

Geometric Approaches to Nonplanar Quadric Surface Intersection Curves

JAMES R. MILLER
Control Data Corporation

Quadric surfaces occur frequently in the design of discrete piece parts in mechanical CAD/CAM. Solid modeling systems based on quadric surfaces must be able to represent intersection curves parametrically and in a fashion that allows the underlying surfaces to be partitioned. An algebraic approach originally developed by Levin meets these needs but is numerically sensitive and based on solutions to fourth-degree polynomial equations. In this paper we develop geometric approaches that are robust and efficient, and do not require solutions to polynomials of degree higher than 2.

Categories and Subject Descriptors: G.4 [**Mathematics of Computing**]: Mathematical Software—*efficiency; reliability and robustness*; I.3.5 [**Computer Graphics**]: Computational Geometry and Object Modeling—*curve, surface, solid, and object representations; geometric algorithms, languages, and systems*

General Terms: Algorithms, Performance, Reliability

Additional Key Words and Phrases: Boundary evaluation, quadric surfaces, solid modeling

1. INTRODUCTION

Quadric surfaces occur frequently in the design of discrete piece parts in mechanical CAD/CAM. Several solid modeling systems are based at least in part on them [1, 2, 6, 8, 13, 14]. When modeling with any type of surface, it is important that the intersection of one surface with another be represented efficiently and accurately. In solid modeling, the ability to represent such curves parametrically is also critical so that a preferred direction can be established and an ordered set of points along the intersection generated.

When two quadric surfaces intersect, the intersection is either a point or a line (if they are tangent) or one or two intersection curve branches. The intersection curve branches will be either conic sections (including straight lines) or fourth-degree nonplanar space curves. It is certainly straightforward to represent conic sections exactly and parametrically. Nonplanar quadric surface intersection curves can also be represented exactly and parametrically using a method pioneered by Levin [9–11] and further developed by Sarraga [14].

Author's present address: Computer Science Department, University of Kansas, Lawrence, KS 66045. Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

© 1987 ACM 0730-0301/87/1000-0274 \$01.50

ACM Transactions on Graphics, Vol. 6, No. 4, October 1987, Pages 274–307.

Before these methods are discussed, some background remarks on representations are needed. Two primary approaches to the representation of quadric surfaces have evolved: an algebraic one and a geometric one [4]. The algebraic approach is summarized in Section 2 and is characterized by the representation of all quadric surfaces in a single form. A single surface–surface intersection algorithm suffices in this approach. The geometric approach contrasts with the algebraic one primarily in that surfaces are represented as type-dependent combinations of scalars, points, and vectors, and algorithms for surface–surface intersections are dependent upon the types of surfaces involved [4, 7, 12].

A number of problems exist with the exclusive use of the algebraic approach. These are well documented [4, 7, 15]. Indeed, the discovery of these problems in practice led to the development and use of the geometric approach. The problems relate primarily to a lack of numerical robustness, and we amplify on some of them in Section 2 after we have developed some requisite background material.

Although geometric approaches work well when conic sections arise [5, 12], adequate methods based on these approaches when nonplanar intersection curves result have not been described in the literature. Therefore, it has been suggested that geometric approaches be used to detect and describe conic sections when they arise, and that algebraic ones be used only after it has been determined that a nonplanar curve will result [12]. In this paper we describe how geometric approaches can be used for nonplanar intersections as well, and we note several advantages that arise from using these approaches.

We consider here only the so-called *natural quadrics* [7], that is, the sphere, cylinder, cone, and plane. These are by far the most commonly occurring quadric surfaces used in modeling mechanical objects. The methods described herein can be employed with many of the remaining quadrics as well. As we observe later, however, some additional techniques will be needed for some of them, and there may well come a point at which a purely geometric approach ceases to be practical or even possible.

The work involved in intersecting quadric surfaces consists of the following three steps (the second and third are only required if the intersection is nonplanar):

- Detect and represent conic sections.
- Derive the parametric representation of the curve using the parameter space of a ruled quadric surface.
- Determine critical points in parameter space such as where the curve turns on itself, intersects itself, or tends to infinity.

The first step is really optional in the algebraic approach, but it is typically performed since these curves are more effectively handled in this fashion [14]. As we shall see, the second step is the easiest. Finally, the third step involves determining the topology and extent of the intersection curve and is probably the most important aspect of ensuring a robust intersection algorithm.

Section 2 summarizes the salient aspects of the algebraic approach and provides important background for the geometric approaches discussed later. Section 3 introduces the geometric representations with which we shall be dealing, as well

as some notation that will be used throughout. The possibilities of hybrid approaches are briefly considered (and then dismissed) in Section 4, and the role of conic sections are described in Section 5. The heart of the paper is Section 6, where the geometric approaches to nonplanar intersection curves are fully developed. Section 7 is a summary of the results.

2. THE ALGEBRAIC APPROACH

2.1 Representations and Methods

Any quadric surface can be represented as the set of all points in three-dimensional space satisfying the second-degree polynomial equation

$$Ax^2 + By^2 + Cz^2 + 2Dxy + 2Eyz + 2Fxz + 2Gx + 2Hy + 2Jz + K = 0 \quad (1)$$

for suitable values of the coefficients $A \cdots K$. Setting $A = B = C = -K = 1$ and $D = E = F = G = H = J = 0$, for example, defines a unit sphere at the origin. The coefficients “numerically encode” the type of the surface, as well as its position, orientation, and size. Dresden [3] discusses methods for determining the type of a quadric surface given its coefficients.

Equation (1) is typically written in matrix form as

$$PQP^T = 0, \quad (2)$$

where

$$P = (x, y, z, 1),$$

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

Given any two quadric surfaces represented by their matrices Q_1 and Q_2 , we can construct a family of quadric surfaces that are linear combinations of the two surfaces. This family is called the *pencil* of the two original surfaces and is defined as

$$Q = Q_1 - \lambda Q_2 \quad (3)$$

for arbitrary real λ [3]. For our purposes there are two interesting properties related to the pencil. If the surfaces corresponding to Q_1 and Q_2 intersect, then each quadric surface in the pencil intersects Q_1 and Q_2 along their curve of intersection. (This is easily verified by considering some point P on the intersection curve of Q_1 and Q_2 and showing that it must also be on the quadric surface Q defined by (3) for any arbitrary real value λ .) The second property of interest is that there is at least one ruled surface in the pencil of any two quadrics. (A ruled surface is one that can be characterized as a collection of straight lines, for example, a cylinder or a cone.) This is not intuitively obvious. For a proof, see [9] or [11].

Ruled quadrics are of interest since they are easily parameterized, and one can express quadric surface intersection curves exactly and parametrically by using

the parameterization scheme for a ruled quadric in the pencil. The two most common parameterizations used on the ruled quadric are based on rational polynomials [10, 11] and trigonometric functions [14]. A rational polynomial parameterization for a unit cylinder whose axis is the z -axis, for example, is

$$P(s, r) = (x(r), y(r), z(s)) = \left(\frac{1 - r^2}{1 + r^2}, \frac{2r}{1 + r^2}, s \right),$$

and a trigonometric parameterization for the same cylinder is

$$P(s, t) = (x(t), y(t), z(s)) = (\cos(t), \sin(t), s).$$

Computing polynomials is faster than computing trigonometric functions, and it is easier algebraically to locate zeros of polynomial functions. However, the polynomial parameter must vary from $r = -\infty$ to $r = +\infty$ to sweep out entire circular cross sections. It is therefore common in practice to use two such polynomials, each varying the parameter from $r = -1$ to $r = +1$ and sweeping out half of the circular cross section. This makes intersection curve bookkeeping more complex, however, since artificial breaks in the parameterization are introduced.

A trigonometric parameterization need only vary its parameter from $t = -\pi$ to $t = +\pi$ to sweep out entire circular cross sections. Moreover, the periodicity of trigonometric functions actually simplifies bookkeeping related to calculating and storing intersections. On the subject of finding zeros of functions, geometric approaches will be possible for most of the situations that we shall encounter. When an algebraic approach must be used, there are straightforward mappings between the trigonometric and rational polynomial parameters. (For the cylinder example, we have $r = \tan(t/2)$.)

Largely in the interests of simplified bookkeeping, trigonometric parameterizations are used for the geometric schemes developed in this paper. Since we are only dealing with the natural quadrics and nonplanar intersections, we need only consider the cylinder and cone as parameterization surfaces. The algebraic approach deals with these surfaces in canonical position and orientation; they are therefore described as [14]

Cylinder of radius r (Figure 1):

$$\begin{aligned} x &= r * \cos(t), \\ y &= r * \sin(t), \\ z &= s. \end{aligned} \tag{4}$$

Cone with half-angle α (Figure 2):

$$\begin{aligned} x &= s * \tan(\alpha) * \cos(t), \\ y &= s * \tan(\alpha) * \sin(t), \\ z &= s. \end{aligned} \tag{5}$$

Intersection curves can be described in terms of the parameterization surface as

$$\mathbf{a}(t)s^2 + \mathbf{b}(t)s + \mathbf{c}(t) = 0. \tag{6}$$

To compute the functions \mathbf{a} , \mathbf{b} , and \mathbf{c} once the parameterization surface has been selected, one of the intersecting quadrics is transformed to the local coordinate

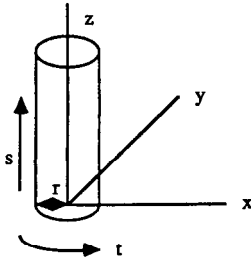


Fig. 1. Surface parameters on a canonical cylinder of radius r .

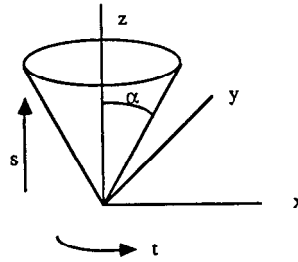


Fig. 2. Surface parameters on a canonical cone with half-angle α .

system defined by the parameterization surface. The ruled surface parameterization ((4) or (5)) is then substituted for x , y , and z in the transformed implicit equation (1) of the quadric. The resulting equation is then manipulated into the form of (6).

The \mathbf{a} , \mathbf{b} , and \mathbf{c} functions derived in this fashion for the algebraic approach are [14]

Cylinder (from (4) substituted into (1)):

$$\begin{aligned} \mathbf{a} &= C, \\ \mathbf{b}(t) &= 2Er \sin(t) + 2Fr \cos(t) + 2J, \\ \mathbf{c}(t) &= Ar^2 \cos^2(t) + Br^2 \sin^2(t) + 2Dr^2 \sin(t) \cos(t) \\ &\quad + 2Gr \cos(t) + 2Hr \sin(t) + K. \end{aligned} \tag{7}$$

Cone (from (5) substituted into (1)):

$$\begin{aligned} \mathbf{a}(t) &= A \tan^2(\alpha) \cos^2(t) + B \tan^2(\alpha) \sin^2(t) + C + 2D \tan^2(\alpha) \sin(t) \cos(t) \\ &\quad + 2E \tan(\alpha) \sin(t) + 2F \tan(\alpha) \cos(t), \\ \mathbf{b}(t) &= 2G \tan(\alpha) \cos(t) + 2H \tan(\alpha) \sin(t) + 2J, \\ \mathbf{c} &= K. \end{aligned} \tag{8}$$

Once these functions have been developed, the range(s) of t values on the parameterization surface that correspond to points on the intersection curve must be determined. These are bounded by *critical points* on the curve, that is, places in the t direction of the parameter space where the intersection curve turns back on itself, crosses itself, or goes to infinity. In the algebraic approach, these points must be determined by an algebraic analysis of eqs. (7) or (8), which seeks values of t for which s tends to infinity or the discriminant ($\mathbf{b}^2 - 4\mathbf{ac}$) vanishes. This leads to a partitioning of parameter space on the basis of the solutions to fourth-degree polynomials.

Once these critical points are determined, one marches along the intersection curve by selecting successive values for t between the critical points, substituting each value into (7) or (8), determining corresponding values for s by solving the quadratic equation (6), and calculating a point on the intersection curve using the resultant (s, t) pairs. When substituted into (4) or (5), (s, t) will yield a point

in the local coordinate system of the ruled quadric, which must then be transformed back to world coordinates by the inverse of the transformation used earlier.

The form of (6) will be simplified if one or more of \mathbf{a} , \mathbf{b} , and \mathbf{c} is constant or identically equal to zero. We have already seen constants arising in (7) and (8). Other situations yielding constants will appear in Section 6. If either \mathbf{a} or \mathbf{c} is identically zero, the intersection curve parameterization is linear in s . We shall see examples of this later as well. Whether or not any of these simplifications occurs depends not only on the relative position and orientation of the intersecting quadrics, but also on the specific parameterization surface used. As a result, some ruled surfaces are better suited as parameterization surfaces than others. Levin [10, 11] outlines an algebraic technique that will find the best such ruled quadric in the pencil under many conditions. A natural fallout of the geometric techniques we shall examine later is that the best such surface will always be determined from a purely geometric analysis. Indeed, in our context it will always be one of the two surfaces involved.

2.2 Problems with the Algebraic Approach to Intersections

Consider first conic section curves in the xy -plane. Conic sections can be represented algebraically via (1) with $z = 0$. (Equivalently, they can be represented with the 3×3 matrix determined by removing the third row and third column of Q in (2).) Wilson [15] cites experience with IGES, which has shown that algebraic representations of conic curves are extremely sensitive to small perturbations in the coefficients. This is manifested in two critical ways. First, small changes in coefficient values can cause large changes in the locations of points satisfying the conic equation. Second, it is often important to be able to determine the type of a conic section (circle, ellipse, etc.) from its algebraic coefficients. Unfortunately, this determination is numerically unstable because it depends critically on whether various functions of the coefficients are precisely zero, positive, or negative. Small perturbations in the coefficient values will therefore not only induce large changes in the points satisfying the conic equation, but will also actually change the *type* of the conic section.

Transformations are common operations in modeling systems, which, because of their floating-point implementation on a computer, produce the effects of perturbations in coefficients. Wilson's results show a continuing degradation in numerical accuracy with each transformation applied. Moreover, conics that were correctly classified according to their type before any transformations were applied were often incorrectly classified after the application of transformations.

Wilson's results are specifically for conics, but there is every reason to believe that they will be similar for quadrics [15]. The representations and analytical approaches are exactly analogous. This is all particularly troublesome in the context of doing intersections using the algebraic approach. Aside from the normal (and frequent) use of transformations during the course of modeling, transformations are an integral part of the intersection operation in the algebraic approach. Recall that an intersecting quadric must be transformed into the local coordinate system of the parameterization surface before eqs. (7) or (8) and corresponding critical points can be determined. Points generated from the

application of these equations must then be transformed back to global coordinates. Worse yet, the parameterization surface type must be determined from the algebraic coefficients determined by (3) (again in the context of transformations) in order to determine which of (7) or (8) applies. As noted above, this determination is numerically unstable.

Geometric approaches for representing and intersecting quadrics minimize these numerical problems. Small changes to the data stored in the geometric approach induce correspondingly small changes in the positions of points on the surface. Moreover, it is impossible for such perturbations to change the type of a surface since, as we shall see in the next section, surface types are stored directly and need not be determined by algebraic analyses. We shall also see in Section 6 that transformations are never required for any aspect of the intersection operation.

In addition to the numerical problems described above, the selection of reasonable tolerances to zero is also quite difficult in the algebraic approach. As a simple example, consider the important problem of determining whether a particular intersection curve represented by (7) or (8) is linear in s . As we shall see in Section 6, C in (7) is related to differences involving squares of cosines. K in (8) is related to differences involving squares of distances. Armed only with the algebraic equations, however, one would tend to use the same tolerance to zero for both. This is a simple example of a more difficult general problem with the algebraic approach. It is often necessary to determine when various complex expressions involving the algebraic coefficients $A \dots K$ are zero. Determining an appropriate tolerance to zero for these expressions is extremely difficult since it is generally impossible to relate these expressions to anything spatial. If the geometric approach is used, however, these expressions are developed in terms of spatial quantities (vectors between points, radii, etc.) for which meaningful tolerances to zero are immediately obvious.

3. GEOMETRIC REPRESENTATIONS

In the geometric approach, surfaces are defined as a surface type code plus a type-dependent combination of scalars, points, and vectors. These parameters size, position, and orient the surfaces. The three surfaces of interest in this paper are specified as (see Figures 3–5)

“sphere,”	B, r	(center, B ; radius, r),
“cylinder,”	B, w, r	(axis point, B ; unit axis vector, w ; radius, r),
“cone,”	B, w, α	(vertex, B ; unit axis vector, w ; half-angle, α).

We need additional vectors for the cylinder and cone to support parameterizations on those surfaces. These vectors will be identified as u and v . The only requirement on them is that u , v , and w form a mutually perpendicular set of unit vectors. These extra vectors can be calculated and stored when needed; in particular, they need not be provided as input to the surface–surface intersection algorithms.

We shall follow the basic approach of Levin and use a ruled surface in the pencil of the two quadrics being intersected to parameterize the intersection

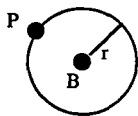


Fig. 3. Sphere centered at B with radius r . P represents an arbitrary point on the sphere.

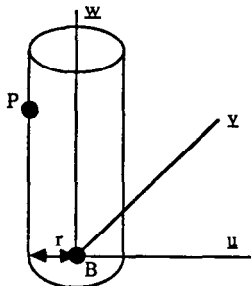


Fig. 4. Cylinder with axis defined by point B and unit vector w with radius r . u and v are arbitrary unit vectors perpendicular to w and each other. P represents an arbitrary point on the surface.

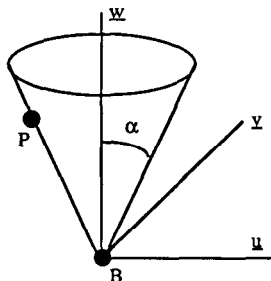


Fig. 5. Cone with vertex B , unit axis vector w , and half-angle α . u and v are arbitrary unit vectors perpendicular to w and each other. P represents an arbitrary point on the surface.

curve. The similarity ends there, however, since we shall use a coordinate-system-independent parameterization for the ruled surface and surface-type-dependent implicit equations for the other surface (instead of the generic equation (1)), and perform purely geometric analyses to determine the intersection curve topology, critical points, and various special cases. The a , b , and c functions that will result from this process will have obvious geometric interpretations, which will be exploited to predict, for example, the situations under which one or more of the functions is constant or identically zero.

Figure 4 illustrates the geometric parameters for a cylindrical parameterization surface. Its parametric representation is

$$P(s, t) = \gamma(t) + sw, \tag{9}$$

where

$$\gamma(t) = B + r(\cos(t)u + \sin(t)v).$$

For each t , $\gamma(t)$ is a point on a circle of radius r centered at B with normal w .

The geometric parameters for a conical parameterization surface are pictured in Figure 5. Its parametric representation is

$$P(s, t) = B + s(\delta(t) + w), \tag{10}$$

where

$$\delta(t) = \tan(\alpha)(\cos(t)u + \sin(t)v).$$

For each t , $\delta(t)$ is a vector of length $\tan(\alpha)$ perpendicular to the cone axis w .

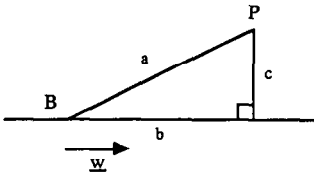


Figure 6

$$a^2 = (P - B) \cdot (P - B),$$

$$b^2 = ((P - B) \cdot w)^2,$$

$$c^2 = a^2 - b^2.$$

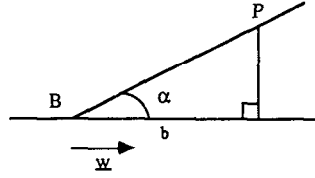


Figure 7

$$b = (P - B) \cdot w \Rightarrow b^2 = ((P - B) \cdot w)^2,$$

$$b = |P - B| \cos \alpha \Rightarrow b^2 = \cos^2 \alpha (P - B) \cdot (P - B).$$

With P being used as the variable position, the implicit equations of the three surfaces of interest can be written as (see Figures 3–5)

Sphere: $(P - B) \cdot (P - B) - r^2 = 0,$ (11)

Cylinder: $(P - B) \cdot (P - B) - ((P - B) \cdot w)^2 - r^2 = 0,$ (12)

Cone: $((P - B) \cdot w)^2 - \cos^2(\alpha)(P - B) \cdot (P - B) = 0,$ (13)

Equation (11) simply states that the square of the distance between P and B must equal the square of the sphere radius. Equation (12) uses the Pythagorean theorem to express that the square of the distance between the point P and the line (B, w) must equal the square of the cylinder radius (Figure 6). Finally, eq. (13) expresses the requirement that the angle between the vectors $(P - B)$ and w must be α (Figure 7).

4. HYBRID APPROACHES

One could develop a hybrid algebraic/geometric approach. This would take one of two forms. The coordinate-system-independent parameterizations of (9) and (10) could be substituted into (1) and manipulated into the form of (6). We would retain the advantage of handling all quadric surfaces in a unified fashion (therefore minimizing the amount of code required), and we would have eliminated the need to transform surfaces into and generate points out of the canonical coordinate system of the parameterization surface. Unfortunately this yields \mathbf{a} , \mathbf{b} , and \mathbf{c} functions, which still cannot be understood geometrically and would therefore still require algebraic analyses for critical point determination. It also turns out to be more expensive to generate points along the curve when it is represented in this manner. Moreover, Wilson's results [15] indicate that this could be the worst choice from a numerical point of view, depending on how curves and surfaces were actually stored in the database.

Alternatively, the canonical coordinate-system parameterizations (4)–(5) could be used in the surface-type-dependent implicit equations (11)–(13). This totally sacrifices the objective of common treatment for all quadrics, however, and once again requires transformations into and out of the canonical coordinate system. Whereas some geometric insight can be gleaned from the resulting \mathbf{a} , \mathbf{b} , and \mathbf{c} functions, it is somewhat obscured because of the need to express the equations

using the x , y , and z point coordinates and vector components instead of the coordinate-system-independent form of (11)–(13).

Of the four possible approaches (purely algebraic, the two forms of hybrids just mentioned, and purely geometric), only the first and last offer benefits outweighing their disadvantages. We therefore conclude that neither of the hybrid approaches merit further consideration.

5. DETECTION OF CONIC SECTION AND ISOLATED POINT INTERSECTIONS

Whereas the thrust of this paper is the development of new methods for the computation of nonplanar quadric surface intersection curves, some remarks are in order relative to the larger environment in which these computations must exist. As noted earlier, when quadric surfaces intersect, either conic sections or nonplanar curves will result. Conic sections occur quite frequently and should be detected and reported as the result whenever possible. There are a variety of reasons for this, including the following:

—It is frequently the case that points and derivatives must be computed along quadric surface intersection curves (e.g., for drawing, numerically intersecting the curve with another, offsetting during tool path generation). If the actual intersection curves are conic sections, these quantities can generally be computed faster and more accurately if the curves are directly represented as conic sections.

—The representation of conic sections is generally more compact than the general form of eq. (6).

—Several graphics devices directly support higher order graphical primitives such as circles and ellipses. Communication with these devices can be simplified, and common local graphics operations, such as zooming, can be performed with more acceptable results if the application knows that it is dealing with conic sections.

—Machine tools often have circular interpolation capability. If portions of a pocket in a part have areas of circular cross section, then the N/C machining interface can exploit that capability if the underlying modeling system has detected that a circular intersection curve resulted from the quadric surface intersection operations used to define the geometry of the pocket.

—These intersections are a critical operation in boundary evaluation in solid modelers. It is important during boundary evaluation to detect when two different pairs of surfaces have intersected in the same curve. The most frequently occurring of such situations involve conic sections. If these curves are detected and stored as conics, this determination can be performed more reliably since (1) the type of the curve is directly captured, (2) the representation is not a function of the intersecting surfaces, and (3) scalar parameters (e.g., radii) can often be copied directly to the generated curve records, thereby minimizing the effect of round-off error on subsequent comparisons.

The system developer is the recipient of another nice side benefit. As we shall see, there are many special cases regarding the parametric form of general quadric surface intersection curves (e.g., situations in which \mathbf{a} and \mathbf{b} are both zero, either

Table I. Sample Configurations Yielding Conic Sections

Surface combination	Geometric configuration	Resulting conic section and/or point
Cylinder-sphere	Sphere center on cylinder axis; identical radii	Tangent circle
	Sphere center on cylinder axis; Sphere radius > cylinder radius	Two-circle intersection
	DCA ^a = sum of radii OR DCA = cylinder_radius - sphere_radius (radii not identical)	Tangent point
Cylinder-cylinder	Axes parallel; DBA ^b = sum of radii OR DBA = max radius - min radius	Tangent line
	Axes parallel; max radius - min radius < DBA < sum of radii	Two-line intersection
	Axes intersect; identical radii (Figure 8)	Two-ellipse intersection
Cylinder-cone	Axes identical	Two-circle intersection ^c
	Angle between axes = cone angle; vertex on cylinder (Figure 9)	Ellipse and tangent line
	Axes intersect; a pair of rulings on each is tangent to the other (Figure 10)	Two-ellipse intersection
Cone-sphere	Sphere center on cone axis; Distance sphere center to vertex < sphere radius	Two-circle intersection ^c
	Sphere center on cone axis; Sphere radius < distance sphere center to vertex < sphere radius/sin(cone angle)	Two-circle intersection ^d
Cone-cone	Axes identical; half-angles different	Two circles
	Shared tangential ruling	Ellipse and tangent line

^a DCA = distance between sphere center and cylinder axis.

^b DBA = distance between axes.

^c One on each half of the double cone.

^d Both on the same half of the double cone.

identically or for a specific value of t). Most of these special cases arise in conjunction with intersections involving conic section curves. If conic sections were simply represented as any other quadric surface intersection curve, the code generating points along the curve would have to be much more complex in order to detect and handle each of these special cases adequately.

Several situations in which spheres, cylinders, and cones intersect in conic sections or isolated points are given in Table I. Some of the configurations are illustrated in Figures 8–10. Table I is not complete; it is simply an attempt to illustrate some of the ways in which conics can appear. Conics arise in quadric surface intersections under a wide variety of conditions, many of which are by no means obvious. A complete characterization of when they do arise and algorithms for computing them are described in [5], where the algebraic approach is exploited as an analytical tool to determine when planes arise in the pencil of two intersecting quadrics.

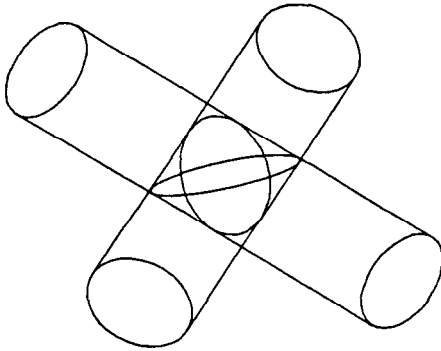


Fig. 8. Cylinder/cylinder \Rightarrow two ellipses.

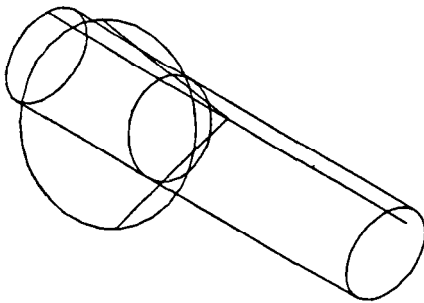


Fig. 9. Cone/cylinder \Rightarrow ellipse + line.

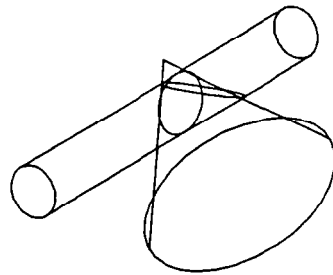


Fig. 10. Cone/cylinder \Rightarrow two ellipses.

6. GEOMETRIC APPROACHES TO QUADRIC SURFACE INTERSECTION CURVE ANALYSIS

In this section we develop geometric techniques for handling intersections involving cylinders, spheres, and cones. Section 6.1 deals with the cylinder as a parameterization surface, and Section 6.2 considers the cone. When a cone and a cylinder are being intersected, either surface may be used as the parameterization surface. As we shall see, the preferred choice depends on the specific cone and cylinder involved. We therefore address cone-cylinder intersections separately in Section 6.3. (We need not consider the plane as a parameterization surface since the only situations in which it can be used yield conic sections, which are separately detected as described in Section 5.)

For each of the possible pairs of surfaces, our analysis will proceed as follows. We begin by substituting the parametric form of the parameterization surface as presented in (9) and (10) into the implicit equations for the other surface (11)–(13). We then manipulate the resulting equation until it is in the form of (6). The resulting functions **a**, **b**, and **c** will have obvious geometric interpretations. For now we note the following:

—**a** basically depends on the relative orientations of the intersecting quadrics. Since spheres have no orientation, they will never influence the form of **a**.

— \mathbf{b} will always be written as $\mathbf{b}(t) = 2\mathbf{b}' \cdot \mathbf{X}(t)$, where $\mathbf{X}(t)$ is a vector (identically the cylinder axis for cylinder–sphere intersections) and \mathbf{b}' is a constant vector that is normal to the surface not being used as the parameterization surface, at least in some geometric configurations. It is often a normal in all but degenerate configurations. For this reason zeros of \mathbf{b} are sometimes important in determining topology and critical points.

— \mathbf{c} is the implicit equation of the other surface evaluated at $\gamma(t)$ (when the parameterization surface is the cylinder) or the cone vertex (when the parameterization surface is the cone). This is the most concrete of the three general classifications. The expressions for \mathbf{c} in (7) and (8) can easily be interpreted in this fashion. Since $\gamma(t)$ is the circle on the parameterization cylinder at $s = 0$, we can substitute (4) (with $z = 0$) into (1) and obtain $\mathbf{c}(t)$ as in (7). The vertex of the canonical cone described in (5) is $(x = 0, y = 0, z = 0)$. Plugging this into (1) yields K , the value of \mathbf{c} shown in (8).

Under certain conditions, important special cases of the \mathbf{a} , \mathbf{b} , and \mathbf{c} functions arise. For example, the intersection curve is linear in s if one of \mathbf{a} or \mathbf{c} is identically zero. The situations giving rise to these special cases are discussed in the context of the geometric interpretations behind the \mathbf{a} , \mathbf{b} , and \mathbf{c} functions for the specific pair of surfaces.

Finally, the analysis for a given pair of surfaces is concluded by showing how the intersection curve topology and critical points can be determined from a geometric analysis of the surfaces involved without the need to solve polynomials of degree higher than 2. There are three possible intersection curve topologies: a single closed branch, a figure eight, or two closed branches (see Figure 11). In cone–cylinder and cone–cone intersections, the branches can be open, tending to infinity at the ends. In Sections 6.2.2 and 6.3.3 we shall see how to handle these situations properly.

To distinguish the scalars, points, and vectors defining the parameterization surface from those defining the other surface, a “p” subscript is used for the parameterization surface and an “o” subscript for the other surface.

6.1 Using the Cylinder as the Parameterization Surface

6.1.1 *Cylinder–Sphere*. Substituting (9) into (11) results in

$$(\gamma(t) - B_o + s\mathbf{w}_p) \cdot (\gamma(t) - B_o + s\mathbf{w}_p) - r_o^2 = 0.$$

Rearranging into the form of (6), we find

$$\begin{aligned} \mathbf{a} &= 1, \\ \mathbf{b} &= 2\mathbf{b}' \cdot \mathbf{w}_p, \\ \mathbf{b}' &= B_p - B_o, \\ \mathbf{c}(t) &= (\gamma(t) - B_o) \cdot (\gamma(t) - B_o) - r_o^2. \end{aligned} \tag{14}$$

The curve can be linear in s only if $\mathbf{c}(t)$ is identically zero. Since $\mathbf{c}(t)$ is the implicit equation of the sphere evaluated at $\gamma(t)$, it can be identically zero only

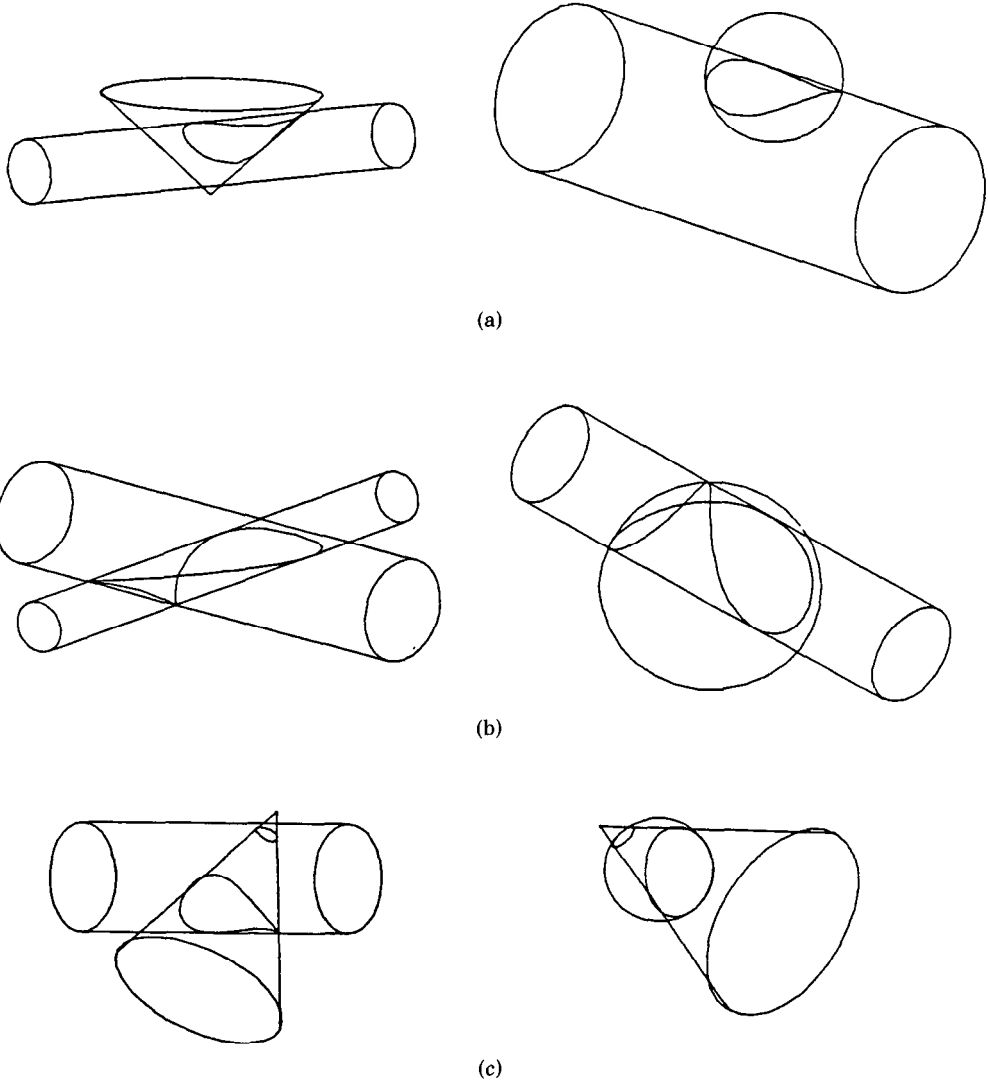


Fig. 11. (a) One-branch intersections. (b) Figure-eight intersections. (c) Two-branch intersections.

if the entire circle defined by $\gamma(t)$ is on the sphere. This is only possible if the center of the sphere is on the axis of the cylinder, but that would have resulted in a conic section intersection which would have been detected and handled earlier. As long as B_p and B_o are distinct points, \mathbf{b}' will always be a normal to the sphere at B_p (or at the perpendicular projection of B_p onto the sphere).

Once we have determined that the cylinder and sphere intersect in a space curve, we compute the distance (d) between the sphere center and the cylinder

axis and proceed as follows:

```

if  $r_o > r_p$  then
  if  $d = (r_o - r_p)$  then
    figure-eight intersection using all of cylinder parameter
    space (Figure 11b)
  else if  $d < (r_o - r_p)$  then
    two disjoint intersection curve branches using all of
    cylinder parameter space
  else
    single intersection curve branch;
    determine critical points (see below)
else
  single intersection curve branch;
  determine critical points (see below)

```

When critical points must be determined, there will be two of them, and they correspond to the two cylinder rulings that are tangent to the sphere. We can determine these t values by considering a plane containing the sphere center that is perpendicular to the cylinder axis. This plane will cut the cylinder and sphere in the two circles illustrated in Figure 12. A standard circle-circle intersection yields the points P_1 and P_2 . We compute the values of t corresponding to these points, and we are done.

6.1.2 *Cylinder-Cylinder*. Substituting (9) into (12) results in

$$(\gamma(t) - B_o + s\mathbf{w}_p) \cdot (\gamma(t) - B_o + s\mathbf{w}_p) - ((\gamma(t) - B_o + s\mathbf{w}_p) \cdot \mathbf{w}_o)^2 - r_o^2 = 0.$$

From this we obtain

$$\begin{aligned}
 \mathbf{a} &= 1 - (\mathbf{w}_p \cdot \mathbf{w}_o)^2, \\
 \mathbf{b}(t) &= 2\mathbf{b}' \cdot (\gamma(t) - B_o), \\
 \mathbf{b}' &= \mathbf{w}_p - (\mathbf{w}_p \cdot \mathbf{w}_o)\mathbf{w}_o, \\
 \mathbf{c}(t) &= (\gamma(t) - B_o) \cdot (\gamma(t) - B_o) - ((\gamma(t) - B_o) \cdot \mathbf{w}_o)^2 - r_o^2.
 \end{aligned} \tag{15}$$

If $\mathbf{w}_p = \pm\mathbf{w}_o$, then \mathbf{a} , \mathbf{b}' , and \mathbf{b} are identically zero, and we are left with a quadratic equation in t (i.e., $\mathbf{c}(t) = 0$). Geometrically, this means that the two cylinders are parallel, in which case we know that they are either identical, disjoint, or intersect in one or two lines. The lines, if any, correspond to the solutions of $\mathbf{c}(t) = 0$. If $\mathbf{w}_p \neq \pm\mathbf{w}_o$, \mathbf{a} is positive, and no difficulties could arise during curve evaluation.

When nonzero, \mathbf{b}' is a normal to the cylinder along rulings contained in the plane containing \mathbf{w}_o and \mathbf{w}_p . The function $\mathbf{c}(t)$ can be identically zero (independent of \mathbf{a} and \mathbf{b}) only if the circle defined by $\gamma(t)$ lies entirely on the other cylinder, that is, only if the cylinders are identical.

Note that either cylinder can be used as the parameterization surface. Two branch and figure-eight intersections are the only situations for which there will be a significant difference. In both of these cases, the radii of the two cylinders must be different or a two-ellipse intersection would have resulted. If the parameterization cylinder is the larger of the two, then three (or four) critical points exist on the parameterization cylinder, depending on whether the topology is a figure-eight (or two-branch) intersection. The two values of s corresponding

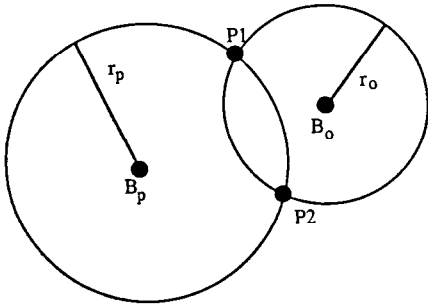


Fig. 12. Cutting a cylinder and a sphere with a plane perpendicular to the cylinder axis.

to a given t yield two points on the same branch. If the parameterization cylinder is the smaller of the two, then the only critical point is the tangent point of a figure-eight intersection, and the two values of s corresponding to a given t yield two points on opposite branches. Using the smaller cylinder is therefore easier initially for generating the representation of the intersection curve (all of its parameter space is used), but the larger cylinder will likely lead to more efficient point generation algorithms, since each t will yield two points on the same intersection curve branch.

As we did with cylinder–sphere intersections, we consider the two radii and the distance between the two axes to determine the topology. Critical point determination is slightly more complex, however. Consider a plane containing the axis of the cylinder not being used as the parameterization surface and the vector perpendicular to the two axes. (Recall that we know that the axes are not parallel.) This plane cuts the cylinder not being used as the parameterization surface into two lines. These lines will either fail to intersect the parameterization cylinder, or they will intersect it (either tangentially or in two distinct points). If one or both of the lines intersect the parameterization surface, the t values for the intersection point(s) are the desired critical points. If neither line intersects, then we have a two-branch intersection with the smaller cylinder as the parameterization surface. No critical points need to be determined in this case, since all of its parameter space corresponds to positions on the curve. The way in which the two line–cylinder intersection results indicate the critical points is summarized in Table II.

6.2 Using the Cone as the Parameterization Surface

Recall that when the cone is used as the parameterization surface and its vertex is on the other surface, the parameterization is linear in s (because $\mathbf{c} = 0$). Partly for this reason, our analysis of intersections involving cones is based on considering, in turn, the three possible relative vertex locations: on, inside, or outside the other surface.

It is often important to determine when $\mathbf{b}(t)$ is zero. Since $\mathbf{b}(t)$ is equivalent to a rational quadratic, the zeros could simply be determined by solving the equivalent polynomial. There is a geometric approach, however. Whenever the cone is the parameterization surface, $\mathbf{b}(t)$ is of the form $(\mathbf{b}(t) = 2\mathbf{b}' \cdot (\delta(t) + \mathbf{w}_p))$. If \mathbf{b}' is the zero vector, $\mathbf{b}(t)$ will be identically zero. Otherwise, $\mathbf{b}(t)$ will be

Table II. Determination of Critical Points for Cylinder-Cylinder Intersections

Line 1	Line 2		
	Do not intersect	Tangent	Two-point intersection
Do not intersect	Two-branch intersection; all of parameterization cylinder t values used ^a	Figure-eight intersection; all of parameterization cylinder t values used; tangent point yields t value of figure-eight cross point ^a	One-branch intersection; intersection points yield starting/ending t values on parameterization cylinder ^b
Tangent		Impossible; this would have yielded a two-ellipse intersection	Figure-eight intersection; Line 2 intersections yield starting/ending t values for curve on parameterization cylinder; tangent point yields t value of figure-eight cross point ^c
Two-point intersection			Two-branch intersection; first branch is between t_1 and t_2 , second is between t_3 and t_4 . (t_1 and t_2 are t values for Line 1; t_3 and t_4 are for Line 2) ^c

^a Parameterization cylinder radius < other cylinder radius.

^b Either radius may be larger than or equal to the other.

^c Other cylinder radius < parameterization cylinder radius.

zero at values of t corresponding to cone rulings that are perpendicular to \mathbf{b}' . These rulings, if any, can be located directly by intersecting the cone with the plane (B_p , \mathbf{b}'). Zero, one, or two lines will result, the t values of which will be the zeros of $\mathbf{b}(t)$.

In mechanical CAD/CAM, one is normally only interested in one-half of the double cone defined by the geometric parameters. (We assume this half is the one in the direction of \mathbf{w} .) Since the mathematical representations describe both halves, extra work is required to describe only those portions of intersections with the appropriate half. There are two ways that this could be done. The intersection curve could be generated in its entirety and later intersected with (trimmed to) the plane (B , \mathbf{w}). Since curve-surface intersections are a common operation in boundary evaluation anyway, this is certainly a viable approach. Alternatively, extra work could be expended in the initial intersection operation to throw away intersection curve branches or portions thereof that do not lie on the desired half of the cone(s).

Methods for the former approach are not the subject of this paper, so they will not be considered further. We describe methods for the latter approach in the remainder of this paper, but do not necessarily advocate this approach in practice. The extra work generally consists of operations performed after critical points have been determined and is therefore easily ignored if the former approach is to be used in a given implementation.

6.2.1 *Cone-Sphere*. Substituting eq. (10) into (11) we obtain

$$((B_p - B_o) + s(\delta(t) + w_p)) \cdot ((B_p - B_o) + s(\delta(t) + w_p)) - r_o^2 = 0.$$

Since $\delta(t)$ is a vector of length $\tan(\alpha_p)$, $\delta(t) \cdot \delta(t) = \tan^2(\alpha_p)$ and we obtain

$$\begin{aligned} \mathbf{a} &= \tan^2(\alpha_p) + 1, \\ \mathbf{b}(t) &= 2\mathbf{b}' \cdot (\delta(t) + w_p), \\ \mathbf{b}' &= B_p - B_o, \\ \mathbf{c} &= (B_p - B_o) \cdot (B_p - B_o) - r_o^2. \end{aligned} \tag{16}$$

As with cylinder-sphere intersections, \mathbf{a} is never zero. $\mathbf{b}(t)$ is identically zero only if the vertex of the cone and the center of the sphere are coincident. This yields a pair of circles (one on each half of the double cone) and is therefore not of interest to us here. The \mathbf{c} term is, of course, identically zero if the cone vertex is on the sphere. This yields a linear parameterization and is the only special case of (16) we need consider. As long as B_p and B_o are distinct points, \mathbf{b}' will always be a normal to the sphere at B_p (or at the perpendicular projection of B_p onto the sphere).

6.2.1.1 *Cone Vertex on Sphere*. The parameterization is linear in s , and all of parameter space will be used to trace out the curve. We shall exploit the fact that \mathbf{b}' is the normal to the sphere at the cone vertex to determine the intersection curve topology and whether the curve passes through the vertex. Note that $\mathbf{b}(t_o) = 0$ implies that the ruling defined by $t = t_o$ is tangent to the sphere at the cone vertex. Since $\mathbf{b}(t)$ is equivalent to a rational quadratic, there can be at most two zeros of \mathbf{b} . If there are no \mathbf{b} zeros, the intersection curve is disjoint from the vertex and of course lies entirely on one-half of the cone, as in Figure 13. If there is one \mathbf{b} zero, then the curve passes through the vertex once, but remains on one-half of the cone, as in Figure 14. If there are two zeros, then the curve passes through the vertex twice and is a figure eight with one loop on each cone half, as in Figure 15.

6.2.1.2 *Cone Vertex inside Sphere*. This is the simplest cone-sphere intersection scenario. The sphere intersects both halves of the double cone, and all of the cone parameter space is used to sweep out the intersection. It is straightforward to return the branch on the desired half of the cone, that is, the one with positive values of s .

6.2.1.3 *Cone Vertex outside Sphere*. The analysis for this case proceeds by considering the relationships among three angles (see Figure 16):

- $0 < \alpha < 90$: cone angle,
- $0 < \theta < 90$: in the plane of the cone axis and sphere center, an angle between a line from the vertex tangent to the sphere and a line from the vertex through the sphere center,
- $0 < \phi < 180$: angle from the cone axis to the line from the vertex through the sphere center.

Fig. 13. No **b** zeros \Rightarrow no tangent rulings.

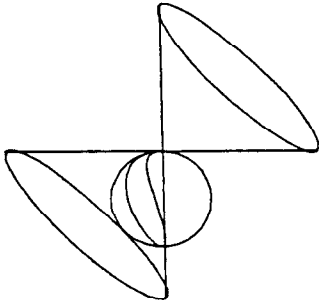
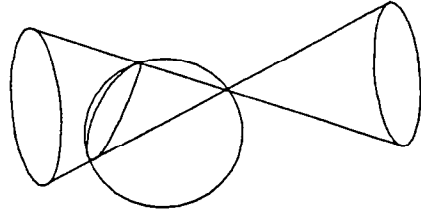


Fig. 14. One **b** zero \Rightarrow one tangent ruling.

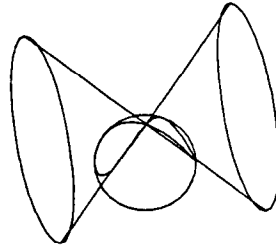


Fig. 15. Two **b** zeros \Rightarrow two tangent rulings.

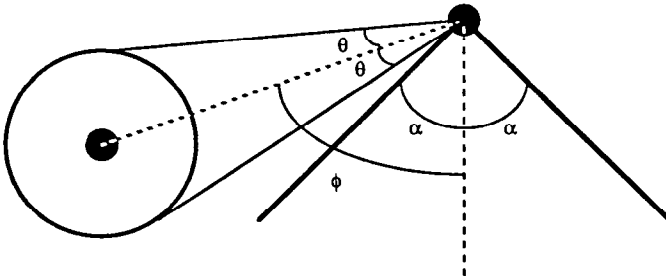


Fig. 16. View of sphere and cone in plane of cone axis and sphere center.

There are five major cases, one of which has three subcases:

$\phi = \alpha + \theta$: There is a single tangent point of intersection; the sphere is outside the cone.

$\phi = \alpha - \theta$: There is a single tangent point of intersection; the sphere is inside the cone.

$\phi > \alpha + \theta$: The sphere and cone do not intersect; the sphere is outside the cone.

$\phi < \alpha - \theta$: The sphere and cone do not intersect; the sphere is inside the cone.

Otherwise: The sphere and cone intersect in a space curve. The exact nature of the intersection is determined by

$\theta = \alpha + \phi$: figure-eight intersection occupying all of the cone parameter space (see Figure 17);

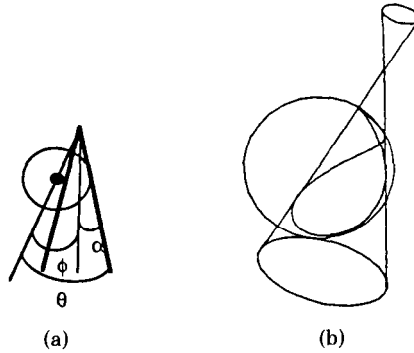


Fig. 17. (a) $\theta = \alpha + \phi$. (b) Result.

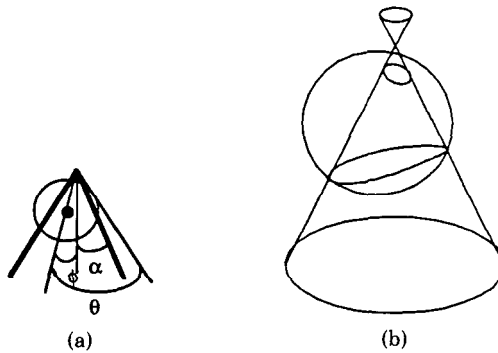


Fig. 18. (a) $\theta > \alpha + \phi$. (b) Result.

- $\theta > \alpha + \phi$: two-branch intersection occupying all of the cone parameter space (see Figure 18);
- $\theta < \alpha + \phi$: one-branch intersection or the sphere intersects both halves of the double cone (see Figure 19). This is the only situation for which limits of the cone parameter space are needed. These critical points correspond to the values of t defining cone rulings tangent to the sphere. The point of tangency, the sphere center, and the cone vertex define a right triangle from which it is straightforward to determine the distance from the vertex to the tangent point along a ruling. This distance determines a pair of circles (one on each half of the cone) whose intersection with the sphere yields the desired points.

6.2.2 *Cone-Cone*. When two cones are intersected, either can be used as the parameterization surface. If the vertex of one is on the other, we use it since the curve parameterization will be linear in s . If neither vertex lies on the other cone, but the vertex of one is inside the other, we choose it as the parameterization

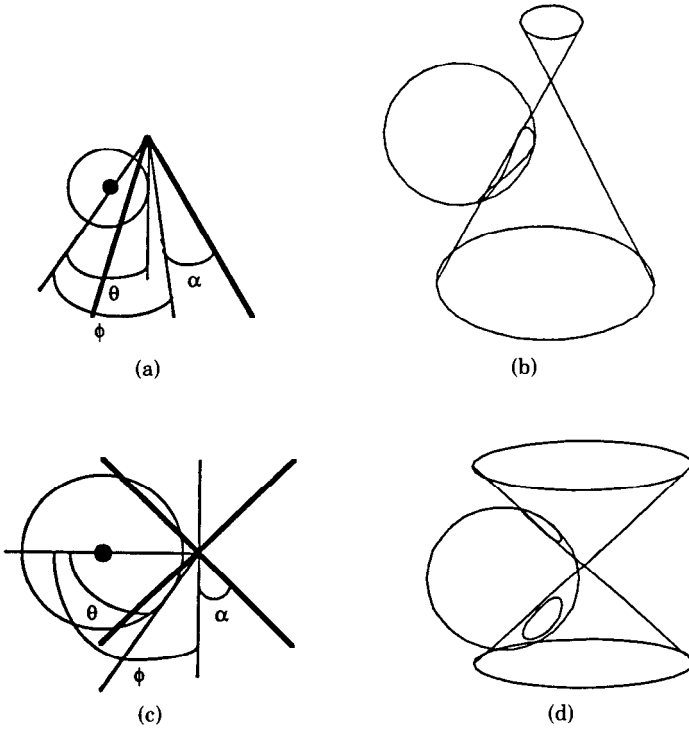


Fig. 19. (a) $\theta < \alpha + \phi$. (b) Result. (c) $\theta < \alpha + \phi$. (d) Result.

surface since critical point determination is easier. Finally, if neither vertex lies on or in the other cone, we arbitrarily select one to be the parameterization surface.

We derive the general parametric form of the cone-cone intersection curve by substituting (10) into (13) to obtain

$$(((B_p - B_o) + s(\delta(t) + w_p)) \cdot w_o)^2 - \cos^2(\alpha_o)((B_p - B_o) + s(\delta(t) + w_p)) \cdot ((B_p - B_o) + s(\delta(t) + w_p)) = 0.$$

From this we determine

$$\begin{aligned} \mathbf{a}(t) &= ((\delta(t) + w_p) \cdot w_o)^2 - \cos^2(\alpha_o)(\tan^2(\alpha_p) + 1) \\ &= ((\delta(t) + w_p) \cdot w_o)^2 - \frac{\cos^2(\alpha_o)}{\cos^2(\alpha_p)}, \\ \mathbf{b}(t) &= 2\mathbf{b}' \cdot (\delta(t) + w_p), \\ \mathbf{b}' &= ((B_p - B_o) \cdot w_o)w_o - \cos^2(\alpha_o)(B_p - B_o), \\ \mathbf{c} &= ((B_p - B_o) \cdot w_o)^2 - \cos^2(\alpha_o)(B_p - B_o) \cdot (B_p - B_o). \end{aligned} \tag{17}$$

$\mathbf{a}(t)$ is identically zero if and only if the axis vectors are parallel and the angles are the same; that is, the cones are identical except for position. This will yield a conic section [5], so we do not consider it further here. $\mathbf{a}(t)$ is equivalent to a rational quartic; there can therefore be up to four zeros. At these zeros, the

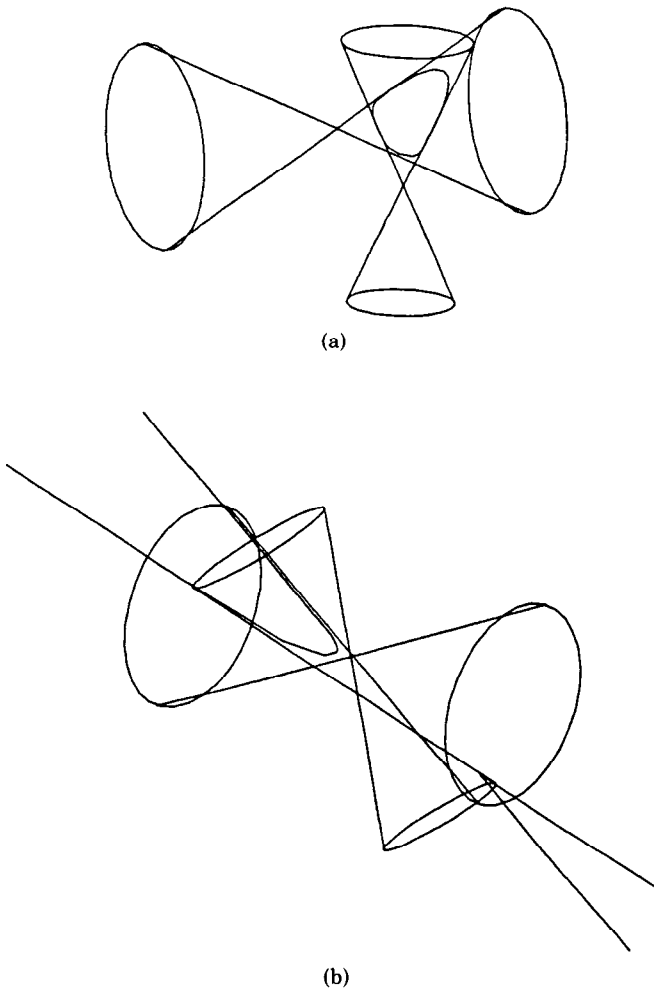


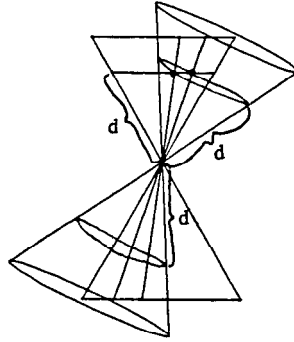
Figure 20

intersection curve transitions from infinity on one half to infinity on the other half of both cones. These zeros, therefore, subdivide the normal one-branch, two-branch, or figure-eight intersections into pieces which connect at infinity (see Figure 20). Geometrically, the zeros of \mathbf{a} correspond to parameterization cone rulings that intersect the other cone only once. Of course, it is important to find these zeros during the course of intersecting the cones, and we shall see shortly how these zeros can be located geometrically without solving a fourth-degree polynomial equation.

$\mathbf{b}(t)$ is identically zero if and only if the cone vertices are identical. In this case, the intersection is just a point or a collection of up to four rulings. We are therefore not interested in this case per se; however, we shall return to it shortly in the context of locating the zeros of $\mathbf{a}(t)$.

It can be shown that, if B_p is on the other cone, \mathbf{b}' is parallel to the cone normal along the ruling containing B_p . Parameterization cone rulings tangent to

Figure 21



(or contained by) the other cone therefore force $\mathbf{b}(t)$ to zero when B_p is on the other cone. When B_p is not on the other cone, \mathbf{b}' is merely a vector in the plane containing B_p , B_o , and \mathbf{w}_o , but it is not normal to the cone.

As usual, \mathbf{c} is zero if and only if B_p is on the other cone. Note that $\mathbf{a}(t)$ and $\mathbf{b}(t)$ can go to zero simultaneously only if \mathbf{c} is zero. We shall see in Section 6.2.2.1 that this happens at the t corresponding to a shared ruling between the cones.

We again proceed by considering the three possible relative locations of the parameterization cone vertex: on, inside, and outside the other cone. In all three cases, we need to know where $\mathbf{a}(t)$ is zero. We can locate these points geometrically as follows. On examining (17), we observe that, if the cones share a vertex (i.e., $B_p = B_o$), then $\mathbf{b}(t)$ is identically zero and \mathbf{c} is zero. Equation (6) describing the quadric surface intersection curve then becomes $(\mathbf{a}(t)s^2 = 0)$. This is obviously satisfied when $s = 0$, but it is also satisfied for rulings defined by values of t at which $\mathbf{a}(t) = 0$. Therefore, finding the zeros of \mathbf{a} is equivalent to finding the lines of intersection of the two cones $(B_p, \mathbf{w}_p, \alpha_p)$ and $(B_p, \mathbf{w}_o, \alpha_o)$. We can find these lines geometrically in the following manner: Construct a circle on the first cone at some distance d along a ruling, as in Figure 21. Each line of intersection between the cones must intersect this circle exactly once. The intersection points will, of course, also lie on the other cone and must be at the same distance d from the vertex. This suggests the construction of two circles on the other cone, each at a distance d from the vertex along a ruling, one on either side of the vertex (Figure 21). Each of these circles can intersect the original circle up to two times. The resulting points (a maximum of four) can be used along with the vertex to find the intersection lines (equivalently the t values satisfying $\mathbf{a}(t) = 0$).

6.2.2.1 Parameterization Cone Vertex on Other Cone. When the parameterization cone vertex is on the other cone, the intersection curve parameterization is, of course, linear in s . There are two primary subcases to consider on the basis of whether the vertex of the other cone is also on the parameterization cone. If it is, the cones share a ruling and also intersect in another curve that passes through the shared ruling. If the ruling is shared tangentially, the other curve is an ellipse [5], so we do not consider that case further here. If it is not shared tangentially, then the other curve is a space cubic. We now consider the original two subcases in turn, assuming for the former that the ruling is not shared tangentially.

When the vertex of the other cone is on the parameterization cone, there will be at least two zeros of $\mathbf{a}(t)$.¹ One will be at the shared ruling (t_1), and the remainder will be at places where the space cubic goes to infinity. $\mathbf{b}(t)$ will have precisely two zeros: one at t_1 and one where the space cubic passes through the vertex of the parameterization cone (t_2). All of parameter space will be used to trace out the curve, with the exception of the \mathbf{a} zeros not equal to t_1 . These zeros, therefore, define intervals on the intersection curve. To restrict the intersection curve to the positive half of both cones, we consider an expanded set of intervals defined by

- the \mathbf{a} zeros not equal to t_1 ,
- t_2 ,
- the t value of the other cone's vertex on the parameterization cone.

Each of these intervals will be entirely on one or the other half of both cones.

When the vertex of the other cone is not on the parameterization cone, we have a situation analogous to that described in Section 6.2.1.1. That is, the basic type of the intersection curve is determined by the number of \mathbf{b} zeros. If there are none, then the intersection curve is totally disjoint from the parameterization cone vertex. If there is one, then the curve passes through B_p once, but stays on one half of the cone in the neighborhood of the vertex. If there are two, then the curve passes through the vertex twice and is a figure eight with one loop on each cone half.

We next consider the zeros of \mathbf{a} . If there are none, then all of parameter space is used to trace the curve, and we need only test a single point on the curve (two for a figure eight) to determine what portion of the curve lies on the positive half of both cones. If there are \mathbf{a} zeros, then these zeros cause the intersection curve (or branches of a figure eight) to be split across both halves of both cones, connecting at infinity. The zeros therefore define intervals, as described earlier, that are entirely on one or the other half of both cones.

6.2.2.2 Parameterization Cone Vertex inside Other Cone. In this case it is not important where the vertex of the other cone is with respect to the parameterization cone. (We do assume, however, that it is not on the parameterization cone or we would be in the case covered by Section 6.2.2.1.) This case is individually considered, since we know a priori that all of the parameter space will be used to sweep out the curve (with the exception of the \mathbf{a} zeros, if any); so the generation of the result is relatively easy. The intersection will basically be two disjoint branches. If there are no \mathbf{a} zeros, all that needs to be done is to test a point on each branch to see whether it lies on the positive half of both cones (Figure 22a and b).

If there are \mathbf{a} zeros, then one or both branches will again be split across both halves of both cones, connecting at infinity (Figure 22c). Every parameterization cone ruling will intersect the other cone at least once. There can be up to four

¹ Note that $\mathbf{a}(t)$ does not depend on vertex positioning. Here we are dealing with the case in which *both* vertices are on the other cone, however. This imposes a constraint on relative orientation (two rotational degrees of freedom are lost), and $\mathbf{a}(t)$ does depend on the relative orientation of the cones.

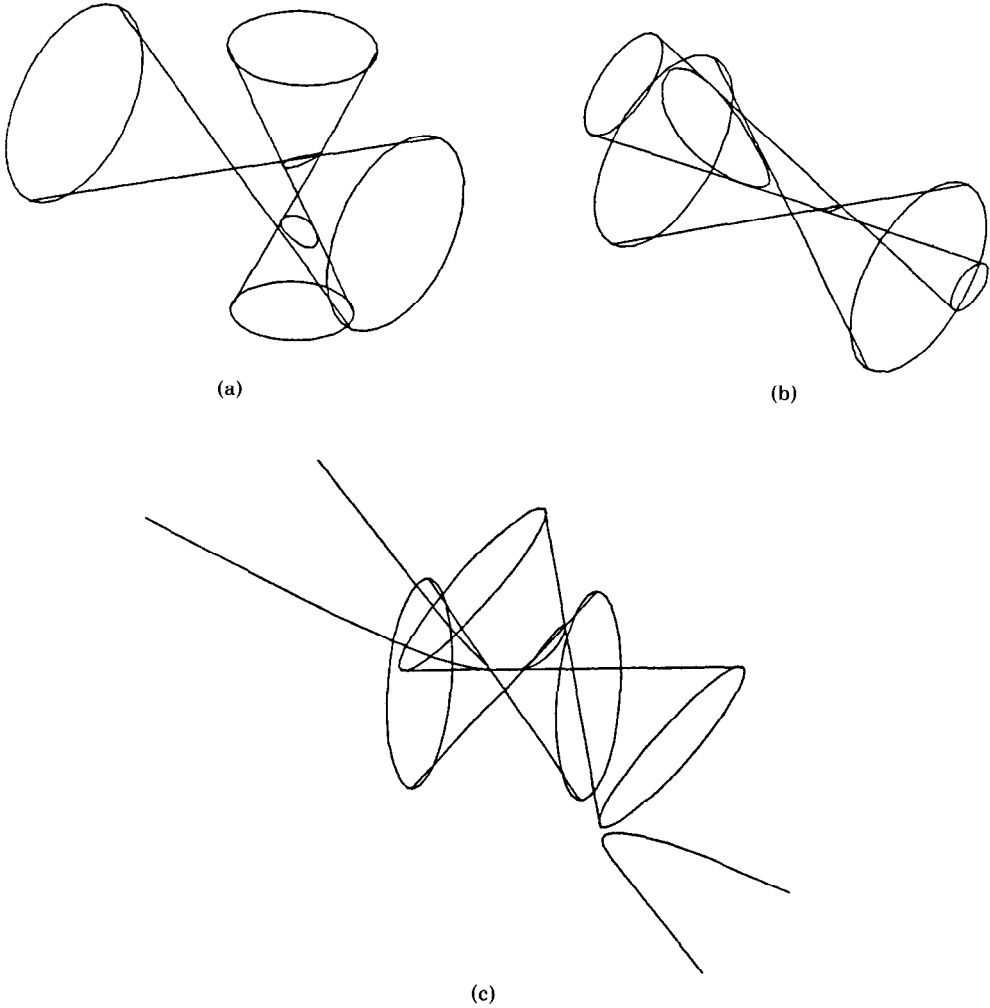


Fig. 22. (a) No \mathbf{a} zeros. (b) No \mathbf{a} zeros. (c) Two \mathbf{a} zeros.

rulings that *only* intersect once; these correspond to the zeros of \mathbf{a} . Each parameterization cone ruling that intersects the other cone twice is intersecting both intersection curve branches; that is, it determines a point on each branch. The four or fewer rulings that only intersect once determine a point on one of the branches and are asymptotic to the other. Intervals are therefore defined by pairs of \mathbf{a} zeros at which the sign of \mathbf{b} is the same. The rationale for this is as follows. \mathbf{a} and \mathbf{b} cannot be simultaneously zero since $\mathbf{c} \neq 0$ (recall (6)). For a given value of t , we obtain s from the quadratic formula as

$$s = -\mathbf{b}(t) \pm \frac{\sqrt{(\mathbf{b}(t))^2 - 4\mathbf{a}(t)\mathbf{c}}}{2\mathbf{a}(t)}. \quad (18)$$

Suppose that $\mathbf{a}(t_0) = 0$ and $\mathbf{b}(t_0) > 0$. The two values of s are $s = 0/0$ and $s = -2\mathbf{b}(t_0)/0$. The latter is the root for the branch that has gone to infinity, and

the former corresponds to the single intersection point of the parameterization cone ruling with the other cone, which can be determined as $s = -\mathbf{c}/\mathbf{b}(t_0)$. Once the intervals have been determined, the branches are defined by an interval along with a \pm flag indicating which sign to use when evaluating s from (18) along the interval. Each branch can then be tested for being on the positive half of both cones.

6.2.2.3 Both Vertices outside Other Cone. Unlike the previous two cases, in this case there may be intervals of parameter space not used to sweep out the intersection curve. Geometrically, there will be sets of parameterization cone rulings that do not intersect the other cone at all. Algebraically, the discriminant $(\mathbf{b}(t))^2 - 4\mathbf{a}(t)\mathbf{c}$, which was always nonnegative in the previous two cases, may be negative over ranges of t here. Critical points are therefore determined not only from the zeros of \mathbf{a} , but also from the zeros of the discriminant. The discriminant is again equivalent to a rational quartic, but we can avoid solving the quartic algebraically by considering its geometric significance.

A parameterization cone ruling defined by $t = t_0$ will either fail to intersect the other cone, intersect it once nontangentially, intersect it once tangentially, or intersect it twice. (By the assumptions of this section, it cannot be contained by the other cone.) If it does not intersect, the discriminant is negative. If it intersects once nontangentially, the angle between the line direction vector and the cone axis vector must be the cone angle, which means that $\mathbf{a}(t_0) = 0$. Intersecting once tangentially implies there is a double root at the intersection point; therefore the discriminant is zero. Finally, if the ruling intersects twice, the discriminant must be strictly greater than zero.

Since tangent rulings correspond to discriminant zeros, we can find the zeros of the discriminant geometrically by finding the tangent parameterization cone rulings, if any. We parametrically define a tangent plane to the cone not being used as the parameterization surface as follows. The plane base point is B_0 . To determine the plane normal, we recall that $(\delta(t) + \mathbf{w}_0)$ is a vector along a ruling where it is understood that $\delta(t)$ here involves \mathbf{u}_0 , \mathbf{v}_0 , and α_0 . The derivative of $(\delta(t) + \mathbf{w}_0)$ with respect to t yields a tangent vector along the ruling. If we cross the vector along the ruling with the tangent vector, we get a normal to the cone as a function of t , which we shall use as the tangent plane normal:

$$\begin{aligned} \mathbf{n}(t) &= (\delta(t) + \mathbf{w}_0) \times \frac{d}{dt} (\delta(t) + \mathbf{w}_0) \\ &= (\tan(\alpha_0)(\cos(t)\mathbf{u}_0 + \sin(t)\mathbf{v}_0) + \mathbf{w}_0) \times (\tan(\alpha_0)(-\sin(t)\mathbf{u}_0 + \cos(t)\mathbf{v}_0)) \\ &= \tan^2(\alpha_0)\mathbf{w}_0 - (\tan(\alpha_0)\mathbf{u}_0 \cos(t) + \tan(\alpha_0)\mathbf{v}_0 \sin(t)). \end{aligned}$$

This can be expressed as the equivalent rational quadratic

$$\mathbf{n}(r) = \tan^2(\alpha_0)\mathbf{w}_0 - \left(\tan(\alpha_0)\mathbf{u}_0 \left[\frac{1 - r^2}{1 + r^2} \right] + \tan(\alpha_0)\mathbf{v}_0 \left[\frac{2r}{1 + r^2} \right] \right).$$

Any line tangent to the other cone must be contained in a plane $(B_0, \mathbf{n}(r))$ for some r . Since we are looking for parameterization cone rulings tangent to the other cone, we seek tangent planes containing B_p . That is, we solve

$$(B_p - B_0) \cdot \mathbf{n}(r) = 0.$$

This is a quadratic equation that will yield 0, 1, or 2 planes tangent to the other cone containing B_p . If n is the number of planes determined and $p[]$ is the array of planes, we can proceed as follows to compute the m discriminant zeros and store them in an array $d[]$:

```

m := 0
for i := 1 to n do
  intersect p[i] with the parameterization cone
  if intersection is a single point (i.e., the vertex) then
    ignore plane p[i]
  else { intersection must be one or two lines }
    for each intersection line, L do
      m := m + 1
      d[m] := t value of L on parameterization cone
    end for each
end for

```

Intersection curve branches will lie between the discriminant zeros; these zeros therefore define intervals that will be further subdivided by \mathbf{a} zeros, if any. Intervals or subintervals with negative discriminants are discarded. The remaining ones can then be checked for being on the positive half of both cones.

6.3 Intersecting a Cylinder and a Cone

In cylinder-cone intersections, either the cylinder or the cone can be used as the parameterization surface. As we shall see, the “better” choice (in terms of computational efficiency, one parameterization or the other being linear, or ease of analysis) depends on their relative position and orientation. We shall therefore decide which surface to use as the intersection operation proceeds. This section begins by developing the general form of the intersection curve parameterization using first the cylinder, then the cone as the parameterization surface. Then we examine an approach to performing cone-cylinder intersections that includes selection of the better parameterization surface.

6.3.1 Using the Cylinder as the Parameterization Surface. We derive the general parametric form of the cone-cylinder intersection curve using the cylinder as the parameterization surface by substituting (9) into (13). This results in

$$\begin{aligned}
 & ((\gamma(t) - B_o + s\mathbf{w}_p) \cdot \mathbf{w}_o)^2 - \cos^2(\alpha_o)(\gamma(t) - B_o + s\mathbf{w}_p) \\
 & \cdot (\gamma(t) - B_o + s\mathbf{w}_p) = 0.
 \end{aligned}$$

From this we find

$$\begin{aligned}
 \mathbf{a} &= (\mathbf{w}_p \cdot \mathbf{w}_o)^2 - \cos^2(\alpha_o), \\
 \mathbf{b}(t) &= 2\mathbf{b}' \cdot (\gamma(t) - B_o), \\
 \mathbf{b}' &= (\mathbf{w}_p \cdot \mathbf{w}_o)\mathbf{w}_o - \cos^2(\alpha_o)\mathbf{w}_p, \\
 \mathbf{c}(t) &= ((\gamma(t) - B_o) \cdot \mathbf{w}_o)^2 - \cos^2(\alpha_o)(\gamma(t) - B_o) \cdot (\gamma(t) - B_o).
 \end{aligned} \tag{19}$$

\mathbf{a} will be zero if and only if the angle between the axes is the same as the cone angle. It is only possible for $\mathbf{b}(t)$ to be identically zero if B_o is in the plane of $\gamma(t)$, and \mathbf{b}' is perpendicular to this plane (i.e., parallel to \mathbf{w}_p). This is only possible if \mathbf{w}_o and \mathbf{w}_p are either parallel or perpendicular, but \mathbf{a} cannot be zero in either case. Therefore, no special cases arise when $\mathbf{b}(t)$ is identically zero. Note,

however, that when \mathbf{a} is zero and the cone vertex is on the cylinder, there will be a t_0 such that $\mathbf{b}(t_0) = \mathbf{c}(t_0) = 0$. This is also observed in Section 6.3.2 and discussed in Section 6.3.3.

It can be shown that if \mathbf{w}_p is parallel to a cone ruling, then \mathbf{b}' is the normal to the cone along that ruling. Otherwise, \mathbf{b}' is not a normal to the cone.

Since $\mathbf{c}(t)$ is the cone implicit equation evaluated at $\gamma(t)$, it can be identically zero only if the whole circle defined by $\gamma(t)$ is on the cone. This would have been detected earlier in the intersection operation, since the cone–cylinder intersection would have been a pair of circles, one on each half of the cone.

6.3.2 Using the Cone as the Parameterization Surface. To derive the parametric form of the intersection curve using the cone as the parameterization surface, we substitute (10) into (12) to obtain

$$\begin{aligned} & ((B_p - B_o) + s(\delta(t) + \mathbf{w}_p)) \cdot ((B_p - B_o) + s(\delta(t) + \mathbf{w}_p)) \\ & - (((B_p - B_o) + s(\delta(t) + \mathbf{w}_p)) \cdot \mathbf{w}_o)^2 - r_o^2 = 0. \end{aligned}$$

From this we find

$$\begin{aligned} \mathbf{a}(t) &= \tan^2(\alpha_p) + 1 - (\mathbf{w}_o \cdot (\delta(t) + \mathbf{w}_p))^2, \\ \mathbf{b}(t) &= 2\mathbf{b}' \cdot (\delta(t) + \mathbf{w}_p), \\ \mathbf{b}' &= (B_p - B_o) - ((B_p - B_o) \cdot \mathbf{w}_o)\mathbf{w}_o, \\ \mathbf{c} &= (B_p - B_o) \cdot (B_p - B_o) - ((B_p - B_o) \cdot \mathbf{w}_o)^2 - r_o^2. \end{aligned} \tag{20}$$

$\mathbf{a}(t)$ is nonnegative for all t and *could* only be identically zero if $(\mathbf{w}_o \cdot (\delta(t) + \mathbf{w}_p))^2$ were identically $(\tan^2(\alpha_p) + 1)$. The only vectors whose dot product with $(\delta(t) + \mathbf{w}_p)$ could be *any* constant are those parallel to \mathbf{w}_p (hence $\perp \delta(t)$). If \mathbf{w}_o were such a vector, \mathbf{a} would be identically $\tan^2(\alpha_p)$, not zero.

$\mathbf{b}(t)$ is identically zero only if the cone vertex is on the cylinder axis, but $\mathbf{a}(t)$ and \mathbf{c} are both nonzero in this case, so it is not of great interest. \mathbf{b}' is a normal to the cylinder along rulings contained in the plane containing B_p , B_o , and \mathbf{w}_o . It will be zero if B_p is on the cylinder axis.

If the angle between the axes is equal to the cone angle, then there will be a t_0 such that $(\delta(t_0) + \mathbf{w}_p)$ is parallel to \mathbf{w}_o , and $\mathbf{a}(t_0) = \mathbf{b}(t_0) = 0$. We examine this case further in Section 6.3.3.

6.3.3 An Approach to Cone–Cylinder Intersections. In this section we present an approach to intersecting a cone with a cylinder. The main steps in the analysis are independent of the parameterization surface to be used. Indeed, one of the objectives of the analysis here is to determine which surface should be used. For the purposes of this section (6.3.3 and its subsections) only, we subscript labels for the various defining parameters with “cyl” and “con” instead of the usual “p” and “o.”

There are two observations that we exploit in this section. The first is that $(\delta(t) + \mathbf{w}_{\text{con}})$ defines a “cone of vectors,” which is the set of all vectors at an angle α_{con} from \mathbf{w}_{con} . The second is that the following three statements are identical (i.e., if any is true, it implies that the other two are true):

- The angle between the cone and cylinder axes is α_{con} .
- There is a t such that $(\delta(t) + \mathbf{w}_{\text{con}}) \parallel \mathbf{w}_{\text{cyl}}$.
- If the cone vertex is on the cylinder, the cylinder and the cone share a ruling.

6.3.3.1 *Cone Vertex on Cylinder.* The cone parameterization is always linear in this case, so we use it as the parameterization surface. All of its parameter space will be used to sweep out the curve. Once again, we have a situation analogous to that described in Sections 6.2.1.1 and 6.2.2.1. That is, the number of \mathbf{b} zeros determines the intersection curve topology and whether the curve passes through the vertex. We compute the angle between the axes (ϕ) and proceed as follows:

```

if no  $\mathbf{b}$  zeros then
  intersection is bounded, disjoint from the vertex, and lies
  entirely on one half of the cone
else if one  $\mathbf{b}$  zero then
  if  $\phi = \alpha_{\text{con}}$  then
    intersection is a shared tangential ruling plus an ellipse
    (Figure 9)
  else
    intersection is bounded, passes once through the vertex,
    and lies entirely on one half of the cone
else { two  $\mathbf{b}$  zeros }
  if  $\phi = \alpha_{\text{con}}$  then
    intersection is a shared nontangential ruling plus a space
    cubic which passes once through the vertex and goes to in-
    finity on both cone halves (Figure 23)
  else
    intersection is a figure eight with loops on each cone
    half, connecting at the vertex

```

The t values corresponding to the shared rulings in the previous discussion are the t_0 values noted in Sections 6.3.1 and 6.3.2.

6.3.3.2 *Cone Vertex inside Cylinder.* Both halves of the double cone will intersect the cylinder in this case. The cone parameterization cannot be linear, but there are no critical points to determine on it since all of its parameter space will correspond to actual positions on the intersection curve. Our choice of parameterization surface is therefore the cone, unless the angle between the axes (ϕ) is α_{con} , in which case we chose the cylinder since its parameterization will be linear. The analysis proceeds as follows:

```

if  $\phi = \alpha_{\text{con}}$  then { use the cylinder }
  intersection is two disjoint unbounded branches, one on each
  cone half (Figure 24)
  { compute cylinder parameter space  $t$  values where curve goes
  to infinity }
   $\mathbf{n} :=$  component of  $\mathbf{w}_{\text{con}}$  perpendicular to  $\mathbf{w}_{\text{cy1}}$ 
  intersect cylinder with the plane  $(B_{\text{con}}, \mathbf{n}) \Rightarrow$  two rulings
  compute the  $t$  values for the rulings on the cylinder
else { use the cone }
  intersection is two disjoint bounded branches, one on each
  cone half

```

6.3.3.3 *Cone Vertex outside Cylinder.* Our analysis in this case begins by considering the angle ϕ between the axes. Consider Figure 25, where we are

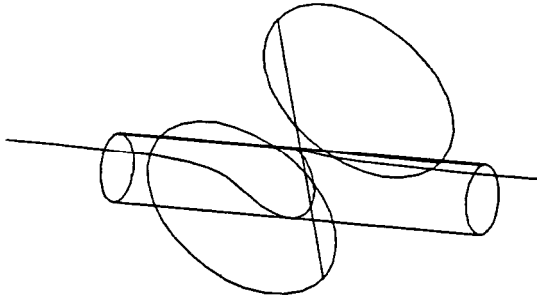


Figure 23

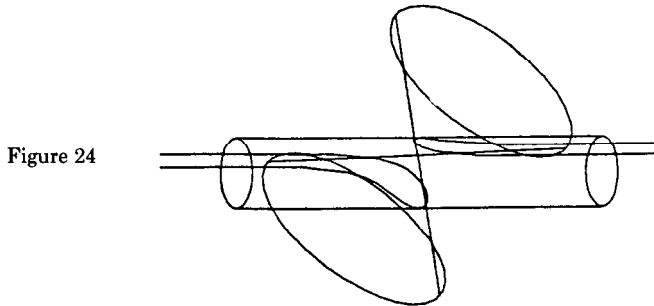


Figure 24

looking along a vector perpendicular to the two axis vectors:

```

if  $\phi < \alpha_{con}$  then
  { every cylinder ruling intersects both cone halves }
  intersection is two disjoint bounded branches, one on each
  cone half, using all of cylinder parameter space
else if  $\phi = \alpha_{con}$  then { cylinder parameterization is linear }
  intersection is one bounded branch on one cone half using all
  of cylinder parameter space
else {  $\phi > \alpha_{con}$  }
  intersection is bounded and is determined by explicitly com-
  puting discriminant zeros in a manner analogous to that de-
  scribed in Section 6.2.2.3 (see below)
    
```

When explicitly computing discriminant zeros, as in Section 6.2.2.3, we could use either surface as the parameterization surface. Let us first consider the cone. We need to find cone rulings tangent to the cylinder, so we begin by specifying a generic cylinder tangent plane. Such a plane can be defined as $(\gamma(r), \gamma(r) - B_{cyl})$, where the use of “ r ” as the parameter signifies use of the rational polynomial parameterization. In order for one of these tangent planes to contain cone rulings, the vertex of the cone would have to be on the plane. This leads us to seek solutions to the equation

$$(\gamma(r) - B_{con}) \cdot (\gamma(r) - B_{cyl}) = 0.$$

This is a quartic polynomial, so we turn our attention to the cylinder to see whether we can do better. To use the cylinder as the parameterization surface, we need to find cylinder rulings tangent to the cone. We saw in Section 6.2.2.3

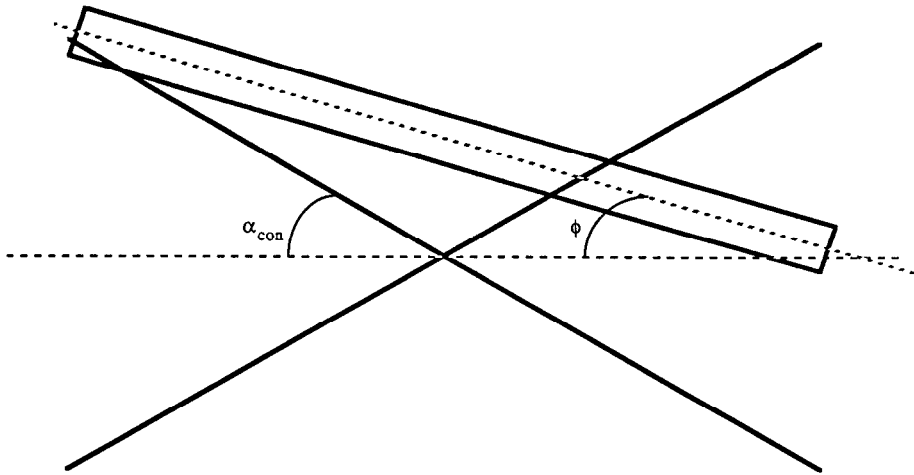


Fig. 25. View of cone-cylinder intersection in plane with normal derived from cross product of the two axis vectors.

that the generic cone tangent plane is $(B_{\text{con}}, n(r))$. For one of these planes to contain a cylinder ruling, we only require that $n(r) \perp w_{\text{cyl}}$. This leads us to seek solutions to the equation

$$n(r) \cdot w_{\text{cyl}} = 0.$$

This is a quadratic equation determining 0, 1, or 2 cone tangent planes that could contain cylinder rulings. If n is the number of planes determined and $p[]$ is the array of planes, we can proceed as follows to compute the m discriminant zeros and store them in an array $d[]$:

```

m := 0
for i := 1 to n do
  intersect p[i] with the cylinder
  if there is an intersection then
    { intersection must be one or two rulings }
    for each intersection line, L do
      m := m + 1
      d[m] := t value of L on cylinder
    end for each
  end for
end for

```

The remainder of the analysis is then identical to that described in Section 6.2.2.3.

7. SUMMARY

We have examined two approaches to the representation and intersection of quadric surfaces: an algebraic one and a geometric one. Both are viable only because we are dealing with quadrics. The number of intersection possibilities is relatively small for these surfaces, and we know from Levin that there is a ruled quadric in the pencil of any two intersecting quadrics.

In the algebraic approach, a single representation is used (the symmetric Q matrix of (2)), and a single intersection algorithm that produces intersection curves represented in the form of (6) is developed. The types of the quadric surfaces, the topology of intersection curves, and critical points on those curves are all determined by an algebraic analysis of these common forms. Although this is attractive mathematically, these models must be captured inside a computer, where floating-point numbers are not represented with infinite precision. Unfortunately, many of the basic decisions that must be made relative to these surfaces and their intersection curves in modeling systems depend critically on solutions to fourth-degree polynomial equations and on whether certain quantities calculated from the surface equation coefficients and the \mathbf{a} , \mathbf{b} , and \mathbf{c} functions are zero, positive, or negative. Since expressions involving the surface equation coefficients have no apparent geometric significance, meaningful tolerances are at best extremely difficult to derive and apply to these tests. As a direct result, these approaches have been fraught with numerical instabilities.

In the geometric approach, unique representations are employed for each surface type. This is a combination of points, vectors, and scalars that define the position, orientation, and size of the surface. For each surface pair, tailored algorithms that exploit the knowledge of the types and instancing parameters of the surfaces involved are developed to determine not only the type of the result, but also critical point information. All tests for various topological conditions are in terms of geometric parameters for which meaningful tolerances are easily derived. Moreover, scalar parameters such as radii are not numerically embedded in the representation, as they are in the algebraic approach. This means, for example, that they are not affected by round-off error during transformations. This further contributes to numerical stability.

The primary disadvantage of the geometric approach is an increased amount of code and special-case handling. Experience with a C implementation of these methods does not indicate an extraordinary volume of code, however. Intersections involving all possible combinations of the natural quadrics (including the plane) were implemented along with some conic-conic and conic-quadric intersections as needed by the analysis. The implementation required approximately 7000 lines of code.

When developing methods based on the enumeration of cases, it is obviously important to make sure that all of them are covered. There are three basic types of special cases that must be considered: detecting conic sections (and/or isolated point intersections), detecting linear parameterizations, and locating parameter values at which the intersection tends to infinity. (These are not mutually exclusive.) It is also important to capture the topology of the intersection correctly. Formal methods based on the use of the algebraic approach as an analytical tool are presented in [5] for the detection of conics in quadric surface intersections. Unfortunately it is not clear how one could be as formally rigorous for these other aspects of the intersection operation. The strong geometric insight realizable with these approaches, as well as experience with a wide range of test cases ranging from pathological to "normal" to randomly generated, provide some measure of confidence.

There are many different geometric algorithms that could have been developed for each of the situations described. The point is that such approaches are

possible and, at least with the natural quadrics, avoid the need to solve polynomials of degree higher than 2.

These methods can be generalized somewhat. When dealing with the natural quadrics, either the cone or the cylinder is always involved when nonplanar intersections arise. We therefore do not need to appeal to the pencil (3) to find a ruled parameterization surface. Extending these methods by supporting additional ruled quadrics would be fairly straightforward. Introducing additional quadrics that are not ruled, however, requires the development of methods to determine an appropriate parameterization surface by examining the pencil of two intersecting quadrics.

The advantages of the geometric approaches to quadric surface intersections as described in this paper can be summarized as follows:

—Intersection curve topology and critical point information can be determined from geometric analyses, avoiding the need to solve fourth-degree polynomials. This is not only faster computationally, but also more robust numerically since we can directly compute geometrically the roots, if any, of critical functions.

—We can employ direct geometric reasoning instead of algebraic analyses to determine intersection topology, critical points on curves, the presence of conic sections, and degenerate situations, such as one or more of \mathbf{a} , \mathbf{b} , and \mathbf{c} being zero, either identically or for a given value of t . Tricky topological situations, such as the vertex of a cone being on another surface with a disjoint intersection elsewhere, can be detected without danger of interpreting the region of the vertex as another small intersection curve branch.

—No additional work is required to find the best parameterization surface. In the algebraic approach, considerable work is required to find a good (not necessarily the best) surface. Note that the determination of the best surface depends not only on the *types* of the surfaces involved, but also on their *instancing parameters*.

—There is no need to determine transformations to represent a quadric in the local coordinate system of the parameterization surface. This minimizes round-off error, storage requirements, algorithm complexity, and execution time.

—When generating points on the intersection curve, fewer floating-point operations are required for the geometric approach than for the algebraic approach. The difference is not overwhelming. It varies from a couple of percentage points for cylinder–cylinder and cylinder–cone (using the cylinder as the parameterization surface) to about 40 percent for cylinder–sphere and cone–sphere. It is therefore very much a secondary consideration. The presence of this speed advantage should not be surprising since there are many constants in (14)–(17) and (19)–(20) and many other terms that appear in more than one of \mathbf{a} , \mathbf{b} , and \mathbf{c} . Furthermore, while evaluating the \mathbf{a} , \mathbf{b} , and \mathbf{c} functions, $\gamma(t)$ or $(\delta(t) + \mathbf{w}_p)$ is computed and of course need not be recomputed when finding $P(s, t)$ from (9) or (10). For the algebraic approach, recall that a final matrix multiplication is required after an (x, y, z) is computed from (4) or (5). The algebraic approach is a general scheme that works when any quadric surface intersects the given ruled parameterization quadric. It pays a penalty for this generality by, in some sense, always handling the most complex case.

REFERENCES

1. BOYSE, J. W., AND GILCHRIST, J. E. GMSolid: Interactive modeling for design and analysis of solids. *IEEE Comput. Graph. Appl.* 2, 2 (Mar. 1982).
2. BROWN, C. M. PADL-2: A technical summary. *IEEE Comput. Graph. Appl.* 2, 2 (Mar. 1982).
3. DRESDEN, A. *Solid Analytic Geometry and Determinants*. Dover Publ., New York, 1964.
4. GOLDMAN, R. N. Two approaches to a computer model for quadric surfaces. *IEEE Comput. Graph. Appl.* 3, 6 (Sept. 1983).
5. GOLDMAN, R. N., AND MILLER, J. R. Detecting and calculating conic sections in quadric surface intersections, in preparation.
6. GOLDSTEIN, R., AND MALIN, L. 3D modeling with the Synthavision system. In *Proceedings of the 1st Annual Conference on Computer Graphics in CAD/CAM Systems* (Apr. 1979).
7. HAKALA, D. G., HILLYARD, R. C., NOURSE, B. E., AND MALRAISON, P. J. Natural quadrics in mechanical design. In *Proceedings of Autofact West 1* (Nov. 1980).
8. HILLYARD, R. The build group of solid modelers. *IEEE Comput. Graph. Appl.* 2, 2 (Mar. 1982).
9. LEVIN, J. A parametric algorithm for drawing pictures of solid objects composed of quadric surfaces. *Commun. ACM* 19, 10 (Oct. 1976).
10. LEVIN, J. Mathematical models for determining the intersections of quadric surfaces. *Comput. Graph. Image Process.* 11, 1 (1979).
11. LEVIN, J. QUISP: A computer processor for the design and display of quadric-surface bodies. Ph.D. dissertation, Electrical and Systems Engineering Dept., Rensselaer Polytechnic Institute, Troy, N.Y., Jan. 1980.
12. MILLER, J. R. Analysis of quadric surface based solid models. Introduction to Solid Modeling. Short Course Notes. ACM SIGGRAPH '86 (Dallas, Tex., Aug. 18-22). ACM, New York, 1986.
13. REQUICHA, A. A. G., AND VOELCKER, H. B. Solid modeling: A historical summary and contemporary assessment. *IEEE Comput. Graph. Appl.* 2, 2 (Mar. 1982).
14. SARRAGA, R. F. Algebraic methods for intersections of quadric surfaces in GMSOLID. *Comput. Vision Graph. Image Process.* 22, 2 (May 1983).
15. WILSON, P. R. Conic representations for shape description. *IEEE Comput. Graph. Appl.* 7, 4 (Apr. 1987).

Received January 1987; revised August 1987; final revision accepted September 1987