Parameterization In Finite Precision

Chandrajit L. Bajaj
Department of Computer Sciences and TICAM,
University of Texas at Austin,
Austin, Texas 78712

Andrew V. Royappa
Department of Mathematics & Computer Studies,
Millsaps College,
Jackson, Mississippi 39210

Abstract

Certain classes of algebraic curves and surfaces admit both parametric and implicit representations. Such dual forms are highly useful in geometric modeling since they combine the strengths of the two representations. 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. In this work, we investigate the behaviour of published parametrization algorithms in a finite precision domain and derive succint 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.

Keywords: curves and surfaces, geometric modeling, numerical methods, computational algebraic geometry.

1 Introduction

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 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. Dual forms can have the best of both worlds.

Mathematical techniques from algebraic geometry have recently been applied to the problem of converting between the two forms. While *implicitizing* a parametric curve or surface is always possible, the converse (*rational parameterization*) is not always possible. That is, all algebraics are not rational. However, important classes of curves and surfaces are rational and algorithms for their rational parameterization based on algebraic geometry have been given in [1],[2],[3], [4],[18], and some will be analyzed here. There is also a numerical method due to Jacobi which works by iteratively converting a conic or quadric to standard form (see [12]).

Functionally, rational parameterization takes one implicit equation in n variables, and for each implicit variable returns a rational function in $n-1$ parameters. Since the rational functions have a common denominator, the output can be viewed as consisting of $n+1$ polynomials.

While the input implicit equation is assumed to have rational coefficients, the output polynomials may require algebraic number coefficients, which are (informally speaking) roots of polynomials, such as $\sqrt{2}$. The algorithms based on algebraic geometry assume exact computations. While techniques exist for manipulating algebraic numbers exactly, they are expensive. In this work, we consider parameterization algorithms in a finite precision domain.

This paper is organized as follows. We choose a finite precision numerical domain and explain our general approach to rederiving a parameterization algorithm to work in this domain. First, we analyze algorithms for conics and quadrics, and then analyze an algorithm for singular cubic curves. The error in each algorithm is described algebraically. We then use the algebraic error analysis to derive simple geometric error bounds for conics and quadrics. Finally, we consider singular cubic parameterization from another standpoint, showing that they can in fact be parameterized exactly using only rational arithmetic. Finally, we conclude by briefly discussing extensions of this approach, e.g. to cubic surfaces.

2 Approach and Numerical Model

To examine this problem when exact arithmetic is not allowed, we focus on the use of algebraic numbers. We stop short of allowing floating-point arithmetic; instead, the algorithms will use rational arithmetic throughout. Algebraic numbers will be approximated by rationals. This allows us, as a first study, to isolate the effects of the error caused by rational approximations to algebraic numbers.

Recall that parameterization algorithms take a polynomial with rational coefficients as input, and output several polynomials. The algorithms can be restructured so that each coefficient of an output polynomial is given as a *formula* in the (symbolic) input coefficients, and some additional symbols. Every algebraic number required by the parameterization will be represented by one symbol in the formulas.

If the algebraic numbers themselves are substituted for their symbols into the output formula, the output will be exact. However, we only allow rational approximations to algebraic numbers. Substituting these numbers will yield only an approximate output. This output will converge to the exact one as the rational approximations converge to the algebraic numbers.

Thus, given a certain precision to which algebraic numbers are to be rationally approximated, and a bound on the size of the rational input coefficients, one can calculate from the formulas a bound on the rational output coefficients, and hence finally a bound on the precision required to carry out the entire calculation, if fixed finite precision is desired.

Our approach to restructuring the algorithms consists of examining them step by step, and eliminating from each step every subexpression that must vanish if exact arithmetic was used. As it turns out, there are two benefits of this approach: it is often possible to carry through the computation so that the output is expressed as a formula in the input, and error formulas are easily derived. While this method seems to work very well for parameterization algorithms, its applicability in other settings is likely to be limited, where *repeated* computations with algebraic numbers may be required.

3 Conic Parameterization

We restructure the algorithm in [1] for conic parameterization. The algorithm is given for conics in homogeneous form; this allows the use of both projective and affine transformations. The algorithm is then analyzed for the error in its output when approximations are used for algebraic numbers.

Given the equation of a conic plane curve, parameter functions for the curve are derived. The parameter functions are given as closed form formulas in the parameter t, the coefficients of the curve, and the coordinates of a point on the curve.

INPUT. An irreducible conic curve given by $f(x,y) = a_{20}y^2 + a_{11}xy + a_{02}x^2 + a_{10}y + a_{01}x + a_{00} = 0$.
OUTPUT. Rational functions $(x(t), y(t))$ of degree at most two, such that $f(x(t), y(t)) = 0$.
ALGORITHM.
1. Homogenize the conic. This yields the homogeneous equation $F(X,Y,W) = a_{20}Y^2 + a_{11}XY + a_{02}X^2 + a_{10}YW + a_{01}XW + a_{00}W^2 = 0$. If the X^2, Y^2 or W^2 term is missing from the conic's equation, then it will be linear in the corresponding variable, and can be immediately parameterized. Compute quadratic polynomials $X(t), Y(t)$ and $W(t)$ such that $F(X,Y,W) = 0$, and go to step 4.
2. If all squared terms are present, apply a linear transformation to cancel one of these terms.
3. Parameterize the transformed conic, and apply the inverse transformation to the parameterization; yielding three quadratic polynomials$X(t), Y(t)$ and $W(t)$ such that $F(X,Y,W) = 0$.
4. The parameterization for the affine conic is then given by $x(t) = X(t)/W(t), y(t) = Y(t)/W(t)$.

TRANSFORMATIONS. If all three squared terms are present, then any one of the following three transformations may be used in step 2 of the conic parameterization algorithm. The transformations to cancel X^2 and Y^2 are more general than that for W^2. Hence we explain the X^2 case in most detail (the Y^2 case is similar and omitted).

- To cancel the X^2 term, use the transformation

$$\begin{array}{rcllll} X & = & bX_1 & & & \\ Y & = & cX_1 & + & Y_1 & \\ W & = & dX_1 & & & + \; W_1 \end{array} \tag{1}$$

where (b,c,d) are the homogeneous coordinates of some point on the curve. For the transformation to be well-defined, b must be non-zero. Then, if $d \neq 0$, the transformation is affine; otherwise it is projective. Since proportional projective coordinates represent the same point, we make the restriction $d = 0$ or $d = 1$. If $d = 0$, then we should also make a restriction $b = 1$ or $c = 1$; since $b \neq 0$ is required for the transformation to be well-defined, we will restrict $b = 1$ in this case.

Transforming F yields a new conic curve with implicit equation

$$\begin{aligned} F_1(X_1, Y_1, W_1) &= F(bX_1, cX_1 + Y_1, dX_1 + W_1) \\ &= F(b,c,d)X_1^2 + F_2(X_1, Y_1, W_1) \end{aligned}$$

Since the subexpression $F(b,c,d)X_1^2$ must vanish, we only need to parameterize

$$\begin{aligned} F_2(X_1, Y_1, W_1) = \; & (a_{10}d + 2a_{20}c + a_{11}b)X_1Y_1 + \\ & (2a_{00}d + a_{10}c + a_{01}b)X_1W_1 + \\ & a_{20}Y_1^2 + a_{10}Y_1W_1 + a_{00}W_1^2 = 0 \end{aligned}$$

The curve $F_2 = 0$ passes exactly through the point $(1, 0, 0)$ and can be parameterized by intersecting it with the pencil of lines $Y_1 = tW_1$ which pass through this point, yielding

$$\begin{aligned} X_1(t) &= a_{20}t^2 + a_{10}t + a_{00} \\ Y_1(t) &= -(a_{10}d + a_{11}b + 2a_{20}c)t^2 - \\ &\quad (2a_{00}d + a_{01}b + a_{10}c)t \\ W_1(t) &= -(a_{10}d + a_{11}b + 2a_{20}c)t - \\ &\quad (2a_{00}d + a_{01}b + a_{10}c) \end{aligned} \tag{2}$$

This symbolic parameterization for F_2 is independent of the specific values for b, c and d, i.e., it is always exact, since only rational operations in the coefficients of F_2 are used.

Since $F(b, c, d) = 0$, $F_1(X_1, Y_1, W_1) = F_2(X_1, Y_1, W_1)$, and hence the parameterization (2) also applies to F_1. Applying the inverse linear transformation to this parameterization immediately yields a formula for the original conic:

$$\begin{aligned} X(t) &= b(a_{20}t^2 + a_{10}t + a_{00}) \\ Y(t) &= -(a_{10}d + a_{11}b + a_{20}c)t^2 - \\ &\quad (2a_{00}d + a_{01}b)t + a_{00}c \\ W(t) &= a_{20}dt^2 - (a_{11}b + 2a_{20}c)t - \\ &\quad (a_{00}d + a_{01}b + a_{10}c) \end{aligned} \tag{3}$$

• The transformation cancelling the W^2 term is always affine (i.e. $d = 1$); it is the translation taking the point $(b, c, 1)$ to the affine origin $(0, 0, 1)$. The parameterization formulas derived are

$$\begin{aligned} X(t) &= -(a_{10}d + a_{20}c + a_{11}b)t^2 - \\ &\quad (a_{01}d + 2a_{02}b)t + a_{02}c \\ Y(t) &= a_{20}bt^2 - (a_{10}d + 2a_{20}c)t + (a_{01}d + \\ &\quad a_{11}c + a_{02}b) \\ W(t) &= d(a_{20}t^2 + a_{11}t + a_{02}) \end{aligned} \tag{4}$$

3.1 Backward Error Analysis

The only computation in the algorithm given above is to derive the coordinates of a point on the input conic curve. Once these coordinates are found, the parameterization is given as a closed form formula in terms of those numbers and the coefficients of the input curve. The output parameter functions $x(t)$ and $y(t)$ are formulas in algebraic numbers b and c (d is always either 0 or 1), satisfying $f(x(t), y(t)) = 0$. When approximations $\tilde{b}$ and $\tilde{c}$ are used for b and c, the algorithm will output approximate parameter functions $\tilde{x}(t)$ and $\tilde{y}(t)$ such that $f(\tilde{x}(t), \tilde{y}(t)) \neq 0$. These parameter functions also correspond to some algebraic curve. We would like to find the implicit equation of this new curve and compare it to the original input curve. This is the approach of backward error analysis.

LEMMA. Let the first transformation above be used in computing the parameterization. Then the output parametric curve exactly satisfies the perturbed implicit equation $\tilde{f}(x, y) = a_{20}y^2 + a_{11}xy + (a_{02} - \delta)x^2 + a_{10}y + a_{01}x + a_{00} = 0$, where the value δ is given by

$$\delta = \begin{cases} \dfrac{f(\tilde{b}, \tilde{c})}{\tilde{b}^2} & \text{if } d = 1 \\ a_{20}\tilde{c}^2 + a_{11}\tilde{c} + a_{02} & \text{if } d = 0 \end{cases}$$

PROOF. The analysis begins by computing the value of the expression $f(\tilde{x}(t), \tilde{y}(t))$. This value must vanish when exact arithmetic is used, since every point on the output (parametric) curve must be on the input (implicit) curve. However, in the presence of numerical approximations, it will be non-zero, and can be found symbolically. It depends on which transformations above was used. In the following, we use the relationship $f(X/W, Y/W) = F(X, Y, W)/W^2$. We now compute $f(\tilde{x}(t), \tilde{y}(t))$ directly:

$$
\begin{aligned}
f(\tilde{x}(t), \tilde{y}(t)) &= f(\frac{\tilde{X}(t)}{\tilde{W}(t)}, \frac{\tilde{Y}(t)}{\tilde{W}(t)}) \\
&= \frac{F(\tilde{X}(t), \tilde{Y}(t), \tilde{W}(t))}{\tilde{W}^2(t)} \\
&= \frac{F(b\tilde{X}_1(t), c\tilde{X}_1(t) + \tilde{Y}_1(t), d\tilde{X}_1(t) + \tilde{W}_1(t))}{\tilde{W}^2(t)} \\
&= \frac{F_1(\tilde{X}_1(t), \tilde{Y}_1(t), \tilde{W}_1(t))}{\tilde{W}^2(t)} \\
&= \frac{F(\tilde{b}, \tilde{c}, d)\tilde{X}_1^2(t) + F_2(\tilde{X}_1(t), \tilde{Y}_1(t), \tilde{W}_1(t))}{\tilde{W}^2(t)} \\
&= \frac{F(\tilde{b}, \tilde{c}, d)\tilde{X}_1^2(t)}{\tilde{W}^2(t)} \\
&= \frac{F(\tilde{b}, \tilde{c}, d)}{\tilde{b}^2} \frac{\tilde{X}^2(t)}{\tilde{W}^2(t)} \\
&= \frac{F(\tilde{b}, \tilde{c}, d)}{\tilde{b}^2} \tilde{x}^2(t)
\end{aligned}
$$

The key is that $F_2(\tilde{X}_1(t), \tilde{Y}_1(t), \tilde{W}_1(t)) = 0$ even when approximations are used.

Thus each point on the output curve evidently satisfies the equation $f(x, y) - (F(\tilde{b}, \tilde{c}, d)/\tilde{b}^2)x^2 = 0$. Since $F(\tilde{b}, \tilde{c}, 1) = f(\tilde{b}, \tilde{c})$ and $F(1, \tilde{c}, 0) = a_{20}\tilde{c}^2 + a_{11}\tilde{c} + a_{02}$, the lemma follows. □

Similarly, one can show that if the second transformation is used, the approximate output parameterization satisfies the equation $f(x, y) - \delta y^2 = 0$ with $\delta = f(\tilde{b}, \tilde{c})/\tilde{c}^2$, for $d = 1$, and $\delta = a_{20} + a_{11}\tilde{b} + a_{02}\tilde{b}^2$ for $d = 0$.

Finally, if the third transformation was used, then the output parametric curve satisfies the implicit equation $f(x, y) - \delta = 0$, where $\delta = f(\tilde{b}, \tilde{c})$.

Thus the effect of approximating (b, c) by rationals is an output parametric curve that corresponds to the input implicit curve, perturbed in precisely one of the coefficients a_{02}, a_{20}, a_{00}, depending on the transformation used.

We note that the above discussion remains valid under scaling of the input equation by a constant.

3.2 Geometric Error Bounds

The algebraic error analysis tells us the implicit equation of the approximate output curve; it is natural to investigate the relationship between the input (exact) and output (approximate) curves. For conics and quadrics, we can derive geometric error bounds in terms of the magnitude of the coefficient perturbation.

In [11], general bounds are given for local geometric perturbations at a point on a curve due to random perturbations in the coefficients of its equation. However, the perturbations that appear as a result of approximations in the parameterization process have a definite structure, which we exploit to derive global geometric error bounds.

We investigate the geometric effects of perturbing a single coefficient in the equation of a conic curve. The perturbations yield an entire family of conics. In particular, the effect of perturbing the constant coefficient is investigated.

It will be shown that perturbing the constant coefficient gives rise to a conic similar to the original conic. We then bound the maximum orthogonal distance between the original and perturbed conic.

We first list some relevant facts about conics, from [19] and [16]. Consider the affine quadratic equation of a conic curve $\mathbf{C}$, in the form

$$F(x, y) = ax^2 + by^2 + 2hxy + 2gx + 2fy + c = 0$$

The *discriminant* of $\mathbf{C}$ is

$$\Delta = \begin{vmatrix} a & h & g \\ h & b & f \\ g & f & c \end{vmatrix} = abc + 2fgh - af^2 - bg^2 - ch^2$$

The following facts about conics are known:

1. $\mathbf{C}$ degenerates to a pair of lines when $\Delta = 0$
2. $\mathbf{C}$ is a parabola when $ab - h^2 = 0$, an ellipse when $ab - h^2 < 0$, and a hyperbola when $ab - h^2 > 0$.
3. When $\mathbf{C}$ is not a parabola, its center is given by $(\frac{hf-bg}{ab-h^2}, \frac{gh-af}{ab-h^2})$.
4. The axes of the conic are given by the equation $h(x^2 - y^2) - (a - b)xy = 0$.
5. The conic can be translated to have its center at the origin, and axes rotated to the principal axes. In this coordinate system its equation is

$$F(x, y) = (a + b + R)x^2 + (a + b - R)y^2 + \frac{2\Delta}{ab - h^2} = 0$$

where $R^2 = (a - b)^2 + 4h^2$. It is clear that perturbing the constant term c in the equation of a conic will produce a new conic of the same type that is concentric and coaxial with the original (see also [8]). Perturbing the coefficients of x^2 or y^2, on the other hand, can change all these quantities: Figure 1 shows a family of conics perturbed only in the coefficient of x^2; they vary in type, center, and axis. We will therefore only consider the third transformation of the conic algorithm, which only perturbs the constant coefficient.

Even when only the constant coefficient is perturbed, the conic could still degenerate into a pair of lines. A large enough perturbation could turn a hyperbola into one that is concentric and coaxial to the original, but with transverse and conjugate axes reversed. Hence, an upper bound must be imposed on the perturbation. Since the constant coefficient c appears linearly in the discriminant Δ, so will the perturbed coefficient $c + \delta$, and hence one can immediately bound $|\delta|$ to avoid this case. If this bound is very small the conic will already be close to degenerate.

For perturbations smaller than this bound, then, we wish to geometrically describe the error. Define the (orthogonal) distance from a point p on one conic to the other conic as the shortest distance along the normal vector at p to the other conic. Then the maximum orthogonal distance from a point on one conic to the other will occur at one of the extreme points of the conic along its semi-axes, if ellipse, or transverse axis, if hyperbola.

Figure 1: Conics Perturbed in the Higher Order Coefficients

Now suppose one is given two conics $\mathbf{C}, \tilde{\mathbf{C}}$, where the second conic is derived by perturbing the constant coefficient in the equation of the first (if $\mathbf{C}$ is a parabola, some slight modifications will apply to the arguments below). Then they will be concentric and coaxial, and we can consider their equations in a coordinate system where their center is at the origin and their axes are aligned with the primary axes. In this coordinate system their equations will take the form $f(x, y) = Ax^2 + By^2 + C_1 = 0$ and $\tilde{f}(x, y) = Ax^2 + By^2 + C_2 = 0$. Let $d_x, \tilde{d}_x$ be the distances along the x-axis from the origin (which is the center) to $\mathbf{C}, \tilde{\mathbf{C}}$ respectively. Likewise, let $d_y, \tilde{d}_y$ be the distances along the y-axis. That is, d_x an d_y are simply the lengths of the semi-axes of the conic (in the case of a hyperbola, only one of these distances is finite). Then $\tilde{d}_x = d_x + p_x$ and $\tilde{d}_y = d_y + p_y$. One of p_x and p_y will be the maximum orthogonal distance between the two curves. We can solve directly for p_x and p_y, the maximum and minimum geometric error.

To solve for p_x, put $y = 0$ in the curve equations. Then $d_x^2 = -\frac{C_1}{A}$ and $\tilde{d}_x^2 = -\frac{C_2}{A}$. Hence $\tilde{d}_x^2 - d_x^2 = (d_x + p_x)^2 - d_x^2 = \frac{C_1 - C_2}{A}$. So $p_x^2 + 2d_x p_x = \frac{C_1 - C_2}{A}$, and since $p_x = 0$ when $C_1 - C_2 = 0$, we find that $p_x = -d_x + \sqrt{d_x^2 + \frac{(C_1 - C_2)}{A}}$.

Revert to the original coordinate system, where the conics have equations $ax^2 + by^2 + 2hxy + 2gx + 2fy + c_1 = 0, ax^2 + by^2 + 2hxy + 2gx + 2fy + c_2 = 0$; then, by the coordinate transformations of the previous section, and some algebra, $C_1 - C_2 = 2(c_1 - c_2)$. Using the definitions for A and B, and R as given in previously, and putting $\delta = c_1 - c_2$, we find that

$$p_x = -d_x + \sqrt{d_x^2 + \left(\frac{2}{a + b + R}\right)\delta}$$

Now suppose it is desired that $|p_x| < \epsilon$ for some $\epsilon > 0$, and we wish to bound $|\delta|$. To have $|p_x| < \epsilon$, it is necessary that $p_x < \epsilon$ and $p_x > -\epsilon$. For reasons that will soon be clear, we will require $\epsilon < d_x$. Considering each case separately,

1. $p_x < \epsilon$ implies that $\left(-d_x + \sqrt{d_x^2 + \left(\frac{2}{a+b+R}\right)\delta}\right) < \epsilon$.

2. $p_x > -\epsilon$ implies $\left(-d_x + \sqrt{d_x^2 + \left(\frac{2}{a+b+R}\right)\delta}\right) > -\epsilon$.

After considering both possibilities for the sign of $a+b+R$, the requirements above may be satisfied by taking

$$\begin{aligned} |\delta| &< \epsilon(2d_x + \epsilon)\,|(a+b+R)/2| \\ |\delta| &< \epsilon(2d_x - \epsilon)\,|(a+b+R)/2| \end{aligned}$$

for cases (1) and (2) respectively.

Finally, recalling that $\epsilon < d_x$, choices (1) and (2) can be simultaneously satisfied by choosing

$$|\delta| < \epsilon \cdot d_x \cdot \left|\frac{a+b+R}{2}\right|$$

This is the only simplification made in the calculation, and at most a factor of two of accuracy (one bit) is lost.

The error along the y axis is bounded in an identical way. The quantities d_x and d_y are independent of any scaling of the coefficients of the original conic by a constant, but the scale factor will be linearly present in the quantities $a+b+R$ and $a+b-R$. Hence, if δ is defined as in the backward error analysis for conics, these bounds correct for the scale factor automatically. Keeping this in mind, it suffices to compute δ such that

$$|\delta| < \epsilon \cdot \frac{\min(d_x \cdot |a+b+R|, d_y \cdot |a+b-R|)}{2}$$

4 Quadrics

The results for conics generalize directly to quadrics. It is possible to derive explicit formulas of degree two for the parameterization, in a pair of parameters s, t. The formulas are small; the only computation required is that of finding a point (a, b, c, d) on the homogeneous conic. There are four choices of transformations, one to cancel each squared term. A corresponding error analysis holds. For instance, if the W^2 term is cancelled using an approximate (affine) point $(\tilde{a}, \tilde{b}, \tilde{c})$, then the output parameterization will satisfy $f(\tilde{x}(s,t), \tilde{y}(s,t), \tilde{z}(s,t)) - f(\tilde{a}, \tilde{b}, \tilde{c}) = 0$, i.e., the original input equation perturbed in the constant coefficient.

This raises an important point. In general, a parametric curve of degree n corresponds to a curve of algebraic (implicit) degree n, but a parametric surface of degree n may correspond to a surface of algebraic degree up to n^2. Thus when using approximations in a parameterization, one might legitimately question whether the algebraic degree of the output is the same as that of the input. In the case of quadrics ($n = 2$), it would be unpleasant if a parameterization algorithm could actually output a cubic or quartic surface. Fortunately, the error analysis above allows us to answer this question in the negative, since a set of rational parametric equations of a surface satisfy a unique irreducible algebraic surface.

4.1 Geometric Error Bounds

As for conics, there is a quadric discriminant. The sign of the discriminant, among other quantities, distinguishes amongst the various quadric surfaces. Essentially, perturbing the constant coefficient

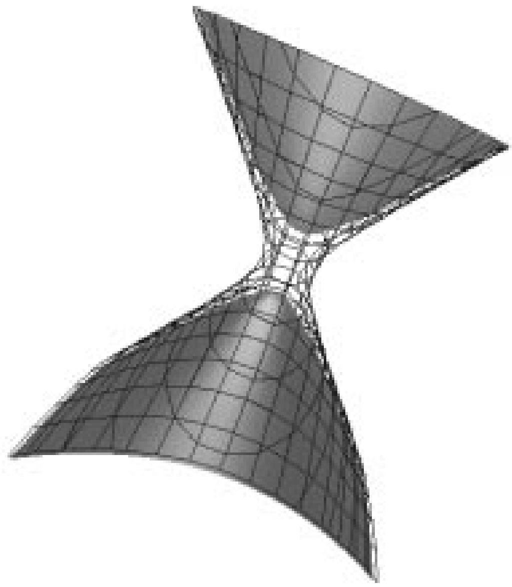

Figure 2: Quadrics Perturbed in the Constant Coefficient

preserves the center and orientation, although the quadric could degenerate from a hyperboloid of one sheet to a cone to a double-sheeted hyperboloid (see Figure 2). 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 3). 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. Vital information regarding quadrics was taken from [20].

Let two quadrics that differ only in their constant coefficient be given. Generalizing the notation from the conic case, let d_x, d_y, d_z be the distances from the origin to the unperturbed conic (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 δ, 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 δ such that

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

where expressions for λ_i are the roots of a cubic polynomial $\phi(\lambda)$ whose coefficients are expressions in the coefficients of the quadrics. From data in [20], 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. We omit the details here.

Only considering real values of d_x, d_y, d_z, then, we can bound the geometric error for a quadric due to approximate parameterization.

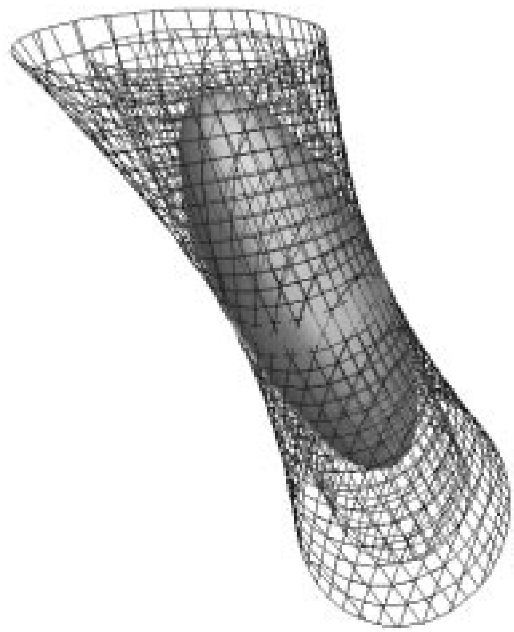

Figure 3: Quadrics Perturbed in the Higher Order Coefficients

5 Singular Cubic Curves

For an irreducible, singular cubic plane curve, a parameterization algorithm is given in [2], which we analyze. While output formulas exist for this case, they are unwieldy, and instead we show how they can be derived, and the error in the parameterization.

INPUT. A cubic plane curve given by the cubic equation $f(x,y) = a_{30}y^3 + a_{21}xy^2 + a_{20}y^2 + a_{12}x^2y + a_{11}xy + a_{10}y + a_{03}x^3 + a_{02}x^2 + a_{01}x + a_{00} = 0$.

OUTPUT Rational functions $(x(t), y(t))$ of degree at most four, such that $f(x(t), y(t)) = 0$.

ALGORITHM. As in the conic case, the curve is transformed into a birationally equivalent one that is readily parameterizable. Several transformations are used. The steps are detailed below. If the cubic has a zero x^3 or y^3 term, the first step is omitted, otherwise the first step cancels y^3. The computation is symmetric with respect to x.

1. Apply a transformation that removes the y^3 term of f. This can be done via the linear transformation $x = x_1 + qy_1, y = y_1$. When applied to the cubic equation $f(x,y) = 0$, this yields a new cubic curve with equation $f_1(x_1,y_1) = 0 = f(x_1+qy_1, y_1) = L(q)y_1^3 + f_2(x_1,y_1)$ where $L(q) = a_{03}q^3 + a_{12}q^2 + a_{21}q + a_{30}$. Choose q to be a root of L, i.e. $L(q) = 0$. Then the subexpression $L(q)y_1^3$ must vanish, so we only need to parameterize the curve $f_2(x_1, y_1) = 0$

2. Parameterize the cubic with equation $f_2(x_1,y_1) = 0$, which has no y_1^3 term, by transforming it into a transformed into a quadratic curve. f_2 is of the form

$$f_2(x_1,y_1) = g_1(x_1)y_1^2 + g_2(x_1)y_1 + g_3(x_1) \tag{5}$$

where g_1,g_2,g_3 have degrees equal to their subscripts. The *discriminant* of f_2 (with respect to y_1) is simply $g_4(x_1) = g_2(x_1)^2 - 4g_1(x_1)g_3(x_1)$. It can be shown that $g_4(x_1)$ must have a multiple root of the original cubic is singular, as assumed. By performing the following substitution

$$y_2 = 2g_1y_1 + g_2 \tag{6}$$

we have

$$\begin{aligned} 4g_1f_2 &= 4g_1^2y_1^2 + 4g_1g_2y_1 + 4g_1g_3 \\ &= (2g_1y_1 + g_2)^2 - (g_2^2 - 4g_1g_3) \\ &= y_2^2 - g_4 \end{aligned} \tag{7}$$

Note that $g_4(x_1)$ is a polynomial in x_1 of degree at most four. The curve is singular (and hence rational) if and only if $g_4(x_1)$ has a multiple root. This repeated root can be real or complex; only the real case is considered. Now for any number r, expand the polynomial $g_4(x_1)$ in a Taylor series at r: $g_4(x_1) = \sum_{i=0}^{4} \frac{g_4^{(i)}(r)}{i!}(x_1 - r)^i$. The terms of order higher than 4 are identically zero, g_4 being a polynomial of degree 4. Collecting coefficients of $(x_1 - r)^2$ yields

$$g_4(x_1) = q_2(x_1)(x_1 - r)^2 + g_4'(r)(x_1 - r) + g_4(r) \tag{8}$$

where $q_2(x_1)$ is of degree two. Now apply the substitution $y_3 = y_2/(x_1 - r)$ together with (8) into the right-hand side of (5); this leads to

$$\begin{aligned} &4g_1f_1 = y_2^2 - g_4(x_1) \\ &= (y_3^2 - q_2(x_1))(x_1 - r)^2 + g_4'(r)(x_1 - r) + g_4(r) \\ &= f_3(x_1, y_3) \end{aligned} \tag{9}$$

Choose r to be a multiple root of $g_4(x_1)$: then $g_4(r) = g_4'(r) = 0$, and the subexpression $g_4'(r)(x_1 - r) + g_4(r)$ must vanish. Therefore, to parameterize $f_3(x_1, y_3)$ we can simply parameterize the conic curve corresponding to the quadratic factor $C(x_1, y_3) = y_3^2 - q_2(x_1) = 0$.
3. Parameterize the conic with equation $C(x_1, y_3) = 0$ using the methods of the previous section. This yields a pair of rational functions $(x_1(t), y_3(t))$ that satisfy $C(x_1(t), y_3(t)) = 0$. Applying all the transformations in reverse yields one for the input cubic.

The cubic parameterization calls for computing a root q of the cubic polynomial $L(q)$, a multiple root r of the quartic polynomial $g_4(x_1)$, and a parameterization $(x_1(t), y_3(t))$ of the conic with equation $C(x_1, y_3) = 0$. Assuming, say, that the third conic transformation section was used, a pair of algebraic numbers (b, c) need to be computed.

5.1 Backward Error Analysis

If all computations were exact, i.e. $L(q) = 0$, $g_4(r) = 0$, and $C(x_1(t), y_3(t)) = 0$, then the output will be correct. However, one may need to use approximations $\tilde{q}, \tilde{r}$ and $(\tilde{b}, \tilde{c})$, which will lead to an approximate output parameterization $(\tilde{x}(t), \tilde{y}(t))$. In this case one must measure the error incurred. Once again, a backward error analysis will be performed, beginning with back-substitution.
LEMMA. The output parameterization will satisfy the implicit equation

$$f(x, y) - L(\tilde{q})y^3 - \frac{C(\tilde{b}, \tilde{c})(x - \tilde{q}y - \tilde{r})^2 + g_4'(\tilde{r})(x - \tilde{q}y - \tilde{r}) + g_4(\tilde{r})}{4g_1(x - \tilde{q}y)} = 0$$

PROOF. Given the approximate output parameter functions $(\tilde{x}(t), \tilde{y}(t))$, we compute $f(\tilde{x}(t), \tilde{y}(t))$. The subscript (t) is dropped for convenience. Then $f(\tilde{x}, \tilde{y}) = f(x_1 + \tilde{q}y_1, y_1) = L(\tilde{q})y_1^3 + f_2(x_1, y_1)$.

Applying transformations in reverse and performing some algebraic manipulation, we find that

$$f_2(\tilde{x_1}, \tilde{y_1}) = f_2(\tilde{x_1}, \frac{\tilde{y_2} - g_2(\tilde{x_1})}{2g_1(\tilde{x_1})}) = \frac{\tilde{y_2}^2 - g_4(\tilde{x_1})}{4g_1(\tilde{x_1})}$$
$$= \frac{(\tilde{y_3}^2 - q_2(\tilde{x_1}))(\tilde{x_1} - \tilde{r})^2 + g_4'(\tilde{r})(\tilde{x_1} - \tilde{r}) + g_4(\tilde{r})}{4g_1(\tilde{x_1})}$$

Now $C(\tilde{x_1}(t), \tilde{y_3}(t)) = \tilde{y_3}^2 - q_2(\tilde{x_1})$, and since we assumed that the third conic transformation was used to parameterize C, it follows that there is a point $(\tilde{b}, \tilde{c})$ such that $C(\tilde{x_1}(t), \tilde{y_3}(t)) = C(\tilde{b}, \tilde{c})$. The lemma follows by substituting and expanding previous identities. □

If the values $\tilde{q}, \tilde{r}, \tilde{b}, \tilde{c}$ are exact, then $L(\tilde{q}) = g_4(\tilde{r}) = g_4'(\tilde{r}) = C(\tilde{b}, \tilde{c}) = 0$, and it is clear that the parametric output curve coincides with the implicit input curve.

However, if the values are *not* exact, the output curve differs from the input curve. The coefficient perturbations are now present in many terms, not just one.

5.2 Exact Solutions

Finally, we show that in some cases, algebraic number computation is unnecessary for exact rational parameterization. A fact that appears to be known in Diophantine analysis is that a rational cubic curve with rational coefficients has a rational singular point. [1] This was apparently not well-known in the geometric modeling community; it is mentioned in a book on Diophantine equations ([15]).

Every rational cubic has a singular point. It is well-known (see, e.g. [21] for details) that 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:

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

Therefore, if extended precision rational arithmetic is allowed, one can theoretically 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. One way to compute the singularity rationally is as follows.

An affine singular point is found as a solution to the system of equations $f(x, y) = f_x(x, y) = f_y(x, y) = 0$. The x-coordinate of this solution will be a multiple rational root of the degree six polynomial $p(x) = resultant(f(x, y), f_x(x, y), y)$. The rational roots of a polynomial can be computed by applying the algorithm in ([14]). The resultant is computed using a subresultant remainder sequence; this may then be used to compute the y-coordinate ([4]). Each (x, y) pair found this way can be tested whether it additionally satisfies $f_x(x, y) = 0$; only one pair will satisfy the test.

[1] We are grateful to Allan Adler for alerting us to this fact.

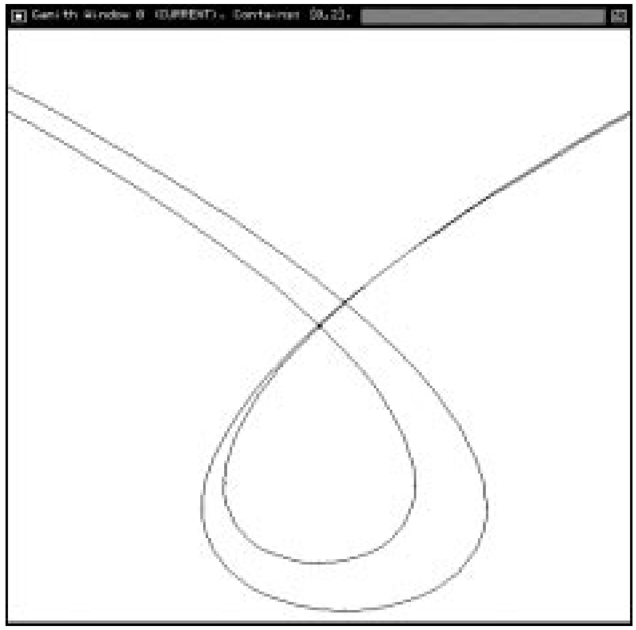

Figure 4: Exact and Perturbed Singular Cubics

6 Extensions to Cubic Surfaces and Monoid Hypersurfaces

We have reformulated algorithms for rational parameterizations conics, quadrics and singular cubic curves in a finite precision domain. Algebraic numbers are approximated by rationals to produce an approximate parameterization of an implicit curve or surface. For each method, we isolated the error due to the algebraic number approximation. The error formulas have useful geometric interpretations, some examples of which were given. In ongoing research, we find that the parameterization algorithms are quite stable. For instance, monoid parameterizations depend on the computation of the singular point (b, c) of a monoid curve (similarly for a monoid hypersurface[2] in any dimension). If a rational approximation $(\tilde{b}, \tilde{c})$ is calculated instead, we show that the monoid algorithm can be formulated so that the approximate output parametric curve will have a singularity at $(\tilde{b}, \tilde{c})$. In general, if a point is computed for a parameterization (conic, quadric, monoidal curve or surface), we show that the approximate point and the approximate output mimic the relationship of the exact point and the (exact) input (see Figure 4).

The method of rederiving the algorithms to work properly in finite precision arithmetic works well for low degree curve and surface parameterizations. In fact, we have been able to generate a formula to parameterize a cubic surface, in terms of its coefficients and certain algebraic numbers derived from extracting skew straight lines on its surface[7]. The rational parametric equations derived are of the fourth degree, comparable to [2], [7], [18].

Acknowledgements. This research was supported in part by AFOSR contract F49620-93-10138 and NSF grant CCR 92-22467. An earlier version of this paper appeared in the Proceedings of the 1992 Canadian Graphics Interface Conference.

[2]A monoid hypersurface is one which has a singular point of multiplicity one less than its degree.

References

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

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

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

[4] Abhyankar, S., and Bajaj, C., (1989), "Automatic Rational Parameterization of Curves and Surfaces IV: Algebraic Space Curves", *ACM Transactions on Graphics*, (1989), 8, 4, 325-334.

[5] Bajaj, C., (1990), "Rational Hypersurface Display," *Computer Graphics*, 24, 2, 117-128.

[6] Bajaj, C., and Royappa, A., (1990) "The Ganith Algebraic Geometry Toolkit", *Proceedings of the First International Symposium on the Design and Implementation of Symbolic Computation Systems*", Lecture Notes in Computer Science No. 429", 268-269.

[7] Bajaj, C., Holt, R., Netravali, A., (1995), "Rational Parameterization of Nonsingular Real Cubic Surfaces," *ACM Transactions on Graphics*, (1998), 1, 17, 1 - 31.

[8] Bookstein, F. L., (1979), "Fitting Conic Sections to Scattered Data," *Computer Graphics and Image Processing*, 9, 56-71.

[9] Canny, J. (1988) "Elimination Theory", *The Complexity of Robot Motion Planning*, ACM Doctoral Dissertation Series, MIT Press, chapter 3.

[10] Collins, G. (1967), "Subresultants and Reduced Polynomial Remainder Sequences," *JACM*, Vol. 14, No. 1, Jan. 1967, pp. 128-142.

[11] Farouki, R.T., (1987) On the Numerical Condition of Algebraic Curves and Surfaces 1. Implicit Equations, IBM Research Report RC 13263 (#59267), 1987.

[12] Golub, G., and van Loan, C. (1983), *Matrix Computations*, Johns Hopkins Press, pp 444-448.

[13] Hodge, W.V.D., and Pedoe, D. (1952), *Methods of Algebraic Geometry*, v.2, p. 69, Cambridge University Press.

[14] Loos, R., (1983) "Computing Rational Zeroes of Integral Polynomials by p-Adic Expansion", *SIAM J. Computing*, Vol. 12, No. 2, May 1983, pp 286-293.

[15] Mordell, L.J., (1969) *Diophantine Equations*, Academic Press.

[16] Salmon, G., (1879), *Conic Sections*, Longmans, Green and Co., London.

[17] Schwartz, J. and Sharir, M., (1983), On the Piano Movers Problem: II, General Techniques for Computing Topological Properties of Real Algebraic Manifolds, *Adv. Appl. Math.* Vol. 4, pp. 298–351.

[18] Sederberg, T., and Snively, J., (1987), "Parameterization of Cubic Algebraic Surfaces", *The Mathematics of Surfaces II*, ed. R. Martin, Oxford University Press, pp 299-321.

[19] Sommerville, D.M.Y., (1924), *Analytical Conics*, G. Bell and Sons, London.

[20] Sommerville, D.M.Y., (1951), *Analytical Geometry of Three Dimensions*, G. Bell and Sons, Cambridge University Press.

[21] Walker, R., (1950), *Algebraic Curves*, Springer Verlag 1978, New York