SOME APPLICATIONS OF CONTRUCTIVE REAL ALGEBRAIC GEOMETRY

Chandrajit L. Bajaj

CSD-TR-93-056
September 1993

To appear in Algebraic Geometry and Applications, Special Issue of Symposium on the occasion of Shreeram Abhyankar's 60th Birthday, edited by C. Bajaj, Springer Verlag, (1993).

Some Applications of Constructive Real Algebraic Geometry

Chandrajit L. Bajaj

ABSTRACT In this short article we summarize a number of recent applications of constructive real algebraic geometry to geometric modelling and robotics, that we have been involved with under the tutelage of Abhyankar.

1 Introduction

Macaulay, Tarski and Seidenberg [16, 17, 18] set the tone for current day researchers of constructive methods in algebraic geometry over real closed fields. Constructive methods are clearly at the heart (and soul) of Abhyankar's papers in algebraic geometry and amply evident in his teaching[1]. In this short article we summarize a number of recent applications of constructive real algebraic geometry to geometric modeling and robotics, that we have been involved with under the tutelage of Abhyankar. First is parameterizations, useful for computing intersections, sweeps, offsets etc., required in robotic software simulation systems. Here we consider constructive methods for both local and global real parameterizations of curves and surfaces. Next we look at intersections between curves and surfaces which are fundamental for solid modeling systems based on Boolean set operations. Finally we look at surface fitting with algebraic surface patches, a technique used for both complicated interactive geometric design as well as scattered data fitting.

2 Global Parameterization

Certain classes of algebraic curves and surfaces admit both parametric and implicit representations. Algebraic curves and surfaces are the most common representations for curved objects in geometric modeling. Algebraics satisfy *polynomial* equations, usually with rational coefficients. A *rational* algebraic curve or surface is one whose points can be represented as rational functions in some parameters. Each form has certain benefits and

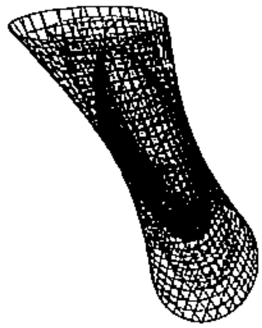

FIGURE 1. Global Parameterization of Quadrics using Finite Precision Arithmetic

drawbacks. The parametric form is better for rapid display and interactive control; the implicit form defines a half-space naturally and is suited for modeling. The class of all algebraics is also much larger than the class of rational algebraics. Having dual forms are highly useful in geometric modeling since they combine the strengths of the two representations[7].

In [13] we consider the problem of computing the rational parameterization of an implicit curve or surface in a finite precision domain. Known algorithms for this problem are based on classical algebraic geometry, and assume exact arithmetic involving algebraic numbers[2, 3, 4, 5, 6]. In this work, we investigate the behaviour of parametrization algorithms in a finite precision domain and derive succinct algebraic and geometric error characterizations. We then indicate numerically robust methods for parameterizing curves and surfaces which yield no error in extended finite precision arithmetic and alternatively, minimize the output error under fixed finite precision calculations

For example, one can obtain succinct bounds on the geometric error incurred in parameterizing quadratic surfaces (quadrics) by mapping (in fixed precision arithmetic) either the constant coefficient or one the squared term coefficients to infinity. The sign of the discriminant, among other quantities, distinguishes amongst the various quadric surfaces. Essentially, perturbing the constant coefficient preserves the center and orientation, although the quadric could degenerate from a hyperboloid of one sheet to a cone to a double-sheeted hyperboloid. Perturbing the highest order coefficients could cause an ellipsoid to change to a cylinder to a one-sheeted hyperboloid, for example, in addition to changing its orientation and center (Figure 1). Since the geometric errors find their extrema along the axes when the center and orientation are fixed, we can bound the errors easily in this case. We simply state the results, for brevity.

Let two quadrics that differ only in their constant coefficient be given and let d_x, d_y, d_z be the distances from the origin to the unperturbed quadric (some may not be finite). Given a number $\epsilon > 0$ that also satisfies $\epsilon < \min(d_x, d_y, d_z)$, and a difference in the constant coefficients of a quantity d_c, if the geometric perturbations p_x, p_y, p_z are to satisfy

$$\max(|p_x|, |p_y|, |p_z|) \quad < \quad \epsilon$$

then it suffices to choose d_c such that

$$|d_c| < \epsilon \cdot \min(d_x \cdot |\lambda_1|, d_y \cdot |\lambda_2|, d_z \cdot |\lambda_3|)$$

where expressions for λ_i are the roots of a cubic polynomial $\phi(\lambda)$ whose coefficients are expressions in the coefficients of the quadrics. The quadric can be put in standard form in terms of the roots of $\phi(\lambda)$, allowing the the quantities d_x, d_y, d_z to be efficiently calculated.

WIth the parameterization of singular cubic curves, algebraic number computation is unnecessary for exact rational parameterization. Every rational cubic with rational coefficients has a rational singular point. Such a cubic can be parameterized by a pencil of lines through the singularity, which then intersect the cubic at exactly one other point. The coordinates of the latter point parameterized by the slope of the line give parameter functions for the cubic curve. The parameter functions are given as closed form formulas in the parameter t, the coefficients of the curve, and the coordinates (b, c) of the singularity, as shown below:

$$\begin{aligned}
X(t) \quad &= a_{30}bt^3 - (3a_{30}c + a_{20})t^2 - \\
&(2a_{21}c + a_{12}b + a_{11})t - (2a_{03}b + a12c + a_{02}) \\
Y(t) \quad &= -((2a_{30}c + a_{21}b + a_{20})t^3 + \\
&(a_{21}c + 2a_{12}b + a_{11})t^2 + (3a_{03}b + a_{02})t - a_{03}c) \\
W(t) \quad &= a_{30}t^3 + a_{21}t^2 + a_{12}t + a_{03}
\end{aligned}$$

Therefore, if extended precision rational arithmetic is allowed, one can parameterize an irreducible rational cubic curve without error and without algebraic number computation, by computing the singular point exactly, and substituting the coordinates in the above formula.

3 Local Parameterization

In [14, 15] we use a combination of both algebraic and numerical techniques to construct C^1-continuous, piecewise (m, n) rational ϵ-approximation of real algebraic curves of degree d. For example, Figure 3, shows a C^1 continuous $(3, 3)$-rational approximation of the curve $(x^2 + y^2)^3 - 4x^2y^2 = 0$ for values of $\epsilon = 0.1, 0.05, 0.025$. At singular points we use the classical Weierstrass Preparation Theorem and Newton power series factorizations, based on the technique of Hensel lifting[1]. These, together with modified rational Padé approximations, are used to efficiently construct locally approximate, rational parametric representations for all real branches of an algebraic curve.

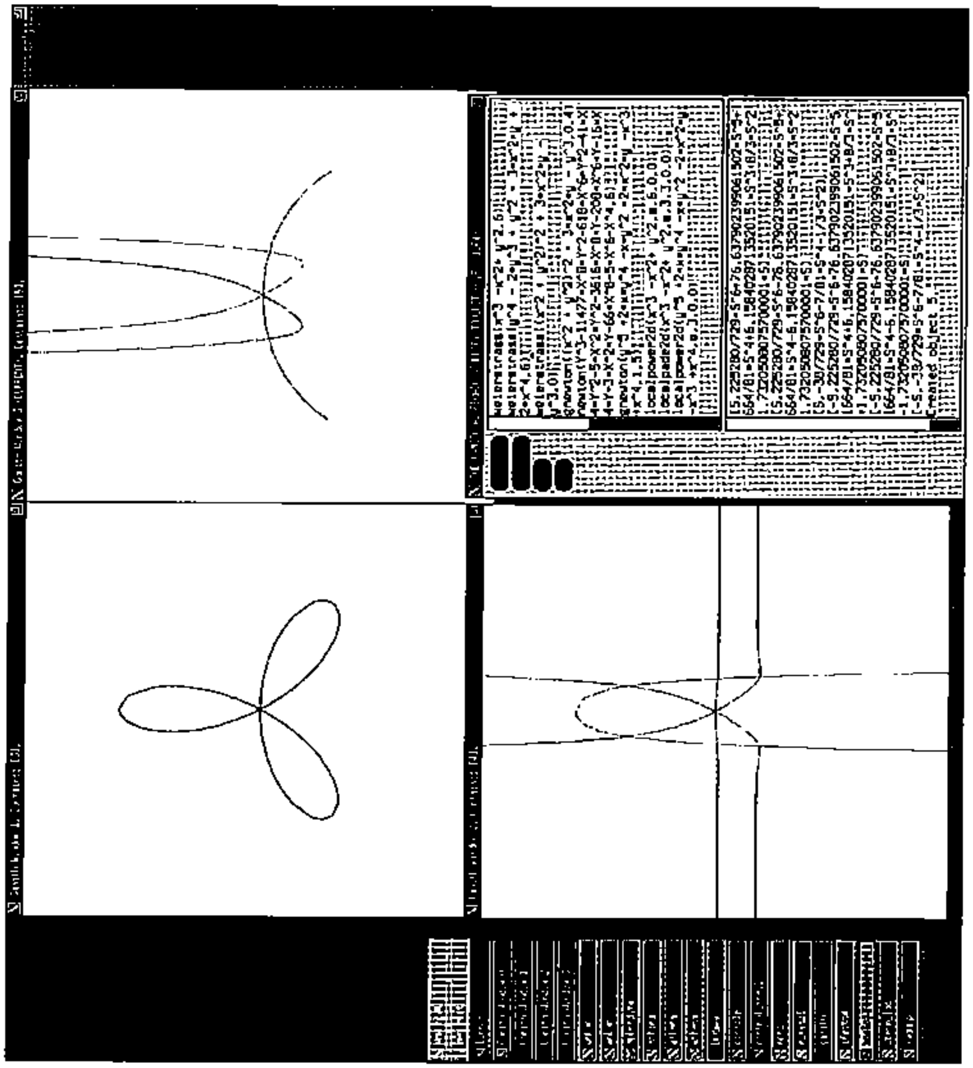

FIGURE 2. Resolution of Singularities: Newton and Weierstrass Factorizations

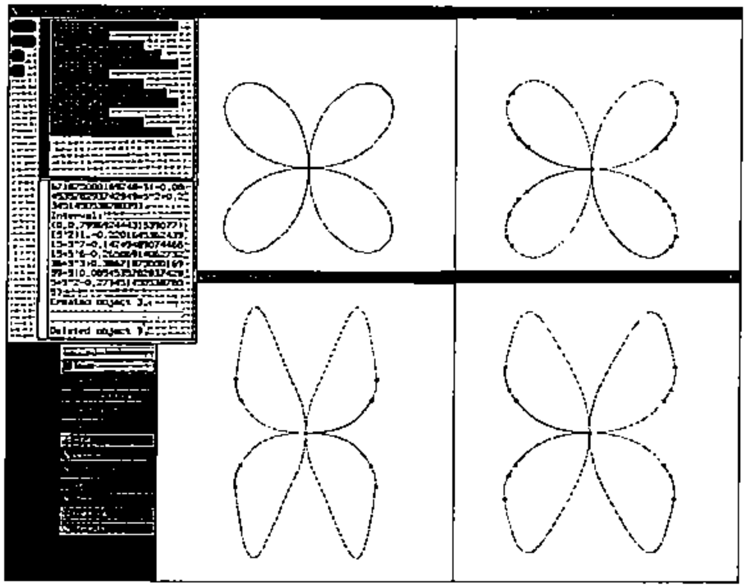

FIGURE 3. Piecewise Rational Approximations of Real Algebraic Plane Curves

A Weierstrass power series factorization is of the form $f(x,y) = g(x,y)\underbrace{(y^e + a_{e-1}(x)y^{e-1} + \cdots + a_0(x))}_{h(x,y)}$ where $g(x,y)$ is a unit power series, i.e., $g(0,0) \neq 0$ while $h(x,y)$ is a "distinguished" polynomial in y with coefficients $a_i(x)$, $i = 0 \ldots e-1$ being non-unit power series, i.e., $a_i(0) = 0$. In Figure 2 the lower right picture shows the two Weierstrass power series factors (truncated to degree 14 in x) of the plane curve $f(x,y) = (x^2 + y^2)^2 + 3x^2y - y^3 = 0$: the unit $g(x,y) = y + 121632404x^14 + 5254746x^12 + 237526x^10 + 11477x^8 + 618x^6 + 41x^4 + 5x^2 - 1 = 0$ which represents the part of the curve away from the origin, and the "distinguished" polynomial in y $h(x,y) = y^3 + (-121632404x^14 - 5254746x^12 - 237526x^10 - 11477x^8 - 618x^6 - 41x^4 - 5x^2)y^2 - (35422240x^14 - 1556448x^12 - 72080x^10 - 3616x^8 - 208x^6 - 16x^4 - 3x^2)y - 498162x^14 - 23038x^12 - 1153x^10 - 66x^8 - 5x^6 - x^4 = 0$ which represents the part of the curve at the origin.

The "distinguished" factor $h(x,y)$ is again shown in the upper left picture of Figure 2 where it is split via Newton factorization into real linear factors of the type $h(x,y) = \Pi_{i=1}^{e}\ (y - \eta_i((t)))$ with $t^m = x$ and m a positive integer and $\eta_i((t))$ a real power series or real meromorphic series.

Besides singular points we obtain an adaptive selection of simple points about which the curve approximations yield a small number of pieces yet achieve C^1 continuity between pieces.

The rational approximation algorithms have been implemented in its entirety as part of GANITH, an X-11 based interactive algebraic geometry

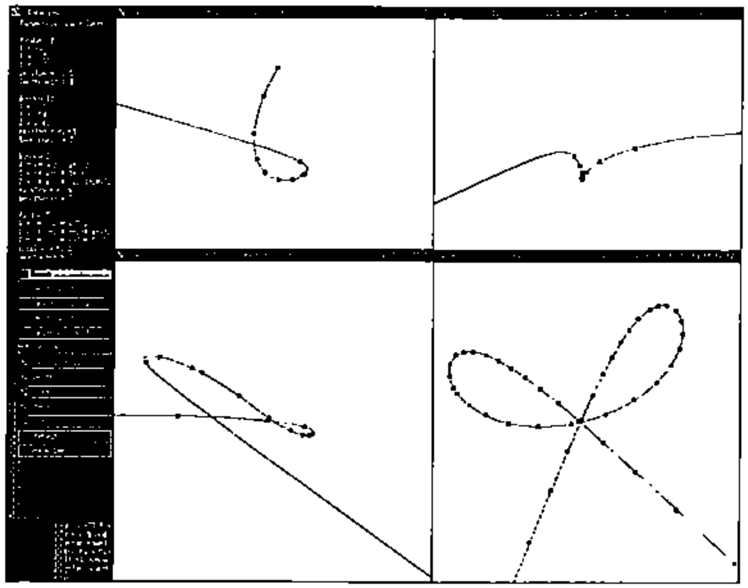

FIGURE 4. Piecewise Rational Approximation of Real Algebraic Space Curves

toolkit, using Common Lisp for the symbolic computation and C for all numeric and graphical computation. The Hensel power series computations as well as its use in Weierstrass and Newton factorizations are based on a robust implementation of the fast euclidean HGCD algorithm. Rational Padé approximants are also computed based on the same HGCD algorithm. Power Series are stored as truncated sparse polynomials, as are the polynomials representing the original algebraic curves, in recursive canonical form. In this form, a polynomial in the variables $x_1, \ldots, x_n$ is represented either as a constant, or as a polynomial in x_n whose coefficients are (recursively) polynomials in the remaining variables $x_1, \ldots, x_{n-1}$. A strength of this form (for purposes of implementation) is that multivariates "look like" univariates, making it easy to modify algorithms for univariate polynomials to handle multivariates.

Floating point coefficients are allowed in the input curve representations, which are then converted to rational numbers for the GCD and power series computations. In Newton factorizations, user options are provided to compute only real branch factorizations. This is achieved by not allowing complex conjugate roots of the appropriate univariate polynomial, to split in the base case of the Henselian computation. Singularity computations and intersection with the bounding box are done in GANITH using multivariate resultants and based on the method of birational maps [8]. Details of this are given in the next section. Examples from the software implementation, are shown in Figures 3 and 4.

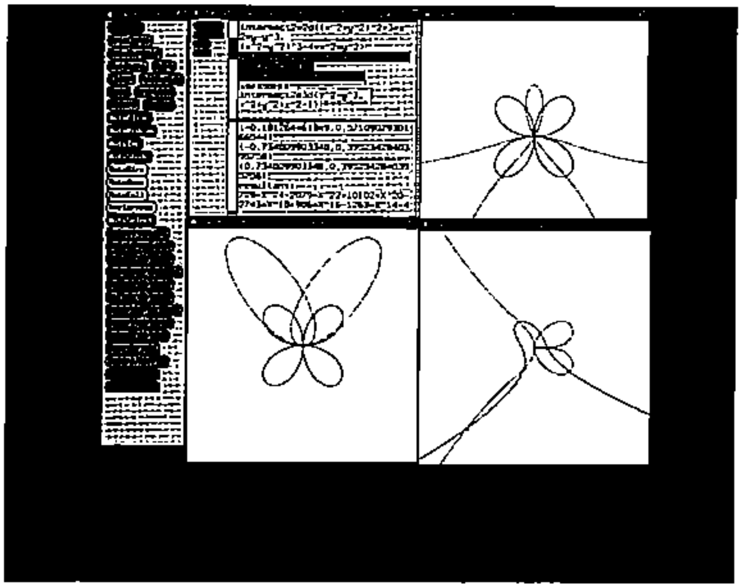

FIGURE 5. Computation of Curve-Curve Intersections

4 Intersection

The set of solutions (or *zero* set $Z(S)$) of a collection S of polynomial equations

$$S_1 : f_1(x_1, ..., x_n) = 0$$
$$...$$
$$S_m : f_m(x_1, ..., x_n) = 0 \tag{1.1}$$

is referred to as an ***algebraic set***. Algebraic curves and surfaces are algebraic sets of dimension 1 and 2 respectively. Problems dealing with zero sets $Z(S)$, such as the intersection of curves and surfaces, or the decision whether a surface contains a set of curves, are often first versed in an ideal-theoretic form and then solved using Gröbner basis manipulations. In [8] we present an alternative technique based on constructing bi-rational mappings between algebraic varieties and hypersurfaces. questions of intersection and parameterization of algebraic varieties. The bi-rational mapping technique deals directly with the *zero sets* of polynomial equations (rather than just the combinatorial structure of the polynomials), and provides simpler solutions to questions of intersection and parameterization of algebraic varieties.

Given m independent equations in n variables (1.1), let S be the algebraic variety of dimension $n - m$ defined by these equations. Then the bi-rational map construction of [8] produces a new "triangulated" polynomial system of equations

$$\tilde{f}(x_1, ..., x_{n-m+1}) = 0$$

$$x_{n-m+2} = \frac{h_{2m-4}(x_1, \ldots, x_{n-m+1})}{h_{2m-3}(x_1, \ldots, x_{n-m+1})}$$

$$\ldots$$

$$x_{n-1} = \frac{h_2(x_1, \ldots, x_{n-2})}{h_3(x_1, \ldots, x_{n-2})}$$

$$x_n = \frac{h_0(x_1, \ldots, x_{n-1})}{h_1(x_1, \ldots, x_{n-1})} \tag{1.2}$$

This bi-rational map construction is based on the multi-polynomial resultant[16] and multi-polynomial remainder sequences.

Cases of intersection computation of interest in geometric and solid modeling are those of plane curve-curve intersections, surface-surface intersections and three algebraic surface intersections [7]. All these are special cases of the bi-rational map construction. The two prevalent representations of algebraic curves in geometric modeling are the implicit and the rational parametric. Both implicitly and parametrically defined algebraic plane curve-curve intersections reduce to the special case of (1.1) for $n = 2$ and $m = 2$. The common intersection points (x_1, x_2) of the two curves are then obtained from the special case "triangulated" system (1.2) by first computing the zeros of the univariate polynomial $\tilde{f}(x_1) = 0$ and then substituting these into $x_2 = \frac{h_0(x_1)}{h_1(x_1)}$. Examples from the software implementation in GANITH, are shown in Figure 5.

Implicitly defined algebraic surface-surface intersections reduce to the special case of (1.1) for $n = 3$ and $m = 2$. Points (x_1, x_2, x_3) on the common intersection space curve of the two surfaces are then obtained from the special case "triangulated" system (1.2) by first computing points on the plane curve $\tilde{f}(x_1, x_2) = 0$ and then substituting these into $x_3 = \frac{h_0(x_1,x_2)}{h_1(x_1,x_2)}$. Parametrically defined algebraic surface-surface intersections reduce to the special case of (1.1) for $n = 4$ and $m = 3$. Points (x_1, x_2, x_3, x_4) on the common intersection space curve of the two surfaces are then obtained from the special case "triangulated" system (1.2) by first computing points on the plane curve $\tilde{f}(x_1, x_2) = 0$ and then substituting these into $x_3 = \frac{h_0(x_1,x_2)}{h_1(x_1,x_2)}$ and $x_4 = \frac{h_2(x_1,x_2)}{h_3(x_1,x_2)}$. An example from the software implementation in GANITH, involving a sphere and a quartic algebraic surface is shown in Figure 6.

Implicitly defined three algebraic surfaces intersection reduces to the special case of (1.1) for $n = 3$ and $m = 3$. Common intersection points (x_1, x_2, x_3) of the three surfaces are then obtained from the special case "triangulated" system (1.2) by first computing the zeros of the univariate

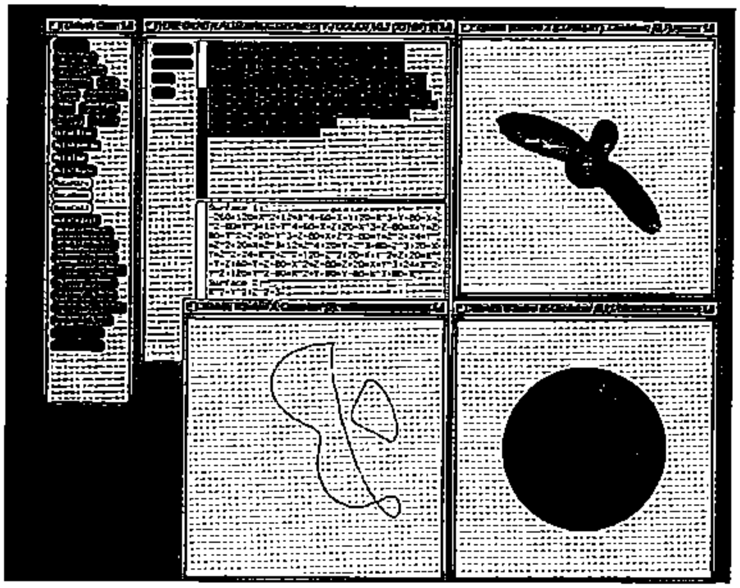

FIGURE 6. Computation of Surface-Surface Intersections

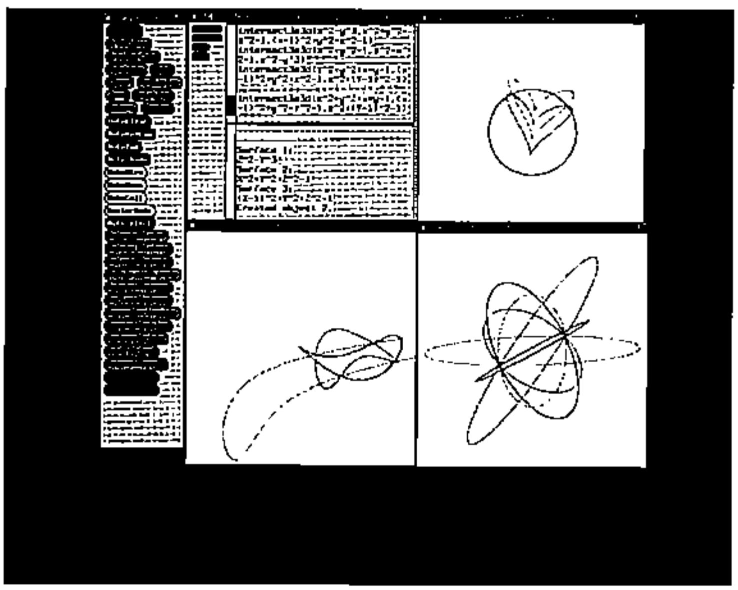

FIGURE 7. Computation of Intersections of Three Surfaces

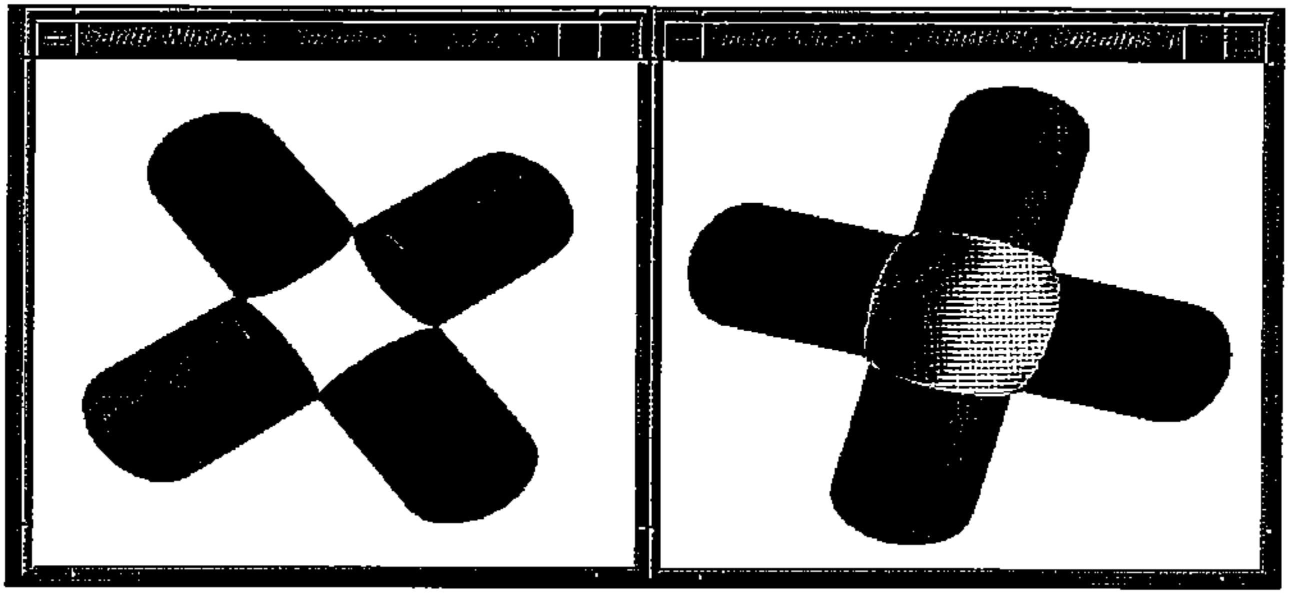

FIGURE 8. C^1 Join of Cylinders with a Quartic Surface

polynomial $\tilde{f}(x_1) = 0$ and then substituting these into $x_2 = \frac{h_0(x_1)}{h_1(x_1)}$, and $x_3 = \frac{h_2(x_1)}{h_3(x_1)}$. Parametrically defined three algebraic surfaces intersection reduces to the special case of (1.1) for $n = 6$ and $m = 6$. Common intersection points $(x_1, x_2, x_3, x_4, x_5, x_6)$ of the two surfaces are then obtained from the special case "triangulated" system (1.2) by first computing the zeros of the univariate polynomial $\tilde{f}(x_1, x_2) = 0$ and then substituting these into $x_2 = \frac{h_0(x_1)}{h_1(x_1)}$, $x_3 = \frac{h_2(x_1)}{h_3(x_1)}$, $x_4 = \frac{h_4(x_1)}{h_5(x_1)}$, $x_5 = \frac{h_6(x_1)}{h_7(x_1)}$, and $x_6 = \frac{h_8(x_1)}{h_9(x_1)}$. Examples from the software implementation in GANITH, are shown in Figure 7. The three surface intersection points are shown as the common intersections of the space curves for each pair of surfaces.

5 Interpolation and Approximation

The generation of a mesh of smooth real algebraic surface patches or *splines* that interpolate or approximate *triangulated space data* is one of the central topics of geometric design. Prior work on splines have traditionally worked with a given planar triangulation using a piecewise polynomial function basis or over triangulations in three dimensions using parametric surface patches. Little work has been done on spline basis for implicitly defined real algebraic surfaces.

I report briefly on some ongoing work in this extremely interesting and

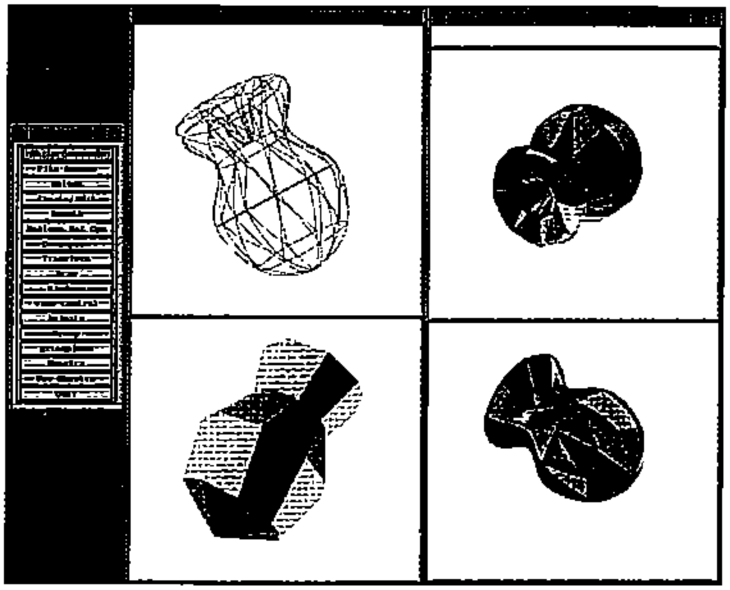

FIGURE 9. Smoothing of a Polyhedron with Triangular Interpolatory Splines

fundamental area of research for geometric design [9, 10]. In [11] we show how low degree blending and joining algebraic surfaces can be computed via C^1 interpolation and least-squares approximation. The algebraic surface fitting scheme reduces to the solution of a finite system of linear equations, based on a proper normalization of the coefficients of the surface. Both the finiteness bound and the linear equations are derived from various invocations of Bezout's theorem. In the example shown in Figure 8 and implemented in GANITH, the joining surface of the four cylindrical surfaces is computed by a C^1 interpolation of the four circular cross-sections of the cylinders and the gradient vectors along these cross-sections. Least-squares approximation from a variable radius sphere centered at the mid point of the junction helps select the desired bulge of the joining surface.

In [12] we consider an arbitrary spatial triangulation $\mathcal{T}$ consisting of vertices (x_i, y_i, z_i) in $\mathbb{R}^3$ (or more generally a simplicial polyhedron $\mathcal{P}$ when the triangulation is closed), with possibly "normal" vectors at the vertex points. An algorithm is given to construct a C^1 continuous mesh of low degree real algebraic surface patches S_i , which respects the topology of the triangulation $\mathcal{T}$ or simplicial polyhedron $\mathcal{P}$, and C^1 interpolates all the vertices (x_j, y_j, z_j) in $\mathbb{R}^3$. The technique uses a single implicit surface patch for each triangular face of $\mathcal{T}$ of $\mathcal{P}$, i.e. no local splitting of triangular faces. Each triangular surface patch has local degrees of freedom which are used to provide local shape control. This is achieved by use of weighted least squares approximation from points (x_k, y_k, z_k) generated locally for each triangular patch from the original patch data points and normal directions on them. Examples of this smoothing process are shown in Figure 9 and implemented in GANITH.

6 References

[1] Abhyankar, S., (1990) "Algebraic Geometry for Scientists and Engineers", *Mathematical Surveys and Monographs*, Number 35, American Mathematical Society, Providence, Rhode Island.

[2] Abhyankar, S., and Bajaj, C., (1987) "Automatic Parameterization of Rational Curves and Surfaces I: Conics and Conicoids", *Computer Aided Design*, 19, 1, pp. 11 - 14.

[3] Abhyankar, S., and Bajaj, C., (1987) "Automatic Parameterization of Rational Curves and Surfaces II: Cubics and Cubicoids", *Computer Aided Design*, 19, 9, pp. 499 - 502.

[4] Abhyankar, S., and Bajaj, C., (1988) "Automatic Parameterization of Rational Curves and Surfaces III: Algebraic Plane Curves", *Computer Aided Geometric Design*, 5, (1988), 309 - 321.

[5] Abhyankar, S., and Bajaj, C., (1988) "Computations with Algebraic Curves", (Proc. of Intl. Symposium on Symbolic and Algebraic Computation), (ISSAC8), *Lecture Notes in Computer Science*, No. 358, Springer-Verlag, (1989), 279 - 284.

[6] Abhyankar, S., and Bajaj, C., (1989) "Automatic Parameterization of Rational Curves and Surfaces IV: Algebraic Space Curves", *ACM Transactions on Graphics*, 8, 4, (1989), 324 - 333.

[7] Bajaj, C., (1989) "Geometric Modeling with Algebraic Surfaces" *The Mathematics of Surfaces III*, ed., D. Handscomb, Oxford University Press, pp. 3 - 48.

[8] Bajaj, C., (1990) "Geometric Computations with Algebraic Varieties of Bounded Degree", Proc. of the Sixth ACM Symposium on Computational Geometry, Berkeley, California, 148-156.

[9] Bajaj, C., (1992) "Surface Fitting with Implicit Algebraic Surface Patches" *Topics in Surface Modeling*, edited by H. Hagen, SIAM Publications, (1992), in press.

[10] Bajaj, C., (1992) "The Emergence of Algebraic Curves and Surfaces in Geometric Design", *Directions in Geometric Computing 1991*, edited by R. Martin, Information Geometers Press, United Kingdom (1992), in press.

[11] Bajaj, C. and Ihm, I., (1992), "Algebraic Surface Design with Hermite Interpolation" *ACM Transactions on Graphics*, 19, 1, (1992), 61-91.

[12] Bajaj, C., and Ihm, I., (1992), "C^1 Smoothing of Polyhedra with Implicit Algebraic Splines", *Computer Graphics*, 26, 2, SIGGRAPH'92, 79 - 88.

[13] Bajaj, C., and Royappa, A., (1992) "Parameterization in Finite Precision", Proc. of the Graphics Interface'92 Conference, (1992), Vancouver, Canada, 29-36.

[14] Bajaj, C. and Xu, G., (1991) "Piecewise Rational Approximations of Real Algebraic Curves", Submitted to *Advances in Computational Mathematics*.

[15] Bajaj, C. and Xu, G., (1992) "NURBS Approximation of Surface-Surface Intersection Curves", Submitted to *Advances in Computational Mathematics*.

[16] Macaulay, F. (1916), "The Algebraic Theory of Modular Systems", *Cambridge University Press*.

[17] Seidenberg, A., (1974) "Constructions in Algebra", *Transactions of the AMS*, 197:273–313.

[18] Tarski, A., (1948) "A Decision Method for Elementary Algebra and Geometry", *University of California Press*, Berkeley.