Department of Computer Science Technical Reports

Department of Computer Science

1989

Algebraic and Numerical Techniques for Offsets and Blends

Christoph M. Hoffmann

Purdue University, cmh@cs.purdue.edu

Report Number:

89-895

ALGEBRAIC AND NUMERICAL TECHNIQUES
FOR OFFSETS AND BLENDS

Christoph M. Hoffmann

CSD TR-895
July 1989

ALGEBRAIC AND NUMERICAL TECHNIQUES FOR OFFSETS AND BLENDS

CHRISTOPH M. HOFFMANN[1]
Computer Science Department
Purdue University
West Lafayette, IN 47907
USA

ABSTRACT. We examine some techniques and results from algebraic geometry, and assess how and to what extent they are of use in computer-aided geometric design (CAGD). Focusing on offset and blending surface construction, we illustrate how to apply and assess algebraic methods. We also examine some numerical techniques for interrogating offsets and blending surfaces constructed using the algebraic approach.

1. INTRODUCTION

In his thesis, Sederberg (1983) demonstrated that it is possible to convert any given parametric curve and surface into implicit form, at least in principle. Sederberg used *resultants*, a classical algebraic technique developed at the end of the 19th and the beginning of the 20th century; e.g., Netto (1892) and Macaulay (1902 and 1916). Although the convertibility from parametric to implicit form, using resultants, was known to geometers of those times, it fell subsequently into oblivion, until Sederberg's work renewed awareness of it.

Sederberg's thesis stimulated much interest in algebraic methods in general and in resultant-based computations in particular. A number of algorithms have since been devised that use resultants to reduce many geometric operations to the problem of determining the roots of univariate polynomials. See, for example, Abhyankar and Bajaj (1989), Farouki (1986), and Geisow (1983).

Resultants can indeed be used as a tool in many geometric computations. However, since then, it has been recognized that using resultants may not be practical for the following reasons:

---

1. Polynomials derived from resultant operations may generate additional solutions that do not satisfy the original problem. Such phantom solutions are intrinsic to the method, and are not always easy to recognize.

2. The evaluation of a resultant may entail a huge amount of computation that cannot be done in a reasonable amount of time and space.

3. Finally, when using floating-point arithmetic, the accuracy of solutions obtained with resultants is not fully understood. For example, could a well-conditioned intersection of three surfaces be reduced to finding the roots of an ill-conditioned univariate polynomial?

These difficulties motivate examining whether, and how, to apply resultants or other methods from algebraic geometry, and how to obtain a correct perspective of what these methods can achieve in the context of CAGD.

The problems that must be faced when using resultants are typical in the following sense: When examining a method from algebraic geometry, three questions should be asked:

1. Is the method faithful?

2. Is the method efficient?

3. Is the method numerically well-behaved?

In the following, we will look at these questions focusing on offsetting and blending curves and surfaces, given parametrically or implicitly.

While the algebraic approach delivers good techniques for defining offsets and blends with precision, subsequent elimination to arrive at closed-form expressions for such curves and surfaces may require excessively long and space-intensive computations. There has been recent progress, however, and techniques have been discovered that are a substantial step forward. This raises hopes that future improvements can give algebraic methods wider applicability. Nevertheless, it is realistic also to seek numerical techniques for interrogating and analyzing complex offsets and blends. So, we describe some numerical approaches as well, to tracing surface intersection and to locally approximating surfaces.

2. FAITHFULNESS

We illustrate the problem of faithfulness. More precisely, we concentrate on faithfulness in problem formulation and defer faithfulness in symbolic computation until Section 5.

Figure 1: Semicubic Parabola (s^2, s^3)

2.1: DIFFERENTIAL-OFFSET-FORMULATION

Consider offsetting a parametric curve, such as the semicubic parabola shown in Figure 1. The curve is given parametrically by

$$\begin{aligned} x(s) &= s^2 \\ y(s) &= s^3 \end{aligned}$$

and implicitly by the equation

$$y^2 - x^3 = 0$$

We seek a curve consisting of all points q whose Euclidean distance from a corresponding point p on the curve is r, where r is a constant. More precisely, let $p = (x(s), y(s))$ be a point on the semicubic parabola, and let

$$n(s) = \frac{(-y'(s), x'(s))}{\sqrt{x'(s)^2 + y'(s)^2}}$$

be the signed unit normal at p. Then

$$q = p + rn(s) \tag{1}$$

is a point of the offset curve. Let us define the *r-offset* of a curve to be the set of all points given by formula (1). It would be nice if we could represent the offset of a parametric curve as another parametric curve. However, mathematically this is not possible. For example, the offset of an ellipse is a curve of degree 4 that cannot be parameterized, in general.

3

Figure 2: Differential Offset of the Semicubic Parabola (s^2, s^3)

Equation (1) constitutes a *differential* representation of offsets, in terms of the coordinate functions and their derivatives. That this representation is not necessarily algebraic is shown in Figure 2: At the singularity, at $s = 0$, the curve normal n(0) is undefined. The offset curve, as defined by (1), has a discontinuity at that point, and so consists of two separate branches that end abruptly at the points $(0, r)$ and $(0, -r)$. It is known, however, that an algebraic curve cannot end abruptly; e.g., Hilbert and Cohn-Vossen (1952). Hence, the offset curve, as defined by (1), is not an algebraic curve in general. It follows, that the offset of an algebraic curve, when so defined, cannot be represented faithfully as an algebraic curve, let alone as a parametric one.

A shortcoming of the differential offset formulation is the fact that the r-offset of a parametric curve representation may differ qualitatively from the r-offset of the implicit representation of the same curve. Consider the cubic curve of Figure 3. The implicit representation is

$$y^2 - x^2 - x^3 = 0$$

and the parametric form is

$$x(s) = s^2 - 1$$
$$y(s) = s^3 - s$$

The differential r-offset of the implicit form is therefore described by

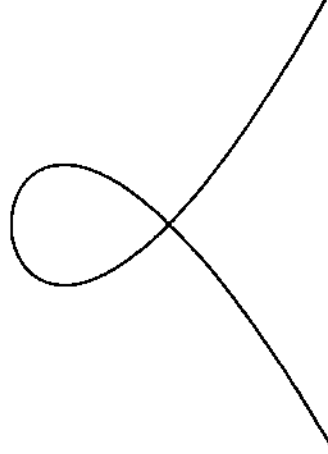$$(x, y) = (u, v) + \frac{r}{N}(-2u - 3u^2, 2v)$$

4

Figure 3: Cubic Curve Given by $(s^2 - 1, s^3 - s)$ or by $y^2 - x^2 - x^3$

where (u, v) is a curve point, i.e., $v^2 - u^2 - u^3 = 0$, and N is the length of the curve gradient, $N = \sqrt{(-2u - 3u^2)^2 + 4v^2}$. The differential r-offset of the parametric form is similarly given by

$$(x, y) = (u, v) + \frac{r}{M}(-v', u')$$

where $u = s^2 - 1$, $v = s^3 - s$, $u' = 2s$, $v' = 3s^2 - 1$, and $M = \sqrt{u'^2 + v'^2}$. Figure 4 shows the differential offset of the implicit form, and Figure 5 the differential offset of the parametric form. The qualitative difference of the two offsets demonstrates that the differential offset formulation depends critically on the representation of the curve, rather than on the intrinsic geometry of the curve.

2.2. ALGEBRAIC OFFSET FORMULATION

Customarily one desires an algebraic representation, of which the parametric representation is a special case. If we want to represent offsets algebraically, then we must redefine them in the sense that additional points are included. Figure 6 shows the algebraic offset of the semicubic parabola. It is obtained by considering the *bi-sided* offset, consisting of the points given by

$$q = p \pm r\mathbf{n}(s) \qquad (2)$$

Both normal directions must be considered, because of the square root in the expression for $\mathbf{n}(s)$.
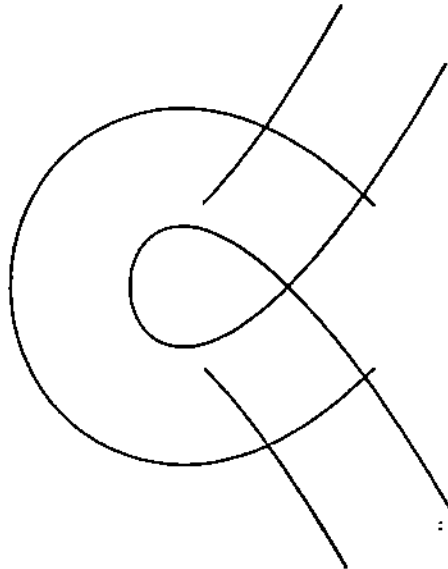
Figure 4: Offset of the Implicit Form

From differential geometry, we obtain the following method for defining the algebraic offset of a parametric curve; e.g., Spivak (1975). We formulate the equations

$$(x - x(s))^2 + (y - y(s))^2 - r^2 = 0 \qquad (3)$$
$$x'(s)(x - x(s)) + y'(s)(y - y(s)) = 0 \qquad (4)$$
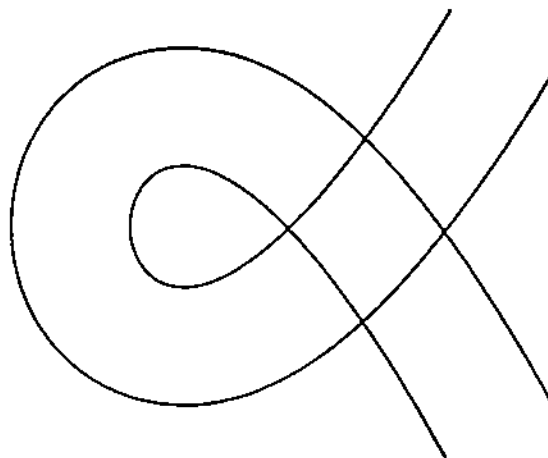

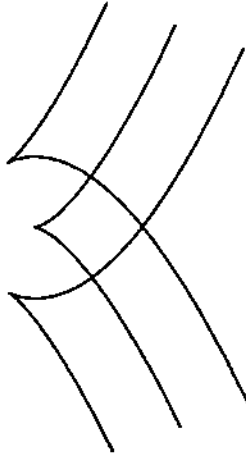Figure 5: Offset of the Parametric Form

Figure 6: Algebraic Offset of the Semicubic Parabola (s^2, s^3)

From these two equations, the parameter s is eliminated, resulting in a single algebraic equation, in x and y, that describes the r-offset of the parametric curve $(x(s), y(s))$. Note that r is a constant.

The significance of equations (3) and (4) is as follows. Equation (3) states that a point (x, y), on the offset curve, has the Euclidean distance r from the corresponding curve point, $(x(s), y(s))$. Equation (4) states that the direction vector $(x - x(s), y - y(s))$, from the curve point $(x(s), y(s))$ through the offset point (x, y), is perpendicular to the curve tangent $(x'(s), y'(s))$. In consequence, the point (x, y) is on the *envelope* of a family of circles, of radius r, whose centers lie on the curve $(x(s), y(s))$. See also Figure 7.
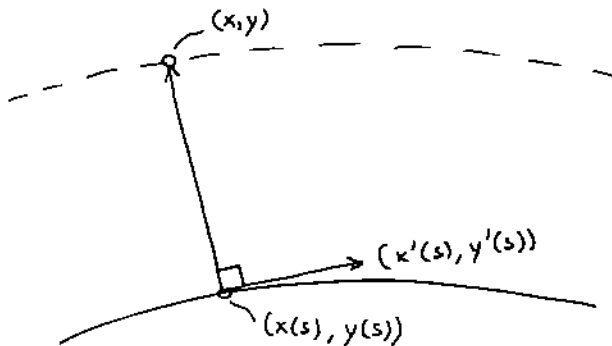

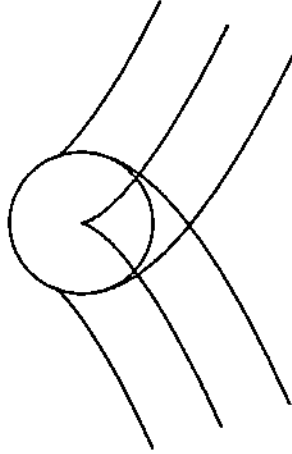Figure 7: Interpretation of Second Offset Equation

Figure 8: Extraneous Points in Offset of the Semicubic Parabola

For example, in the case of the semicubic parabola, we obtain the following equations

$$(x - s^2)^2 + (y - s^3)^2 - r^2 = 0 \qquad (5)$$
$$-2s(x - s^2) - 3s^2(y - s^3) = 0 \qquad (6)$$

Elimination of s yields then the offset equation.

The implicit offset of the semicubic parabola obtained from (5) and (6) is not as shown in Figure 6, but looks as shown in Figure 8. When eliminating s from (5) and (6), a polynomial of the form

$$(x^2 + y^2 - r^2)\,h(x, y, z)$$

is obtained, where h has degree 8. The problem is that the equation (6) fails for $s = 0$: At that point, (6) vanishes identically, so that all points of the circle $x^2 + y^2 - r^2$ satisfy the two equations. This fact is responsible for the presence of the extraneous factor $x^2 + y^2 - r^2$.

Note that the cuspidal singularities on the offset curve are not generated by the cusp of the semicubic parabola, but by those points at which the curvature equals the offset distance. Moreover, the offset cusps lie inside the circle of extraneous points. That is, their Euclidean distance from the cusp of the semicubic parabola is smaller than the offset distance.

In the case of parametric curves, such extraneous factors can be avoided by factoring out the greatest common divisor (gcd) of $x'(s)$ and $y'(s)$; Farouki and Neff (1989). We conclude, that a *faithful* formulation of the algebraic offset of a parametric curve
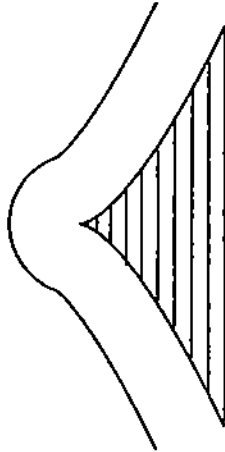
8

Figure 9: Offset of Halfspace Bounded by (s^2, s^3)

is of the form[2]

$$
\begin{aligned}
(x - x(s))^2 + (y - y(s))^2 - r^2 &= 0 \\
(x'(s)(x - x(s)) + y'(s)(y - y(s)))/\gcd(x'(s), y'(s)) &= 0
\end{aligned}
\tag{7}
$$

2.3. FAITHFUL PROBLEM FORMULATION

We have discussed faithfulness of the problem formulation. Clearly, this concept of faithfulness depends on what the object is we wish to determine, and what is possible to describe mathematically. Here, we require that the offset of an irreducible algebraic curve be an algebraic curve, of minimum degree. Some applications may require different offset definitions, and thus the notion of faithful problem formulation depends on the context. The semicubic parabola again illustrates the matter: If we consider the curve as the boundary of a halfspace, then an offset of the form shown in Figure 9 might be wanted. We saw above that such an offset curve must composite, consisting of arcs of the algebraic offset and of the circle of extraneous points. Moreover, these arcs do not join with tangent continuity.

Given a mathematical definition of a geometric object, such as (7), we often have to do some computation to derive a more convenient description of the result. In the offsetting example, this computation is the elimination of the parametric variable s. Depending on the method of computation, the elimination process might introduce additional solutions not entailed by the original system of equations. We will give an example of this phenomenon later, in Section 5. For now, we note that this is another

---

[2]The formulation is faithful for polynomial parametric curves. See also Section 4.

9

aspect of faithfulness; i.e., the algorithm used to eliminate variables may or may not be faithful.

3. EFFICIENCY AND NUMERICAL STABILITY

Formulating systems of equations such as (7) is evidently a simple and efficient operation. In contrast, the subsequent elimination of the parameter s, done to arrive at a closed-form description of the offset curve, is an expensive symbolic computation. It is precisely this inefficiency that hinders wider acceptance of symbolic computation in geometric applications, and an important research challenge is to customize general symbolic computations to the specific geometric problem at hand, in an effort to improve efficiency. We discuss some experiments that amplify this point and mention some of the implications later in Section 5.

From a geometric perspective, resultant computations are projections. As an example, consider the intersection of two implicit surfaces, $f(x, y, z) = 0$ and $g(x, y, z) = 0$. From the two implicit equations, we can eliminate any variable; for example, z. In effect, we project the intersection curve onto the (x, y)-plane. The resulting plane algebraic curve $h(x, y) = 0$ is in birational correspondence with the surface intersection. That is, there is a mapping between the points of the curve $h = 0$ and the points of the intersection of f and g, and this mapping is bijective except at finitely many points.

The mapping is provided by a rational function, that is, by the ratio of two polynomials. The situation is exactly analogous to the mapping between the (s, t)-plane and the points on a parametric surface, in (x, y, z)-space; it is also analogous to the mapping between the s-line, and the points of a parametric curve $(x(s), y(s))$.

Elimination of the variable z is a projection in the direction of the z-axis. Other projection directions may be obtained by changing the coordinate system prior to the elimination step. That is, general projections of the surface intersection onto a plane is achieved by

1. Transformation of f and g by a nonsingular, linear transformation.

2. Elimination of one of the variables.

For almost all directions we obtain a plane algebraic curve that is in birational correspondence with the surface intersection. There may be, however, special projection directions for which the projection degenerates. A simple example illustrates this possibility.

Consider the intersection of two circular cylinders, $x^2 + y^2 - 4 = 0$ with $x^2 + z^2 - 1 = 0$. One cylinder is about the z-axis, of radius 2, and the other about the y-axis, of radius 1. Their intersection is an irreducible space curve of degree 4, and is shown
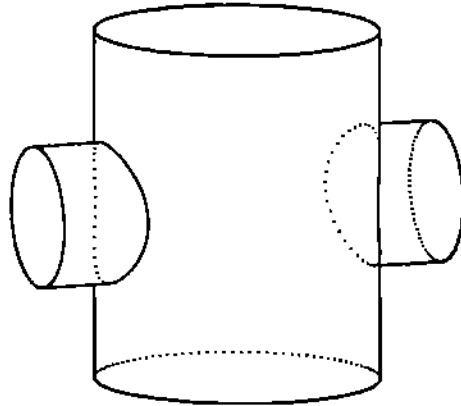
Figure 10: Intersection Curve of Two Cylinders

in Figure 10. Elimination of z yields $(x^2 + y^2 - 4)^2$, i.e., two coincident circles. So, this projection is degenerate since there is no 1-1 correspondence between the points of the circle and the points on the intersection of the two cylinders.

Now consider a projection in a direction that is nearly parallel to the z-axis. Qualitatively, a curve of the form shown in Figure 11 is obtained. It is evident that we can increase indefinitely the curvature at the four marked points, by diminishing the angle between the projection direction and the z-axis. In consequence, a numerical evaluation of the projected curve becomes increasingly more difficult, even though, in 3-space, a numerical evaluation of the intersection of the two cylinders is not difficult. That is, we may have projected a numerically well-conditioned problem into an ill-conditioned one. There seems to be no mention in the literature of this important problem.

Figure 11: Projection of Cylinder Intersection Curve

11

4. OFFSETS AND CONSTANT-RADIUS BLENDING

4.1. OFFSETS OF CURVES AND SURFACES

We described before how to formulate the r-offset of a parametric curve. If the curve is given by $(x(s), y(s))$, the basic description of its r-offset is

$$
\begin{aligned}
(x - x(s))^2 + (y - y(s))^2 - r^2 &= 0 \\
x'(s)(x - x(s)) + y'(s)(y - y(s)) &= 0
\end{aligned}
\tag{8}
$$

To eliminate extraneous factors, the second equation in the system should be divided by the gcd of $x'(s)$ and $y'(s)$. As shown in Farouki and Neff (1989), if $x(s)$ and $y(s)$ are polynomials, then the formulation (8) is faithful. However, when the coordinate functions $x(s)$ and $y(s)$ are rational, then division by the gcd need not eliminate all extraneous solutions.

The r-offset of implicit curves and of parametric and implicit surfaces is defined analogously. Given the parametric surface

$$
\begin{aligned}
x &= x(s, t) \\
y &= y(s, t) \\
z &= z(s, t)
\end{aligned}
$$

its r-offset is defined by the three equations

$$
\begin{aligned}
X^2 + Y^2 + Z^2 - r^2 &= 0 \\
x_s(s, t)\, X + y_s(s, t)\, Y + z_s(s, t)\, Z &= 0 \\
x_t(s, t)\, X + y_t(s, t)\, Y + z_t(s, t)\, Z &= 0
\end{aligned}
\tag{9}
$$

where $X = x - x(s, t)$, $Y = y - y(s, t)$, and $Z = z - z(s, t)$. The subscripts denote partial differentiation.

The interpretation of the equations is exactly as the interpretation of equations (3) and (4) before. Again, the presence of singularities introduces extraneous factors. Their elimination seems to be much more complicated. This occurs when the two tangent vectors

$$
\begin{aligned}
(x_s(s, t), y_s(s, t), z_s(s, t)) \\
(x_t(s, t), y_t(s, t), z_t(s, t))
\end{aligned}
$$

become linearly dependent or vanish outright.

12

Given the implicit curve $f(x,y) = 0$, we obtain its r-offset from the three equations

$$
\begin{aligned}
(x-u)^2 + (y-v)^2 - r^2 &= 0 \\
f(u,v) &= 0 \\
f_v(x-u) - f_u(y-v) &= 0
\end{aligned}
\tag{10}
$$

The first two equations specify that an offset point (x,y) has the Euclidean distance r from its corresponding "footpoint" (u,v), on the curve f. The last equation says that the direction vector from footpoint to offset point is perpendicular to the curve tangent $(f_v, -f_u)$. At singularities, where f_u and f_v vanish simultaneously, extraneous factors are generated in this formulation.

Finally, given the implicit surface $f(x,y,z) = 0$, we obtain its r-offset from the following four equations

$$
\begin{aligned}
(x-u)^2 + (y-v)^2 + (z-w)^2 - r^2 &= 0 \\
f(u,v,w) &= 0 \\
f_v(x-u) - f_u(y-v) &= 0 \\
f_w(y-v) - f_v(z-w) &= 0
\end{aligned}
\tag{11}
$$

Here, we have chosen $(f_v, -f_u, 0)$ and $(0, f_w, -f_v)$ as two tangent directions to which the direction vector $(x-u, y-v, z-w)$, from the footpoint (u,v,w) to the offset point (x,y,z), must be perpendicular. Singularities introduce extraneous factors, as do those curves on the surface $f = 0$ along which the tangent vectors $(f_v, -f_u, 0)$ and $(0, f_w, -f_v)$ become linearly dependent.

In summary, it is relatively simple to formulate offsets to parametric or implicit curves and surfaces, but the formulations are as systems of equations, rather than a single implicit equation. Moreover, the presence of singularities, and special relationships between partial derivatives of the surface, introduce extraneous factors, so that these formulations are not faithful. For the purpose of elimination, the extraneous factors are unwanted, and further research is needed to uncover simple ways to eliminate them conveniently, in a manner analogous to the gcd computation done in the case of polynomial parametric curves.

4.2. CONSTANT-RADIUS BLENDING SURFACES

Given two surfaces f and g, a *blending surface* is any surface that is tangent to f and to g, along two specified link curves. If no additional requirements are placed on a blending surface, then there are fairly simple methods to construct such surfaces for implicit primary surfaces, to any prescribed order of continuity, as described in Hoffmann and Hopcroft (1985, 1986, and 1987a).

It may be required that a blending surface have circular cross-sections, of fixed

13

radius. Such blending surfaces are called *constant-radius blends*, and arise in the design of ducting and many mechanical parts, as well as in certain problems arising in numerically-controlled machining. Constant-radius blends may serve to facilitate air or fluid flow, or to relieve stress. See, e.g., Rossignac and Requicha (1984). Variable-radius blends can also be defined, and are used in engineering applications; e.g., Chandru, Dutta and Hoffmann (1989a,b). We do not consider them here.

Constant-radius blends can be conceptualized as follows. Consider a family of spheres of fixed radius. Position the spheres such that each one touches both surfaces to be blended. Then the envelope surface of the family defines a surface that, after suitable trimming, is the desired constant-radius blending surface.

This conceptual procedure can be formalized in algebraic terms as follows. Let f and g be the surfaces to be blended. Note that each sphere in the family has its center on the r-offset of f, since it touches f. Similarly, each sphere has its center also on the r-offset of g. In consequence, the center of each sphere must lie on the intersection curve of the r-offsets of f and of g. This curve is called the *spine* of the blending surface. Thus, we place on each point of the spine a sphere of radius r, and describe the envelope surface of these spheres by the following equations:

$$(x - u)^2 + (y - v)^2 + (z - w)^2 - r^2 = 0 \tag{12}$$

$$F(u, v, w) = 0 \tag{13}$$

$$G(u, v, w) = 0 \tag{14}$$

$$(x - u, y - v, z - w) \cdot \mathbf{t}(u, v, w) = 0 \tag{15}$$

Here, F is the equation of the r-offset of f, and G is the equation of the r-offset of g. Moreover, $\mathbf{t}(u, v, w)$ is the tangent to the spine of the blend, at the point (u, v, w). We explain how to obtain suitable expressions for these quantities later.

The equations are closely analogous to the equations for offsetting curves and surfaces. They are interpreted as follows. Equation (12) states that the distance of the point (x, y, z) on the envelope surface, from the center of the sphere at (u, v, w), is r. It is essentially the equation of one of the spheres of the family to which we are computing the envelope.

Equations (13) and (14) say that the center of each sphere lies on the r-offsets of f and of g. Later, we will replace these equations with other equations that collectively describe both offsets. Equation (15), finally, states that the points on the envelope are precisely those points of the sphere that lie in a direction perpendicular to the spine tangent. In the literature on envelope surfaces this fact has been expressed by stating that the envelope points are on the intersection of two consecutive spheres. The curve tangent $\mathbf{t}$ is simply the cross product of the normals of F and of G.

In simple cases, the offset equations of F and of G are known explicitly. When they are not available, however, then F and G can be replaced by the systems of equations that we derived before as the description of r-offsets. Of course, this requires renaming

14

some of the variables. In this case, the surface normals to F and to G must be determined indirectly. But this is easy because at nonsingular offset points the normal of the r-offset is equal to the normal of the original surface, at the corresponding footpoint. Since footpoints are explicit in the equation systems describing offsets, the needed offset normals can be readily formulated.

The description of a constant-radius blend by this method is fairly involved, so we give an example to firm up the underlying ideas. Consider blending the implicit surface $f(x,y,z) = 0$ and the parametric surface $g = (x(s,t), y(s,t), z(s,t))$, with a constant-radius blend. The offset of f is formulated as the system (11), and the offset of g as the system (9). Renaming variables as needed, equations (12), (13), and (14) are then:

$$(x - u)^2 + (y - v)^2 + (z - w)^2 - r^2 = 0$$

$$
\begin{aligned}
(u - u_1)^2 + (v - v_1)^2 + (w - w_1)^2 - r^2 &= 0 \\
f(u_1, v_1, w_1) &= 0 \\
f_{v_1}(u - u_1) - f_{u_1}(v - v_1) &= 0 \\
f_{w_1}(v - v_1) - f_{v_1}(w - w_1) &= 0
\end{aligned}
\tag{16}
$$

$$
\begin{aligned}
X^2 + Y^2 + Z^2 - r^2 &= 0 \\
x_s(s,t)\,X + y_s(s,t)\,Y + z_s(s,t)\,Z &= 0 \\
x_t(s,t)\,X + y_t(s,t)\,Y + z_t(s,t)\,Z &= 0
\end{aligned}
$$

where $X = u - x(s,t)$, $Y = v - y(s,t)$, and $Z = w - z(s,t)$.

We now formulate equation (15) explicitly. At the point (u,v,w), the normal to the r-offset of f is $(f_{u_1}, f_{u_2}, f_{u_3})$. By the construction of offsets, it is equal to the vector from the footpoint to (u,v,w), and is therefore $(u - u_1, v - v_1, w - w_1)$. Similarly, the normal to the r-offset of g is (X, Y, Z), at (u, v, w). In consequence, the tangent to the spine of the blend is

$$\mathbf{t} = (u - u_1, v - v_1, w - w_1) \times (X, Y, Z)$$

Therefore, the last equation for specifying the constant-radius blend is

$$(x - u, y - v, z - w) \cdot (u - u_1, v - v_1, w - w_1) \times (X, Y, Z) = 0 \tag{17}$$

Equations (16) and (17) together describe the constant-radius blend between the implicit surface f and the parametric surface g. This description involves 9 equations in 11 variables. It is clear that blends between two implicit or two parametric surfaces may be constructed in much the same way.

15

5. VARIABLE ELIMINATION

A possible strategy for obtaining closed-form expressions for offsets and blends is to formulate the equation systems from before, and to eliminate all additional variables. Several techniques for variable elimination are available; namely, resultant-based methods, and Gröbner bases methods. On theoretical grounds, both methods have exponential behavior. Moreover, many practical problem instances do lead to excessive running times, so that this general approach is not attractive. On the other hand, these methods yield research insights, and in those cases where elimination is a preprocessing step, long running times may be acceptable. Therefore, we give a brief description. For details see Hoffmann (1989b).

5.1. RESULTANT METHODS

By far the best-known approach to eliminate variables is based on resultants. Several formulations are known; e.g., Sederberg (1983). In particular, the *Sylvester resultant* requires evaluating a determinant whose entries are coefficients of the powers of the variable to be eliminated. Several variables can be eliminated by successive phases of single-variable eliminations.

As illustration of the Sylvester resultant, consider eliminating y from the two polynomial equations $f(x,y) = 0$ and $g(x,y) = 0$. We write the polynomials f and g as polynomials in y. The coefficient of each power of y is a polynomial in x. For example, the polynomial $x^2y^2 - 2xy^2 + y - x^2 + 1$ is viewed as the polynomial $a_2y^2 + a_1y + a_0$, where $a_2 = x^2 - 2x$, $a_1 = 1$, and $a_0 = 1 - x^2$. Having written f as

$$f = a_ny^n + a_{n-1}y^{n-1} + \cdots + a_1y + a_0$$

and g as

$$g = b_my^m + b_{m-1}y^{m-1} + \cdots + b_1y + b_0$$

we form the $(m + n) \times (m + n)$ determinant

$$Res_y(f,g) = \begin{vmatrix} a_n & a_{n-1} & \cdots & a_0 & 0 & \cdots & 0 \\ 0 & a_n & \cdots & a_1 & a_0 & \cdots & 0 \\ \vdots & & \ddots & & & \ddots & \\ 0 & \cdots & 0 & a_n & a_{n-1} & \cdots & a_0 \\ b_m & b_{m-1} & \cdots & b_0 & 0 & \cdots & 0 \\ 0 & b_m & \cdots & b_1 & b_0 & \cdots & 0 \\ \vdots & & \ddots & & & \ddots & \\ 0 & \cdots & 0 & b_m & b_{m-1} & \cdots & b_0 \end{vmatrix}$$

16

$Res_y(f,g)$ is a polynomial in x alone, and is called the *Sylvester resultant* of f and g. It can be shown that, for every point (u,v) satisfying simultaneously $f(u,v) = 0$ and $g(u,v) = 0$, the polynomial $Res_y(f,g)$ has the root u. The polynomial $Res_y(f,g)$ may have additional roots that do not have corresponding solutions of the original system.

We give an example that shows that the Sylvester resultant may generate extraneous solutions that do not correspond to solutions of the original system. Consider the following three equations

$$
\begin{aligned}
x^2 + y^2 + 2x &= 0 & (18) \\
x^2 + y^2 - 2yz &= 0 & (19) \\
x + 2z &= 0 & (20)
\end{aligned}
$$

Equation (18) defines a cylinder, equation (19) defines a cone, and equation (20) an inclined plane. The cylinder and the cone intersect in a curve consisting of the line $x = y = 0$ and a twisted cubic with the parameterization

$$
\begin{aligned}
x(s) &= \frac{-2s^2}{1+s^2} \\
y(s) &= \frac{2s}{1+s^2} & (21) \\
z(s) &= s
\end{aligned}
$$

As explained later, the inclined plane intersects this curve in three points; namely, the point $(0,0,0)$, and two other points with complex coordinates. The intersection of the cylinder and the cone, and the intersection of the plane and the cylinder are shown in Figure 12.

We eliminate x from equations (18) and (19) using the Sylvester resultant, and obtain the equation

$$
4y(yz^2 + y - 2z) = 0 \qquad (22)
$$

This is the equation of the orthogonal projection of the cylinder/cone intersection, into the (y,z)-plane. Next, we eliminate x from the two equations (18) and (20), obtaining

$$
4z^2 - 4z + y^2 = 0 \qquad (23)
$$

This is the equation of the projection of the cylinder/plane intersection, also into the (y,z)-plane. Both projections are shown in Figure 13. They have more than one real intersection, but two of them, marked in the figure, do not correspond to a surface intersection in 3-space.
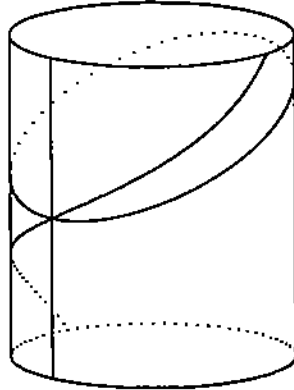
Figure 12: Intersections of the Cone and the Inclined Plane with the Cylinder

We eliminate z from the derived equations, (22) and (23), and obtain

$$16y^3(y-2)^2(y^3 + 4y^2 + 4y - 8) = 0 \qquad (24)$$

We analyze the roots of this equation. The triple root $y = 0$, due to the first factor, corresponds to the double intersection of the curves at the origin, plus the apparent intersection of (22) and (23), at $(y, z) = (0, 1)$. The latter solution does not extend to a solution of the system (18), (19), and (20). The double root $y = 2$ extends to two complex solutions of the original system. The last factor, finally, has a root $y \approx 0.9311425$ that again is an apparent intersection of (22) and (23), but cannot be extended to a solution of the original system. Its other two roots are also extraneous, but are complex.

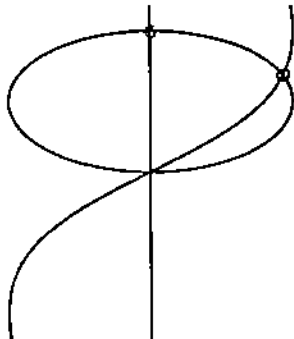It is not hard to verify that the original system has three distinct solutions, one of

Figure 13: Projection into the (y, z)-Plane

18

which is a double intersection. The cubic component of the cylinder/cone intersection, parameterized by formula (21), intersects the plane (20) in exactly three points. Two of them are complex points, the third is the origin. The other component, the line $x = y = 0$, intersects the plane at the origin. Thus, four points should have been described by the univariate polynomial obtained from eliminating x and z. However, the polynomial (24) has degree 8, thus contains an extraneous factor of degree 4. This extraneous factor is

$$y(y^3 + 4y^2 + 4y - 8)$$

As we discuss later, Gröbner bases methods do not generate such factors.

The possibility of additional solutions is the *extraneous factors problem*, and it implies that the Sylvester resultant does not provide a faithful method for eliminating variables. Techniques have been developed to compensate for this fact by determining several resultants, each computed after a different linear transformations of the original system, as described in Garrity and Warren (1988). Efficiency considerations would cast doubt on the practicality of such techniques.

5.2. GRÖBNER BASES METHODS

Given a system of linear equations, manipulations such as LU-decomposition bring the system into a form that facilitates finding its solutions. In analogy, a Gröbner basis of a system of algebraic equations can be viewed as an equivalent system of algebraic equations that facilitates finding the solutions of the nonlinear system. Gröbner bases were introduced in Buchberger (1965).

Gröbner bases methods can be used to eliminate variables. In contrast to resultant-based methods, Gröbner bases methods are faithful and do not introduce additional solutions to the original problem. A detailed description of Gröbner bases methods goes beyond the scope of these notes. For a general introduction to the subject see Buchberger (1985). A self-contained introduction that discusses applications to geometric modeling can be found in Hoffmann (1989b), Chapter 7.

Two basic elimination algorithms using Gröbner bases are known. The first algorithm constructs a Gröbner basis with respect to a term ordering known as the *elimination order*. Its effect on a system of algebraic equations is analogous to Gaussian elimination in linear equations; Lazard (1983).

As an example of the method, we reconsider finding solutions to the equations (18), (19), and (20). We construct the Gröbner basis using the elimination order with

19

$y \prec z \prec x$, obtaining the four polynomials

$$-y^3 + 2y^2$$
$$(y - 2)z$$
$$4z^2 - 4z + y^2$$
$$x + 2z$$

The first polynomial, $-y^3 + 2y^2$, has the roots 0 and 2. Each value is substituted into the remaining polynomials, and the common roots define the solutions to the original system. For example, the z-values belonging to $y = 2$ are the roots of $4z^2 - 4z + 4$, namely $(-1 \pm i\sqrt{3})/2$. For each z-value, substitution into $x + 2z$ yields the x-coordinate. For more details see Hoffmann (1989b), Chapter 7, Section 7.4. Note that the method will generate only coordinate values that correspond to solutions of the original system. It is therefore a faithful elimination method.

The second elimination algorithm, described in Hoffmann (1989b), Chapter 7, Section 7.8, requires additional information to ensure termination. It evolved from a basis conversion algorithm described in Faugère, Gianni, Lazard, and Mora (1989), and is also faithful.

The algorithm first constructs a Gröbner basis with respect to a term ordering that is different from the elimination order and is much more efficient. Thereafter, the algorithm constructs the final polynomial in which all variables have been eliminated, by constructing it term-by-term. The algorithm seems to be the fastest elimination method for geometry applications that has been implemented to date.

Although elimination based on Göbner bases is an improvement over previously known methods, variable elimination, with this approach, must not be considered a routine step in geometric modeling applications. The following figures illustrate the present situation. Three problems are considered: implicitizing a parametric quadratic, a parametric cubic, and a bicubic surface. In each case, the two parametric variables must be eliminated from three algebraic equations in five variables. Three elimination methods are compared:

Method 1: Implicitization using the Sylvester resultant.

Method 2: Implicitization with a Gröbner basis using the elimination order.

Method 3: Implicitization using basis conversion.

The parametric quadric is

$$
\begin{aligned}
x &= 3t^2 + 4s^2 + st - 2s - 5t + 4 \\
y &= 6s^2 - st + 8t + 7 \\
z &= 9st + 12s - 15t + 34
\end{aligned}
$$

Problem	Method 1	Method 2	Method 3
quadratic	21	22	6
cubic	10^5	∞	315
bicubic	∞	∞	10^5

Table 1: Implicitization Times in Seconds

The parametric cubic is

$$
\begin{aligned}
x &= -t^3 + 3st + s^3 + s \\
y &= ts^2 - 3t + 1 \\
z &= 2t^3 - 5st + t - s^3
\end{aligned}
$$

The bicubic surface, finally, is

$$
\begin{aligned}
x &= 3t(t-1)^2 + (s-1)^3 + 3s \\
y &= 3s(s-1)^2 + t^3 + 3t \\
z &= -3s(s^2 - 5s + 5)t^3 - 3(s^3 + 6s^2 - 9s + 1)t^2 \\
  &\quad + t(6s^3 + 9s^2 - 18s + 3) - 3s(s-1)
\end{aligned}
$$

The timings are as shown in the table. All computations were done on a Symbolics 3650 Lisp machine with 16MB main memory and 120 MB virtual memory. Methods 1 and 2 are the standard implementation of resultants and Gröbner bases offered by Macsyma 414.62. Method 3 is an experimental implementation by J.-H. Chuang and B. Bouma. The entry ∞ indicates that the computation could not be completed due to insufficient virtual memory.

5.3. EFFICIENCY OF VARIABLE ELIMINATION

Table 1 shows very clearly that the traditional methods, Method 1 and 2, do not have satisfactory efficiency. In contrast, Method 3 improves efficiency significantly, viz. the implicitization of the parametric cubic. One reason why this is so is that Methods 1 and 2 are more general: Suppose we want to eliminate 2 variables from 3 algebraic equations in n variables. Unless additional information is given, there might not exist a single algebraic equation in $n - 2$ variables. Methods 1 and 2 will uncover this fact, while Method 3 will not.[3] The table shows quite dramatically that there is a huge

---

[3]Method 3 loops forever in such a situation. The method is applicable precisely in those situations in which the existence of a polynomial in fewer variables is known beforehand.

21

payoff in efficiency for this diminished generality. It remains an important challenge to explore such trade-offs in greater depth. The basis conversion algorithm of Faugère et al. (1989) and its variant, Method 3, are examples of such exploration.

6. AVOIDING COMPLETE ELIMINATION

On the one hand, we can formulate offsets and blending surfaces fairly routinely. On the other hand, deriving closed-form algebraic equations for these surfaces remains an expensive operation at this time. So, we explore the possibility of bypassing variable elimination as a step in specifying and working with offsets and blends.

In particular, we view the systems of equations formulated in Sections 2 and 4 as the final representation of a curve or surface. So, we consider a system of $n - 2$ algebraic equations in n variables as the representation of a surface, and a system of $n - 1$ algebraic equations in n variables as the representation of a curve. For such representations to be a sensible alternative, we need to develop algorithms for the basic manipulations and interrogations of surfaces so represented. We consider several of these operations now.

6.1. SURFACE INTERSECTION

In Bajaj, Hoffmann, Hopcroft and Lynch (1988), a method for tracing surface intersections has been presented. The method is an adaptive marching scheme that proceeds as follows:

1. At the point p on the surface intersection, an underdetermined linear system is formulated whose solution determines a local parametric approximation of the intersection curve. Since the system is underdetermined, choices can be made which determine, for instance, the speed of the parameterization.

2. Using the approximant, and a step length, a new curve point estimate is derived. A heuristic analyzes the relative magnitude of higher-order terms in the approximant, and estimates a safe step length.

3. The curve point estimate is refined using Newton iteration, thereby obtaining a new point on the surface intersection.

Careful attention to the details of this procedure results in a tracing algorithm that is quite robust and flexible.

The method is easily generalized to the situation at hand here. That is, given $n - 1$ algebraic hypersurfaces in n-dimensional space, their intersection is, in general, a curve in n-dimensional space, and this curve can be traced by the method from

22

Bajaj, Hoffmann, Hopcroft and Lynch (1988) without any essential modifications, whether $n = 2$, $n = 3$, or $n > 3$.

Hoffmann (1989a) reports a number of experiments with surface intersection in n-dimensional space. It was found that curves of algebraic degrees well over 150 can be traced with 10 significant decimals using ordinary double-precision arithmetic. Using second-order approximants to derive new point estimates, typically two to three Newton iterations achieve such accuracy. We review the details of the method in Section 7.

6.2. STARTING POINTS

In unpublished work, Chiang and Hoffmann investigate methods to find initial points on surface intersections in higher-dimensional spaces. Briefly, by introducing one additional variable, all implicit equations in the system can be converted into Bernstein-Bezier form; see, e.g., Waggenspack and Anderson (1986). Then, ordinary subdivision methods can be applied to find initial points on the various branches. One of the critical aspects is to preserve the sparseness a polynomial might have, since otherwise the number of needed control points becomes too large.

6.3. LOCAL SURFACE APPROXIMATION

In unpublished work, Chuang and Hoffmann have investigated techniques construct local surface approximants from the system of equations. That is, given $n-2$ algebraic equations in n variables, where $n > 3$, algorithms have been developed that construct parametric approximants to the surface in (x, y, z)-space described by the system of equations. These approximants have the form

$$
\begin{aligned}
x &= h_1(s, t) \\
y &= h_2(s, t) \\
z &= h_3(s, t)
\end{aligned}
\tag{25}
$$

or the form

$$
z = h_4(x, y)
$$

The derivation of the parametric approximant (25) is remarkably similar in detail to the derivation of the approximant used in the curve tracing algorithm. We describe some of the details in Section 8. The special case of locally approximating a parametric surface with an implicit form has been considered in Chuang and Hoffmann (1989).

23

7. TRACING SURFACE INTERSECTIONS NUMERICALLY

We are given $n - 1$ algebraic equations in n variables,

$$
\begin{aligned}
f_1(x_1, x_2, ..., x_n) &= 0 \\
f_2(x_1, x_2, ..., x_n) &= 0 \\
&\vdots \\
f_{n-1}(x_1, x_2, ..., x_n) &= 0
\end{aligned}
\tag{26}
$$

We are also given a point $p = (u_1, u_2, ..., u_n)$ that lies on each hypersurface f_i. We assume that the hypersurfaces f_i are not singular at p, and that they intersect transversally. Under these assumptions, the algebraic set, described by (26), is locally a space curve and is analytic in the neighborhood of p. We consider tracing this curve numerically, in the manner outlined before.

Technically, tracing involves the process of setting up and solving a system of linear equations from which a curve approximant is derived. Thereafter, a step size is determined adaptively, a new curve point estimate constructed, and the new point estimate is refined iteratively to a point on the intersection of the hypersurfaces. We sketch the derivation of this procedure. More detail can be found in Hoffmann (1989b), Chapter 6.

7.1. FINDING THE CURVE APPROXIMANT

We consider the intersection curve in the vicinity of the known point p, and expand the curve using Taylor's theorem. In vectorial notation, the curve is locally given by

$$
\mathbf{r}(s) = \mathbf{r}(0) + s\mathbf{r}'(0) + \frac{s^2}{2}\mathbf{r}''(0) + \frac{s^3}{6}\mathbf{r}'''(0) + \cdots
$$

where $p = \mathbf{r}(0)$, the first derivative at p is $\mathbf{r}'(0)$, and so on. By determining the derivatives $\mathbf{r}'$, $\mathbf{r}''$, etc., up to a certain order, we obtain a local approximation of the intersection curve.

Consider the Taylor expansion of each surface f_i, at p. Let $f_i^{(k)}$ denote the partial derivative of f_i by x_k, and $f_i^{(k,j)}$ denote the partial derivative of $f_i^{(k)}$ by x_j. Recall

that $f_i^{(k,j)} = f_i^{(j,k)}$. Then

$$
\begin{aligned}
f_i(x_1,...,x_n) &= f_i(u_1 + \delta_1, u_2 + \delta_2, ..., u_n + \delta_n) \\
&= f_i(u_1,...,u_n) \\
&\quad + f_i^{(1)}\delta_1 + \cdots + f_i^{(n)}\delta_n \\
&\quad + [f_i^{(1,1)}\delta_1^2 + \cdots + f_i^{(n,n)}\delta_n^2]/2 \\
&\quad + f_i^{(1,2)}\delta_1\delta_2 + \cdots + f_i^{(1,n)}\delta_1\delta_n \\
&\quad + f_i^{(2,3)}\delta_2\delta_3 + \cdots + f_i^{(n-1,n)}\delta_{n-1}\delta_n \\
&\quad + \text{higher order terms}
\end{aligned}
$$

for real numbers $\delta_1,...,\delta_n$. Note that all partial derivatives of f_i are evaluated at $p = (u_1,...,u_n)$.

Let $\mathbf{r}_i$ denote the ith component of $\mathbf{r}$. We set

$$
\begin{aligned}
\delta_1 &= \mathbf{r}_1's + \mathbf{r}_1''s^2/2 + \cdots \\
\delta_2 &= \mathbf{r}_2's + \mathbf{r}_2''s^2/2 + \cdots \\
&\vdots \\
\delta_n &= \mathbf{r}_n's + \mathbf{r}_n''s^2/2 + \cdots
\end{aligned}
$$

We substitute for δ_k in the Taylor expansion of f_i, for $i = 1,...,n-1$. Since $f_i(\mathbf{r}(s)) \equiv 0$, the coefficient of each power s^a should be zero, from which we obtain equations of the form

$$
f_i^{(1)}\mathbf{r}_1' + f_i^{(2)}\mathbf{r}_2' + \cdots + f_i^{(n)}\mathbf{r}_n' = 0 \tag{27}
$$

$$
f_i^{(1)}\mathbf{r}_1'' + f_i^{(2)}\mathbf{r}_2'' + \cdots + f_i^{(n)}\mathbf{r}_n'' = -b_i \tag{28}
$$

for $i = 1, 2, ..., n-1$. The quantities b_i are given by

$$
\begin{aligned}
b_i = \quad & f_i^{(1,1)}\mathbf{r}_1'^2 + f_i^{(2,2)}\mathbf{r}_2'^2 + \quad \cdots \quad + f_i^{(n,n)}\mathbf{r}_n'^2 \\
+ 2[\, & f_i^{(1,2)}\mathbf{r}_1'\mathbf{r}_2' + f_i^{(1,3)}\mathbf{r}_1'\mathbf{r}_3' + \quad \cdots \quad + f_i^{(1,n)}\mathbf{r}_1'\mathbf{r}_n' \\
& + f_i^{(2,3)}\mathbf{r}_2'\mathbf{r}_3' + \quad \cdots \quad + f_i^{(2,n)}\mathbf{r}_2'\mathbf{r}_n' \\
& \ddots \\
& + f_i^{(n-1,n)}\mathbf{r}_{n-1}'\mathbf{r}_n'\,]
\end{aligned}
$$

From these equations we determine the first and second derivative of $\mathbf{r}$.

Similar equations are derived from the coefficients of higher powers of s, and they determine higher-order derivatives of $\mathbf{r}$. But the equations quickly grow in

25

complexity, and the added accuracy in the approximant, obtained from the higher-order derivatives, is offset by the difficulty of evaluating the equations. There seem to be no systematic studies of this accuracy vs. efficiency trade-off.

The linear equations (27) and (28) from which to compute $\mathbf{r}'$ and $\mathbf{r}''$ are under-determined. Their solution takes the form

$$\alpha_1 \nabla f_1 + \alpha_2 \nabla f_2 + \cdots \alpha_{n-1} \nabla f_{n-1} + \beta \mathbf{t}$$

where β can be chosen arbitrarily. Note that $\mathbf{t}$ is the curve tangent, and is perpendicular to the gradients ∇f_i, for $i = 1, ..., n - 1$.

We solve these linear systems numerically using singular value decomposition algorithm; e.g., Golub and Van Loan (1983). The choices for β are as follows. When solving (27), we have $\alpha_1 = \alpha_2 = \cdots = \alpha_{n-1} = 0$. We choose β such that the vector $\mathbf{r}'$ has unit length. The sign of β determines the direction of the trace. When solving (28), not all α_i are zero. Here we may choose $\beta = 0$. In consequence, $\mathbf{r}'$ is perpendicular to $\mathbf{r}''$. After determining $\mathbf{r}'$ and $\mathbf{r}''$, and, possibly, higher-order derivatives of $\mathbf{r}$, we have a local approximant to the surface intersection.

7.2. STEPPING

After constructing an approximant $\mathbf{r}(s)$ to the curve at p, we must choose a step length s_0 to obtain a subsequent curve-point estimate $\mathbf{r}(s_0)$. Choosing a safe step length requires understanding the radius of convergence of the full Taylor series. In practice, estimating the radius of convergence is difficult except in those cases where simple recurrences or closed-form expressions can be given. So, we use a heuristic instead, in which the contribution of the second-order term in the Taylor series, to the next point estimate, is kept small. Since $\mathbf{r}'$ has unit length, we may choose s_0, for example, such that

$$\frac{\|s_0^2 \mathbf{r}''(0)\|}{2} < \frac{|s_0|}{10}$$

This simple strategy does well in many cases. After s_0 has been fixed, we obtain the new curve point estimate $q = \mathbf{r}(s_0)$.

7.3. NEWTON ITERATION

Using Newton iteration, we refine the estimate q until we are on the intersection of the surfaces f_i with acceptable accuracy. Let $q_0 = q$, and set $q_{k+1} = q_k + \Delta_k$, where $\Delta_k = (\delta_1, \delta_2, ..., \delta_n)$. The iteration is based on the following, first-order approximation of the surfaces

$$\nabla f_i(q_k) \cdot \Delta_k = -f_i(q_k)$$

26

Note that this system has the same structure as does system (27). As in the approximant construction, we solve the linear system using singular value decomposition. For the solution Δ_k, we set the undetermined coefficient β of t to zero, since it represents lateral movement that will not improve the quality of the new estimate substantially. We continue the iteration until

$$\|q_{k+1} - q_k\| < 10^{-t}\|q_k\|$$

where t determines the number of significant digits. Typically, we have $t = 10$ for double-precision floating-point computations, and we require two or three iterations to achieve this accuracy. Of course, the number of iterations will depend on the step size s_0 that we have chosen, and on the local geometry of the curve.

8. LOCAL APPROXIMATIONS

We are given $n - 2$ algebraic equations in n variables, and a point $p = (u_1, u_2, ..., u_n)$ satisfying each equation. The given equations are

$$
\begin{aligned}
f_1(x_1, x_2, ..., x_n) &= 0 \\
f_2(x_1, x_2, ..., x_n) &= 0 \\
&\vdots \\
f_{n-2}(x_1, x_2, ..., x_n) &= 0
\end{aligned}
\tag{29}
$$

Assuming the equations are algebraically independent, that p is a regular point on every f_i, and that the corresponding hypersurfaces intersect transversally at p, there is a neighborhood of p in which we can define coordinate functions

$$
\begin{aligned}
x_1 &= h_1(s, t) \\
x_2 &= h_2(s, t) \\
&\vdots \\
x_n &= h_n(s, t)
\end{aligned}
\tag{30}
$$

such that

$$p = (h_1(0,0), ..., h_n(0,0))$$

and

$$f_i(h_1(s,t), h_2(s,t), ..., h_n(s,t)) \equiv 0$$

for $i = 1, ..., n - 2$. We wish to construct $h_1, ..., h_n$ from the given equations.

27

Conceptually, we consider a surface in (x_1, x_2, x_3)-space defined by the system (29). Rather than determining a closed-form equation, by eliminating the variables $x_4, ..., x_n$, we may determine a local approximant to the surface. The approximant consists of the three functions

$$
\begin{aligned}
x_1 &= h_1(s, t) \\
x_2 &= h_2(s, t) \\
x_3 &= h_3(s, t)
\end{aligned}
$$

We might call such an approximant a *local parameterization* of the surface.

The approach to determining the local coordinate functions $x_i = h_i(s, t)$ is closely analogous to the determination of a local approximant to the intersection of $n - 1$ hypersurfaces in n-space, described in Section 7. Consider the Taylor expansion of the hypersurface f_i at the point $p = (u_1, ..., u_n)$

$$
\begin{aligned}
f_i(x_1, ..., x_n) &= f_i(u_1 + \delta_1, u_2 + \delta_2, ..., u_n + \delta_n) \\
&= f_i(u_1, ..., u_n) \\
&\quad + f_i^{(1)}\delta_1 + \cdots + f_i^{(n)}\delta_n \\
&\quad + [f_i^{(1,1)}\delta_1^2 + \cdots + f_i^{(n,n)}\delta_n^2]/2 \\
&\quad + f_i^{(1,2)}\delta_1\delta_2 + \cdots + f_i^{(1,n)}\delta_1\delta_n \\
&\quad + f_i^{(2,3)}\delta_2\delta_3 + \cdots + f_i^{(n-1,n)}\delta_{n-1}\delta_n \\
&\quad + \text{higher order terms}
\end{aligned}
\tag{31}
$$

Moreover, consider the Taylor expansion of the local coordinate functions, assuming a choice of variables s and t such that

$$
p = (h_1(0,0), h_2(0,0), ..., h_n(0,0))
$$

We obtain

$$
\begin{aligned}
h_k(s, t) &= h_k(0,0) \\
&\quad + h_k^{(s)}s + h_k^{(t)}t \\
&\quad + [h_k^{(s,s)}s^2 + 2h_k^{(s,t)}st + h_k^{(t,t)}t^2]/2 \\
&\quad + \text{higher order terms}
\end{aligned}
$$

for $k = 1, ..., n$, where $h_k^{(s)}$ denotes the partial derivative of h_k by s, and so on. All partial derivatives are evaluated at $(s, t) = (0, 0)$. By assumption, there is a neighborhood of p in which

$$
f_i(h_1(s, t), h_2(s, t), ..., h_n(s, t)) \equiv 0
$$

28

We set

$$
\begin{aligned}
\delta_1 &= h_1^{(s)}s + h_1^{(t)}t + [h_1^{(s,s)}s^2 + 2h_1^{(s,t)}st + h_1^{(t,t)}t^2]/2 + \cdots \\
\delta_2 &= h_2^{(s)}s + h_2^{(t)}t + [h_2^{(s,s)}s^2 + 2h_2^{(s,t)}st + h_2^{(t,t)}t^2]/2 + \cdots \\
&\vdots \\
\delta_n &= h_n^{(s)}s + h_n^{(t)}t + [h_n^{(s,s)}s^2 + 2h_n^{(s,t)}st + h_n^{(t,t)}t^2]/2 + \cdots
\end{aligned}
$$

Note that

$$
\begin{aligned}
\delta_k^2 &= (h_k^{(s)})^2 s^2 + 2h_k^{(s)}h_k^{(t)}st + (h_k^{(t)})^2 t^2 + \cdots \\
\delta_k \delta_j &= h_k^{(s)}h_j^{(s)}s^2 + (h_k^{(s)}h_j^{(t)} + h_k^{(t)}h_j^{(s)})st + h_k^{(t)}h_j^{(t)}t^2 + \cdots
\end{aligned}
$$

We substitute these quantities into (31), obtaining a power series in s and t. The coefficient of each term $s^a t^b$ in the power series must be identically zero, so that we obtain the following systems of equations:

$$
f_i^{(1)}h_1^{(s)} + f_i^{(2)}h_2^{(s)} + \cdots + f_i^{(n)}h_n^{(s)} = 0 \tag{32}
$$

$$
f_i^{(1)}h_1^{(t)} + f_i^{(2)}h_2^{(t)} + \cdots + f_i^{(n)}h_n^{(t)} = 0 \tag{33}
$$

$$
f_i^{(1)}h_1^{(s,s)} + f_i^{(2)}h_2^{(s,s)} + \cdots + f_i^{(n)}h_n^{(s,s)} = -c_i \tag{34}
$$

$$
f_i^{(1)}h_1^{(s,t)} + f_i^{(2)}h_2^{(s,t)} + \cdots + f_i^{(n)}h_n^{(s,t)} = -d_i \tag{35}
$$

$$
f_i^{(1)}h_1^{(t,t)} + f_i^{(2)}h_2^{(t,t)} + \cdots + f_i^{(n)}h_n^{(t,t)} = -e_i \tag{36}
$$

where $i = 1, ..., n - 2$. The righthand sides are, respectively,

$$
\begin{aligned}
c_i &= f_i^{(1,1)}(h_1^{(s)})^2 + \cdots + f_i^{(n,n)}(h_n^{(s)})^2 \\
&\quad + 2[f_i^{(1,2)}h_1^{(s)}h_2^{(s)} + \cdots + f_i^{(1,n)}h_1^{(s)}h_n^{(s)} \\
&\quad \cdots + f_i^{(n-1,n)}h_{n-1}^{(s)}h_n^{(s)}]
\end{aligned}
$$

$$
\begin{aligned}
d_i &= f_i^{(1,1)}h_1^{(s)}h_1^{(t)} + \cdots + f_i^{(n,n)}h_n^{(s)}h_n^{(t)} \\
&\quad + f_i^{(1,2)}h_1^{(s)}h_2^{(t)} + \cdots + f_i^{(1,n)}h_1^{(s)}h_n^{(t)} \\
&\quad \cdots + f_i^{(n-1,n)}h_{n-1}^{(s)}h_n^{(t)} \\
&\quad + f_i^{(1,2)}h_1^{(t)}h_2^{(s)} + \cdots + f_i^{(1,n)}h_1^{(t)}h_n^{(s)} \\
&\quad \cdots + f_i^{(n-1,n)}h_{n-1}^{(t)}h_n^{(s)}
\end{aligned}
$$

$$
\begin{aligned}
e_i &= f_i^{(1,1)}(h_1^{(t)})^2 + \cdots + f_i^{(n,n)}(h_n^{(t)})^2 \\
&\quad + 2[f_i^{(1,2)}h_1^{(t)}h_2^{(t)} + \cdots + f_i^{(1,n)}h_1^{(t)}h_n^{(t)} \\
&\quad \cdots + f_i^{(n-1,n)}h_{n-1}^{(t)}h_n^{(t)}]
\end{aligned}
$$

We determine the partial derivatives of the coordinate functions h_i from these systems of linear equations, thereby obtaining an approximate local parameterization of the surface defined by (29).

The systems of equations (32) – (36) are underdetermined, with rank deficiency 2. Their solutions, therefore, take the form

$$
\alpha_1 \nabla f_1 + \cdots + \alpha_{n-2} \nabla f_{n-2} + \beta t_1 + \gamma t_2
$$

Here, t_1 and t_2 are two linearly independent tangent directions to the surface at the point p. The tangents are determined by solution methods such as singular-value decomposition.

To choose suitable values for the β and γ coefficients, we proceed as follows. For systems (32) and (33), we have

$$
\alpha_1 = \alpha_2 = \cdots = \alpha_{n-2} = 0
$$

For (32), we choose $(\beta, \gamma) = (1, 0)$; and for (33), we choose $(\beta, \gamma) = (0, 1)$. Then the iso-parametric lines $(h_1(s, 0), ..., h_n(s, 0))$ and $(h_1(0, t), ..., h_n(0, t))$ intersect transversally at p. For the other three systems we choose $\beta = \gamma = 0$. The approximants for the three coordinate functions h_1, h_2, and h_3, obtained in this way, define a local approximation to the surface in (x_1, x_2, x_3)-space.

30

ACKNOWLEDGEMENTS

BIBLIOGRAPHY

Abhyankar, S., and C. Bajaj (1989)
  "Automatic Rational Parameterization of Curves and Surfaces IV: Algebraic Space Curves," *ACM Trans. on Graphics*, to appear.

Bajaj, C., C. Hoffmann, J. Hopcroft and R. Lynch (1988)
  "Tracing Surface Intersections," *Computer Aided Geometric Design 5*, 285–307.

Barnhill, R.E., and W. Böhm, eds. (1983)
  Surfaces in Computer-Aided Geometric Design, North-Holland, Amsterdam.

Böhm, W., G. Farin and J. Kahmann (1984)
  "A Survey of Curve and Surface Methods in CAGD," *Computer Aided Geometric Design 1*, 1–60.

Buchberger, B., (1965)
  "Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal," Dissertation, Univ. Innsbruck, Austria.

Buchberger, B., (1976)
  "A Theoretical Basis for the Reduction of Polynomials to Canonical Forms," *ACM SIGSAM Bull. 39*, 19–29.

Buchberger, B., (1985)
  "Gröbner Bases: An Algorithmic Method in Polynomial Ideal Theory," in *Multidimensional Systems Theory*, N.K. Bose, ed., D. Reidel Publishing Co., 184–232.

Buchberger, B., G. Collins and B. Kutzler (1988)
  "Algebraic Methods for Geometric Reasoning," *Annl. Reviews in Comp. Science*, Vol. 3.

Chandru, V., D. Dutta, and C. Hoffmann (1989a)
  "On the Geometry of Dupin Cyclides," *The Visual Computer 5*.

Chandru, V., D. Dutta, and C. Hoffmann (1989b)
  "Variable Radius Blending using Dupin Cyclides," in *Geometric Modeling for Product Engineering*, K. Preiss, J. Turner, M. Wozny, eds., North Holland.

Chuang, J.-H., and C. Hoffmann (1989)
"On Local Implicit Approximations of Curves and Surfaces," *ACM Trans. on Graphics*, to appear.

Farin, G., (1988)
Curves and Surfaces for Computer Aided Geometric Design, Academic Press, Boston.

Farouki, R.T., (1986)
"The Characterization of Parametric Surface Sections," *Comp. Vision, Graphics and Im. Proc. 33*, 209–236.

Farouki, R.T., and C.A. Neff (1989)
"Some Analytic and Algebraic Properties of Plane Offset Curves," Rept. RC 14364, IBM Yorktown Heights.

Farouki, R.T., and V.T. Rajan (1987)
"On the Numerical Condition of Algebraic Curves and Surfaces 1. Implicit Equations," IBM Res. Rept. RC–13263.

Faugère, J., P. Gianni, D. Lazard, T. Mora (1989)
"Efficient Change of Ordering for Gröbner Bases of Zero-Dimensional Ideals," manuscript.

Garrity, T., and J. Warren (1988)
"On Computing the Intersection of a Pair of Algebraic Surfaces," *Comp. Aided Geometric Design*, to appear.

Geisow, A., (1983)
"Surface Interrogation," *Ph.D.Diss.*, School of Computing and Accountancy, Univ. of East Anglia.

Golub, G., and C. van Loan (1983)
Matrix Computations, Johns Hopkins Press.

Hilbert, D., and Cohn-Vossen, S., (1952)
Geometry and the Imagination, Chelsea Publ. Co., New York.

Hoffmann, C., (1987)
"Algebraic Curves," in *Mathematical Aspects of Scientific Software*, J. Rice, ed., IMA Volumes in Math. and Applic., Springer Verlag, 101–122.

Hoffmann, C., (1988)
"A Dimensionality Paradigm for Surface Interrogations," TR 88-837, Comp. Sci., Purdue University.

Hoffmann, C., (1989a)

    "The Problems of Accuracy and Robustness in Geometric Computation,"
IEEE Computer 22, 31–42.

Hoffmann, C., (1989b)

    Geometric and Solid Modeling, An Introduction, Morgan Kaufmann
Publishers, San Francisco.

Hoffmann, C., and J. Hopcroft (1985)

    "Automatic Surface Generation in Computer Aided Design," *The Visual
Computer 1*, 92–100.

Hoffmann, C., and J. Hopcroft (1986)

    "Quadratic Blending Surfaces," *Comp. Aided Design 18*, 301–307.

Hoffmann, C., and J. Hopcroft (1987a)

    "The Potential Method for Blending Surfaces and Corners," in *Geometric
Modeling*, G. Farin, ed., SIAM Publications, Philadelphia.

Hoffmann, C., and J. Hopcroft (1987b)

    "Geometric Ambiguities in Boundary Representations," *Comp. Aided Design
19*, 141–147.

Hoffmann, C., and J. Hopcroft (1988)

    "Projective Blending Surfaces," *Artif. Intelligence 37*, 357–376.

Hoffmann, C., J. Hopcroft and M. Karasick (1987)

    "Robust Set Operations on Polyhedral Solids," Tech. Rept. 723, Comp. Sci.,
Purdue University, to appear in *IEEE Comp. Graphics and Applic.*

Hoffmann, C., J. Hopcroft and M. Karasick (1988)

    "Towards Implementing Robust Geometric Computations," 4th *ACM
Symp. on Comp. Geometry*, 106–117.

Lazard, D., (1983)

    "Gröbner Bases, Gaussian Elimination and Resolution of Systems of Algebraic
Equations," *EUROCAL '83*, Springer Lect. Notes in Comp. Sci. 162, 146–156.

Macaulay, F.S., (1902)

    "Some Formulae in Elimination," *Proc. London Math. Soc. 35*, 3–27.

Macaulay, F.S., (1916)

    The Algebraic Theory of Modular Systems, Cambridge University Press, 1916.

Netto, E., (1892)

    Theory of Substitutions and of Integral Functions, The Inland Press, Ann
Arbor, Mich.

Rossignac, J.R., and A.A.G. Requicha (1984)
  "Constant Radius Blending in Solid Modeling," *Comp. Mech. Engr. 3*, 65–73.

Sabin, M.A., (1987)
  "Envelope Curves and Surfaces," in *The Mathematics of Surfaces II*,
  R. Martin, ed., Clarendon Press, Oxford, 413–418.

Sederberg, T.W., (1983)
  "Implicit and Parametric Curves and Surfaces for Computer Aided Geometric
  Design," *Ph.D. Diss.*, Mech. Engr., Purdue University

Sederberg, T.W., and S. Parry (1986)
  "A Comparison of Curve Intersection Algorithms," *Comp. Aided Geometric
  Design 18*, 58–63.

Waggenspack, W.N., and D. Anderson (1986)
  "Converting Standard bivariate Polynomials to Bernstein Form Over
  Arbitrary Triangular Regions," *Comp. Aided Design, 18*, 529–532