Chapter 2: Operations on Polynomial Curves and Surfaces

Chandrajit Bajaj and Andrew Gillette

September 21, 2010

Contents

Piecewise polynomials of fixed degree and continuously differentiable up to some order are known as splines or finite elements. Splines are used in applications ranging from computer-aided design, computer graphics, data visualization, geometric modeling, and image processing to the solution of partial differential equations via finite element analysis. The spline-fitting problem of constructing a mesh of finite elements that interpolate or approximate multivariate data is by far the primary research problem in geometric modeling. Parametric splines are vectors of a set of multivariate polynomial (or rational) functions while implicit splines are zero contours of collections of multivariate polynomials.

The various spline methods may be distinguished by several criteria:

- Implicit or parametric representations.

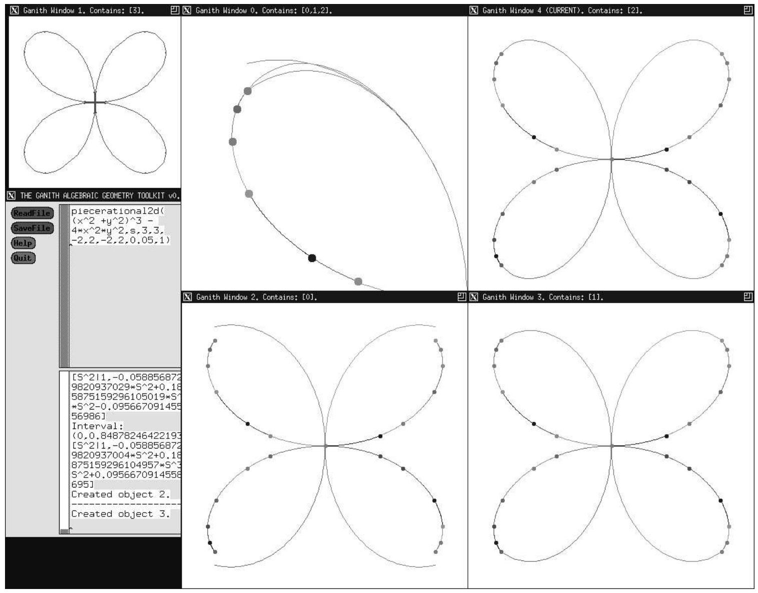


Figure 1: $(x^2 + y^2)^3 - 4x^2y^2 = 0$

- Algebraic and geometric degree of the spline basis.
- Number of surface patches required.
- Computation (time) and memory (space) required.
- Stability of fitting algorithms.
- Local or nonlocal interpolation.
- Splitting or nonsplitting of input mesh.
- Convexity or nonconvexity of the input and solution.
- Fairness of the solution (first- and second-order variation).

1 Plane Curves

1.1 Computation of Topology

Given a real algebraic plane curve $\mathbb{C}$: $f(x, y) = 0$ of degree d and of arbitrary genus, a box B defined by $\{(x, y) | \alpha \leq x \leq \beta, \gamma \leq y \leq \delta\}$, an error bound $\epsilon > 0$, and integers m, n with $m + n \leq d$ construct a C^{-1}, C^0 or C^1 continuous piecewise rational ϵ-approximation of all portions of $\mathbb{C}$ within the given bounding box B, with each rational function $\frac{P_i}{Q_i}$ of degree $P_i \leq m$ and degree $Q_i \leq n$. Here C^{-1} means no continuity condition is imposed between the different pieces, C^0 implies there are no gaps and C^1 implies that the first derivatives are continuous at the common end points of adjacent pieces. The ϵ-approximation here means that the approximation error is within given ϵ. The input curve $f(x, y) = 0$ may be reducible and have several real components but we assume it has components of only single multiplicity i.e. polynomial $f(x, y)$ has no repeated factors.

We use a combination of both algebraic and numerical techniques to construct a C^1-continuous, piecewise (m, n) rational ϵ-approximation by two different approaches, of a real algebraic plane curve. At singular points we rely on the classical resolution of plane curves [1, 29] based on

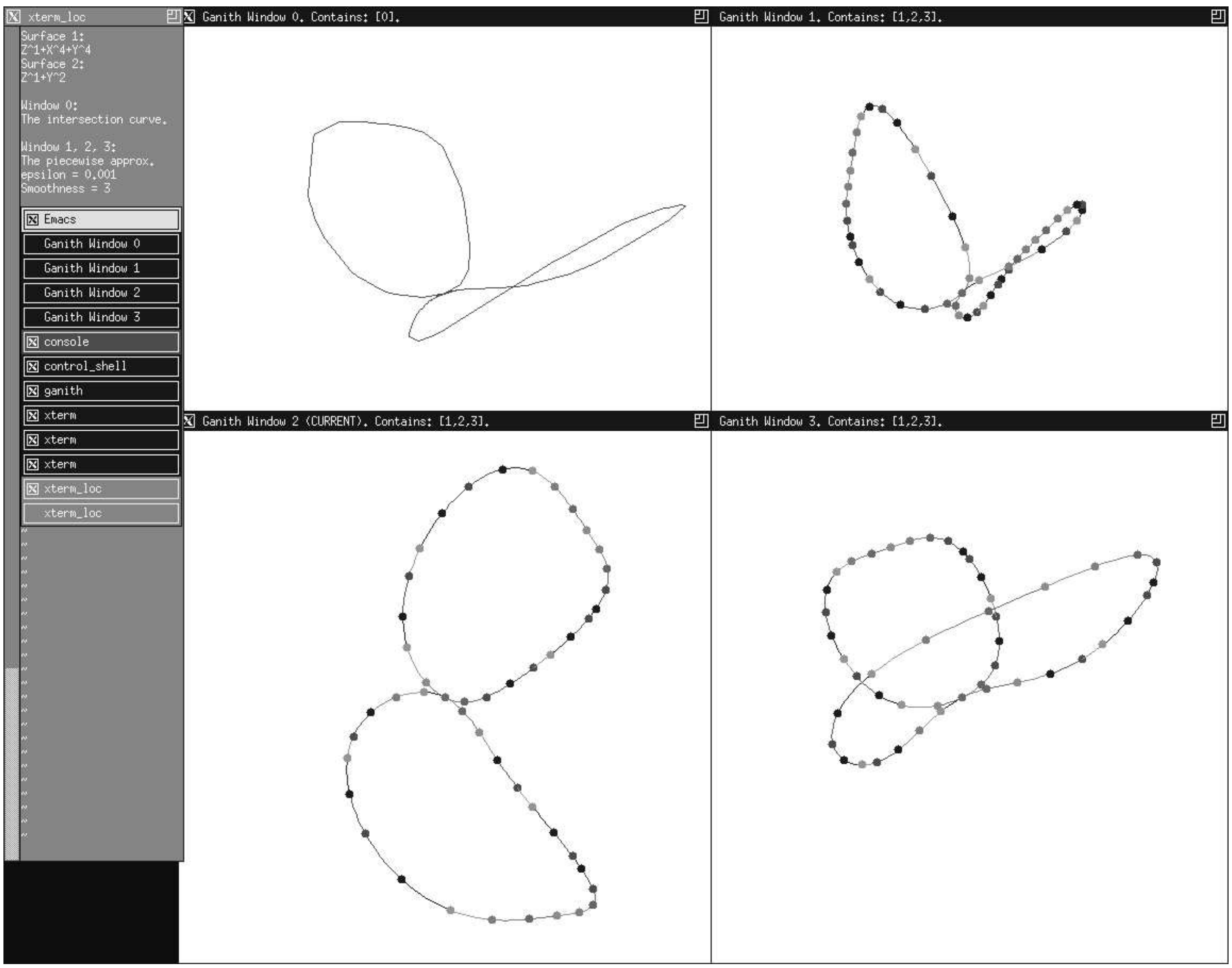


Figure 2: $x^4 + y^4 + z = 0$ and $y^2 + z = 0$

the Weierstrass Preparation Theorem [33] and Newton power series factorizations[21], using the technique of Hensel lifting[15]. These, together with modified Padé approximations [23], are used to efficiently construct locally approximate, rational parametric representations for all real branches of an algebraic plane curve. 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 simpler cases of C^{-1} and C^0 continuity are also handled in a similar manner. Details of the implementation of all these algorithms in GANITH [8] are also provided.

Sketch of Algorithm Input Given a real algebraic curve $\mathbb{C}$ of degree d, a bounding box B, a finite precision real number ϵ and integers m, n with $m + n \leq d$.

Output A C^{-1}, C^0 or C^1 continuous piecewise rational ϵ-approximation of all portions of $\mathbb{C}$ within the given bounding box B, with each rational function $\frac{P_i}{Q_i}$ of degree $P_i \leq m$ and degree $Q_i \leq n$ and $m + n \leq d$.

Algorithm We state the algorithm for a C^1 continuous piecewise rational ϵ-approximation. The C^{-1} and C^0 are similar and simpler.

1. Compute all intersections of the given real plane curve $\mathbb{C}$ within the given bounding box B and also the tracing direction at these points. Let the curve within the box B be denoted by $\mathbb{C}_B$. Next, compute all singular points S and x-extreme points T on the bounded plane curve $\mathbb{C}_B$. The set of points T act as starting points for smooth ovals of the curve $\mathbb{C}$ completely inside B.

2. Compute a Newton factorization for each singular point (x_i, y_i) in S and obtain a power series representation for each analytic branch of $\mathbb{C}$ at (x_i, y_i) and given by

$$\begin{cases} X(s) = x_i + s^{k_i} \\ Y(s) = \sum_{j=0}^{\infty} c_j^{(i)} s^j, \quad c_0^{(i)} = y_i \end{cases} \tag{1}$$

   or

$$\begin{cases} Y(s) = y_i + s^{k_i} \\ X(s) = \sum_{j=0}^{\infty} \tilde{c}_j^{(i)} s^j, \quad \tilde{c}_0^{(i)} = x_i \end{cases} \tag{2}$$

3. Without loss of generality, we only consider the case where the analytic branch at the singularity is of type (1). Compute $\frac{P_{mn}(s)}{Q_{mn}(s)}$ the (m, n) Padé approximation of $Y(s)$. That is $\frac{P_{mn}(s)}{Q_{mn}(s)} - Y(s) = O(s^{m+n+1})$

4. Compute $\beta > 0$ a real number, corresponding to points $(\tilde{x}_i = X(\beta), \tilde{y}_i = Y(\beta))$ and $(\hat{x}_i = X(-\beta), \hat{y}_i = Y(-\beta))$ on the analytic branch of the original curve $\mathbb{C}$, such that $\frac{P_{mn}(s)}{Q_{mn}(s)}$ is convergent for $s \in [-\beta, \beta]$ (see sub-sections (3.3.2) and (3.3.4) for details).

5. Modify $P_{mn}(s)/Q_{mn}(s)$ to $\tilde{P}_{mn}(s)/\tilde{Q}_{mn}(s)$ such that $\tilde{P}_{mn}(s)/\tilde{Q}_{mn}(s)$ is C^1 continuous approximation of $Y(s)$ on $[0, \beta]$, similarly modify $P_{mn}(s)/Q_{mn}(s)$ to $\hat{P}_{mn}(s)/\hat{Q}_{mn}(s)$ such that $\hat{P}_{mn}(s)/\hat{Q}_{mn}(s)$ is C^1 continuous approximation of $Y(s)$ on $[-\beta, 0]$(see subsections (3.3.1) and (3.3.3) for details).

6. Denote the set of all the points $(\tilde{x}_i, \tilde{y}_i)$, $(\hat{x}_i, \hat{y}_i)$, the set T and the boundary points of $\mathbb{C}_B$ by V. The curve $\mathbb{C}_B$ yields a natural multigraph[1] G having V as its vertex set and the set of curve segments of $\mathbb{C}_B$ joining any pair of points in V, as its edge set E. Now starting from each (simple) point (x_i, y_i) in V we trace out the multigraph G, approximating each of its

[1] A graph with perhaps multiple edges between a pair of vertices

edges by C^1 continuous piecewise rational curves as in the following:
Compute the Taylor expansion that yields (without loss of generality) the single analytic branch given, by

$$\begin{aligned} X(s) &= x_i + s \\ Y(s) &= \sum_{j=0}^{\infty} c_j^{(i)} s^j, \quad c_0^{(i)} = y_i \end{aligned}$$

Exactly the same steps as above are used for determining the Padé approximation, β and modified Padé approximants for a C^1 ϵ-approximation of these analytic branches. The C^1 continuity here is achieved at the point $(\tilde{x}_i = X(\beta), \tilde{y}_i = Y(\beta))$ between the original curve and the ϵ-approximation and the multigraph is updated with only this single vertex till all edges are visited exactly once in the Euler tour. For each visited edge the C^1 piecewise approximation rational curves are stored in a separate list and finally output.

Details and Correctness of Algorithm

1.1.1 Expansion at Simple Points

Let $f(x,y) = \sum a_{ij}x^i y^j = 0$ be an algebraic curve and (x_0, y_0) be a simple point on it. By a simple translation $x = \tilde{x} + x_0, \quad y = \tilde{y} + y_0$ we may assume that $(x_0, y_0) = (0,0)$, i.e., $f(0,0) = 0$. Since $(0,0)$ is a simple point of the curve, we assume without loss of generality, that $f_y(0,0) \neq 0$. Consider $f(x,y)$ in its recursive canonical form (RCF) form as a polynomial in y with coefficients polynomials in x:

$$f(x,y) = a_0(x) + a_1(x)y + \cdots + a_d(x)y^d, \tag{3}$$

with $a_i(x) = \sum_{j=0}^{m_i} a_{ij}x^j, \quad i = 0,1,\ldots,d$ and by the earlier assumption $a_1(0) = a_{10} \neq 0$. Let $y(x) = \sum_{i=1}^{\infty} A_i x^i = \sum_{i=1}^{\infty} A(1)_i x^i$ Then

$$y^j(x) = \sum_{i=j}^{\infty} A(j)_i x^i \tag{4}$$

where $A(j)_i = \sum_{i_1+\cdots+i_j=i} A_{i_1}A_{i_2},\ldots,A_{i_j}$. Hence $A(j+1)_i = \sum_{k=j}^{i-1} A(j)_k A_{i-k}$. Substituting (4) into $f(x,y) = 0$, we have

$$\begin{aligned} 0 &= a_0(x) + \textstyle\sum_{j=1}^{d} a_j(x) \sum_{i=j}^{\infty} A(j)_i x^i \\ &= a_0(x) + \textstyle\sum_{j=1}^{d} \quad \sum_{i=j}^{\infty} B(j)_i x^i \\ &= \textstyle\sum_{i=1}^{m_1} a_{0i}x^i + \sum_{i=1}^{\infty} B(1)_i x^i + \sum_{i=2}^{\infty} (\sum_{j=2}^{\min\{i,d\}} B(j)_i) x^i \end{aligned} \tag{5}$$

where

$$B(j)_i = \sum_{s=0}^{\min\{i-j,m_j\}} a_{js}A(j)_{i-s}, \quad j = 1,2,\ldots, \quad i \geq j \tag{6}$$

It follows from (5) that

$$B(1)_1 + a_{01} = 0 \tag{7}$$

$$B(1)_i + a_{0i} + \sum_{j=2}^{\min\{i,d\}} B(j)_i = 0, \ \ i = 2,3,\ldots, \tag{8}$$

Since $B(1)_i = \sum_{s=0}^{\min\{i-1,m_1\}} a_{1s}A_{i-s} = a_{10}A_i + \sum_{s=1}^{min\{i-1,m_1\}} a_{1s}A_{i-s}$, it follows from (7) and (8) that

$$\begin{aligned} &A_1 = -a_{01}/a_{10} \\ &A_i = -\left[\textstyle\sum_{s=1}^{\min\{i-1,m_1\}} a_{1s}A_{i-s} + a_{0i} + \sum_{j=2}^{\min\{i,d\}} B(j)_i\right]/a_{10}, \quad i = 2,3,\ldots \end{aligned} \tag{9}$$

1.1.2 Expansion at Singular Points

We rely on the classical resolution of algebraic plane curves [1, 29] based on the Weierstrass Preparation Theorem [33] and Newton power series factorizations[21], using the technique of Hensel lifting[15]. We repeat it here for the sake of completeness and to point out the modifications required for the special case of computing only the real branches at real singularities of the plane curve.

1.1.3 Newton Factorization

Consider $h(x, y)$, a monic polynomial in y of degree e, with no repeated factors and with coefficients polynomial or power series or meromorphic series in x (like the "distinguished" polynomial of the Weierstrass factorization)

$$h(x,y) \ = \ y^e \ + \ a_{e-1}(x)y^{e-1} \ + \ \cdots \ + \ a_0(x)$$

Then it is possible to factor $h(x, y)$ 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 meromorphic series. This factorization is also achieved via Hensel lifting. We precondition the curve so that it admits a non-trivial base factorization, i.e. having at least two real coprime factors which can be lifted.

Step 1: Cancel the term $a_{e-1}(x)$ via the substitution $\tilde{y} \ = \ y \ + \ \frac{a_{e-1}(x)}{e}$. Note, that the case when all other $a_i(x)$ terms also vanish under this substitution is when the original $h(x, y) = (y \ - \ \frac{a_{e-1}(x)}{e})^e$ (a repeated factor which does not occur for our input curves).

Step 2: Ensure some $a_{e-i}(0) \ \neq \ 0$ for $i \ \geq \ 2$ via the substitution $\breve{y} = \frac{\tilde{y}}{x^\lambda}$ with $\lambda = min_{(2 \leq i \leq e)} \frac{\alpha_i}{i}$ and $\alpha_i \ = \ ord_x a_{e-i}(x)$. Then $h(0, \breve{y}) \ = \ h_0(\breve{y})$ has at least two distinct roots. If the only roots are complex, return "No real branches at the origin" and skip Step 3.

Step 3: Use Hensel lifting to lift the factorization $h_0(\breve{y}) \ = \ g_0(\breve{y}) \ \breve{h}_0(\breve{y})$ with g_0 being linear, to $h(x, \breve{y}) \ = \ g(x, \breve{y}) \ \breve{h}(x, \breve{y})$ and apply the inverse of the coordinate substitutions in Steps 1 and 2. Repeat Steps 1-3 until all factors of h are linear or all real factors are obtained.

1.1.4 Local Parametrization

If the given curve $\mathbb{C}$ has a singularity with non-rational (algebraic) coordinates, we first compute a rational approximation to the singularity as well as determine the multiplicity order of the singularity. This order e is the minimum order of the partials which are greater than machine precision. Next we translate the curve to make the singularity to be at the origin, and also discard all monomial terms of degree less than e.

Hence, consider an implicit plane algebraic curve $\mathbb{C} : f(x, y) = 0$, with no repeated factors, and with a singularity at the origin. To compute a local parametric approximation of each of the curve's branches incident at the origin, we execute the following steps:

1. Compute a Weierstrass power series factorization of $f(x, y)$ into $f = gh$, where $g(x, y)$ is a unit power series and $h(x, y)$ is a polynomial in y with coefficients non-unit power series in x. The equation $h = 0$ corresponds to the curve's branches at the origin while the power series equation $g = 0$ corresponds to the portion of the plane curve away from the origin.

2. Recursively apply the Newton factorization to $h(x, y)$ till all factors are linear in y or all real factors are obtained. Each of these power series factors represent a local branch parametrization of the type $x = t^m$ and $y = b_i((t))$ where b_i is a power series. The minimum of m and $ord_t((b_i))$, say k, is known as the order of the branch, with $k > 1$ implying a singular branch of the curve.

1.1.5 C^1 Continuous Padé Approximation

C^1-continuity—Approach I Let $P_{mn}(s)/Q_{mn}(s)$ be the (m, n) Padé approximation of $Y(s)$, That is $\frac{P_{mn}(s)}{Q_{mn}(s)} - Y(s) = O(s^{m+n+1})$. Let $\beta > 0$ be a real number, corresponding to a point on the analytic branch of the original curve C, such that $\frac{P_{mn}(s)}{Q_{mn}(s)}$ is convergent for $s \in [0, \beta]$. This β is determined in Section 1.1.5.

Consider

$$\frac{\tilde{P}_{mn}(s)}{Q_{mn}(s)} = \frac{P_{mn}(s) + s^k(a + bs)}{Q_{mn}(s)}, \qquad 2 \le k < m \tag{10}$$

Note that the above choice of $\tilde{P}_{mn}(s)$ change neither the degree of the approximation nor the order of the approximation error (shown in subsection 1.1.5). On the other hand, it is easy to see that

$$Y(s) - \frac{\tilde{P}_{mn}(s)}{Q_{mn}(s)} = O(s^k)$$

Any choice of k within the allowed range suffices and we have currently left that as a parameter in our implementation. For a fixed choice of k, determine a and b such that

$$\frac{\tilde{P}_{mn}(\beta)}{Q_{mn}(\beta)} = Y(\beta), \qquad C^0 \text{ continuity} \tag{11}$$

and further

$$\left(\frac{\tilde{P}_{mn}}{Q_{mn}}\right)'(\beta) = Y'(\beta), \qquad C^1 \text{ continuity} \tag{12}$$

Hence, for C^0 continuity, we have

$$a = \frac{Y(\beta)Q_{mn}(\beta) - P_{mn}(\beta)}{\beta^k}, \tag{13}$$

and $b = 0$. For C^1 continuity, it follows from (11) and (12) that

$$a + b\beta = \frac{Y(\beta)Q_{mn}(\beta) - P_{mn}(\beta)}{\beta^k}, \tag{14}$$

$$ka + (k+1)b\beta = \frac{(YQ_{mn} - P_{mn})'(\beta)}{\beta^{k-1}}. \tag{15}$$

Since the matrix $\begin{bmatrix} 1 & \beta \\ k & (k+1)\beta \end{bmatrix}$ is nonsingular for $\beta \neq 0$, equations (14) and (15) have a unique solution and

$$a = \frac{(k+1)(YQ_{mn} - P_{mn})(\beta) - \beta(YQ_{mn} - P_{mn})'(\beta)}{\beta^k},$$

$$b = \frac{\beta(YQ_{mn} - P_{mn})'(\beta) - k(YQ_{mn} - P_{mn})(\beta)}{\beta^{k+1}}$$

For C^{-1} continuity , i.e no continuity constraints, $a = b = k = 0$. For C^0 continuity , i.e no gaps, $b = 0$ and a is computed as (13) for some fixed k such that $1 \le k \le m$.

Approximation Error Bound—Approach I We now compute $\beta > 0$ a real number, corresponding to a point on the analytic branch of the original curve $\mathbb{C}$, such that the segment $\frac{P_{mn}(s)}{Q_{mn}(s)}$ is convergent for $s \in [0, \beta]$. The following error analysis is based on the functional distance between the curve branch and the approximating segment. Similar error analysis can also be achieved for more geometric distance measures.

Note that $\tilde{P}_{mn}(s) = P_{mn}(s) + s^k(a + bs)$, where a and b are chosen to enforce C^1 continuity. Since

$$Y(s) - \frac{\tilde{P}_{mn}(s)}{Q_{mn}(s)} = \frac{Y(s)Q_{mn}(s) - P_{mn}(s) - s^k(a + bs)}{Q_{mn}(s)},$$

$s^k(a + bs)$ can be regarded as an C^1 interpolating polynomial of $Y(s)Q_{mn}(s) - P_{mn}(s)$ at points 0 and β. Hence we have

$$Y(s) - \frac{\tilde{P}_{mn}(s)}{Q_{mn}(s)} = \frac{(YQ_{mn} - P_{mn})^{(k+2)}(\xi)}{Q_{mn}(s)(k + 2)!} s^k(s - \beta)^2, \quad \xi \in (0, \beta).$$

where $(YQ_{mn} - P_{mn})^{(k+2)}$ is the $(k + 2)^{th}$ derivative of the power series. Since

$$|s^k(s - \beta)^2| \le \frac{4k^k\beta^{k+2}}{(k + 2)^{(k+2)}}, \qquad s \in [0, \beta],$$

we have

$$\left|Y(s) - \frac{\tilde{P}_{mn}(s)}{Q_{mn}(s)}\right| \le \left|\frac{(YQ_{mn} - P_{mn})^{(k+2)}(\xi)}{Q_{mn}(s)(k + 2)!}\right| \frac{4k^k\beta^{k+2}}{(k + 2)^{(k+2)}}$$

From $(YQ_{mn} - P_{mn})(s) = \sum_{i=m+n+1}^{\infty} e_i s^i$, we have

$$\left|\frac{(YQ_{mn} - P_{mn})^{(k+2)}(\xi)}{(k + 2)!}\right| \le \sum_{i=m+n+1}^{\infty} |e_i| a^i_{k+2} \beta^{i-k-2}.$$

Let $Q^{-1}_{mn}(s) = \sum_{i=0}^{\infty} q_i s^i$ and $|Q^{-1}_{mn}(s)| \le \sum_{i=0}^{\infty} |q_i|\beta^i$, then

$$\left|\frac{(YQ_{mn} - P_{mn})^{(k+2)}(\xi)}{Q_{mn}(s)(k + 2)!}\right| \le \left|\frac{(YQ_{mn} - P_{mn})^{(k+2)}(\xi)}{(k + 2)!}\right| |Q^{-1}_{mn}(s)| = (\sum_{i=0}^{\infty} r_i\beta^i)\beta^{m+n-k-1}$$

Therefore from the above analysis and the previous subsection we have

Theorem 1. *Let*

$$\sum_{i=0}^{\infty} r_i\beta^i = (\sum_{i=m+n+1}^{\infty} |e_i| a^i_{k+2} \beta^{i-m-n-1})(\sum_{i=0}^{\infty} |q_i|\beta^i).$$

Then

$$1^* \quad \left(\frac{\tilde{P}_{mn}}{Q_{mn}}\right)^{(i)}(0) = Y^{(i)}(0), \quad i = 0, 1, \cdots, k - 1.$$

$$2^* \quad \left(\frac{\tilde{P}_{mn}}{Q_{mn}}\right)(\beta) = Y(\beta), \quad \left(\frac{\tilde{P}_{mn}}{Q_{mn}}\right)^{(1)}(\beta) = Y^{(1)}(\beta).$$

$$3^* \quad \left| Y(s) - \frac{\tilde{P}_{mn}(s)}{Q_{mn}(s)} \right| \le (\sum_{i=0}^{\infty} r_i \beta^i) \frac{4k^k \beta^{m+n+1}}{(k+2)^{(k+2)}}, \quad s \in [0, \beta]. \tag{16}$$

An interesting property of (16) is that the order of the approximation does not depends on k. Further, the error bound depends on P_{mn}/Q_{mn} but not a and b. Hence (16) can be used to compute the approximation range β after Padé approximation is obtained. For C^0 continuity, a similar bound can be obtained.

In our implementation, we take $\sum_{i=0}^{\infty} r_i \beta^i \approx r_0 + r_1 \beta$ and then determine β_1 such that

$$(r_0 + r_1 \beta_1) \frac{4k^k \beta_1^{m+n+1}}{(k+2)^{(k+2)}} \le \epsilon$$

Next compute the smallest pole of the rational function, i.e.

$$\beta_2 = \zeta * \min\{z_i : Q_{mn}(z_i) = 0, \quad z_i \in I\!R\}$$

for some positive constant $\zeta < 1$ and take $\beta_3 = \min\{\beta_1, \beta_2\}$. From the point on the Padé approximation $(X(\beta_3), P_{mn}(\beta_3)/Q_{mn}(\beta_3))$ we compute via Newton's method the nearest point $(\tilde{x}_i, \tilde{y}_i)$ on the analytic branch

$$\begin{aligned} X(s) &= x_i + s^{k_i} \\ Y(s) &= \sum_{j=0}^{\infty} c_j^{(i)} s^j \end{aligned}$$

of the original curve $f(x, y) = 0$. Finally we determine β from the equation $\beta^{k_i} = \tilde{x}_i - x_i$.

C^1-continuity—Approach II The second method for getting C^1 (or C^0) continuous Padé approximation is to first modify the $Y(s)$ as $\tilde{Y}(s)$:

$$\tilde{Y}(s) = \sum_{i=0}^{\infty} \tilde{c}_i s_i = \begin{cases} Y(s) & \text{for } C^{-1} \\ Y(s) + a s^{m+n} & \text{for } C^0 \\ Y(s) + s^{m+n-1}(b + as) & \text{for } C^1 \end{cases} \tag{17}$$

and then compute (m, n) Padé approximation $P_{mn}(s)/Q_{mn}(s)$ for $\tilde{Y}(s)$. After that determine a and b such that $P_{mn}(s)/Q_{mn}(s)$ is C^1 (or C^0) continuous on $[0, \beta]$. If $n = 0$, the problem is reduced to the approach one of last subsection with $k = m - 1$. Now assume $n > 1$. From the expression of Padé approximation:

$$P_{mn}(s, a, b) = det \begin{bmatrix} c_{m+1} & c_m & \cdots & c_{m-n+1} \\ \cdots & \cdots & \cdots & \cdots \\ c_{m+n} + a & c_{m+n-1} + b & \cdots & c_m \\ \sum_{i=0}^{m} \tilde{c}_i s^i & \sum_{i=0}^{m-1} \tilde{c}_i s^{i+1} & \cdots & \sum_{i=0}^{m-n} \tilde{c}_i s^{i+n} \end{bmatrix}$$

$$Q_{mn}(s, a, b) = det \begin{bmatrix} c_{m+1} & c_m & \cdots & c_{m-n+1} \\ \cdots & \cdots & \cdots & \cdots \\ c_{m+n} + a & c_{m+n-1} + b & \cdots & c_m \\ 1 & s & \cdots & s^n \end{bmatrix},$$

we know that $P_{mn}(s, a, b)$, $Q_{mn}(s, a, b)$ is linear in a, degree at most 2 in b. Then by

$$\left(\frac{P_{mn}(s)}{Q_{mn}(s)} \right)_{s=\beta} = Y(\beta), \quad \left(\frac{P_{mn}(s)}{Q_{mn}(s)} \right)'_{s=\beta} = Y'(\beta)$$

We have

$$\begin{cases} P_{mn}(\beta,a,b) - Y(\beta)Q_{mn}(\beta,a,b) = 0 \\ P'_{mn}(\beta,a,b) - Y(\beta)Q'_{mn}(\beta,a,b) - Y'(\beta)Q_{mn}(\beta,a,b) = 0 \end{cases} \tag{18}$$

From the first equation, we get $a = d_0b^2 + d_1b + d_2$ for some constant $d's$. Substitute a into the second equation, we get degree 2 equation $a_0b^2 + a_1b + a_2 = 0$ of b. It has two solutions. At this moment one may immediately ask the question: Which solution do you choose from? Is the problem of modified Padé approximation equivalent to rational Hermite interpolation problem at points 0 and β? Now we try to answer the questions.

Lemma 1. *Denote*

$$A(m/n) = det \begin{bmatrix} c_m & c_{m-1} & \cdots & c_{m-n+1} \\ \cdots & \cdots & \cdots & \cdots \\ c_{m+n-1} & c_{m+n-2} & \cdots & c_m \end{bmatrix}$$

Then if $A(m-1/n-1)$ is nonsingular when $n > 1$, then there exists uniquely a and b such that

$$rank \begin{bmatrix} c_{m+1} & c_m & \cdots & c_{m-n+1} \\ \cdots & \cdots & \cdots & \cdots \\ c_{m+n} + a & c_{m+n-1} + b & \cdots & c_m \end{bmatrix} < n, \quad for \ \ n > 0$$

Proof. If $n = 1$, take $a = -c_{m+n}, b = -c_{m+n-1}$. Then the matrix considered is reduced to zero. Hence the needed conclusion holds. Now suppose $n > 1$ and let A be the above matrix and A_1 be the last n columns of A. Then by the assumption of the lemma we know that there exists uniquely

$$b = (-1)^n \frac{A(m/n)}{A(m-1/n-1)}$$

such that the matrix A_1 is singular. For such b there exists uniquely a such that the last row of A can be expressed by the first $n-1$ rows and this complete the proof of the lemma. $\Diamond$

From the expression of Padé approximation and this lemma we know that there exists uniquely a and b such that $P_{mn} = Q_{mn} = 0$. Therefore equation (18) is satisfied by this trivial solution and this solution is not what we wanted. We should choose the solution which does not make Q_{mn} to be zero. After knowing which solution we should choose, we can answer the second question. Since the solution of the rational Hermite interpolation problem is unique, the modified Padé approximation must be the rational Hermite interpolant. On the other hand, expanding the rational Hermite interpolant into power series at origin and then computing the Padé approximant of the power series, we would get the same rational function by the uniqueness of the Padé approximation. Hence the modified Padé approximation problem is equivalent to the rational Hermite interpolation problem. The computation approach here is easier than the known methods for computing the rational Hermite interpolant.

Theorem 2. *Let $R_{mn}(s,a,b) = P_{mn}(s,a,b)/Q_{mn}(s,a,b)$ be the Padé approximation of $\tilde{Y}(s) = \sum_{i=0}^{\infty} \tilde{c}_i s^i$ defined as (17) and $A(m-1/n-1)$ and $A(m/n)$ are not zero.*

1. *Then for any b, exists a such that*

$$R_{mn}(\beta,a,b) = Y(\beta), \quad \beta \neq 0 \tag{19}$$

if $Q_{m-1,n-1}(\beta) \neq 0$ and $r_{m-1,n-1}(\beta) \neq 0$.

2. *If the above condition is satisfied, then there exist a and b such that*

$$R'_{mn}(\beta,a,b) = Y'(\beta), \quad \beta \neq 0 \tag{20}$$

if $\left(\frac{sr_{m-2,n-2}(s)}{r_{m-1,n-1}(s)}\right)'_{s=\beta} \neq 0$ *and*

$$A^2(m-1/n-1)\left(\frac{r_{m,n}(s)}{sr_{m-1,n-1}(s)}\right)'_{s=\beta} \neq A^2(m/n)\left(\frac{sr_{m-2,n-2}(s)}{r_{m-1,n-1}(s)}\right)'_{s=\beta} \tag{21}$$

where $r_{mn}(s) = Y(s)Q_{mn}(s) - P_{mn}(s)$ *and* $A^2 = A * A$.

Proof. From the determinant expression of Padé approximation, we have

$$\begin{aligned} P_{mn}(s,a,b) &= (-1)^{n-1}asP_{m-1,n-1}(s) + b^2 s(sP_{m-2,n-2}(s)) + \cdots + P_{mn}(s) \\ Q_{mn}(s,a,b) &= (-1)^{n-1}asQ_{m-1,n-1}(s) + b^2 s(sQ_{m-2,n-2}(s)) + \cdots + Q_{mn}(s) \end{aligned}$$

where $\cdots$ part is a linear monomial in b and $(sP_{m-2,n-2}(s), sQ_{m-2,n-2}(s)) = (-s^{m-1}, 0)$ for $n = 1$. Then (19) holds if $ar_{m-1,n-1}(\beta) = (-1)^n[b^2\beta r_{m-2,n-2}(\beta) + \cdots + r_{mn}(\beta)]$. Hence there exists a such that this equality holds for any b if $r_{m-1,n-1}(\beta) \neq 0$. This is conclusion one.

Substitute P'_{mn} and Q'_{mn} into the second equation of (18), after some computations, one get the following equation

$$(-1)^{n-1}a(sr_{m-1,n-1}(s))' + b^2(s^2 r_{m-2,n-2}(s))' + \cdots + (r_{mn}(s))' = 0.$$

It follows from

$$a = (-1)^n\left[\frac{\beta^2 r_{m-2,n-2}(\beta)}{\beta r_{m-1,n-1}(\beta)}b^2 + \cdots + \frac{r_{mn}(\beta)}{\beta r_{m-1,n-1}(\beta)}\right]$$

that

$$\left(\frac{s^2 r_{m-2,n-2}(s)}{sr_{m-1,n-1}(s)}\right)'_{s=\beta} b^2 + \cdots + \left(\frac{r_{mn}(s)}{sr_{m-1,n-1}(s)}\right)'_{s=\beta} = 0 \tag{22}$$

From Lemma 1 we know that this equation has a trivial solution which makes $P_{mn} = Q_{mn} = 0$. It has two solutions if and only if the highest coefficient of the equation is not zero. Let the quadratic equation (22) be denoted as $\alpha_2 b^2 + \alpha_1 b + \alpha_0 = 0$ with roots r_1 and r_2 then since $r_1 * r_2 = \frac{\alpha_0}{\alpha_2}$, we know that in order to make the two solutions of (22) be distinct, it suffices to satisfy the inequality (21). $\diamondsuit$

1.1.6 The Computation of the Singularity

The computation of the singularity consists of two sub-problems. One is to find the singular points, the other is to determine the order of the singular points. The singular points computed should have good accuracy such that the correct order can be determined from which.

For finding the singular points, we solve the equations

$$\begin{cases} f(x,y) = 0 \\ \alpha f_x(x,y) + \beta f_y(x,y) = 0 \end{cases} \tag{23}$$

using multivariate resultants and based on the method of birational maps [3], where the constants α and β are chosen such that f and $\alpha f_x + \beta f_y$ are coprime. In this method, we are led to solving a system of equations in the following form:

$$\begin{cases} \phi_0(X) = 0 \\ Y = \phi_1(X) \end{cases} \tag{24}$$

with (X,Y) and (x,y) are linearly related, where ϕ_0 is a polynomial and ϕ_1 is a rational function. The first equation of (24) can be solved by calling C-library to get the initial approximate values and then using iterative methods to get the higher precision solutions. Then the solution of (23) is received by the second equation of (24) and the linear relation between (X, Y) and (x, y).

In this approach, the equation (24) is produced by symbolic computation. After the initial value is got by numerical methods, the refinement afterwards is also symbolic. In the development of the following, we shall determine the required precision of the refinement. This precision will guarantee that the order of the singular point is correctly determined.

Let $p^* = (x^*, y^*)$ be a singular point of $f(x, y) = 0$. Then the order of it is the minimal integer k, for which $f_{ij}(p^*) \neq 0$ for at least one pair (i, j) with $i + j = k$, where $f_{ij} = \frac{\partial^{i+j} f}{\partial x^i \partial x^j}$. In order to determine the correct order from the approximate singular point of p^*, we need to know the minimal value of $|f_{ij}(p^*)|$ if it is nonzero. The lower bound of this value can be estimated by the following gap theorem:

Gap Theorem. *Let $\mathcal{P}(d, c)$ be the class of integral polynomials of degree d and maximum coefficient magnitude c. Let $f_i(x_1, \cdots, x_n) \in \mathcal{P}(d, c)$, $i = 1, \cdots, n$ be a collection of n polynomials in n variables which has only finitely many solutions when homogenized. If $(\alpha_1, \cdots, \alpha_n)$ is a solution of the system, then for any j either $\alpha_j = 0$, or $|\alpha_j| > (3dc)^{-nd^n}$*

For a given integer pair (i, j), let $z^* = f_{ij}(p^*)$, then using Gap Theorem to the following system:

$$\begin{cases} f(x, y) = 0 \\ f_x(x, y) = 0 \\ z - f_{ij}(x, y) = 0 \end{cases} \tag{25}$$

with a known solution (x^*, y^*, z^*), we know that if $z^* \neq 0$, then $|z^*| > (3dc_{ij})^{-3d^3}$, where d is the degree of f and c_{ij} is the maximum coefficient magnitude of the left hand side of (25). By this inequality, we can get the following criterion for testing whether $f_{ij}(p^*)$ is zero:

Test Criterion. *Let $gap = (3dc_{ij})^{-3d^3}$*

$$|f_{ij}(p) - f_{ij}(p^*)| < \frac{1}{2} gap \tag{26}$$

then $f_{ij}(p^*) = 0$ if and only if $|f_{ij}(p)| < \frac{1}{2} gap$.

Proof. If $|f_{ij}(p)| < \frac{1}{2} gap$, then

$$|f_{ij}(p^*)| \leq |f_{ij}(p)| + |f_{ij}(p) - f_{ij}(p^*)| < gap$$

Hence $f_{ij}(p^*) = 0$. On the other hand, if $f_{ij}(p^*) = 0$, then (26) implies the required inequality.

The Precision of the Singular Points. It follows from the test criterion above that, for knowing whether $f_{ij}(p^*) = 0$ or not, the computed singular point should have such a precision that the inequality (26) holds. Suppose the singular point is in the given bounding box $B = \{(x, y) : a_1 \leq x \leq a_2, b_1 \leq y \leq b_2\}$ in which the rational approximations are constructed, then by the Mean Value theorem, we have

$$\begin{aligned} |f_{ij}(p) - f_{ij}(p^*)| &\leq \sqrt{f_{i+1,j}(p + \theta p^*) + f_{i,j+1}(p + \theta p^*)} \|p - p^*\| \\ &\leq M_{ij} \|p - p^*\| \end{aligned}$$

where $\theta \in (0, 1)$, $M_{ij} = \sqrt{\sum_{ij} (|b_{i+1,j}| s + |b_{i,j+1}| t) s^i t^j}$, b_{ij} is defined by $f_{ij}(x, y) = \sum_{ij} b_{ij} x^i y^j$, $s = max\{|a_1|, |a_2|\}$ and $t = max\{|b_1|, |b_2|\}$. Therefore, (26) holds if

$$\|p - p^*\| \leq \frac{gap}{2M_{ij}}$$

That is, the computation of p^* should make p^* to have the accuracy $\frac{gap}{2M_{ij}}$ and therefore the computation should use $-log(\frac{gap}{2M_{ij}})/log2$ binary bits.

1.1.7 Implementation Issues

The rational approximation algorithms has been implemented in its entirety as part of GANITH, an X-11 based interactive algebraic geometry toolkit, using Common Lisp for the symbolic computation and C for all numeric and graphical computation. The input curve is assumed to have integral coefficients, which are considered to be exact. Floating point coefficients are allowed in the input curve representations, which are then converted to rational numbers and then converted to integers.

The Hensel power series computations of section 3, as well as its use in sections Weierstrass and Newton factorizations are based on a robust implementation of the fast Euclidean HGCD algorithm [9]. Rational Padé approximants are also computed based on the same HGCD algorithm, [9]. 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. All these computations can be numerical.

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(see section 4) as well as the extreme points computations are done in GANITH using multivariate resultants and based on the method of birational maps [3]. The intersection points of the curve with the bounding box are computed by letting x or y to be constant and then solving one unknown equation. In these computations, symbolic as well as numerical computations are used.

1.2 Newton Iterations

While tracing a surface-surface intersection curve SC, at simple (regular) points of SC we need to solve an undetermined nonlinear system that has more unknowns than equations. At singular points on the curve, we need to solve an overdetermined nonlinear system that has more equations and less unknowns. Consider in general an arbitrary system of nonlinear equations

$$F(x) = \begin{bmatrix} f_1(x_1, \ldots, x_m) \\ \cdots\cdots\cdots\cdots \\ f_n(x_1, \ldots, x_m) \end{bmatrix}$$

We need to determine solution of the system $F(x) = 0$ by Newton iterations from a given initial value $p_0 \in \mathbb{R}^m$. In our tracing procedure these initial values are points on the local expansion curves, within an adaptively computed step length. These initial values are then refined back to the original intersection curve SC to yield the actual interpolating points for the rational curve segment approximation. The Newton iteration used is

$$\nabla F(p_k)\Delta_k = -F(p_k), \quad p_{k+1} = p_k + \Delta_k \tag{27}$$

where $\nabla F = \begin{bmatrix} \frac{\partial F}{\partial x_1} & \frac{\partial F}{\partial x_2} & \cdots & \frac{\partial F}{\partial x_m} \end{bmatrix} = \begin{bmatrix} \nabla f_1 \\ \vdots \\ \nabla f_n \end{bmatrix}$ is a $n \times m$ matrix.

Case A: $\mathbf{m = n + 1}$. Here equation (27) is a under-determined linear system. Suppose the set of ∇f_i is linearly independent, then the general solution of (27) is

$$\Delta_k = \alpha_k t + \nabla F(p_k)^T \beta_k \tag{28}$$

where $t \in \nabla F(p_k)^{\perp}$, $\alpha_k \in I\!R$ is arbitrary and $\beta_k \in I\!R^n$ satisfies the following equation

$$\nabla F(p_k) \nabla F(p_k)^T X = -F(p_k) \tag{29}$$

This has a unique solution since $\nabla F(p_k)$ is of full rank. Finally, α_k is chosen as follows:

1. IIS case
   In this case, $m = 3$, $n = 2$ and t in (28) is the tangent direction of the curve. The change of p_k in the direction of t should be as small as possible. Therefore, we set $\alpha_k = 0$.
2. IPS case
   Now $m = 4$, $n = 3$ and $p = (x_1, x_2, x_3, x_4)^T := (u_1, v_1, u_2, v_2)^T$

   $$f_i(x_1, x_2, x_3, x_4) = G_{i1}(x_1, x_2) - G_{i2}(x_3, x_4), \qquad i = 1, 2, 3$$

   The initial value is given by (59), i.e., $p_0 = (Q_1(s_0)^T, Q_2(s_0)^T)^T$, where s_0 is the step length of the approximation of $r(s)$. In order to determine α_k in (28), we project $p_{k+1} \in I\!R^4$ (domain space) into $I\!R^3$ (value space) by

   $$X_1(p_k^{(1)}) + \nabla X_1(p_k^{(1)}) \Delta_k^{(1)}$$

   where $p_k^{(1)}$ (or $p_k^{(2)}$) and $\Delta_k^{(1)}$ (or $\Delta_k^{(2)}$) are the first (or last) two components of p_k and Δ_k, respectively. Let

   $$n_1 = X_{1u_1}(p_k^{(1)}) \times X_{1v_1}(p_k^{(1)}), \qquad n_2 = X_{2u_2}(p_k^{(2)}) \times X_{2v_2}(p_k^{(2)})$$

   and $n_3 = n_1 \times n_2$. Then there exist $\tilde{\alpha}_k \in I\!R$, $\tilde{\beta}_k \in I\!R^2$ such that

   $$X_1(p_k^{(1)}) + \nabla X_1(p_k^{(1)}) \Delta_k^{(1)} = \tilde{\alpha}_k n_3 + [n_1, n_2] \tilde{\beta}_k$$

   and $\tilde{\beta}_k$ is determined uniquely by

   $$\begin{bmatrix} n_1^T X_1(p_k^{(1)}) \\ n_2^T X_2(p_k^{(2)}) \end{bmatrix} = [n_1, n_2]^T [n_1, n_2] \tilde{\beta}_k$$

   and

   $$\begin{aligned} \tilde{\alpha}_k &= n_3^T X_1(p_k^{(1)}) + n_3^T \nabla X_1(p_k^{(1)}) \Delta_k^{(1)} \\ &= n_3^T X_1(p_k^{(1)}) + n_3^T \nabla X_1(p_k^{(1)}) [\alpha_k t^{(1)} + \nabla F^{(1)}(p_k)^T \beta_k] \\ &= a(p_k) \alpha_k + b(p_k) \end{aligned}$$

   where $a(p_k)$ and $b(p_k)$ are constants depending on p_k. For the same reason as IIS, we take $\tilde{\alpha}_k = 0$. Hence $\alpha_k = -\frac{b(p_k)}{a(p_k)}$.

Case B: $\mathbf{n > m}$. This case happens when we arrive at a singular point on the intersection curve SC (see Section 9). Now system (27) is over-determined. So we find the least squares approximate solution, i.e.,

$$\nabla F(p_k)^T \nabla F(p_k) \Delta_k = -\nabla F(p_k)^T F(p_k) \tag{30}$$

1.3 Rational Curve Hermite Interpolation between Simple Points

Let $r_1(u)$, $r_2(v)$ be two space curves, where u and v are arc lengths of the curves measured from some point on the respective curve. At point $u = u_0$, $v = v_0$, if

$$r_1^{(i)}(u_0) = r_2^{(i)}(v_0), \qquad i = 0, 1, \dots, k$$

we say that r_1 and r_2 are k-frame connected, or the composite curve is k-frame continuous. In particular, if $k = 3$, we say the curve is frame continuous.

Given a point p_0 on the curve $r(s)$, which is either IIS, IPS or PC, the arc length s is measured from p_0 (i.e., $r(0) = p_0$) in the given direction.

Step Length

From the approximation $r(s) \approx \sum_{i=0}^{k+1} r^{(i)}(0)\frac{s^i}{i!}$, we compute a trial step length $\beta > 0$ such that

$$\frac{\|r^{(k+1)}(0)\|\beta^{k+1}}{(k+1)!} \; / \; \|\sum_{i=0}^{k} \frac{r^{(i)}(0)\beta^i}{i!}\| < \epsilon \tag{31}$$

For such a β, using $\sum_{i=0}^{k+1} r^{(i)}(0)\beta^i/i!$ (for IIS), or $[\sum_{i=0}^{k+1} Q_1^{(i)}(0)^T\beta^i/i!, \; \sum_{i=0}^{k+1} Q_2^{(i)}\beta^i/i!]^T$ (for IPS) as initial value, we compute a new point p_1 on the curve by Newton iterations (section 6).

We then construct rational approximations as follows:

A. Rational Hermite interpolation

Let m, n be two nonnegative integers and $m + n = 2k + 1$. We construct a rational vector function $R(s) = [R_1(s), R_2(s), R_3(s)]^T$, where $R_i(s) = P_{mi}(s)/Q_{ni}(s)$, $i = 1, 2, 3$ are (m,n) type rational functions, such that

$$R^{(i)}(s) = r^{(i)}(s), \quad i = 0, 1, \dots, k \tag{32}$$

for $s = 0$ and $s = \beta$.

If either $Q_{ni}(s)$ has zeros in $[0, \beta]$ or the error $\max_{s\in[0,\beta]} \|r(s) - R(s)\| > \epsilon$, we halve the β. The approximation error is bounded in the following way:

Since

$$e_i(s) = r_i(s)Q_{ni}(s) - P_{mi}(s) = O(s^{k+1}(s-\beta)^{k+1}),$$

by the remainder formula of Hermite interpolation, we have

$$e_i(s) = [s(s-\beta)]^{k+1}(r_iQ_{ni})[\underbrace{0, \dots, 0}_{k+1}\; \underbrace{\beta, \dots, \beta}_{k+1},\; s],$$

where $f[t_0, \dots, t_r]$ stands for divided difference of f on $t_0, \dots, t_r$. Hence

$$|r_i(s) - R_i(s)| \le \left(\frac{\beta}{2}\right)^{2k+2} \frac{|D_{ki}(s)|}{\min_{s\in[0,\beta]} |Q_{ni}(s)|} \tag{33}$$

where $D_{ki}(s) = (r_iQ_{ni})[0, \dots, 0, \beta, \dots, \beta, s]$ is a function in s. That can be bounded approximately by either

$$|D_{ki}(0)| + |D_{ki}(\beta)| \quad \text{or} \quad \max_{s\in[0,\beta]} |\tilde{D}_{ki}(s)|$$

where $\tilde{D}_{ki}(s)$ is the interpolation polynomial of degree 2 at $D_{ki}(0)$, $D_{ki}(\frac{\beta}{2})$ and $D_{ki}(\beta)$. Let $g = r_i/Q_{ni}$. Then the divided difference can be computed by the following well known recurrence

$$g[t_0, \ldots, t_k] = \begin{cases} g^{(k)}(t_0)/k! & \text{if } t_0 = \ldots = t_k \\ \dfrac{g[t_0, \ldots, t_{r-1} t_{r+1}, \ldots, t_k] - g[t_0, \ldots, t_{s-1}, t_{s+1}, \ldots, t_k]}{t_s - t_r} & \text{if } t_r \neq t_s \end{cases}$$

B. Rational Vector Hermite Interpolation

We construct a rational function

$$R(s) = [P_{m1}(s), P_{m2}(s), P_{m3}(s)]^T / Q_n(s)$$

such that (32) holds and

$$m + n/3 = 2k + 1 \tag{34}$$

where n is divisible by 3. Now each component of the vector rational function has the same denominator. But the degree $m + n$ of each component is higher than the previous case. However, if we transform the vector rational function in case 1 into a rational function that has common denominator, then the degree is higher than in case 2. This transform is necessary when we represent the curve in rational Bernstein-Bézier form.

The error bound of the approximation can be estimated in the same way as before.

C. Two Point Padé Approximation

The two point Padé approximation method discussed here consists of the following two steps. First, compute the Padé approximation $P_{m_1 i}(s)/Q_{n_1 i}(s)$ at $s = 0$, such that

$$r_i(s) - P_{m_1 i}(s)/Q_{n_1 i}(s) = O(s^{k+1}), \qquad i = 1, 2, 3$$

and

$$m_1 + n_1 = k. \tag{35}$$

Second, compute the Padé approximation $P_{m_2 i}(s)/Q_{n_2 i}(s)$ at $s = \beta$ to the function $\tilde{r}_i(s) = (r_i(s)\, Q_{n_1 i}(s) - P_{m_1 i}(s))/s^{k+1}$ such that

$$\tilde{r}_i(s) - P_{m_2 i}(s)/Q_{n_2 i}(s) = O((s-\beta)^{k+1}), \qquad i = 1, 2, 3$$

and

$$m_2 + n_2 = k \tag{36}$$

The required two point approximation is

$$R_i(s) = \frac{P_{m_1 i}(s) Q_{n_2 i}(s) - s^{k+1} P_{m_2 i}(s)}{Q_{n_1 i}(s) Q_{n_2 i}(s)}$$

which is $(\max\{m_1 + n_2, k + m_2 + 1\},\ n_1 + n_2)$ type rational function and satisfies condition (32). For example, if $k = 3$, take $m_1 = m_2 = 2$, $n_1 = n_2 = 1$, then $R_i(s)$ is a (28) type rational function. Since the denominator of $R_i(s)$ is a product of two polynomials, it is easy to check the appearance of the poles of $R_i(s)$ in $[0, \beta]$ when n_i is small, say $n_i \leq 2$.

Denote $Q_{ni}(s) = Q_{n_1 i}(s) Q_{n_2 i}(s)$ $(n = n_1 + n_2)$, the error can be estimated as in the rational Hermite interpolation case.

D. Two Point Vector Padé Approximation

Similar to the rational vector Hermite interpolation, we can also consider a two point vector Padé approximation. Now conditions (35) and (36) should be replaced by

$$m_1 + n_1/3 = k, \qquad m_2 + n_2/3 = k$$

respectively, and further we require that n_1 and n_2 are divisible by 3. The error can be computed as before.

1.4 Rational B-spline Representation

To interactively control the shape of the piecewise approximating curve or to interface to existing B-spline modelers, we represent each of the rational functions as rational B-splines. The first step is to transform the rational function into Bernstein-Bézier form. Let

$$r(s) = [x(s),\ y(s),\ z(s)]^T / w(s)$$

be a space curve on the interval $[a, b]$, where $x(s), y(s), z(s)$ and $w(s)$ are polynomials of degree n. Since

$$t^i = \sum_{j=i}^{n} \frac{C_i^j}{C_i^n} B_j^n(t)$$

with

$$t = \frac{s-a}{b-a} \in [0,1], \quad B_j^n(t) = C_i^n t^j (1-t)^{n-j}, \quad C_i^n = \frac{n!}{i!(n-i)!}$$

we have, for any polynomial $p(s)$ of degree n

$$\begin{aligned} p(s) &= \textstyle\sum_{i=0}^{n} c_i t^i \\ &= \textstyle\sum_{i=0}^{n} (\sum_{j=0}^{i} \frac{C_j^i}{C_j^n} c_j) B_i^n(t) \\ &= \textstyle\sum_{i=0}^{n} b_i' B_i^n(t) \end{aligned}$$

where $b_i' = \sum_{j=0}^{i} \frac{C_j^i}{C_j^n} c_j$. Therefore $r(s)$ can be expressed as

$$r(s) = \sum_{i=0}^{n} w_i b_i B_i^n(t) / \sum_{i=0}^{n} w_i B_i^n(t) = \sum_{i=0}^{n} w_i b_i N_i^n(s) / \sum_{i=0}^{n} w_i N_i^n(s)$$

where $w_i \in I\!R,\ b_i \in I\!R^3$ is Bézier point and $N_i^n(s) = B_i^n(t)$.

Let $T = \{t_0, ..., t_n, t_{n+1}, ..., t_{2n+1}\}$, where $t_i = a$ for $i = 0, ..., n$, $t_i = b$ for $i = n+1, ..., 2n+1$. Then it is easy to show that the normalized B-spline over T is $N_i^n(s)$ defined above. Therefore, the Bézier point is also the de Boor point in this special case. For the general B-spline

$$F(s) = \sum_{i=0}^{m} d_i N_i^n(s) \tag{37}$$

over

$$T = \{t_0 = ... = t_n \le t_{n+1} \le ... \le t_{m+1} = ... t_{m+n+1}\}$$

with $m \ge n$ and $t_i < t_{i+n+1}$. Most operations on splines, such as evaluation by the de Boor algorithm and knot insertion, do not need the explicit expression of $N_i^n(s)$ but the knot sequence

T and the de Boor points. So these two sets are enough to represent the B-spline. For example, the evaluation of the B-spline $F(s)$ in (37) goes as follows: For $s \in [t_l, t_{l+1})$,

$$\begin{cases} b_i^0 = d_i, \quad i = 0, 1, ..., m \\ d_i^r = \left(1 - \dfrac{s - t_i}{t_{i+n+1-r} - t_i}\right) d_{i-1}^{r-1} + \dfrac{s - t_i}{t_{i+n+1-r} - t_i} d_i^{r-1} \\ l - n + r \le i \le l, \qquad r = 1, 2, ..., n \\ F(s) = d_l^n \end{cases}$$

and inserting a point t with $t_l \le t < t_{l+1}$ to T, we have the following algorithm for the new de Boor points d_i^*, for $i = 0, 1, ..., m+1$:

$$d_i^* = a_i d_i + (1 - a_i) d_{i-1}$$

where

$$a_i = \begin{cases} 1 & \text{if} \quad i \le l - n \\ \frac{t - t_i}{t_{i+n} - t_i} & \text{if} \quad l - n + 1 \le i \le l \\ 0 & \text{if} \quad l + 1 \le i \end{cases}$$

Standard NURB Representation

Quite often geometric designers and engineers using NURBS (Rational B-splines with non-uniform knot spacing) like to have NURBS in a standard form, where the denominator polynomial has only positive coefficients. This assumption is quite strong, but rids the curve of real poles (roots of the denominator polynomial) and gives the rational B-spline its convex hull property. In this subsection we show how to convert a curve in BB form (or normalized B-spline form) into a finite number of C^∞ standard NURB curve segments. We also show that for a degree d B-spline the number of NURB segments is bounded above by $\frac{n(n-1)}{2}$.

We only need to show the transformation for the denominator polynomial of the rational curve. Given a denominator polynomial $P(t) = \displaystyle\sum_{i=0}^{n} b_i B_i^n(t) \quad t \in [0,1]$ we divide the interval [0,1] into subintervals, say, $0 = t_0 < t_1 < \ldots < t_k = 1$, such that the BB-form of $P(t)$ on each of the subintervals $P(t)|_{[t_i t_{i+1}]} = P_i(t) \to P_i\left(\frac{s - t_i}{t_{i+1} - t_i}\right) = \tilde{P}_i(s) = \sum b_j^i B_i^n(s)$ has positive coefficients. Without loss of generality we assume $P(t) > 0$ over [0,1], as this can be achieved for any polynomial by a simple translation. First we show how to compute the first breakpoint $t_1 = c$. By the subdivision formula $B_i^n(ct) = \sum_{j=0}^n \ B_i^j(c) B_j^n(t)$ We have on [0,c], $(s = ct; \quad t \in [0,1])$

$$\begin{aligned} P(s) = P(ct) &= \textstyle\sum_{i=0}^n \ b_i B_i^n(ct) \\ &= \textstyle\sum_{i=0}^n b_i \sum_{j=0}^n \ B_i^j(c) B_j^n(t) \\ &= \textstyle\sum_{j=0}^n \underbrace{\left(\displaystyle\sum_{i=0}^{n} b_i B_i^j(c) \right)} B_j^n(t) \qquad (B_i^j = 0 \quad if \quad i > j) \\ &= \textstyle\sum_{j=0}^n q_j(c) B_j^n(t) \end{aligned}$$

where $q_j(c) = \sum_{i=0}^j b_i B_i^j(c)$ is a degree j polynomial in BB form.

Note that the $\lim_{c \to 0} \ q_j(c) = b_0$. This is because $B_0^j(0) = 1$, $B_i^j(0) = 0$, $i > 0$. Therefore if we assume $P(t) > 0$ for $t \in [0,1]$ then $p(0) = b_0 > 0$. Hence find a root of $q_j(c)$ in [0,1] and take $c < \min\{\text{all roots of } \mathrm{q_j(c)} \text{ in } [0,1]\}$. This c will guarantee **all** $q_j(c)$ are positive. The number of roots of all $q_j(c)$ is bounded by $\frac{n(n-1)}{2}$ which is then also a bound on the number of subintervals required.

The initial Bézier or de Boor coefficients over [0,1] are

$$bb[0] = 1.000000 \quad bb[1] = -0.200000 \quad bb[2] = 0.200000$$
$$bb[3] = 0.100000 \quad bb[4] = -0.200000 \quad bb[5] = 0.500000$$

of which two coefficients are negative. The above conversion yields two pieces in standard NURB over [0,1] with 0.640072 as the breakpoint. The new coefficients of the two NURB pieces are

$$bb[0] = 1.000000 \quad bb[1] = 0.231913 \quad bb[2] = 0.119335$$
$$bb[3] = 0.111575 \quad bb[4] = 0.060781 \quad bb[5] = 0.060781$$

and

$$bb[0] = 0.060781 \quad bb[1] = 0.060781 \quad bb[2] = 0.076842$$
$$bb[3] = 0.125649 \quad bb[4] = 0.248051 \quad bb[5] = 0.500000$$

1.5 Isolating the Singular Points

During the tracing of an intersection space curve, one may encounter singular points. Near these points, the coefficient matrix of the systems (3.8) for IIS, (62) and (63) for IPS, are nearly singular. When a near singular condition of the coefficient matrix is detected, the tracing procedure is temporarily suspended and the singular point is accurated isolated as follows.

A. Singular Point of IIS

Let $p_0 = (x_0, y_0, z_0)^T \in I\!R^3$ be a singular point of the intersection curve of $f_i(p) = 0$, $i = 1, 2$. That is $f_i(p_0) = 0$, $i = 1, 2$ and

$$\alpha_1 F_1^{(1)}(p - p_0) = \alpha_2 F_2^{(1)}(p - p_0) \tag{38}$$

where α_1, α_2 are constants, $|\alpha_1| + |\alpha_1| \neq 0$, $f_i(p) = \sum_{s=0} F_i^{(s)}(p - p_0)$ and $F_i^{(s)}(u, v, w)$ is a homogeneous polynomial of degree s. If the order of the singularity is greater than one, then equation (38) is replaced by

$$\alpha_1 F_1^{(s)}(p - p_0) = \alpha_2 F_2^{(s)}(p - p_0), \quad s = 1, 2, \ldots, L$$

or equivalently

$$\alpha_1 \frac{\partial^s f_1(p_0)}{\partial x^i \partial y^j \partial z^k} = \alpha_2 \frac{\partial^s f_2(p_0)}{\partial x^i \partial y^j \partial z^k}, \qquad \forall(i, j, k) \tag{39}$$

with $i + j + k = s, \qquad s = 1, 2, \ldots, L$.

In order to eliminate α_1 and α_2, use one equation, of (39 $\alpha_2 \dfrac{\partial f_2(p_0)}{\partial x} = \alpha_1 \dfrac{\partial f_1(p_0)}{\partial x}$ to obtain

$$f_{i,j,k}(p_0) = \frac{\partial f_2(p_0)}{\partial x} \frac{\partial^s f_1(p_0)}{\partial x^i \partial y^j \partial z^k} - \frac{\partial f_1(p_0)}{\partial x} \frac{\partial^2 f_2(p_0)}{\partial x^i \partial y^j \partial z^k} = 0$$

for $\forall(i, j, k) \in \{(i, j, k) : i + j + k = s,\ s = 1, 2, \ldots, L\} \setminus \{1, 0, 0\}$.

Now use Newton iterations (Section 6) to solve the system of equations

$$\begin{cases} f_i(p) & = & 0 \\ f_{i,j,k}(p) & = & 0, \qquad i + j + k \leq s \end{cases} \tag{40}$$

Use $s = 1$ if the resulted matrix is nonsingular, otherwise increase s by 1 until the matrix is nonsingular.

B. Singular Points of IPS

Let $Q_1^*, Q_2^* \in I\!R^2$ be the points such that $X_1(Q_1^*) = X_2(Q_2^*)$, i.e., $p^* = X_1(Q_1^*)$ on the intersection curve. We use the definition of the singularity for IIS curve to define the singularity for an IPS curve. For this we need to determine the partial derivatives of parametric surfaces, as described below. We exhibit this for for surface X_1. Surface X_2 can be treated in the same way.

Suppose $\dfrac{\partial X_1(Q_1^*)}{\partial u_1}$ and $\dfrac{\partial X_1(Q_1^*)}{\partial v_1}$ are linearly independent. For smooth, parametric surfaces with faithful parameterizations, the Jacobian matrix

$$J(G_{11}, G_{21}) = \begin{bmatrix} \frac{\partial G_{11}(Q_1^*)}{\partial u_1} & \frac{\partial G_{11}(Q_1^*)}{\partial v_1} \\ \\ \frac{\partial G_{21}(Q_1^*)}{\partial u_1} & \frac{\partial G_{21}(Q_1^*)}{\partial v_1} \end{bmatrix}$$

is nonsingular and invertible. The inverse functions of

$$x = G_{11}(u_1, v_1), \qquad y = G_{21}(u_1, v_1) \tag{41}$$

also exist and are given by

$$u_1 = \tilde{G}_{11}(x, y) \qquad v_1 = \tilde{G}_{21}(x, y) \tag{42}$$

around Q_1^*. Substitute (42) into $z = G_{31}(u_1, v_1)$, to obtain an implicit representation of the parametric surface.

$$f_1(x, y, z) = G_{31}(\tilde{G}_{11}(x, y), \quad \tilde{G}_{21}(x, y)) - z = 0 \tag{43}$$

Now compute the partial derivatives of f_1. The derivative about z is trivial, so consider $\frac{\partial f_1}{\partial x}$ first. It follows from (43) and (41) that

$$\frac{\partial f_1}{\partial x} = \frac{\partial G_{31}}{\partial u_1} \frac{\partial u_1}{\partial x} + \frac{\partial G_{31}}{\partial v_1} \frac{\partial v_1}{\partial x} \tag{44}$$

and

$$J(G_{11}, G_{21}) \begin{bmatrix} \frac{\partial u_1}{\partial x} \\ \frac{\partial v_1}{\partial x} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{45}$$

Solving (45), we get $\frac{\partial u_1}{\partial x}$, $\frac{\partial v_1}{\partial x}$, from (44) we get $\frac{\partial f_1}{\partial x}$. Similarly, $\frac{\partial f_1}{\partial y}$ can be computed.

Knowing the partials one can compute the singular points as in in the IIS case. For higher order singularities the higher order partial derivatives can be computed similar to the computation of second order derivatives shown below.

From (44), we have

$$\begin{aligned} \frac{\partial^2 f_1}{\partial x \partial y} &= \left(\frac{\partial^2 G_{31}}{\partial u_1^2} \frac{\partial u_1}{\partial y} + \frac{\partial^2 G_{31}}{\partial u_1 \partial v_1} \frac{\partial v_1}{\partial y} \right) \frac{\partial u_1}{\partial x} + \frac{\partial G_{31}}{\partial u_1} \frac{\partial^2 u_1}{\partial x \partial y} \\ &+ \left(\frac{\partial^2 G_{31}}{\partial u_1 \partial v_1} \frac{\partial u_1}{\partial y} + \frac{\partial^2 G_{31}}{\partial v_1^2} \frac{\partial v_1}{\partial y} \right) \frac{\partial v_1}{\partial x} + \frac{\partial G_{31}}{\partial v_1} \frac{\partial^2 v_1}{\partial x \partial y} \end{aligned} \tag{46}$$

and from (41), we have

$$J(G_{11} G_{21}) \begin{bmatrix} \frac{\partial^2 u_1}{\partial x \partial y} \\ \\ \frac{\partial^2 v_1}{\partial x \partial y} \end{bmatrix} = - \begin{bmatrix} v_1 \\ \\ v_2 \end{bmatrix} \tag{47}$$

where

$$v_i = \left(\frac{\partial^2 G_{i1}}{\partial u_1^2} \frac{\partial u_1}{\partial y} + \frac{\partial^2 G_{i1}}{\partial u_1 \partial v_1} \frac{\partial v_1}{\partial y} \right) \frac{\partial u_1}{\partial x} + \left(\frac{\partial^2 G_{i1}}{\partial u_1 \partial v_1} \frac{\partial u_1}{\partial y} + \frac{\partial^2 G_{i1}}{\partial v_1^2} \frac{\partial v_1}{\partial y} \right) \frac{\partial v_1}{\partial x}$$

From (47) we get $\frac{\partial^2 u_1}{\partial x \partial y}$, $\frac{\partial^2 v_1}{\partial x \partial y}$, from (46) we get $\frac{\partial^2 f_1}{\partial x \partial y}$.

1.6 The Local Approximation at Singular Points

At the singular points, simple Taylor series expansions fail and we must use special methods to tackle the approximation problem.

1. IIS.

   Let $p_0 = (x_0, y_0, z_0)^T \in I\!R^3$ be a singular point on the curve. since the matrix $\nabla f_i(p_0) \neq 0$, we may assume, WLG, that $\frac{\partial f_1}{\partial z} \neq 0$. Then we can express z by a power series $z = \phi(x, y)$ in x and y from $f_1(x, y, z) = 0$ around the point p_0. Substitute z into $f_2(x, y, z) = 0$, we get $h(x, y) = f(x, y, \phi(x, y)) = 0$ As in the plane curve case, expanding $h(x, y) = 0$ at point $(x_0, y_0)^T$ by Weierstrass and Newton factorization, we obtain

   $$\left\{ \begin{array}{lcl} x & = & x_0 + t^{k_i} \\ y & = & \psi_i(t) \end{array} \right. \qquad i = 0, 1, \ldots, m$$

   where $\psi_i(t)$ is a power series in t and m is the number of the branches of the curve $h(x, y) = 0$. We then have

   $$\begin{array}{lcl} z & = & \phi(x_0 + t^{k_i}, \psi_i(t)) \\ & = & \theta_i(t), \end{array} \qquad i = 0, 1, \ldots, m$$

   Therefore we get the local parametric form of the space curve as

   $$\left\{ \begin{array}{lcl} x & = & x_0 + t^{k_i} \\ y & = & \psi_i(t) \\ z & = & \theta_i(t) \end{array} \right. \qquad i = 0, 1, \ldots, m$$

   For each branch, use the two point interpolating condition to get a rational approximation.

2. IPS

   Let $Q_1^* = (u_1^*, v_1^*)^T$, $Q_2^* = (u_2^*, v_2^*)$ be the points in $I\!R^2$ such that $X_1(Q_1^*) = X_2(Q_2^*)$ and $X_1(Q_1^*)$ is a singular point of the curve IPS. Since the matrices $\nabla X_1(Q_1^*)$ and $\nabla X_2(Q_2^*)$ are full rank in column, we may assume $J(G_{11}, G_{21})$ is not singular at Q_1^*. By one of the first two equations, say the first, $G_{11}(u_1, v_1) = G_{12}(u_2, v_2)$, we can express u_1 as

   $$u_1 = \phi^{(1)}(v_1, u_2, v_2) \tag{48}$$

   Substituting it into another equation of the first two, we get

   $$v_1 = \phi^{(2)}(u_2, v_2) \tag{49}$$

   Substituting u_1 and then v_1 into the last equation $G_{31}(u_1, v_1) = G_{32}(u_2, v_2)$, we have $\phi^{(3)}(u_2, v_2) = 0$. Now, use plane curve factorization techniques for dealing with the singularities, we get

   $$\begin{array}{lcl} u_2 & = & u_2^* + t^{k_i} \\ v_2 & = & \phi_i(t), \end{array} \qquad i = 0, 1, \ldots, m$$

Substitute then back to (49) and (37), we have

$$\begin{aligned} v_1 &= \phi^{(2)}(u_2^* + t^{k_i}, \phi_i(t)) \\ &= \psi_i(t) \end{aligned}$$

$$\begin{aligned} u_1 &= \phi^{(1)}(\psi_i(t), u_2^* + t^{k_i}, \phi_i(t)) \\ &= \theta_i(t) \end{aligned}$$

Then the local parameterization is obtained by

$$\begin{aligned} r_i(t) &= X_1(\theta_i(t), \psi_i(t)) \\ &\text{or } \; X_2(u_2^* + t^{k_i}, \phi_i(t)), \qquad i = 0, 1, \ldots, m \end{aligned}$$

The next step for getting approximation is the same as IIS.

Implementation Details

1. Starting Points

   In order to trace the intersection curve SC, we need to provide a starting point on each real component of the curve. Besides the boundary points which are straightforward roots of univariate or coupled bivariate polynomial equations one computes a starting point on each real component completely inside the given box. For IIS this can be done by projecting the intersection curve (via resultant elimination) into a plane and then finding a coordinate axis extreme point on the projection curve of that component. See [4] for details of such resultant elimination schemes. For IPS, papers [10] [20] provide some numerical methods for computing these starting points.

2. Curve Interpolations Points

   When we march along the curve, we encounter precomputed points on the way. An encountered point may be a boundary point, a starting point on a closed loop or may be an end point of the prior segment tracing. Suppose $p_0 \in I\!R^3$ is a point on the curve, $r(s)$ $(s \in [0, \beta])$ is a segment of the curve, which approximates the original curve. Then a possible question is whether $r(s)$ passes through p_0 within the allowable error? We answer this question by computing the distance between p_0 and $r(s)$:

$$\text{dis}(p_0, r) = \min_{s \in [0,\beta]} \|r(s) - p_0\| \tag{50}$$

   Since $r(s)$ is a rational function in s, the minimum point of (50) can be computed by $\frac{d}{ds}\|r(s) - p_0\|^2 = 0$. If $s = s^* \in [0, \beta]$ is the minimum point, then if $\|r(s^*) - p_0\| < \epsilon$, $r(s)$ passes through p_0. Then we modify $r(s)$ such that $r(s)$ is frame continuous at s^* and replaces β by s^*. Otherwise, $r(s)$ does not pass through p_0.

3. Solving Linear System of Equations

   In all the cases we always solve the linear system $Ax = b$ with a positive definite coefficient matrix A. The size of matrix A is as small as one, and as large as four. A stable method to solve this equation is to use singular value decomposition $A = U^T \Sigma U$, where U is an orthogonal matrix and Σ is a diagonal matrix. The solution is $x = U^T \Sigma^{-1} U b$.

4. Tangent Direction

   In Sections 4 and 5, we have mentioned that the sign of the tangent vector t at an expansion point should be properly chosen. Now we will make this point clear.

a. If the expansion point is a boundary point, then t points to the interior of the box.

b. If the point is a starting point on a loop, then the sign can be any.

c. If the point is an end point of a previous approximation $\tilde{r}(s) = \sum_{i=0}^{k+1} r^{(i)}(0)s^i/i!$ ($s \in [0, \beta]$), then we choose the sign of t such that $\tilde{r}'(\beta)^T t \geq 0$.

2 Space Curves

2.1 Piecewise Parameterization of Surface-Surface Intersection Curves

Given a real intersection space curve SC which is either

(a). the intersection of implicit surfaces (IIS) defined by $f_1(x, y, z) = 0, \qquad f_2(x, y, z) = 0$, and within a bounding box $B = \{(x, y, x) : x_0 \leq x \leq x_1, \quad y_0 \leq y \leq y_1, \quad z_0 \leq z \leq z_1\}$

(b). the intersection of parametric surfaces(IPC) defined by

$$\begin{aligned} X_1(u_1, v_1) &= [G_{11}(u_1, v_1)\ G_{21}(u_1, v_1),\ G_{31}(u, v_1)]^T \\ X_2(u_2, v_2) &= [G_{12}(u_2, v_2)\ G_{22}(u_2, v_2),\ G_{32}(u_2, v_2)]^T \end{aligned}$$

and within a bounding box

$$B = \{(u_1, v_1, u_2, v_2) \ : \ \begin{array}{l} u_{10} \leq u_1 \leq u_{11}, \quad v_{10} \leq v_1 \leq v_{11} \\ u_{20} \leq u_2 \leq u_{21}, \quad v_{20} \leq v_2 \leq v_{21}\} \end{array}$$

and an error bound $\epsilon > 0$, a continuity index k, construct a C^k (or G^k) continuous piecewise parametric rational ϵ-approximation of all portions of SC within the given bounding box B.

The Outline of the Algorithm

The approximation process is a tracing procedure along the curve. It consists of the following steps:

1. Form a starting point list (SPL) by computing the boundary points containing the intersection points of the curve SC and the bounding box B. Further SPL is made to contain at least one point for each inner loop component of SC i.e. a curve loop completely inside the given box B. Tracing direction are also provided at each of these points in SPL. (See section 11 for implementation details).

2. Test if SPL is empty. If yes, the tracing is finished. Otherwise, starting from a point p in SPL, trace the curve along the given direction until either of the following tests in step 3 or step 4 are true. The tracing step consists of the following sub-steps:

   (a). Compute an arc length based power series expansion (see sections 4 and 5) up to $k+1$ terms at the given point p.

   (b). Determine a step-length and a point $\hat{q}$ on the above expansion curve in the tracing direction within a step-length of p. and then starting from $\hat{q}$ refine to a new point q on the curve SC by Newton iterations (see section 6).

   (c). Compute an arc length power series expansion up to $k+1$ terms at the new point q.

(d). Construct an approximating rational parametric curve segment by C^k Hermite interpolation (see section 7) of the two end points p and q and convert it into a rational B-spline or standard NURBS with positive denominator polynomial (see section 8).

(e). Add the rational curve approximant into the piecewise approximation list and return to step 2, to continue the tracing from the the newly constructed point q.

3. Test if a singular point is met. If yes, stop the present tracing and put the end point of the tracing into SPL (we may delete a few approximation segments from the present approximation list, because the step length near a singular point is small). Then locate the singular point(see section 9), obtain a finite set of power series expansion at the singular point corresponding to the distinct curve branches (see section 10). Trace each branch one or two steps, and then put the end points of the tracing into SPL. Then return to Step 2.

4. Test if another point in SPL is met(see §12). If yes, we have stitched together one continuous segment of the curve. Delete the two end points of the traced segment from SPL and return to Step 2.

2.2 Notation for Space Curves

We will express a space curve as a power series, locally at point and with its arc length as a parameter. We refer to [12] for some intrinsic parameters of space curves.

Let $r(s) = [x(s), y(s), z(s)]^T$ be a space curve, where s is arc length of the curve measured from some fixed point. The tangent vector $t(s) = r'(s)$ has unit length; $k(s) = \|r''(s)\|$ is the curvature, where $\|\cdot\|$ is the Euclidean norm in $\mathbb{R}^3$. Further, $n(s) = r''(s)/k(s)$ is the principle normal; $b(s) = t(s) \times n(s)$ is the binormal, where $\times$ denotes the cross product of two vectors. Finally, the number $T(s)$ defined by $b'(s) = -T(s)n(s)$ is the torsion. The three orthogonal vectors $t(s), n(s)$ and $b(s)$ form the so called Frenet frame. These vectors are related by the following Frenet formulas

$$t' = kn, \qquad b' = -Tn, \qquad n' = -kt + Tb$$

The derivatives of $r(s)$ are therefore given by

$$r'(s) = t, \qquad r''(s) = kn, \qquad r'''(s) = k'n + kTb - k^2 t \tag{51}$$

Since $t = r'(s)$, $k = \|r''(s)\|$ and $T = r'(s) \times r''(s) \cdot r'''(s)/\|r''(s)\|^2$ then the curve is obviously tangent, or curvature or torsion continuous if $r'(s)$, or $r'(s)$ and $r''(s)$, or $r'(s)$, $r''(s)$ and $r'''(s)$ is continuous respectively. We construct a piecewise approximation of the given curve such that the composite curve is tangent ($t(s)$), normal ($n(s)$) and binormal ($b(s)$) continuous.

Among the various local parameterizations of the space curve, taking arc length as parameter has several advantages.

A. If $r(s)$ is the parametrization of the given curve and s is arc length start from some point, the $r'(s)$, $r''(s)$, $r'''(s)$ is equivalent to $t(s)$, $n(s)$, $b(s)$ in the sense that the continuity of $r'(s)$, $r''(s)$, $r'''(s)$ are equivalent to the continuity of $t(s)$, $n(s)$, $b(s)$ where the triple $t(s)$, $n(s)$, $b(s)$ is the Frenet frame. Therefore, we need only to force the composite curve's first three derivatives to be continuous at the break points without considering the connection matrix as in the case of geometric continuity.

B. Since the arc length of the curve is independent of any coordinate system, then the expansion of power series may have larger convergence radius. This will, in turn, lead to less segments of approximation.

C. In geometry point of view, the frame of Frenet continuous is the most natural and useful requirement. It keeps the tangent, principle normal and binormal varying continuously, while other geometric continuity can not achieve this conclusion.

2.3 Local Expansion of the Intersection Curve of Implicit Surfaces

Let $f_1(p), f_2(p)$ be two algebraic polynomials with $p = [x, y, z]^T \in I\!R^3$. The intersection of implicit defined the surfaces(IIS) $f_1(p_1) = 0$, $f_2(p_1) = 0$ is defined by $f_1(p) = f_2(p) = 0$. We assume the defining surfaces are smooth, i.e., the normals of the surfaces are not equal to zero at any point on the surface. Now let $F(p) = [f_1(p), f_2(p)]^T$, $p_0 \in I\!R^3$ be a point on the intersection curve $r(s)$, where s is the arc length measured from $p_0 = r(0)$ with prescribed direction. Then, $r'(0), r''(0)$ and $r'''(0)$ are computed as follows:

$$F(r)(s) = F(r)(0) + s\frac{dF(r)(0)}{ds} + \frac{s^2}{2!}\,\frac{d^2F(r)(0)}{ds^2} + \ldots \tag{52}$$

where

$$\frac{d^kF(r)(0)}{ds^k} = V_k(0) + \nabla F(p_0)r^{(k)}(0) \tag{53}$$

$$\begin{aligned} V_1(s) &= 0 \\ V_k(s) &= V'_{k-1}(s) + [\nabla F(r)]' r^{(k-1)}(s), \quad k = 1, 2, \ldots \\ \nabla F(p) &= \left[\ \tfrac{\partial F(p)}{\partial x},\ \tfrac{\partial F(p)}{\partial y},\ \tfrac{\partial F(p)}{\partial z}\ \right] \ \in I\!R^{2\times 3} \end{aligned} \tag{54}$$

It follows from $F(r(s)) \equiv 0$ that

$$\nabla F(p_0)r^{(k)}(0) = -V_k(0) \tag{55}$$

The system of equation (55) has three unknowns and two equations. It has in general infinite many solutions. Now we assume $\nabla f_1(p_0)$ and $\nabla f_2(p_0)$ are linearly independent and illustrate how to get $r^{(k)}(0)$ such that the equations in the last section are satisfied.

Let t be a vector such that

$$t \in \nabla F(p_0)^{\perp}, \qquad \|t\| = 1 \tag{56}$$

and its sign is so chosen that t gives the correct direction along the same line we are going. Then for any vector $x \in I\!R^3$ there exist unique $\alpha \in I\!R$ and $y \in \text{range}(\nabla F(p_0)^T)$, such that $x = \alpha t + y$. Let

$$r^{(k)}(0) = \alpha_m t + \nabla F(p_0)^T \beta_m \tag{57}$$

Then by (55), we have β_m is uniquely defined by

$$\nabla F(p_0)\nabla F(p_0)^T \beta_m = -V_k(0) \tag{58}$$

and α_m is arbitrary. Now we determine α_m $(m = 1, \ldots, 4)$, such that $r^{(m)}(0)$ $(m = 1, \ldots, 4)$ satisfy (51).

A. m = 1. Since $V_1(0) = 0$, then $\beta_1 = 0$. Hence $r'(0) = \alpha_1 t$. According to the definition of t, we choose $\alpha_1 = 1$.

B. m = 2. Since we want the $r''(0)$ orthogonal to $r'(0) = t$, i.e., $r''(0) \in \text{range}(\nabla F(p_0)^T)$, the only choice is $\alpha_2 = 0$. We then have $k = \|r''(0)\|$.

C. m = 3. It follows from (51) that $k'n + kTb \in \text{range}(\nabla F(p_0)^T)$. Then $\alpha_3 = -k^2$, and further $k' = r''(0)^T r'''(0)/k$

D. m = 4. From (51)

$$r^{(4)}(s) = (k'' - kT^2 - k^3)n + [k'T + (kT)']b - 3kk't.$$

Then $\alpha_4 = -3kk'$.

Finally we obtain the approximate expansion $r(s) \approx \sum_{i=0}^{4}(r^{(i)}(0)/i!)s^i$

2.4 Local Expansion of the Intersection Curve of Parametric Surfaces

Let

$$\begin{aligned} X_1(u_1, v_1) &= [G_{11}(u_1, v_1)\ G_{21}(u_1, v_1),\ G_{31}(u, v_1)]^T \\ X_2(u_2, v_2) &= [G_{12}(u_2, v_2)\ G_{22}(u_2, v_2),\ G_{32}(u_2, v_2)]^T \end{aligned}$$

be two parametric surface, where G_{ij} are given smooth functions. The intersection curve of the parametric surface (IPS) is defined by

$$r(s) = X_1(u_1(s), v_1(s)) \qquad (\text{or } X_2(u_2(s), v_2(s)))$$

with $X_1(u_1(s), v_1(s)) = X_2(u_2(s), v_2(s))$ where the parameter s is the arc length measured from some point on the curve.

Let $Q_1 = (u_1, v_1)^T$, $Q_2 = (u_2, v_2)^T$, and Q_1^*, Q_2^* be the points in $I\!R^2$ such that $X_1(Q_1^*) = X_2(Q_2^*)$. At point $X_1(Q_1^*)$, we want to expand $r(s)$ into power series $r(s) = r(0) + r'(0)s + \frac{r''(0)}{2}s^2 + \ldots$ On the curve $r(s)$, Q_1 and Q_2 are functions of s, we can express them as

$$Q_j(s) = \sum_{i=0}^{\infty} \frac{Q_j^{(i)}(0)}{i!}\ s^i, \qquad j = 1, 2 \tag{59}$$

As the case of IIS, we expand $X_j(Q_j(s))$

$$X_j(Q_j(s)) = X_j(Q_j)(0) + \frac{dX_j(Q_j)(0)}{ds}\ s + \frac{d^2X_j(Q_j)(0)}{ds^2}\ \frac{s^2}{2} + \ldots$$

for $j = 1, 2$,

$$\frac{d^kX_j(Q_j)(0)}{ds^k} = V_{kj}(0) + \nabla X_j(Q_j^*)Q_j^{(k)}(0), \qquad j = 1, 2 \tag{60}$$

where

$$\begin{aligned} V_{1j}(s) &= 0, \qquad j = 1, 2 \\ V_{kj}(s) &= \tfrac{d}{ds}V_{k-1,j}(s) + \tfrac{d}{ds}[\nabla X_j(Q_j)]Q_j^{(k-1)}(s). \end{aligned}$$

By $X_1(Q_1(s)) \equiv X_2(Q_2(s))$, we have

$$\nabla X_1 Q_1^{(m)} - \nabla X_2 Q_2^{(m)}(0) = V_{m1}(0) - V_{m2}(0).$$

Let

$$n_i = X_{iu_i} \times X_{iv_i}, \qquad X_{iu_i} = \left[\frac{\partial G_{1i}}{\partial u_i},\ \frac{\partial G_{2i}}{\partial u_i},\ \frac{\partial G_{3i}}{\partial u_i}\right]$$

Then n_1, n_2 are the normals of the two surfaces. Suppose n_1 and n_2 are linearly independent. Let $t \in I\!R^3$ such that $t \in [n_1 n_2]^\perp, \quad \|t\| = 1$, and its sign is properly chosen such that it points to the correct direction. Then we have the expression

$$\begin{aligned} \nabla X_1 Q_1^{(m)}(0) + V_{m1}(0) &= \nabla X_2 Q_2^{(m)}(0) + V_{m2}(0) \\ &= \alpha_m t + [n_1, n_2]\beta_m \end{aligned} \tag{61}$$

Since $n_1^T \nabla X_1 = 0, \quad n_2^T \nabla X_2 = 0$, we have from (61)

$$[n_1, n_2]^T\ [n_1, n_2]\beta_m = \begin{bmatrix} n_1^T V_{m1} \\ n_2^T V_{m2} \end{bmatrix}. \tag{62}$$

Therefore β_m is uniquely determined by the nonsingularity of the matrix $[n_1, n_2]^T[n_1, n_2]$, and α_m is arbitrary. From (61), we determine α_m, such that $r^{(m)}(0) = \alpha_m t + [n_1, n_2]\beta_m$. This can be done exactly the same as the case of IIS by regarding $[n_1, n_2]$ as $\nabla F(p_0)^T$.

After $r^{(m)}(0)$ are received, we can compute $Q_j^{(m)}(0)$, $j = 1, 2$. From (61),

$$\nabla X_j^T \nabla X_j Q_j^{(m)}(0) = \nabla X_j^T (r^{(m)}(0) - V_{mj}(0)), \qquad j = 1, 2. \tag{63}$$

Solving these equations, we get $Q_j^{(m)}(0)$.

The purpose of computing $Q_j^{(m)}(0)$ is to compute the approximate value of $Q_j(s)$ by (59). This approximate value serves as the initial value for the Newton method to get accurate value Q_j on the curve.

3 Surfaces

3.1 Blending by Hermite Interpolation / Approximation

3.1.1 Hermite Interpolation using Real Algebraic Surfaces

The Problem: Construct a real algebraic surface S, which *smoothly* interpolates a collection of k points $\mathbf{p}_i$ in $\mathbb{R}^3$ with associated fixed "normal" unit vectors $\mathbf{m}_i$, and l given space curves C_j in $\mathbb{R}^3$ also with associated "normal" unit vectors $\mathbf{n}_j$, varying along the entire span of the curves, $(i = 1 \ldots k,\ j = 1 \ldots l)$. Both points and space curves have an infinity of potential "normal" vector directions. While for points the $\mathbf{m}_i$ may be chosen arbitrarily, for space curves C_j, the varying unit vectors $\mathbf{n}_j$ are chosen to be always orthogonal to the tangent vector $\mathbf{t}_j$, that is, $\mathbf{t}_j \cdot \mathbf{n}_j = 0$, along the entire curve. Our emphasis being algebraic space curves, the variance of the curves "normals" are restricted to univariate polynomials of some degree. Also, we assume that any of the vectors $\mathbf{m}_i$ and $\mathbf{n}_j$ are never identically zero, a phenomenon that occurs at point and curve singularities. By *smoothly* interpolates we shall mean that S contains each of the points and curves and furthermore has its gradient in the same direction as the "normal" vectors $\mathbf{m}_i$ and $\mathbf{n}_j$. This is a natural generalization of Hermite interpolation, applied to fitting curves through point data, and equating derivatives at those points. As we shall see later, the choice of the associated "normal" direction, in each case is dictated by the use of the Hermite interpolated surface, (eg, in "blending" or "joining" or "fleshing").
Related Work: Sarraga in [26] presents techniques for constructing a C^1-continuous surface of rectangular Bézier (parametric) surface patches, interpolating a net of cubic Bézier curves. Other approaches to parametric surface fitting and transfinite interpolation are also mentioned in that paper, as well as in [34]. An excellent exposition of exact and least squares fitting of algebraic surfaces through given data points, is presented in [24]. Meshing of given algebraic surface patches using control techniques of joining Bézier polyhedrons is shown in [27]. Surface blending consisting of "rounding" and "filleting" surfaces (smoothing the intersection of two primary surfaces), a special case of Hermite interpolation, has been considered for polyhedral models in [13] and for algebraic surfaces in [17, 16, 19, 22, 25, 31, 32, 34].

Results: We show in Sections 3, 4 and 5 that the problem of generalized Hermite interpolation of points and curves with algebraic surfaces, reduces to solving systems of linear equations, albeit at times with symbolic coefficients. In particular for an algebraic surface of degree n, to *smoothly contain* k points and l space curves of degree d with assigned "normal" directions, varying as a polynomial of degree m, the number of linear equations to be satisfied is $3k + (2n + m - 1)dl + 2l$. This number reduces to $3k + (2n - 1)dl + ml + 2l$ when all the space curves and "normals" are represented parametrically. Since the number of independent coefficients (unknowns) of a general algebraic surface of degree n is $\binom{n+3}{3} - 1$, the number of linear equations stated above, yields both necessary and sufficiency conditions on Hermite interpolated algebraic surfaces, for a variety of point and curve data configurations. As applications of this simple vector space characterization of Hermite interpolated algebraic surfaces, we show, in section *5.*, for example, that:

- Two space lines with constant-direction normals can be Hermite interpolated with a real quadric if and only if the lines are parallel or intersect at a point, and the normals are not orthogonal to the plane containing them. The real quadric is a "cylinder" when the lines are parallel and a "cone" when the lines intersect.

- Two skewed lines with constant-direction normals cannot be Hermite interpolated with real quadrics. The only real quadratic surface which satisfies both containment and tangency conditions reduces into two planes.

- The minimum degree of a real algebraic surface, which Hermite interpolates two lines in space, one with a constant direction normal, the other with a linearly varying normal is three.

- Two lines with linearly varying normals can be Hermite interplated by a quadric in only some special cases. In general, a surface of at least degree three is needed. When real quadric surface interpolation is possible, the real quadric is either a hyperboloid of one sheet (the two lines may be parallel, intersecting, or skewed) or a hyperbolic paraboloid (the two lines can only be intersecting or skewed).

Lines in space with constant-direction normals, occur naturally as edges of polyhedra, with the Hermite interpolating surfaces being used to "smooth" planar faces containing those edges. Lines with linearly-varying normals occur on real quadric and cubic surfaces. Similar results to the ones above, are also derived in sections*5.* and *6.*, for Hermite interpolation of conics and cubics in space. Since these rational curves lie on quadrics, cubic surfaces and higher degree algebraic surfaces, our method gives a powerful way of automatically, generating low degree "blending" and "joining" and "fleshing" surfaces with tangent continuity at intersections.

Preliminaries For any multivariate polynomial f, partial derivatives are written by subscripting, for example, $f_x = \partial f/\partial x$, $f_{xy} = \partial^2 f/(\partial x \partial y)$, and so on. Since we consider algebraic curves and surfaces, we have $f_{xy} = f_{yx}$ etc. Vectors and vector functions are denoted by bold letters. The *inner product* of vectors $\mathbf{a}$ and $\mathbf{b}$ is denoted $\mathbf{a} \cdot \mathbf{b}$. The *length* of the vector $\mathbf{a}$ is $\|\mathbf{a}\| = \sqrt{\mathbf{a} \cdot \mathbf{a}}$.

The *gradient* of $f(x, y, z)$ is the vector $\nabla f = (f_x, f_y, f_z)$. A point $p = (x_0, y_0, z_0)$ is a *simple* point of f if the gradient of f at p is not null; otherwise the point is *singular*. An algebraic surface is non-singular or smooth if all its points are simple.

Definition 3.1. *Let $\boldsymbol{p} = (a, b, c)$ be a point with an associated "normal" $\mathbf{m} = (m_x, m_y, m_z)$ in $\mathbb{R}^3$. An algebraic surface $S : f(x, y, z) = 0$ is said to* smoothly contain $\boldsymbol{p}$ *if*
(1) $f(\boldsymbol{p}) = f(a, b, c) = 0$, **(containment condition)**
and
(2) $\nabla f(\mathbf{p})$ *is not zero and* $\nabla f(\mathbf{p}) = \alpha \mathbf{m}$, *for some nonzero* α. **(tangency condition)**

Definition 3.2. *Let C be an algebraic space curve with an associated varying "normal" $\mathbf{n}(x, y, z) = (n_x(x, y, z), n_y(x, y, z), n_z(x, y, z))$, defined for all points on C. An algebraic surface $S : f(x, y, z) = 0$ is said to* smoothly contain C *if*
(1) $f(\boldsymbol{p}) = 0$ *for all points* $\boldsymbol{p}$ *of* C. **(containment condition)**
and
(2) $\nabla f(\mathbf{p})$ *is not identically zero and* $\nabla f(\mathbf{p}) = \alpha \mathbf{n}(\boldsymbol{p})$, *for some nonzero α and for all points $\boldsymbol{p}$ of C.* **(tangency condition)**

Definition 3.3. *An algebraic surface $S : f(x, y, z) = 0$ is said to Hermite interpolate a given collection of data points with associated "normals", and data curves with associated "normals", if* S smoothly contains *all the data points and curves.*

The following is one form of Bezout's theorem (the oldest theorem of algebraic geometry).

Theorem 3.4. *An algebraic curve C of degree d intersects an algebraic surface S of degree n in at most nd points, or else it must intersect it infinitely often, that is, a component of C must lie entirely on S.*

Interpolation of Points

Containment There exist applications in object reconstruction in geometric design, when there is need to construct a surface which interpolates a given set of data points. From the containment condition of definition 3.1 it directly follows that any algebraic surface $S : f(x, y, z) = 0$, whose coefficients satisfy the linear equation $f(\mathbf{p}) = 0$ will contain the point $\mathbf{p}$. For a set of k data points this yields k linear equations. For an algebraic surface of degree n, having $K = \binom{n+3}{3} - 1$ independent coefficients, various types of exact fits can be obtained by choosing the smallest n such that $K \geq r$, where r, $(\leq k)$ is the rank of the system of k linear equations. Details of methods for constructing such real algebraic surfaces can be found in [24].

Containment with Tangency Quite often one also needs a low degree algebraic surface which not only contains a set of data points but is also tangent to a prespecified plane at each of those points. A point $\mathbf{p} = (a, b, c)$ with a "normal" $\mathbf{m} = (m_x, m_y, m_z)$ determines a unique plane $P : m_x x + m_y y + m_z z - (m_x a + m_y b + m_c c) = 0$, at the point $\mathbf{p}$. An algebraic surface $S : f(x, y, z) = 0$ of degree n that Hermite interpolates a point $\mathbf{p}$, can be computed as follows:

1. **(containment condition)** For point $\mathbf{p}$ set up the linear equation $f(\mathbf{p}) = 0$ in the unknown coefficients of S.
2. **(tangency condition)** One of the following cases is selected.
   (a) If $m_x \neq 0$, use the equations $m_x f_y(\mathbf{p}) - m_y f_x(\mathbf{p}) = 0$ and $m_x f_z(\mathbf{p}) - m_z f_x(\mathbf{p}) = 0$.
   (b) If $m_y \neq 0$, use the equations $m_y f_x(\mathbf{p}) - m_x f_y(\mathbf{p}) = 0$ and $m_y f_z(\mathbf{p}) - m_z f_y(\mathbf{p}) = 0$.
   (c) If $m_z \neq 0$, use the equations $m_z f_x(\mathbf{p}) - m_x f_z(\mathbf{p}) = 0$ and $m_y f_z(\mathbf{p}) - m_z f_y(\mathbf{p}) = 0$.
3. We also ensure that the coefficients of $f(x, y, z) = 0$ satisfying the above three linear equations, additionally satisfy the linear constraints $\nabla f(\mathbf{p}) \neq 0$, since non-tangency at $\mathbf{p}$ may occur if S turns out to be singular at $\mathbf{p}$.

The proof of correctness of the above algorithm follows from the following lemma.

Lemma 3.5. *The equations of the above algorithm satisfy defintion 2.1 of point containment and tangency.*

Proof : The first linear equation $f(\mathbf{p}) = 0$ satisfies containment by definition. We now show that the remaining equations satisfy $\nabla f(\mathbf{p}) = \alpha \cdot \mathbf{m}$ for a nonzero α. Since $\mathbf{m}$ is never taken to be the $(0, 0, 0)$ vector, without loss of generality we may assume that $m_x \neq 0$ in step *2.* above. Other cases of $m_y \neq 0$ or $m_z \neq 0$ can be handled symmetrically. Now let $\alpha = \frac{f_x}{m_x}$, assuming $m_x \neq 0$. Then $f_x = \alpha \cdot m_x$ and susbtituting it in the selected linear equation $m_x f_y - m_y f_x = 0$ yields $f_y = \alpha \cdot m_y$ and substituing it again in the other selected linear equation $m_x f_z - m_z f_x = 0$ yields $f_z = \alpha \cdot m_z$. Hence $\nabla f(\mathbf{p}) = \alpha \cdot \mathbf{m}$. Finally, note that $f_x = 0$ for $m_x \neq 0$, in the selected linear equations of step *2 (a).*, would cause $\nabla f(\mathbf{p}) = 0$, which we ensured would not happen in step *3* of the algorithm. Hence $f_x \neq 0$ and so $\alpha \neq 0$ and the lemma is proved. $\square$

Interpolation of Curves The varying "normal" associated with a space curve C can be defined implicitly by the triple $\mathbf{n}(x,y,z) = (n_x(x,y,z), n_y(x,y,z), n_z(x,y,z))$ where n_x, n_y and n_z are polynomials of maximum degree m and defined only for all points $\mathbf{p} = (x,y,z)$ along the curve C. For the special case of a rational curve, as defined earlier, and which we shall treat separately in sections *4.1.2* and *4.2.2*, the varying "normals" can also be defined parametrically as $\mathbf{n}(s) = (n_x(s), n_y(s), n_z(s))$, with n_x, n_y and n_z now rational functions in s.

Containment

Algebraic Curves: Implicit Definition Let $\mathbf{C} : (f_1(x,y,z) = 0, f_2(x,y,z) = 0)$ implicitly define an irreducible algebraic space curve of degree d. The irreducibility of the curve is not really a restriction, since reducible curves can be handled similarly by treating each irreducible component in turn. The situation is slightly more complicated if in the real setting, we may wish to achieve separate containment of each real component of an irreducible curve. We defer a solution to this problem, and for the time being consider it reduced to the problem of choosing appropriate clipping surfaces to isolate that real component, after the interpolated surface is computed. Note for parametrically defined curves, this problem does not arise.

An interpolating surface $S : f(x,y,z) = 0$ of degree n for containment of C, is then computed as follows:

1. Choose a set L_c of $nd+1$ points on $\mathbf{C}$, $L_c = \{\mathbf{p}_i = (x_i, y_i, z_i) | i = 1, \cdots, nd+1\}$[2]. The set L_c may be computed, for example, by tracing the intersection of $f_1 = f_2 = 0$, see for e.g., [6].

2. Next, set up $nd+1$ homogenous linear equations $f(\mathbf{p}_i) = 0$, for $\mathbf{p}_i \epsilon L_c$. Any nontrivial solution of this linear system will represent an algebraic surface which interpolates the entire curve $\mathbf{C}$.

The proof of correctness of the above algorithm is captured in the following Lemma.

Lemma 3.6. *To satisfy the containment condition of an algebraic curve C of degree d by an algebraic surface S of degree n, it suffices to satisfy the containment condition of $nd+1$ points of C by S.*

Proof: This is essentially a restatement of Bezout's theorem of section 2. By making S contain $nd+1$ points of $\mathbf{C}$, ensures that S must intersect C infinitely often and since C is irreducible, S must contain the entire curve. $\square$

Remember $S : f(x,y,z) = 0$ of degree n has $K = \binom{n+3}{3} - 1$ independent coefficient unknowns. Let r be the rank of the system of $nd+1$ linear equations. There are non-trivial solutions to this homogeneous system if and only if $K > r$ and a unique non-trivial solution when $K = r$. Hence, again an interpolating surface can be obtained by choosing the smallest n such that $K \geq r$.

Rational Curves : Parametric Definition When a curve is given in rational parametric form, its equations can be used directly to produce a linear system for interpolation, instead of first computing $nd+1$ points on the curve. Let $\mathbf{C} : (x = G_1(t), y = G_2(t), z = G_3(t))$ be a rational curve of degree d. An interpolating surface $S : f(x,y,z) = 0$ of degree n which contains C is computed as follows:

1. Substitute $(x = G_1(t), y = G_2(t), z = G_3(t))$ into the equation $f(x,y,z) = 0$.

2. Simplify and rationalize to obtain $Q(t) = 0$, where Q is a polynomial in t, of degree at most nd, and with coefficients which are linear expressions in the coefficients of f. For Q to be identically zero, each of its coefficents must be zero, and hence we obtain a system of at most $nd+1$ linear equations, where the unknowns are the coefficients of f. Any non-trivial solution of this linear system will represent a surface S which interpolates $\mathbf{C}$.

[2] Thus, alternatively, an algebraic curve may be given as a list of points.

The proof of correctness of the algorithm follows from the lemma below.

Lemma 3.7. *The containment condition is satisfied by step 2. of the above algorithm*

Proof: We omit this here and refer the reader to the full paper. □

Containment with Tangency In order to Hermite interpolate an algebraic curve C with associated "normals" $\mathbf{n}$ by an algebraic surface S, we need to again solve a homogenous linear system, whose equations stem from both the *containment condition* and the *tangency conditions* of definition 3.2.

Algebraic Curves with Normals: Implicit Definition As before, let $C : (f_1(x, y, z) = 0, f_2(x, y, z) = 0)$ implicitly define an irreducible algebraic space curve of degree d, together with associated "normals" defined implicitly by the triple $\mathbf{n}(x, y, z) = (n_x(x, y, z), n_y(x, y, z), n_z(x, y, z))$ where n_x, n_y and n_z are polynomials of maximum degree m and defined for all points $\mathbf{p} = (x, y, z)$ along the curve C. A Hermite interpolating surface $S : f(x, y, z) = 0$ of degree n which smoothly contains C is then computed as follows:

1. Choose a set L_c of $(n+m-1)d+1$ points on C, $L_c = \{\mathbf{p}_i = (x_i, y_i, z_i) | i = 1, \cdots, (n+m-1)d+1\}$. The set L_c may be computed, as before, by tracing the intersection of $f_1 = f_2 = 0$, see for e.g., [6].
2. Construct the list L_t of $(n+m-1)d+1$ point-normal pairs on C, $L_t = \{[(x_i, y_i, z_i), (n_{xi}, n_{yi}, n_{zi})] | i = 1, \cdots, (n+m-1)d+1\}$[3], where $(n_{xi}, n_{yi}, n_{zi}) = \mathbf{n}(\mathbf{p}_i)$ for $\mathbf{p}_i \epsilon L_c$.
3. **(containment condition)** Next, set up $nd+1$ homogenous linear equations $f(\mathbf{p}_i) = 0$, for $\mathbf{p}_i \epsilon L_c$ and $i = 1, \cdots, nd+1$.
4. **(tangency condition)**
   (a) Compute $\mathbf{t}(x, y, z) = \nabla f_1(x, y, z) \times \nabla f_2(x, y, z)$. Note $\mathbf{t} = (t_x, t_y, t_z)$ is the tangent vector to C.
   (b) One of the following cases is selected.
      i. If $t_x \neq 0$, use the equation $f_y \cdot n_z - n_y \cdot f_z = 0$.
      ii. If $t_y \neq 0$, use the equation $f_x \cdot n_z - n_x \cdot f_z = 0$.
      iii. If $t_z \neq 0$, use the equation $f_x \cdot n_y - n_x \cdot f_y = 0$.

      Substitute each point-normal pair in L_t into the above selected equation to yield additionally $(n+m-1)d+1$ linear equations in the coefficients of the $f(x, y, z)$.
5. In total we obtain a homogeneous system of $(2n+m-1)d+2$ linear equations. Any non-trivial solution of the linear system, for which additionally ∇f is not identically zero for all points of C, (that is, the surface S is nonsingular at all points along the curve C), will represent a surface which Hermite interpolates C.

The proof of correctness of the above algorithm follows from Lemma 3.6 and the following lemma, which shows why the selected equation of step *4.(b)* evaluated at $(n+m-1)d+1$ point-normal pairs, are sufficient.

Lemma 3.8. *To satisfy the tangency condition of an algebraic curve C of degree d with "normal" $\mathbf{n}$ of degree m, by an algebraic surface S of degree n, it suffices to satisfy the tangency condition at $(n+m-1)d+1$ points of C by S as in step 4. of the above algorithm.*

[3]Thus, alternatively, an algebraic curve $\mathbf{C}$ and its associated "normals" $\mathbf{n}$ may (either or both) be given as a list of points or point-normal pairs.

Proof : In step *4.(b)* above, assume without loss of generality that $t_x \neq 0$. Then the selected equation

$$f_y \cdot n_z - n_y \cdot f_z = 0 \tag{64}$$

We first show that even though equation (64) is evaluated at only $(n+m-1)d+1$ points of C in step *4.(b)* above, it holds for all points on C. Equation 64 defines an algebraic surface T of degree $(n+m-1)$ which intersects C of degree d at $(n+m-1)d$ real points. Invoking Bezout's theorem, and from the irreducibility of C, it follows that C must lie entirely on the surface T. Hence equation (64) is valid along the entire curve C.

We now show that step 4. of the above algorithm, satisfies the tangency condition as specified in definition 3.2. Since $\mathbf{t}$ of step *4.(a)* is a tangent vector at all points of C, and the surface $S : f = 0$ contains C, the gradient vector ∇f is orthogonal to $\mathbf{t}$, which yields the equation :

$$f_x \cdot t_x + f_y \cdot t_y + f_z \cdot t_z = 0 \tag{65}$$

valid for all points of C. Next, from the definition of a "normal" of a space curve,

$$n_x \cdot t_x + n_y \cdot t_y + n_z \cdot t_z = 0 \tag{66}$$

valid for all points of C. Now it is impossible that both $n_y(x, y, z)$ and $n_z(x, y, z)$ are identically zero along C, since if they were then equation (66) would imply that $n_x \cdot t_x = 0$, and as we had assumed that $t_x \neq 0$, would in turn imply that also $n_x = 0$ along C, which would contradict the earlier assumption that $\mathbf{n}$ is not identically zero. Hence, at least, one of n_y and n_z must also be nonzero. Without loss of generality, let $n_y \neq 0$. Also, let $\alpha(x, y, z) = \frac{f_y}{n_y}$. Then,

$$f_y = \alpha \cdot n_y \tag{67}$$

and substituting into equation (64) yields

$$f_z = \alpha \cdot n_z \tag{68}$$

for all points on C. From equations (65), (67) and (68) we obtain,

$$f_x \cdot t_x + \alpha \cdot n_y \cdot t_y + \alpha \cdot n_z \cdot t_z = 0 \tag{69}$$

By multiplying α to equation (66) and subtracting equation (69) from it, we obtain

$$f_x \cdot t_x = \alpha \cdot n_x \cdot t_x \tag{70}$$

and since $t_x \neq 0$, finally obtain

$$f_x = \alpha \cdot n_x \tag{71}$$

valid at all points of C. Hence equations (67), (68), and (71) together imply that $\nabla f(x, y, z) = \alpha \cdot \mathbf{n}$ for all points C and some nonzero α[4]. Hence, the tangency condition of definition 3.2 is met. $\square$

Rational Curves with Normals : Parametric Definition When both a space curve and its associated "normal" are given in rational parametric form, their equations can be used directly to produce a linear system for interpolation, instead of first computing $(n+m-1)d+1$ points on the curve. Let $\mathbf{C} : (x = G_1(s), y = G_2(s), z = G_3(s))$ be a rational curve of degree d with associated "normals" $\mathbf{n}(s) = (n_x(s), n_y(s), n_z(s))$ of degree m. A Hermite interpolating surface $S : f(x, y, z) = 0$ of degree n which smoothly contains C is computed as follows:

[4] From equation (69) we see that $\alpha(x, y, z)$ must not be identically zero along C, for otherwise, $\nabla f = (0, 0, 0)$ for points along C and would contradict the fact that we chose a non-trivial solution for the surface $S : f = 0$ which was nonsingular at all points along C.

1. **(containment condition)** Substitute $(x = G_1(s), y = G_2(s), z = G_3(s))$ into the equation $f(x, y, z) = 0$. This results in $nd + 1$ homogenous linear equations as in section *4.1.2*.

2. **(tangency condition)**

   (a) Compute $\nabla f(s) = \nabla f(G_1(s), G_2(s), G_3(s))$ and $\mathbf{t}(s) = (\frac{dx}{ds}, \frac{dy}{ds}, \frac{dz}{ds})$. Note $\mathbf{t} = (t_x, t_y, t_z)$ is the tangent vector to C.

   (b) i. If $t_x \neq 0$, use the equation $f_y(s) \cdot n_z(s) - n_y(s) \cdot f_z(s) = 0$.

   ii. If $t_y \neq 0$, use the equation $f_x(s) \cdot n_z(s) - n_x(s) \cdot f_z(s) = 0$.

   iii. If $t_z \neq 0$, use the equation $f_x(s) \cdot n_y(s) - n_x(s) \cdot f_y(s) = 0$.

   In each case, the numerator of the simplified rational function equation is set to zero. This yields ,additionally $(n-1)d+m+1$ linear equations in the coefficients of the surface $S : f(x, y, z) = 0$.

3. In total we obtain a homogeneous system of $(2n-1)d+m+2$ linear equations. Any non-trivial solution of the linear system, for which additionally ∇f is not identically zero for all points of C, (that is, the surface S is not singular along the curve C), will represent a surface which Hermite interpolates C.

The proof of correctness of the above algorithm follows from Lemma 3.7 and the following lemma, which shows why the selected equation of step *2.* satisfies the tangency condition.

Lemma 3.9. *If we choose a nontrivial solution for which the resulting Hermite interpolating surface S is nonsingular along the given curve C, then the step 2. guarantees that the tangency condition of definition 3.2 is met.*

Proof : The proof is similar to the proof of lemma 3.5 with minor modifications. We omit this here and refer the reader to the full paper. □

Applications: Mixed Points and Space Curve Data The basic mechanics of Hermite interpolation using algebraic surfaces, as presented in the algorithms of sections *3.* and *4.*, are

1. properties of a surface to be designed are described in terms of a combination of points, curves, and possibly associated "normal" directions,

2. these properties are translated into a homogeneous linear system of equations with extra surface constraints, and then

3. nontrivial solutions of the above system are computed using the smallest surface degree

In particular the total number of linear equations generated for a possible algebraic surface of degree n to *smoothly contain* k points with fixed constant "normal" directions and also to *smoothly contain* l space curves of degree d with assigned "normal" directions, varying as a polynomial of degree m, is $3k + (2n + m - 1)dl + 2l$. This number reduces to $3k + (2n - 1)dl + ml + 2l$ when all the space curves and associated "normals" are defined parameterically.

For a given configuration of points, curves and "normals" data the above interpolation scheme, allows one to both upper and lower bound the degree of Hermite interpolated surfaces.

1. **Lower Bound** Let k be the rank of a homogenous system of linear equations, derived for the given geometric configuration. The rank tells us the exact number of independent constraints on the coefficients of our desired algebraic surface. Dependencies arise from spatial interrelationships of the given points and curves. From the rank then we can conclude that there exists no algebraic surface of degree less than or equal to n_0 where n_0 is the largest n such that $K < k$ with $K = \binom{n+3}{3} - 1$.

2. **Upper Bound** Alternatively, the smallest n can be chosen such that $K \geq k$, where again K is the number of independent coefficient unknowns and k is the rank of the above linear system. The non-trivial real solutions of the linear system represents a $K-k$ parameter family of algebraic surfaces of degree n which interpolates the given geometric data. We then select suitable real surfaces from this family, which additionally satisfy our nonsingularity and irreducibility constraints[5]

We now enumerate some results which lower bound the degree of feasible Hermite interpolated surfaces.

1. Two skewed lines in space with constant-direction normals cannot be Hermite interpolated with real quadrics. The only real quadric which satisfies both containment and tangency conditions reduces into two planes.

2. Two lines in space with constant-direction normals can be Hermite interpolated with a real quadric if and only if the lines are parallel or intersect at a point, and the normals are not orthogonal to the plane containing them. The real quadric is a "cylinder" when the lines are parallel and a "cone" when the lines intersect.

3. The minimum degree of a real algebraic surface, which Hermite interpolates two lines in space, one with a constant direction normal, the other with a linearly varying normal is three.

4. Two lines with linearly varying normals can be Hermite interpolated by a quadric in only some special cases. In general, a surface of at least degree three is needed. When real quadric surface interpolation is possible, the real quadric is either a hyperboloid of one sheet (the two lines may be parallel, intersecting, or skewed) or a hyperbolic paraboloid (the two lines can only be intersecting or skewed).

We exhibit the method of generating tight upper bounds on the degree, by constructing the lowest degree Hermite interpolated surfaces for "blending" and "joining" primary surfaces of solid models as well as for "fleshing" curved wireframe models of physical objects.

Example 3.10. *A Hyperboloid Patch for Smoothing the Intersection of Two Cylindrical Surfaces*

The case of two circular cylinders is a common test case for "blending" algorithms. Various different ways have been given, (for e.g. see [17, 25, 32]) for computing a suitable surface which "smoothes" or "blends" the intersection of two equal radius cylinders, $S_1 : x^2 + y^2 - 1 = 0$ and $S_2 : x^2 + z^2 - 1 = 0$. We consider an ellipse C_1 on S_1 (it is the intersection with the plane $3x + y = 0$), defined parameterically, $C_1 : (\frac{2t}{1+t^2}, \frac{-6t}{1+t^2}, \frac{1-t^2}{1+t^2})$ with associated rational "normal" $\mathbf{n}_1(t) = (\frac{4t}{1+t^2}, 0, \frac{2-2t^2}{1+t^2})$, and the ellipse C_2 on S_2 defined implicitly, $C_2 : ((y^2+z^2-1 = 0, x+3y = 0)$ with associated "normal" $\mathbf{n}_2(x, y, z) = (0, 2y, 2z)$. Both C_1 and C_2's "normals" are respectively chosen in the same direction as the gradients of thier corresponding containing surfaces S_1 and S_2. This ensures that any Hermite interpolating surface for C_1 and C_2 will also meet S_1 and S_2 smoothly along these curves. As a possible Hermite interpolant we consider a degree two algebraic surface $S : f(x, y, z) = ax^2+by^2+cz^2+dxy+eyz+fzx+gx+hy+iz+j = 0$. Applying the method of section *4.2.2*, to S and C_1 results in 8 equations, 5 from the containment condition and 3 from the tangency condition. (Note: 5 equations are supposed to be generated, but 2 of these turn out to be degenerate). For C_2, we use the method of section *4.2.1*, and first compute L_c = {(0, 0, 1), (-3, 1, 0), (3, -1, 0), (-2.4, 0.8, -0.6), (2.4,, -0.8, -0.6)} and L_t = {[(0, 0, 1), (0, 0, 2)], [(-3, 1, 0), (0, 2,

[5]However some of these interpolating surfaces might still not be suitable for the design application they were intended to benefit. These problems arise when the given points or curves are smoothly interpolated, however lie on separate real components of the same nonsingular, irreducible algebraic surface. We consider this problem again in section *7*.

0)], [(3, -1, 0), (0, -2, 0)], [(-2.4,0.8,-0.6), (0,1.6,-1.2)], [(2.4,-0.8,-0.6), (0,-1.6,-1.2)]}. For these lists, we get 10 equations, 5 from the containment condition and another 5 from the tangency condition. Hence, overall the linear system consists of 9 independent unknowns and 18 equations. The rank of this system is 9, and hence we get the unique surface soution $f(x,y,z) = x^2+y^2-8z^2+6xy+8 = 0$. This real quadric satisfies both the nonsingularity and irreducibility constraints. It is a hyperboloid of one sheet and the lowest degree surface which "blends" the intersection of the two cylinders. See Figure 1. at the end of the paper. □

Example 3.11. *A Cubic Interpolation Surface for Smoothly Joining Two Cylindrical Surfaces*

Another example for an interpolated surfaces arises when we consider computing the lowest degree surface which can smoothly join two truncated circular cylinders $S_1 : x^2 + z^2 - 1 = 0$ for $y \geq 2$ and $S_2 : y^2+z^2-1 = 0$ for $x \geq 2$. Traditionally, this join has been achieved by using a quarter section of a torus (a degree four algebraic surface). This example was considered in [32] where a similar solution was obtained by finding low degree surface members in appropriate product ideals of the section curves. Here, we illustrate the Hermite interpolation technique which also proves that degree three, is the lowest degree algebraic surface to satisfy the smooth-join requirement for this configuration. As before, we take a circle $C_1 : (\frac{2t}{1+t^2}, 2, \frac{1-t^2}{1+t^2})$ on S_1 with the associated rational "normal" $\mathbf{n}_1(t) : (\frac{4t}{1+t^2}, 0, \frac{2-2t^2}{1+t^2})$ and the circle $C_2 : (2, \frac{2t}{1+t^2}, \frac{1-t^2}{1+t^2})$ on S_2 with the associated rational "normal" $\mathbf{n}_1(t) : (0, \frac{4t}{1+t^2}, \frac{2-2t^2}{1+t^2})$. Again, both C_1 and C_2's "normals" are respectively chosen in the same direction as the gradients of their corresponding containing surfaces S_1 and S_2. This ensures that any Hermite interpolating surface for C_1 and C_2 will also meet S_1 and S_2 smoothly along these curves. A degree two algebraic surface does not suffice for Hermite interpolation, since the rank of the resulting linear system is greater than 9, the number of independent unknowns. Next as a possible Hermite interpolant consider a degree three algebraic surface with 19 independent unknown coefficients. Applying the Hermite interpolation method of section *4.2.2*, to the curves results in 24 equations (28 equations are supposed to be generated, but 4 of the 28 are degenerate.). The rank of this linear system is 19, and thus there is a unique cubic Hermite interpolating surface, which is $f(x,y,z) = x^3 + y^3 + x^2y + xy^2 + xz^2 + yz^2 - 4x^2 - 4y^2 - 4z^2 - 4xy + 3x + 3y + 4$. See Figure 2. at the end of the paper. □

Example 3.12. *A Family of Cubic Surfaces which Flesh a Saddle Wireframe*

Consider a wireframe of a solid model consisting of a circle, a parabola , and two lines. Using Hermite interpolation, we find a 2 parameter family of cubic surfaces which "fleshes" this wire frame. Again we can show that no degree two algebraic surface can contain all these curves simultaneously. Consider the circle $C_1 : (\frac{2t}{1+t^2}, \frac{1-t^2}{1+t^2}, -1)$, the parabola $C_2 : (t, -2t^2 + 2, 1)$, the line $C_3 : (1, 0, t)$, and the line $C_4 : (-1, 0, t)$. As a possible Hermite interpolant consider a degree three algebraic surface with 19 independent unknown coefficients. Applying the Hermite interpolation method of section *4.1.2*, we obtain a homogeneous linear system of 22 equations. The rank of this system is 17, so there is a 2 parameter family of cubic Hermite interpolating surfaces which is $f(x,y,z) = -bx^2z + \frac{a-b}{2}y^2z - \frac{a+b+4c}{4}yz^2 - ax^2 - \frac{a-b}{2}y^2 - \frac{a+b}{4}yz + cy + bz + a$. A suitable real solution surface from this family is obtained for $a = 3$, $b = 2$, and $c = 1$, yielding $f(x,y,z) = 12 + 4y - 2y^2 - 9yz^2 - 8x^2z + 2y^2z - 12x^2 - 5yz + 8z = 0$. See Figure 3. at the end of the paper. □

Extensions and Related Techniques: Meshing Quadric Surface Patches Solving a linear system of equations plays a key role in Hermite interpolation. In what follows, we give another approach of algebraic surface design where a *nonlinear* system of equations needs to be solved.

In Hermite interpolation, the linear equations generated represent the constraints to be met by a single interpolating surface. The larger the number of independent containment and tangency constraints, the higher the degree of the resulting interpolating surface. The total number of constraints depends largely on the degrees of the given curves and their "normals".

Since the number of terms in an algebraic surface increases as the cube of its degree, computation with high degree algebraic surfaces gets expensive and error prone. Hence, for good reasons we are advised to keep the degrees of our "blending", "joining" and "fleshing" surfaces as low as possible. The problem considered in this section is to Hermite interpolate, conic curves in space with (not necessarily one), but a combination of quadric surface patches which themselves meet smoothly along thier intersection curves. Such "smooth" meshing has been largely addressed by [26, 27] amongst others, using the Bézier representations of surfaces.

We first state a useful theorem from algebraic geometry, observed and used independently by numerous authors in various alternate forms

Lemma 3.13. *Let $S : f(x,y,z) = 0$ be an irreducible quadric surface, and $Q : q(x,y,z) = 0$ be a plane which intersects S in a conic C. Then, another quadric surface $S_1 : f_1(x,y,z)$ is tangent to S along C if and only if there exists nonzero constants α, β (possibly complex) such that $f_1 = \alpha f + \beta q^2$.*

Proof: The proof may be found for example, in [24, 31]. $\square$

Since we are interested in interpolation with real surfaces, we may restrict α and β to be real numbers.

A related theorem can be derived for the quadric surface interpolation of two conics in space.

Lemma 3.14. *Consider quadrics $S_1 : f_1 = 0$, $S_2 : f_2 = 0$ and planes $Q_1 : q_1 = 0$, $Q_2 : q_2 = 0$. Let $C_1 : (f_1 = 0, q_1 = 0)$ and $C_2 : (f_2 = 0, q_2 = 0)$ be two conics in space. Then C_1 and C_2 can be Hermite interpolated by a quadric surface S if and only if there exist nonzero constants α_1, α_2, β_1, and β_2 (possibly complex) such that $\alpha_1 f_1 + \beta_1 q_1^2 - \alpha_2 f_2 - \beta_2 q_2^2 = 0$.*

Proof: Trivial. (Just apply Lemma 3.13 twice.) $\square$

This theorem is constructive, in that, it again yields a system of equations and a direct way of computing a Hermite interpolating quadric surface. Furthermore a solution to the above equations, linear in the α's and β's, exists if and only if such an interpolating quadric surface exists. Again, when real surfaces are favorable, we require α_1, α_2, β_1, and β_2 to be real numbers.

Example 3.15. *Suppose $C_1 : (x^2 + z^2 - 1 = 0, 3x + y = 0)$, and $C_2 : (y^2 + z^2 - 1 = 0, x + 3y = 0)$. We get the following equation from Lemma 3.14: $(\alpha_1 + 9\beta_1 - \beta_2)x^2 + (\beta_1 - \alpha_2 - 9\beta_2)y^2 + (\alpha_1 - \alpha_2)z^2 + (6\beta_1 - 6\beta_2)xy + (\alpha_1 - \alpha_2) = 0$. This implies $\alpha_1 = \alpha_2$, $\beta_1 = \beta_2$, $\alpha_1 = -8\beta_1$. When $\alpha_1 = -8$ and $\beta_1 = 1$, the interpolating surface is $x^2 + y^2 - 8z^2 + 6xy + 8 = 0$.*

In the Lemma 6.2 and the example, the two conics on the given quadric surfaces, S_1 and S_2, were fixed. If we have freedom to choose different intersecting planes Q_1 and Q_2 then we may be able to find a family of quadric interpolating surfaces. In this case, the equations of planes Q_1 and Q_2 would have unknown coefficients and the use of Lemma 3.14 would result in a nonlinear system of equations, linear in terms of α_1, α_2, β_1 and β_2, and quadratic in terms of the unknowns of the plane's equations. Now, rather than trying to find a single quadric surface, we can also extend the above Lemma, to construct two or more quadrics which smoothly contain two given conics in space, and furthermore themselves intersect in a smooth fashion. The following Lemma, which is constructive tells us how to go about this.

Lemma 3.16. *Let $C_1 : (f_1 = 0, q_1 = 0)$ and $C_2 : (f_2 = 0, q_2 = 0)$ be two conics in space. These two curves can be smoothly contained by two "smoothly intersecting" quadrics $S_1 : g_1 = a_1 f_1 + b_1 q_1^2 = 0$ and quadrics $S_2 : g_2 = a_2 f_2 + b_2 q_2^2$ if and only if there exist nonzero constants a_1, a_2, b_1, b_2, α, β, and a plane $Q : q(x,y,z) = 0$ such that $a_1 f_1 + b_1 q_1^2 - \alpha(a_2 f_2 + b_2 q_2^2) - \beta q^2 = 0$.*

Proof: It follows from Bezout's theorem for surface intersection, that two quadrics always intersect smoothly in a plane curve (either an irreducible conic or straight lines). Let the intersection curve lie on the unknown plane Q, then just apply Lemma 3.13 three times. $\square$

The final equation of the above Lemma results in a nonlinear (cubic) system of equations which is linear in terms of the unknowns a_1, a_2, b_1, b_2, α, and β, and quadratic in terms of the unknown coefficients of the plane $Q : q = 0$.

Example 3.17. *Let conic C_1 be given by $f_1 = x^2 + y^2 - z^2 + 4xy + 4x + 4y + 3 = 0$ (a hyperboloid of one sheet) and $q_1 = x + y + 1 = 0$. Simiarly, let conic C_2 be given by $f_2 = 19x^2 + 10y^2 - 9z^2 + 38xy - 114x - 114y + 180 = 0$ (a hyperbooid of one sheet), $q_2 = x + y - 3 = 0$, and let the unknown plane be $P : ax + by + cz + d = 0$. Then the equation for the system of smooth interpolating quadrics $a_1 f_1 + b_1 q_1^2 - \alpha(a_2 f_2 + b_2 q_2^2) = \beta(ax + by + cz + d)^2$ results in a nonlinear system of 10 equations: $-\beta c^2 + 9a_2\alpha - a_1 = 0$, $-2b\beta c = 0$, $-2a\beta c = 0$, $-2\beta cd = 0$, $-b^2\beta - \alpha b_2 + b_1 - 10a_2\alpha + a_1 = 0$, $-2ab\beta - 2\alpha b_2 + 2b_1 - 38a_2\alpha + 4a_1 = 0$, $-2b\beta d + 6\alpha b_2 + 2b_1 + 114a_2\alpha + 4a_1 = 0$, $-a^2\beta - \alpha b_2 + b_1 - 19a_2\alpha + a_1 = 0$, $-2a\beta d + 6\alpha b_2 + 2b_1 + 114a_2\alpha + 4a_1 = 0$, and $-\beta d^2 - 9\alpha b_2 + b_1 - 180a_2\alpha + 3a_1 = 0$. This nonlinear system has a nontrivial solution (in the sense that a_1, a_2, and α are nonzero) : $a_1 = -a^2\beta$, $b_1 = 2a^2\beta$, $a_2 = -\frac{a^2\beta}{9\alpha}$, $b_2 = \frac{19a^2\beta}{9\alpha}$, and $b = c = d = 0$.[6] Hence, the two conics C_1 and C_2 are smoothly contained by quadrics $g_1 = 0$ and $g_2 = 0$, respectively, and which in turn, smoothly intersect in a conic in the plane Q. The real quadric $g_1 = x^2 + y^2 + z^2 - 1 = 0$ is a sphere, while the other real quadric $g_2 = y^2 + z^2 - 1$ is a cylinder. Note that the above solution implies that there is only one pair of real quadric surfaces which smoothly contain the given conics. Also, for this case, it can be shown that neither a single quadric nor a single cubic surface can Hermite interpolate the two given conics. Geometrically then, the two hyperboloids of one sheet are smoothly joined by a sphere and a cylinder. See Figure 4. at the end of the paper.*

The above method of Lemma 3.16 can also be straightforwardly extended to finding a mesh of n quadric surfaces which smoothly contain two given conics in space. Necessarily the complexity of the nonlinear system of equations also goes up.

3.2 Surfaces of Revolution

From Bezout's theorem[28], we realize that the intersection of two implicit surfaces of algebraic degree d can be a curve of geometric degree $O(d^2)$. Furthermore the same theorem implies that the intersection of two parametric surfaces of algebraic degree d can be a curve of degree $O(d^4)$. Hence, while the potential singularities of the space curve defined by the intersection of two implicit surfaces defined by polynomials of degree d can be as many as $O(d^4)$, the potential singularities of the space curve defined by the intersection of two parametric surfaces defined by polynomials of degree d can be as many as $O(d^8)$[5]. Hence keeping the degree of fitting surfaces as low as possible benefits both the efficiency and the robustness of post processing for modeling and display[2].

Algebraic Surfaces of Revolution Consider an algebraic surface which is obtained by revolving an algebraic curve $f(x, y) = 0$ (on the xy plane) around the y axis. (See Figure 3.) Rather than restricting ourselves to a circular rotation, we consider a more general elliptic revolution where the rotation path is described by an ellipse $E : x^2 + \frac{z^2}{\alpha^2} = \{r(y)\}^2$ with $\alpha > 0$. Here, $r(y)$ is the x coordinate of the point (x, y) on the curve $C : f(x, y) = 0$.

Now, the surface that results from revolving C along E is specified as "$x^2 + \frac{z^2}{\alpha^2} = \{r(y)\}^2$ *subject to* $f(r(y), y) = 0$." The equation $F(x, y, z) = 0$ of the surface S, hence, becomes $F(x, y, z) = f(\sqrt{x^2 + \frac{z^2}{\alpha^2}}, y) = 0$ where $F(x, y, z)$ is not necessarily *algebraic* due to introduction of the square

[6] This nonlinear system was solved with the aid of MACSYMA, on a Symbolics 3650

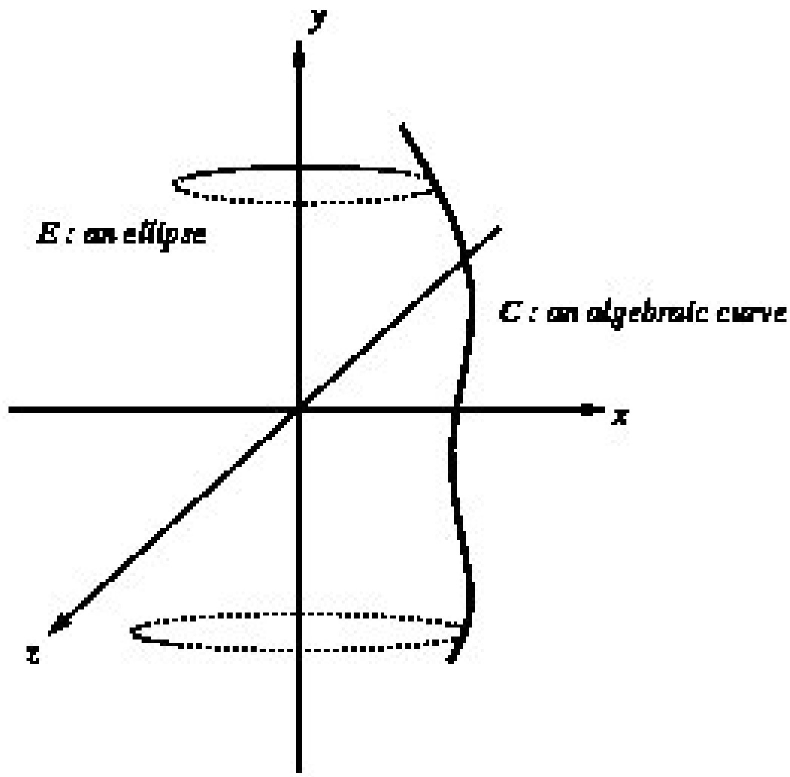


Figure 3: Revolution of an Algebraic Curve along an Ellipse

root. By allowing only even-powered x's (x^0, x^2, x^4, $\cdots$) in $f(x,y)$, we can force $F(x,y,z)$ to be algebraic. Geometrically, this restriction, imposed on the revolved curve, that maintains *algebraicity*, means that the curve $f(x,y)=0$ is symmetric to the y axis.

For quadric curves $f(x,y)=0$, x^2 is the only possible factor of terms in f. Hence, f includes a 4-dimensional vector space V_f^2 of polynomials over real numbers that is spanned by the basis $\{x^2, y^2, y, 1\}$. In case of cubic curves $f(x,y)=0$, the vector space V_f^3 is spanned by the basis $\{x^2y, x^2, y^3, y^2, y, 1\}$ with dimension 6. Quartic curves $f(x,y)=0$ can be chosen from a more abundant vector space V_f^4 of dimension 9, generated by the basis $\{x^4, x^2y^2, x^2y, x^2, y^4, y^3, y^2, y, 1\}$. The bases of vector spaces V_f^d for higher degree curves are formulated in the same fashion.

Each algebraic curve of degree d in V_f^d, revolved around an ellipse, results in an algebraic surface of the same degree. Then we naturally come to the following question : "Is a surface, generated by revolving around an ellipse an algebraic curve that is not in V_f^d, algebraic at all?" In fact, the surface *is* algebraic, though the surface's degree gets doubled. This doubling of the degree arises from the single squaring required to remove the square root from odd-powered x factors. For example, consider a circle $f(x,y)=(x-5)^2+(y-5)^2-1=x^2-10x+y^2-10y+49=0$ of radius 1, centered at $(5,5)$. This conic curve is not in V_f^2 because of the term $10x$. However, by moving $10x$ to the right hand side, and then squaring both sides, we can obtain a quartic curve in V_f^4 which generates a torus (of degree 4) by rotation. Intuitively, the squaring operation has an effect of putting another circle of the same shape to the other side of the y axis in order to artificially make the curve symmetric to the y axis. Any algebraic curve of degree d which is not in V_f^d can be made to be in V_f^{2d} by moving all terms with odd-powered x factors to one side, and squaring both sides.

Remark 3.18. *Let $C : f(x,y)=0$ be an algebraic curve of degree d, and $E : x^2+\frac{z^2}{\alpha^2}=\{r(y)\}^2$ be an ellipse of a rotation path. Then, the algebraic surface $S : F(x,y,z)=0$, resulting from revolving*

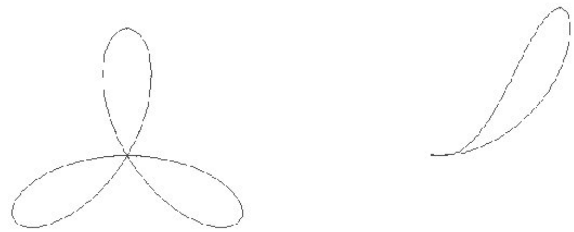


Figure 4: Two Quartic Algebraic Curves

Figure 5: Degree 4 and 8 Algebraic Surfaces of Revolution

C around E, has degree d if C is symmetric around the y axis, or $2d$ otherwise.

A geometric interpretation to Remark 3.18 is as follows : Consider a line on the xy plane parallel to the x axis. This line intersects with C at most d times. Now, imagine the intersection between the line and S. When C is symmetric, the number of intersection remains the same. However, if C is not symmetric, the number of intersection is doubled up because C, rotated by 180 degrees, creates the same number of line-curve intersections.

It is important to understand that, the degrees of freedom, in choosing a curve $f(x, y) = 0$ of degree d from V_f^d, is $\mathsf{dim}(V_f^d) - 1$ where $\mathsf{dim}(*)$ is the dimension of a vector space. Since all the polynomials on a line in V_f^d that passes through f and 0 describe the same curve, we have one less than $\mathsf{dim}(V_f^d)$ degrees of freedom. It is not hard to come up with the expression for $\mathsf{dim}(V_f^d)$:

$$\mathsf{dim}(V_f^d) = \begin{cases} \frac{(d+2)^2}{4} & \text{if } d \text{ is even} \\ \frac{(d+1)(d+3)}{4} & \text{if } d \text{ is odd} \end{cases}$$

In many situations as will be shown later, the curve $f(x, y) = 0$ is to be designed such that it satisfies given geometric requirements. We are interested in designing piecewise curves from given digitized data, and revolving them in a complicated manner to model some class of objects with low degree algebraic surfaces. It will be explained below how the degrees of freedom in piecewise algebraic curves of a given degree limit the geometric continuity between them.

Example 3.19. *Figure 4 (a) and (b) displays two quartic algebraic curves $(x^2+y^2)^2+3x^2y-y^3 = 0$ and $x^4+x^2y^2-2x^2y-xy^2+y^2 = 0$, respectively [30]. The curves, after rotation, result in algebraic surfaces of degree 4 and 8, respectively, and shown in Figure 5 (a) and (b).*

Parametric Surfaces of Revolution Now, we get to a question : "Is it also possible to find a restricted bases of *rational parametric* curves that result in *rational parametric* surfaces of the *same geometric* degree after revolution around an axis?" Consider a rational parametric curve of degree d

$$C(t) = \begin{pmatrix} X(t) \\ Y(t) \end{pmatrix} = \begin{pmatrix} \frac{x(t)}{w(t)} \\ \frac{y(t)}{w(t)} \end{pmatrix}$$

where the degrees of the polynomials $x(t)$, $y(t)$, and $w(t)$ are at most d. The surface obtained by revolving $C(t)$ around y-axis along an ellipse $E : x^2 + \frac{z^2}{\alpha^2} = \{r(y)\}^2$ with $\alpha > 0$ can be represented as $F(s,t) = (X(s,t), Y(s,t), Z(s,t))$, where

$$\begin{aligned} X(s,t) &= \frac{2s}{1+s^2}\frac{x(t)}{w(t)} \\ Y(s,t) &= \frac{y(t)}{w(t)} \\ Z(s,t) &= \frac{\alpha(1-s^2)}{1+s^2}\frac{x(t)}{w(t)}. \end{aligned}$$

First, this representation answers that the revolved surface is always rational parametric. Then, the second question on the degree of $F(s,t)$ must be answered. We are interested in lowering both the *algebraic* degree in the polynomials in $F(s,t)$ and the *geometric* degree of $F(s,t)$ (the maximum possible intersection of $F(s,t)$ with a line). In construction of rational parametric revolved surfaces, we follow the same path we did in the previous subsection. From Remark 3.18, we know that an algebraic curve of degree d generates an algebraic surface of the same degree only when it is symmetric around an axis. Since every rational parametric curve of degree d is an algebraic curve of degree d, we are led to the fact that $F(s,t)$ is of degree d if $C(t)$ is symmetric around the y-axis.

A rational parametric curve is symmetric if there is a parametrization $C(t) = (X(t), Y(t)) = (\frac{x(t)}{w(t)}, \frac{y(t)}{w(t)})$ such that $X(t) = -X(-t)$ and $Y(t) = Y(-t)$. That is,

$$\frac{x(t)}{w(t)} = -\frac{x(-t)}{w(-t)} \tag{72}$$

$$\frac{y(t)}{w(t)} = \frac{y(-t)}{w(-t)} \tag{73}$$

The above conditions are met if *either* $x(t)$ is an odd function (all the terms with nonzero coefficients are odd-powered), and $y(t)$, $w(t)$ are even functions (all the terms with nonzero coefficients are even-powered), *or* $x(t)$ is an even function, and $y(t)$, $w(t)$ are odd functions. It is not difficult to see that the polynomials in the second case can be converted into the first case polynomials by multiplying t to both numerator and denominator, and vice versa. In fact, any polynomials that satisfies the conditions (72) and (73) fall in the above two categories.

Lemma 3.20. *Let $x(t)$, $y(t)$, and $w(t)$ be polynomials in t such that $x(t)$ and $w(t)$ are relatively prime, and $y(t)$ and $w(t)$ are relatively prime. Then, $x(t)$ is an odd function, and $y(t)$, $w(t)$ are even functions if and only if $\frac{x(t)}{w(t)} = -\frac{x(-t)}{w(-t)}$ and $\frac{y(t)}{w(t)} = \frac{y(-t)}{w(-t)}$.*

PROOF : See [7]. □

From now on, we assume that $x(t)$ is an odd function, and $y(t)$ and $w(t)$ are even functions without loss of generality. Since a degree d curve $C(t) = (X(t), Y(t)) = (\frac{x(t)}{w(t)}, \frac{y(t)}{w(t)})$ is symmetric around y-axis, the surface made by revolving it around y-axis is a surface of geometric degree d. The surface equation $F(s,t)$ given above is represented by degree $d+2$ polynomials. In [7] we show it is possible to reduce the algebraic degree of the parametric surface equations to d by applying a transformation to $F(s,t)$.

Remark 3.21. *Let $C : C(t) = (\frac{x(t)}{w(t)}, \frac{y(t)}{w(t)})$ be a rational parametric curve of degree d where $x(t)$ is an odd function, and $y(t)$, $w(t)$ are even functions, and $E : x^2 + \frac{z^2}{\alpha^2} = \{r(y)\}^2$ be an ellipse of a rotation path. Then, the algebraic surface $S : F(s,t) = (X(s,t), Y(s,t), Z(s,t))$ in the rational parametric form, resulting from revolving C around E, has geometric degree d, and can be parameterized in the way that $X(s,t)$, $Y(s,t)$, and $Z(s,t)$ are degree d rational polynomials.*

The class of the above rational parametric curves contains symmetric parametric curves that intersect with y-axis. The set of all such curves is only a proper subset of all symmetric parametric curves. Another interesting class of symmetric rational parametric curves is defined as $C(t) = (X(t), Y(t)) = (\frac{x(t)}{w(t)}, \frac{y(t)}{w(t)})$ such that $X(t) = -X(-\frac{1}{t})$ and $Y(t) = Y(-\frac{1}{t})$[7]. It still remains open how to specify all the bases of symmetric rational parametric curves of a given degree.

Example 3.22. *Recall the "three-leaf clover" in Example 3.19. Its parametric form is $C(t) = (\frac{t^3-3t}{t^4+2t^2+1}, \frac{t^4-3t^2}{t^4+2t^2+1})$. After circular revolution and the above mentioned reparametrization, the quartic surface is $F(u,v) = (\frac{u(u^2+v^2-3)}{(u^2+v^2)^2+2(u^2+v^2)+1}, \frac{(u^2+v^2)^2-3(u^2+v^2)}{(u^2+v^2)^2+2(u^2+v^2)+1}, \frac{v(u^2+v^2-3)}{(u^2+v^2)^2+2(u^2+v^2)+1})$ and shown in Figure 5 (a).*

3.3 Construction of Piecewise C^k Continuous Revolved Objects

So far we have discussed about revolution of a single algebraic curve, represented in either the implicit or the parametric form. A revolved object with a complicated shape, however, cannot be modeled by rotating only one low degree curve. Instead, it is more appropriate to approximate a revolved object using surface patches meeting together with some order of geometric continuity. Hence, the revolved object design problem leads to the following basic problem: design piecewise C^k continuous algebraic curve segments, with restricted bases.

We focus on the design of piecewise C^k continuous *implicitly represented algebraic* curve segments.[8] Designing with parametric splines is explained in [11] in detail. Also, we shall exhibit that designing with symmetric (restricted bases) implicit algebraic curves is no more difficult than with the complete basis. The corresponding case of designing with symmetric parametric curves does not directly follow from the general parametric case and is a an open problem for further research.

Algebraic Curves and Geometric Continuity In this subsection, we describe how to compute two algebraic curves that meet with C^k continuity at a point. First of all, we assume the geometric information about a point p is expressed in terms of a (truncated) power series $C(t)$ of degree k, where $C(t) = (x(t), y(t)) = p + c_1 t + c_2 t^2 + \cdots + c_k t^k$, and $C(0) = p$. This truncated power series approximates the local geometric property (up to order k) of a curve about the point within a radius of convergence. (We will discuss later how this power series is computed.)

Now, given a (truncated) formal power series $C(t)$ about a point p, we find an algebraic curve $f(x,y) = 0$ whose power series expansion at p is the same as $C(t)$ at p. If all terms upto degree k agree for $f(x,y) = 0$ and $C(t)$ at p then $f(x,y) = 0$ is considered to meet $C(t)$ with C^k continuity at p. Let $f(x,y) = \sum_{i+j \le d} a_{ij} x^i y^j = 0$ be an algebraic curve of degree d, and

$$C(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \begin{pmatrix} p_x + c_{1x} t + c_{2x} t^2 + \cdots + c_{kx} t^k \\ p_y + c_{1y} t + c_{2y} t^2 + \cdots + c_{ky} t^k \end{pmatrix}$$

be a given parametric polynomial such that $C(0) = (p_x, p_y) \equiv p$. The relations on the coefficients of $f(x,y)$ can be extracted by repeatedly differentiating $f(C(t))$ up to order k, making all the

[7] For example, a hyperbola is in this class.

[8] From now on, by "algebraic", we mean "implicit algebraic".

derivatives vanish at $t = 0$ [14]. The first few partial derivatives are :

$$\begin{aligned}
f(C(t)) \mid_{t=0} &= f(p) = 0 \\
\frac{df(C(t))}{dt} \mid_{t=0} &= f_x(p)x^{'}(0) + f_y(p)y^{'}(0) \\
&= c_{1x}f_x(p) + c_{1y}f_y(p) = 0 \\
\frac{d^2f(C(t))}{dt^2} \mid_{t=0} &= f_{xx}(p)x^{'}(0)^2 \\
&\quad +2f_{xy}x^{'}(0)y^{'}(0) \\
&\quad +f_{yy}(p)y^{'}(0)^2 \\
&\quad +f_x(p)x^{''}(0) + f_y(p)y^{''}(0) \\
&= c_{1x}^2 f_{xx}(p) + 2c_{1x}c_{1y}f_{xy}(p) \\
&\quad +c_{1y}^2 f_{yy}(p) + c_{2x}f_x(p) \\
&\quad +c_{2y}f_y(p) = 0 \\
&\cdots
\end{aligned}$$

For each derivative of $f(C(t))$, a linear equation in terms of the unknown coefficients a_{ij} of f is generated, hence, any solution of the homogeneous linear system of $k+1$ equations becomes coefficients of algebraic curves of degree d meeting $C(t)$ with C^k continuity. Since an algebraic curve segment needs to satisfy the C^k conditions at both end points, $2k+2$ linear constraints must be satisfied. Hence, in order for an algebraic curve of degree d to exist, d must be chosen such that $\binom{d+2}{2} - 1 \geq 2k+2$, that is, the number of the degrees of freedom in coefficients of the curve is greater than or equal to the constraints for C^k continuity. Exactly the same process is applied for symmetric implicit algebraic curves of degree d with restricted bases, with the difference being that the number of degrees of freedom is given by $\mathsf{dim}(V_f^d) - 1$ as shown in section 3.2.

Computation of a Truncated Power Series There are various forms of divided-difference methods that extract geometric information around a point, from a given list of points [11]. In our case, we choose a parabola to locally approximate the points about a junction point, and take out tangential information from the parabola. The junction points themselves are for now, computed using the dynamic programming scheme in [18] which minimizes the error for a piecewise linear approximation (with fixed number of segments) to a set of digitized points. Consider a sequence of points $\cdots, p_{i-2}, p_{i-1}, p_i, p_{i+1}, p_{i+2}, \cdots$ around the junction point p_i and an imaginary power series $C(t)$ from which, we assume, the digitized points near p_i arise, and whose parameter value is $t = 0$ for p_i. Then, the tangent vector of $C(t)$ at $t = 0$ can be approximated by the approximation :

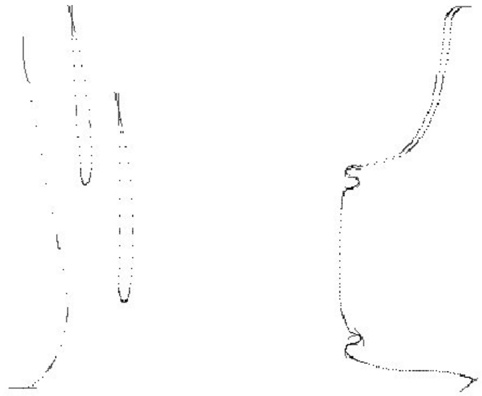
$$\begin{aligned}
C^{'}(0) &\approx \frac{\sigma_i}{\mathsf{dist}(p_i, p_{i+1})}(p_{i+1} - p_i) \\
&+ \frac{1-\sigma_i}{\mathsf{dist}(p_{i-1}, p_i)}(p_i - p_{i-1})
\end{aligned}$$

where $\sigma_i = \frac{\mathsf{dist}(p_{i-1},p_i)}{\mathsf{dist}(p_i,p_{i+1})+\mathsf{dist}(p_{i-1},p_i)}$ and $\mathsf{dist}(*,*)$ is the distance between two points.

Repeatedly applying this approximation formula, we introduce a divided-difference :

$$\Delta^j p_l = \begin{cases} p_l & \text{if } j = 0 \\ \frac{1}{j}\left(\frac{\sigma_l}{\mathsf{dist}(p_l,p_{l+1})}(p_{l+1} - p_l)\right. & \\ \left.+\frac{1-\sigma_l}{\mathsf{dist}(p_{l-1},p_l)}(p_l - p_{l-1})\right) & \text{if } j > 0 \end{cases}$$

Using this divide-difference operator, a truncated power series is represented as $C_i(t) = \Delta^0 p_i + \Delta^1 p_i t + \Delta^2 p_i t^2 + \cdots + \Delta^k p_i t^k$. Note that the geometric information, stored in the coefficients of



(a) (b)

Figure 6: Digitized Engine and Goblet with Truncated Power Series

the power series is extracted from a sequence of $2k+1$ neighboring points, centered at the junction point. This locality in the construction of a power series enables an interactive local modeling operation.

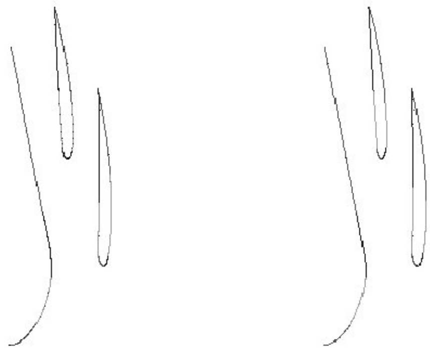
Example 3.23. *In Figure 6, two sets of digitized points are illustrated. (a) shows three lists of points that model engine parts*[9]*, and (b) is a sequence of points that models a goblet. Each point sequence is displayed with truncated power series of order two at junction points.*

Families of Algebraic Curves $f(x, y)$ In order to compute each curve segment $f_i(x, y) = 0$ that interpolates two truncated power series $C_i(t)$ and $C_{i+1}(t)$ at two end points p_i and p_{i+1}, respectively, we construct a linear system $\mathbf{M_I x} = 0$ where the unknowns are coefficients of $f_i(x, y) = 0$. The linear system is made of $2(k+1)$ equations that are generated for both truncated power series. Note that the rank of $\mathbf{M_I}$ must be less than the number of unknowns for a nontrivial solution to exist. Any nontrivial solution represents an algebraic curve that meets $C_i(t)$ and $C_{i+1}(t)$ at p_i and p_{i+1}, respectively, with C^k continuity. One heuristic that we have often used is to select a nice curve segment is to generate a sequence of additional points between the end points that approximate a curve segment, and then, apply least-squares approximation to these additional points. In the case of cubic algebraic curves, we derive a condition on the Bernstein-Bezier coefficients of cubic curves, in either the general or the restricted basis, that guarantees a smooth single curve segment inside a given control triangle.

In case all possible terms of degree d are used as a basis of $f_i(x, y) = 0$, then there are $\binom{d+2}{2}$ unknowns, and hence $\binom{d+2}{2} - 1$ degrees of freedom. However, if we choose a curve from V_f^d, we have fewer degrees of freedom due to restriction in the basis. There are only $\mathsf{dim}(V_f^d) - 1$ degrees of freedom for degree d, and this number must not be less than $2(k+2)$, the maximum possible rank for a homogeneous linear system that needs to be satisfied for order k continuity. For instance, for C^1 continuity, symmetric cubic curves are necessary, while order 2 continuity requires symmetric quartic curves.

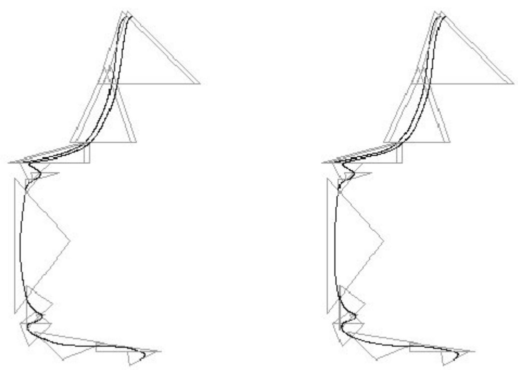
Piecewise C^k Continuous Revolved Objects Figure 7 (a) displays piecewise C^1 approximation with cubic algebraic curves in the restricted basis V_f^3. Note that a symmetric cubic curve in V_f^3 can have a tangent line parallel to x-axis only at points on the y-axis. Hence, the order of geometric continuity is only 0 at the extreme junction points on the cowls around which the curve segments make vertical turnabouts. With symmetric quartic algebraic curves in V_f^4, it is possible to approximate the point data with C^2 continuity everywhere. (See Figure 7 (b).) For the goblet

[9]This data originated from 3D scanned engine data from NASA.



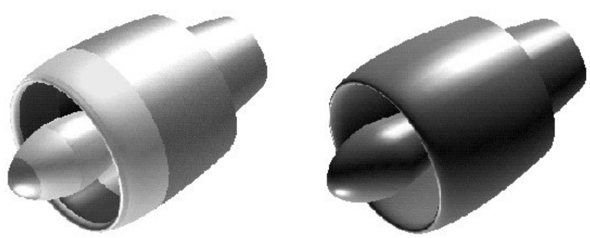
(a) (b)

Figure 7: Symmetric C^1 Cubic and C^2 Quartic Algebraic Splines



(a) (b)

Figure 8: Symmetric C^1 and Arbitrary C^2 Cubic Algebraic Splines



(a) (b)

Figure 9: C^1 Cubic and C^2 Quartic Revolved Surface Models

(a) (b)

Figure 10: C^1 Cubic and C^2 Sextic Revolved Surface Models

data, cubic curves in V_f^3, again, successfully model the data with C^1 continuity in Figure 8 (a). Figure 8 (b) shows a C^2 approximation of the same data with cubic curves in the general basis, which, hence, may not be symmetric about the y-axis.

Once algebraic splines are constructed to fit the digitized data, their revolution surface models are easily obtained, with the appropriate surface degree bounds. C^1 approximation with cubic algebraic surfaces is shown in Figure 9 (a) and are a revolution of the cubic splines in Figure 7 (a). Quartic algebraic surfaces approximate the same object well with C^2 continuity in Figure 9 (b) and are a revolution of the quartic splines in Figure 7 (b). A C^1 cubic algebraic surface goblet is illustrated in Figure 10(a) and is obtained by revolving the symmetric cubic spline in Figure 8 (a). The C^2 goblet in Figure 10(b) is obtained by revolving the arbitrary cubic splines in Figure 8 (b), and is made of degree 6 algebraic surfaces.

References

[1] S. Abhyankar. Algebraic geometry for scientists and engineers. *Math. Surveys and Monographs*, 35, 1990.

[2] C. Bajaj. Geometric Modeling with Algebraic Surfaces. In D. Handscomb, editor, *The Mathematics of Surfaces III*, pages 3–48. Oxford Univ. Press, 1988.

[3] C. Bajaj. Geometric computations with algebraic varieties of bounded degree. In *Proc. of 6th Annual ACM Symposium on Computational Geometry*, pages 148–156, 1990.

[4] C. Bajaj. Geometric modeling with algebraic surfaces. In D. Handscomb, editor, *The Mathematics of Surfaces III*, pages 3 –48. Oxford University Press, 1990.

[5] C. Bajaj. The Emergence of Algebraic Curves and Surfaces in Geometric Design. In R. Martin, editor, *Directions in Geometric Computing.* Information Geometers Press, United Kingdom, 1992.

[6] C. Bajaj, C. Hoffmann, R. Lynch, and J. Hopcroft. Tracing surface intersections. *Computer aided geometric design*, 5(4):285–307, 1988.

[7] C. Bajaj and I. Ihm. Lower Degree Approximation of Surfaces for Revolved Objects. Technical Report 92-084, Computer Sci., Purdue Univ., November 1992.

[8] C. Bajaj and A. Royappa. The ganith algebraic geometry toolkit. In *Lecture Notes in Computer Science*, volume 429, pages 268–26, 1990.

[9] R. Brent, F. Gustavson, and D. Yun. Fast solutions of toeplitz systems of polynomial equations and computation of padé approximants. *Journal of Algorithms*, 1:259–295, 1980.

[10] K. P. Cheng. Using plane vector fields to obtain all the intersection curves oftwo general surfaces. In W. Strasser and H.-P. Seidel, editors, *Theory and Practice of Geometric Modeling*, pages 187–204, 1989.

[11] C. deBoor. *A Practical Guide to Splines.* Springer-Verlag, New York, 1978.

[12] M. do Carmo. *Differential Geometry of Curves and Surfaces.* Prentice-Hall, 1976.

[13] P. Fj
"allstr
"om. Smoothing of polyhedral models. In *Proceedings of the second annual symposium on Computational geometry*, page 235. ACM, 1986.

[14] T. Garrity and J. Warren. Geometric Continuity. *CAGD*, 8:51 – 65, 1991.

[15] K. Hensel. *Theorie der Algebraischen Zahlen.* Teubner, 1908.

[16] C. Hoffmann and J. Hopcroft. The potential method for blending surfaces and corners. Technical report, Cornell University, 1985.

[17] C. Hoffmann and J. Hopcroft. Quadratic blending surfaces. *Computer-Aided Design*, 18(6):301–306, 1986.

[18] I. Ihm and B. Naylor. Piecewise Linear Approximations of Digitized Space Curves with Applications. In N. Patrikalakis, editor, *Scientific Visualization of Physical Phenomena*, pages 545–569. Springer-Verlag, Tokyo, 1991.

[19] A. Middleditch and K. Sears. Blend surfaces for set theoretic volume modelling systems. In *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, page 170. ACM, 1985.

[20] G. Müllenheim. On determining start points for a surface/surface intersection algorithm. *Computer Aided Geometric Design*, 8(5):401–408, 1991.

[21] I. Newton. *The Mathematical Papers of Issac Newton.* Cambridge University Press, 1680.

[22] J. Owen and A. Rockwood. Blending surfaces in solid geometric modeling. *Geometric Modeling: Algorithms and New Trends*, pages 347–366, 1987.

[23] H. Padé. Sur la générlisation des fractions continues algébriques. *Jour. de Math.*, 4(10):291–329, 1894.

[24] V. Pratt. Direct least-squares fitting of algebraic surfaces. *ACM SIGGRAPH Computer Graphics*, 21(4):145–152, 1987.

[25] J. Rossignac and A. Requicha. Constant-radius blending in solid modeling. *Computers in Mechanical Engineering*, 3(1):65–73, 1984.

[26] R. Sarraga. G^1 interpolation of generally unrestricted cubic Bézier curves. *Computer Aided Geometric Design*, 4(1-2):23–39, 1987.

[27] T. Sederberg. Piecewise algebraic surface patches. *Computer Aided Geometric Design*, 2:53–59, 1985.

[28] J. Semple and L. Roth. *Introduction to Algebraic Geometry.* Oxford University Press, Oxford, U.K., 1949.

[29] R. Walker. *Algebraic Curves.* Springer Verlag, New York, 1978.

[30] R. Walker. *Algebraic Curves.* Springer Verlag, New York, 1978.

[31] J. Warren. *On algebraic surfaces meeting with geometric continuity.* PhD thesis, Cornell University, 1986.

[32] J. Warren. Blending quadric surfaces with quadric and cubic surfaces. In *Proceedings of the third annual symposium on Computational geometry*, pages 341–347. ACM, 1987.

[33] K. Weierstrass. Vorbereitungssatz. In *Einige auf die Theorie der Analytischen Funktionen Mehrerer Veranderlichen sich Beziehende*, pages 135–188, 1860.

[34] J. Woodwark. Blends in geometric modelling. In *Proceedings on Mathematics of surfaces II*, page 297. Clarendon Press, 1987.