A method for displaying the intersection curve of two quadric surfaces

M. A. Sabin*

Dynamics and Mathematical Services, British Aircraft Corporation, Commercial Aircraft Division, Weybridge, Surrey, England

A representation of algebraic curves not widely known is shewn to be highly appropriate for calculating views of curves such as the intersections of general second order surfaces (quadrics), a problem approached by Luh and Krolak (1965), by Weiss (1966), and by Woon and Freeman (1971). By allowing the repetitive calculation to be performed in the picture plane it reduces the number of degrees of freedom involved from three to two. This representation gives an algorithm with rather different characteristics from the usual methods of drawing such curves, and these differences are discussed.

(Received November 1973)

1. Notation

Lower case letters denote scalar variables or functions

e.g. $a, b, t, f(a)$

Upper case letters denote tensors if postscripted

e.g. A_{ij}, P^i

and vector variables or functions if not.

e.g. $P, F(a, b)$

2. Background

Several methods of calculating intersections of surfaces are known. If available, the best method is to express the curve as a vector valued rational polynomial in a parameter. (The rational polynomial is the most general function computable without iteration or ambiguity by normally available floating point hardware.) Closely spaced points along the curve can then be generated very cheaply by evaluating the polynomial for successive values of the parameter (**Fig. 1**). It is not difficult to choose the parameter values to minimise the number of segments drawn subject to giving the appearance of a smooth curve (Sabin, 1972). This method is not always available, however, except as an approximation: in particular the intersection of two general quadrics is a quartic curve which can easily take the form of two closed loops (**Fig. 2**), and cannot be represented as a rational polynomial (Sommerville, 1934).

If one of the two surfaces can be represented in the form

$$P = F(u, v)$$

and the other in the form

$$f(P) = 0$$

the problem can be reduced to the solution of

$$f(F(u, v)) = g(u, v) = 0$$

in the two degrees of freedom u and v.

This can be implemented by a lattice method (Payne, 1971) or by stepping along the intersection, iterating in u and v, to each point in turn from a first approximation based on the differential geometry near the previous point (South and Kelly, 1965).

If both surfaces are of the form

$$f(P) = 0$$

it is possible to step along the intersection in three dimensions by estimating iteratively the changes in the three components of P to solve simultaneously

$$f_1(P) = 0$$
$$f_2(P) = 0$$

and a step length equation. If both are of the form

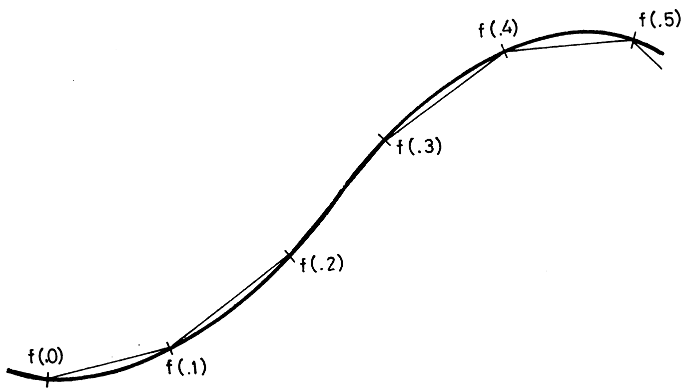

Fig. 1 Drawing generated by chords of a parametric curve

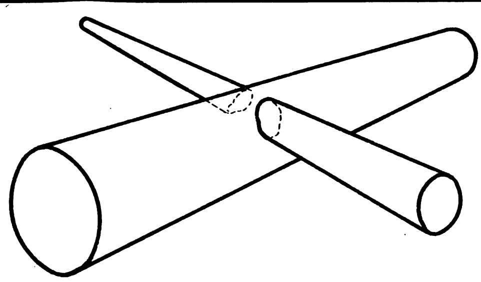

Fig. 2 A curve with two loops

$$P = F(u, v)$$

it is possible to step along the intersection in the four dimensions u_1, v_1, u_2, v_2 by solving simultaneously the three component equations of

$$F_1(u_1, v_1) = F_2(u_2, v_2)$$

and the step length equation. Generally speaking, the fewer degrees of freedom involved the faster and more reliable the process will be.

3. The Cayley form of a curve

A little known geometric technique of some antiquity, which is readily implemented numerically, is also available for this problem of intersecting algebraic surfaces of the form $f(P) = 0$. This uses the Cayley representation of the intersection curve (Cayley, 1860; Semple and Kneebone, 1959), whereby the curve is represented by the function of two points

$$f(P_1, P_2) = 0 .$$

The function is zero if and only if the straight line P_1P_2 cuts

*Now at Computers and Automation Institute, Hungarian Academy of Sciences, Budapest, Hungary.

the curve.

At first sight this appears an abstruse, indirect way of characterising a curve, but in fact it is exactly what is required for displaying views of it, because if we substitute the position of the eye for P_1 the equation

$$f(P_1, P_2) = 0$$

becomes an equation for points of the picture plane lying on the projection of the curve. With this equation we can do all our calculations in the two dimensional picture plane instead of the three dimensional space in which the curve itself lies.

This idea, of performing the repetitive and time consuming calculations in the 2D picture plane instead of the 3D problem space, is a fundamental one. The reader is referred to Warnock (1969) and Newell, Newell and Sancha (1972) to see it applied in a different way in other contexts, those of hidden line removal and half-tone picture representations.

The next question, of course, is how to derive the form and coefficients of the Cayley equation from those of the two surfaces, the eye point, and the picture plane. This we shall describe first for the intersection of a plane and a quadric, then for that of two quadrics.

4. Plane-quadric intersection

For the purposes of algebraic manipulation we shall represent points by four homogeneous coordinates, planes by the corresponding four coefficients, and quadrics by 4 × 4 symmetric matrices (Maxwell, 1961). Using tensor notation (see Appendix, also Coolidge 1945 and Jeffreys, 1969) we have the equations

$F_iP^i = 0$ when the point P lies in the plane F

$A_{ij}P^iP^j = 0$ when the point P lies in the quadric A

Let the eye point be E and the generic point of the picture plane be D. The line DE contains all points of the form

$$P^i = tD^i + (1 - t)E^i \qquad (1)$$

and its intersection with F is given by

$$tF_iD^i + (1 - t)F_iE^i = 0 \qquad (2)$$

or

$$td + e = 0 \qquad (3)$$

where

$$d = F_iD^i - F_iE^i$$

and

$$e = F_iE^i$$

Similarly, the intersections with A are given by the roots of

$$t^2A_{ij}D^iD^j + 2t(1 - t)A_{ij}D^iE^j + (1 - t)^2A_{ij}E^iE^j = 0 \qquad (4)$$

or

$$t^2a + tb + c = 0 \qquad (5)$$

where

$$a = A_{ij}D^iD^j - 2A_{ij}D^iE^j + A_{ij}E^iE^j$$
$$b = 2(A_{ij}D^iE^j - A_{ij}E^iE^j)$$

and

$$c = A_{ij}E^iE^j$$

Eliminating t between equations 3 and 5 gives

$$ae^2 - bed + cd^2 = 0 \qquad (6)$$

which can be re-expressed in terms of D as

$$B_{ij}D^iD^j \doteq 0 \qquad (7)$$

where

$$B_{ij} = (e^2A_{ij} - 2eA_{ik}E^kF_j + cF_iF_j) \ . \qquad (8)$$

The values of c, e and B (another 4 × 4 matrix, actually representing a cone with vertex at E), can be computed once and then used repeatedly in generating the whole profile.

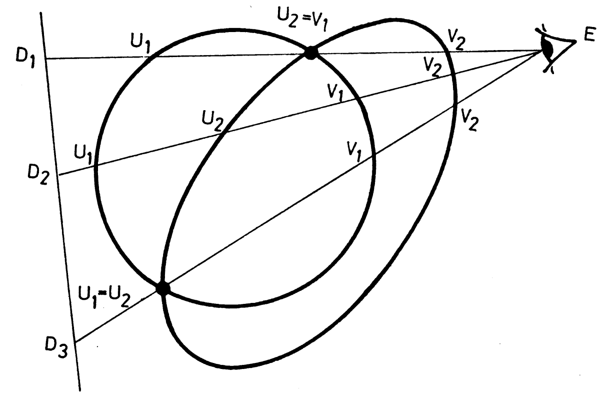

Fig. 3 Intersections and coincidences of roots

5. Quadric-quadric intersection

The procedure for setting up the Cayley function here is similar in principle to the above, but with a few new features which make it worth describing, because they apply equally to the intersections of higher order surfaces. Let the two quadrics be A_1 and A_2 and let D and E be the points as before. The points of intersection of each quadric with DE are given by the roots of the two quadratics

$$t^2a_1 + tb_1 + c_1 = 0 \qquad (9)$$

and

$$t^2a_1 + tb_2 + c_2 = 0 \qquad (10)$$

(cf. equation 5) .

Let the roots of equation (9) be u_1, v_1 and of equation 10 be u_2, v_2.

Now the function

$$(u_1 - u_2)(u_1 - v_2)(v_1 - u_2)(v_1 - v_2)$$

is zero whenever one of u_1, v_1 is equal to one of u_2, v_2 and it can therefore act as our Cayley function, because this will only happen when DE cuts the intersection curve (e.g. for points $D1$ and $D3$ in **Fig. 3**).

This can now be expressed in terms of the sum and product of u_1 and v_1 and those of u_2 and v_2 and thence in terms of the coefficients of the quadratic equations (9) and (10) thus

$$\left(\frac{c_1}{a_1} - \frac{c_2}{a_2}\right)^2 + \left(\frac{b_1}{a_1}\frac{c_2}{a_2} - \frac{b_2}{a_2}\frac{c_1}{a_1}\right)\left(\frac{b_1}{a_1} - \frac{b_2}{a_2}\right) . \qquad (12)$$

We can multiply throughout by $a_1^2a_2^2$ and simplify to give

$$(a_2c_1 - a_1c_2)^2 + (b_1c_2 - b_2c_1)(a_2b_1 - a_1b_2) \qquad (13)$$

which for computation can be replaced by

$$C_{ijkl}D^iD^jD^kD^l = 0 \qquad (14)$$

where

$$C_{ijkl} = c_2^2A_{1ij}A_{1kl} - 2c_1c_2A_{1ij}A_{2kl} + c_1^2A_{2ij}A_{2kl} + 4(c_2A_{1im} - c_1A_{2im})(A_{2jk}A_{1ln} - A_{1jk}A_{2ln})E^mE^n \qquad (15)$$

Now c_1, c_2 and C(a 4 × 4 × 4 × 4 block matrix) can be computed once and for all. If we take advantage of the symmetry of equation (14), C can be stored as 15 numbers, the coefficients of the fourth order polynomial in the two picture plane components of D. The evaluation of the Cayley function for any point D then takes 14 multiplications. To achieve this, all transformations and scaling have to be performed on the quadrics before starting (Forrest, 1969) so that the values of D tried are actual plotter co-ordinates.

6. Higher order surfaces

The same procedure can be applied to surfaces of higher order. The intersection of cubics, for example, leads to a 9th order tensor, reducible to 44 multiplications per trial point.

7. Silhouettes

A related technique, using the condition for equality of two roots of the equation for the intersection of the line DE with one surface gives the projection of the silhouette of the surface. This is hinted at by Luh and Krolak (1965).

8. Comparison with other methods

Two other methods are worth considering. If we parametrise one quadric we can find the solution of the intersection in the parameter plane. This is another quartic (in the two parameters), taking again 14 multiplications per trial point. However, to draw the view of the curve requires a further 23 multiplications and a division for each drawn point, approximately doubling the computation. Windowing may be needed if the curve extends beyond the viewing area. Also the curve may pass through infinity either in the parameter plane or in space while still inside the viewing area. The former can be avoided by keeping the quadric as two separate patches, but it complicates the algorithm.

Another symptom of the same difficulty is that the optimum spacing of the points along the curve is much less easily achieved, and one would expect to compute at least twice as many points as necessary, or else spend an equivalent time getting the spacing right. In all the computation time would typically be 4-5 times that required by the algorithm described here.

The second possibility is to step in three degrees of freedom along the solution of $f_1(P) = 0, f_2(P) = 0$. This requires 18 multiplications per trial point, with a further three and a division per drawn point. Again windowing is necessary and again the curve can go to infinity in space while remaining in view. It is difficult to quantify the increase in trials per drawn point due to stepping in three dimensions instead of two, but if we take a factor of two (correct for lattice methods) this method is likely to be 6-8 times slower than the Cayley function method.

On the other hand, the Cayley function method has three disadvantages compared to the other two.

If a new view of the same curve is required the calculation has to be repeated completely, which makes this method unsuitable if real time rotation is required.

It is difficult to draw just part of the intersection curve if it is limited in space rather than in the picture plane.

Computational difficulties can occur when a curve appears to cross itself. If a coarse lattice method is used the curve may be drawn with incorrect connectivity.

These last two are in fact both symptoms of the same problem, that information about the space configuration has been lost in the projection on to the picture plane. To overcome this requires that some analysis of the arcs to be drawn should be performed before the drawing starts, and so a practical implementation of this algorithm would not be as simple as implied in the description above.

None the less, this representation is of some theoretical importance, and with care the ideas can be applied to good effect.

Acknowledgements

The author wishes to thank the management of British Aircraft Corporation for permission to publish this work, and the referee for several helpful comments and suggestions.

Appendix Tensor Notation

This notation is used to facilitate the otherwise bulky algebraic manipulation. Equation (15), for example, is formally equivalent to 256 equations, each of 19 terms.

If a letter appears as subscript and superscript within one term, summation of that term with the letter given all values between one and (in this case) four is implied. A subscript or superscript appearing on both sides of an equation implies that the equation holds for each value of the letter between one and four. The full content of equation (15) may be illustrated by the equivalent FORTRAN program.

```
  REAL C1, C2, C(4, 4, 4, 4), A1(4, 4), A2(4, 4), E(4)
  C1 = 0.0
  C2 = 0.0
  DO 1 I = 1, 4
  DO 1 J = 1, 4
  C1 = C1 + A1(I, J) × E(I) × E(J)
1 C2 = C2 + A2(I, J) × E(I) × E(J)
  DO 2 I = 1, 4
  DO 2 J = 1, 4
  DO 2 K = 1, 4
  DO 2 L = 1, 4
  C(I, J, K, L) = C2 × C2 × A1(I, J) × A1(K, L)
1                 −2.0 × C1 × C2 × A1(I, J) × A2(K, L)
2                 +C1 × C1 × A2(I, J) × A2(K, L)
  DO 2 M = 1, 4
  DO 2 N = 1, 4
2 C(I, J, K, L) = C(I, J, K, L) + 4.0 × (C2 × A1(I, M)
1                 −C1 × A2(I, M)) × (A2(J, K) × A1(L, N)
                  −A1(J, K) × A2(L, N)) × E(M) × E(N)
```

This is far from efficient coding, but illustrates very precisely what the somewhat forbidding notation means.

The distinction between subscripts and superscripts is essentially that between items of dimension length^{-1}, and those of dimension length. It provides a valuable check when manipulating large equations, such as those encountered when deriving equation (15) from equation (13).

References

Cayley, A. (1860). A new analytical representation of curves in space, *Quarterly Journal of Pure and Applied Mathematics*, 3, 225-236.

Coolidge, J. L. (1945). *A history of the Conic Sections and Quadric Surfaces*, Clarendon Press, Chapter VI.

Forrest, A. R. (1969). *Coordinates, Transformations and Visualizations*, Cambridge University Joint CAD Group Document 23.

Jeffreys, H. (1969). *Cartesian Tensors*, Cambridge University Press, Chapters I and II.

Luh, J. Y. S., and Krolak, R. J. (1965). A Mathematical Model for Mechanical Part Description, *CACM*, Vol. 8, No. 2, pp. 125-129.

Maxwell, E. A. (1961). *General Homogeneous Coordinates in Space of Three Dimensions*, Cambridge University Press, Chapters I, II, IV and VIII.

Newell, M. E., Newell, R. G., and Sancha, T. L. (1972). A solution to the hidden surface problem. *Proc. ACM*, Boston, Mss.

Payne, P. J. (1971). *A contouring program for joined surface patches*, Cambridge University Joint CAD Group Document 58.

Sabin, M. A. (1972). Letter to *The Computer Journal*, Vol. 15, No. 2, p. 104.

Semple, J. G., and Kneebone, G. T. (1959). *Algebraic Curves*, Oxford University Press.

Sommerville, D. M. Y. (1934). *Analytic Geometry of Three Dimensions*, Cambridge University Press, p. 297.

South, N. E., and Kelly, J. P. (1965). *Analytic Surface Methods*, NC Development Unit, Ford Motor Company, December 1965.

Warnock, J. E. (1969). *A hidden surface algorithm for computer-generated half tone pictures*, Tech. Rep. RADC-TR-69-249 Computer Science, University of Utah.

Weiss, R. A. (1966). BE VISION. A package of IBM 7090 FORTRAN programs to draw orthographic views of combinations of planes and quadric surfaces, *JACM*, Vol. 13, No. 2, pp. 194-204.

Woon, P., and Freeman, H. (1971). A procedure for generating visible line projections of solids bounded by quadric surfaces, *Proc IFIP* 1971, TA-6, pp. 81-85.