorem, some of these results have not been well understood by all in the modeling community. We therefore include their statements here for completeness; for rigorous proofs, see [14].

THEOREM 3. *The intersection of a cylinder and a cone degenerates into a line and a space cubic if and only if all the following conditions hold*:

(i) *The angle θ between the axis vectors is the same as the cone half-angle α.*
(ii) *The cone vertex lies on the cylinder.*
(iii) *The axes are skew.*

When the conditions of Theorem 3 are satisfied, the line of intersection can be written as line (V, $\mathbf{w}_{cyl}$). Calculation of the space cubic is discussed in [11]. The best way to visualize the geometry giving rise to a line and a space cubic is to start with the geometry of Fig. 5b. Recall in this case that the cylinder and cone axes intersect, and the intersection curve splits into a tangent line and an ellipse. If we rotate either the cylinder or the cone about their common line, the axes become skew, the ruling is no longer shared tangentially, and the ellipse breaks apart into a space cubic at the point where it intersects the shared ruling. One end of the ellipse then tends toward infinity in one direction, and the other toward infinity in the opposite direction. See Fig. 8.

THEOREM 4. *The intersection of two cones degenerates into a line and a space cubic if and only if all the*

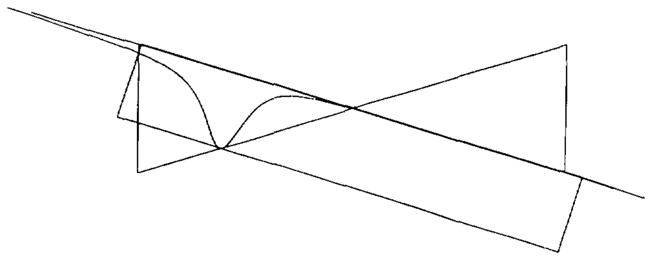

FIGURE 8

following conditions hold:

(i) *Each vertex lies on the other cone.*
(ii) *The vertices are distinct.*
(iii) *The axes are skew.*

When the conditions of Theorem 4 are satisfied, the line of intersection can be written as line (V_1, normalize ($V_2 - V_1$)). Calculation of the space cubic is discussed in [11]. One can visualize this geometry in a manner analogous to that described for the cylinder–cone case above. Start with the geometry of Fig. 7c and twist one of the cones about the common ruling. The result is illustrated in Fig. 9.

Given the geometric data for cylinders and cones, it is easy to implement analytic tests for the geometric conditions in Theorem 3 and 4. A procedure for deciding whether two lines are skew is described in Section 3. Formulas for deciding whether a given point lies on a cylinder or cone are also provided at the end of Section 3.

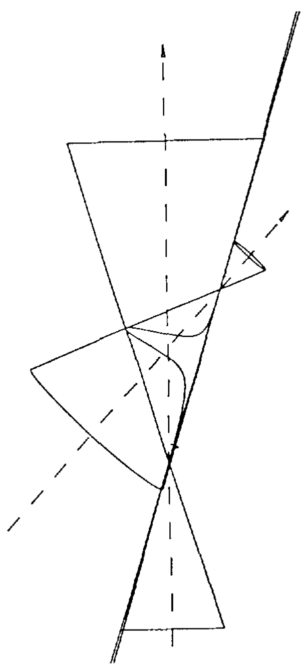

FIGURE 9

7.0. SUMMARY

Detecting the presence of conic sections in quadric surface intersections is important for a variety of reasons including more efficient and reliable representation and analysis [6, 11, 19] and the possibility of blending with low-degree surfaces [23]. Using algebraic geometry, we characterized in [6] all configurations where the intersection of a given pair of natural quadrics is planar. Here we have discussed methods for calculating these planar intersection curves once we have determined that they actually arise. These methods are based solely on the geometric data of natural quadrics in general position and orientation. This combination of algebraic and geometric approaches is ideal since it exploits the rigor of the algebraic method and the numerical reliability of computer implementations based on geometric representations.

All the algorithms described in this paper have been implemented by the first author in a solid modeling system being developed at the University of Kansas and have proven to be quite efficient and highly reliable in practice. The algorithms were originally implemented in C under UNIX on a Silicon Graphics IRIS 4D/60 workstation where some performance evaluation was conducted. This machine runs at about 7 MIPS and is capable of approximately 0.7 MFLOPS. We measured the required execution times for the two computationally intensive examples: the cylinder–cone case of Fig. 5c and the cone–cone case of Fig. 7d. Measurements indicate that the geometry of Fig. 5c can be intersected approximately 480 times per second, while that of Fig. 7d can be performed at the rate of about 330 per second.

The algorithms presented in this paper complement those described elsewhere. How all these procedures fit together is best understood by examining the following prototypical high-level intersection algorithm:

intersect_two_quadrics (**Q**1, **Q**2)

1. Test the relevant conditions from Table 4. If one is satisfied, then execute the corresponding algorithm as presented in Section 5 of this paper (invoking the algorithms described in [13] as appropriate) and return the result.
2. If one of the quadrics is a cone and the other is either a cylinder or a cone, test for the line and space cubic conditions described in, respectively, Theorems 3 and 4 of Section 6. If the relevant conditions are satisfied, compute and return the line defined in Section 6 and the space cubic using the algorithms described in [11].
3. Otherwise the intersection is a nondegenerate quartic curve. Compute the general quadric surface intersection curve (QSIC) as described in [11]. While determining the parametric limits of the QSIC on the parameterization surface, detect single and double points of tangency as discussed briefly in Section 4 and detailed in [11]. Tan-

gent points may arise in conjunction with disjoint QSIC branches.

REFERENCES

1. A. Dresden, *Solid Analytic Geometry and Determinants*, Wiley, New York, 1930.
2. R. T. Farouki, C. A. Neff, and M. A. O'Connor, Automatic parsing of degenerate quadric-surface intersections, *ACM Trans. Graphics* **8**(3), 1989, 174–203.
3. R. N. Goldman, Quadrics of revolution, *IEEE Comput. Graphics Appl.* **3**(2), 1983, 68–76.
4. R. N. Goldman, Intersection of two lines in three-space, in *Graphics Gems* (A. Glassner, Ed.), p. 304, Academic Press, San Diego, 1990.
5. R. N. Goldman, and J. R. Miller, Combining algebraic rigor with geometric robustness for the detection and calculation of conic sections in the intersection of two natural quadric surfaces, *Proceedings of the ACM Conference on Solid Modeling Foundations and CAD/CAM Applications, June 5–7, 1991, Austin, Texas.*
6. R. N. Goldman, and J. R. Miller, Detecting and calculating conic sections in the intersection of two natural quadric surfaces, part I: Theoretical analysis, University of Kansas, Department of Computer Science, Technical Report TR-93-1, January 1993.
7. D. G. Hakala, R. C. Hillyard, B. E. Nourse, and P. J. Malraison, Natural quadrics in mechanical design, *Proc. Autofact West* **1**, November 1980.
8. J. K. Johnstone, and C. K. Shene, Dupin cyclides as blending surfaces for cones, in *Design and Application of Curves and Surfaces: Mathematics of Surfaces V*, (R. B. Fisher, Ed.), pp. 3–24, Oxford University Press, Oxford.
9. J. K. Johnstone, and C. K. Shene, Computing the intersection of a plane and a natural guadric, *Comput. Graphics* **16**, 1992, 179–186.
10. J. Levin, A parametric algorithm for drawing pictures of solid objects composed of quadric surfaces, *Commun. ACM* **19**(10), 1976, 555–563.
11. J. R. Miller, Geometric approaches to nonplanar quadric surface intersection curves, *ACM Trans. Graphics* **6**(4), 1987, 274–307.
12. J. R. Miller, Analysis of quadric-surface-based solid models, *IEEE Comput. Graphics Appl.* **8**(1), 1988, 28–42.
13. J. R. Miller, and R. N. Goldman, Using tangent balls to find plane sections of natural quadrics, *IEEE Comput. Graphics Appl.* **12**(2), 1992, 68–82.
14. J. R. Miller, and R. N. Goldman, Detecting and calculating conic sections in the intersection of two natural quadric surfaces, part II: Geometric constructions for detection and calculation, University of Kansas, Department of Computer Science, Technical Report TR-93-2, January 1993.
15. S. Ocken, J. T. Schwartz, and M. Sharir, Precise implementation of CAD primitives using rational parameterizations of standard surfaces, in *Solid Modeling By Computers: From Theory to Applications* (M. S. Pickett and J. W. Boyse, Eds.), Plenum Press, New York, 1984.
16. M. A. O'Connor, Natural quadrics: Projections and intersections, *IBM J. Res. Dev.* **33**(4), 1989, 417–446.
17. L. Piegl, Geometric method of intersecting natural quadrics represented in trimmed surface form, *CAD* **21**(4), 1989, 201–212.
18. M. J. Pratt, Cyclides in computer aided geometric design, *Comput. Aided Geom. Des.* **7**, 1990, 221–242.
19. R. F. Sarraga, Algebraic methods for intersections of quadric surfaces in GMSOLID, *Comput. Vision Graphics, Image Process.* **22**(2), 1983, 222–238.
20. C. K. Shene, and J. K. Johnstone, On the planar intersection of natural quadrics, *Proceedings of the ACM Symposium on Solid Modeling Foundations and CAD/CAM Applications, June 5–7, 1991, Austin, Texas*, pp. 233–242.
21. C. K. Shene, Planar intersection and blending of natural quadrics, Ph.D. Dissertation, Department of Computer Science, John Hopkins University, Baltimore, MD, 1992.
22. J. Warren, On algebraic surfaces meeting with geometric continuity, Ph.D. Dissertation, Department of Computer Science, Cornell University, Ithaca, NY, 1986.
23. J. Warren, Blending algebraic surfaces, *ACM Trans. Graphics* **8**(4), 1989, 263–278.