Rational Hypersurface Display

*Chanderjit L. Bajaj**
Computer Science Department,
Purdue University,
West Lafayette, IN 47907

Abstract

Algorithms are presented for polygonalizing implicitly defined, quadric and cubic hypersurfaces in $n \geq 3$ dimensional space and furthermore displaying their projections in 3D. The method relies on initially constructing the rational parametric equations of the implicitly defined hypersurfaces, and then polygonalizing these hypersurfaces by an adaptive generalized curvature dependent scheme. The number of hyperpolygons used are optimal, in that they are the order of the minimum number required for a smooth Gouraud like shading of the hypersurfaces. Such hypersurface projection displays should prove useful in scientific visualization applications. The curvature dependent polygonal meshes produced, should also prove very useful in finite difference and finite element analysis programs for multidimensional domains.

1 Introduction

Man has always strived to vault beyond the visual handicap of three dimensions. The power of algebra has allowed him to mathematically define and manipulate geometric objects in any dimensions. The advent of sophisticated graphics workstations with true 3D rendering capabilites may perhaps provide the springboard to visualizing higher dimensional objects.

This paper deals with algebraic hypersurfaces in $n \geq 3$ dimensional space. An algebraic hypersurface is simply the set of zeros of a single multivariate polynomial equation, $f(x_1, x_2, \cdots, x_n) = 0$ [17]. We further restrict ourselves to only polynomials of degree 2 and 3, for then these hypersurfaces are also *rational*. Rationality of the algebraic hypersurface is a restriction where advantages are obtained from having both the implicit and rational parametric representations. For example, an algebraic surface, in three dimensional space, is represented implicitly by the single polynomial equation $f(x, y, z) = 0$ and parameterically by the three equations $(x = G_1(s,t), y = G_2(s,t), z = G_3(s,t))$. When the G_i, $i = 1, \ldots, 3$ are rational functions, i.e. ratio of polynomials. Simpler algorithms for geometric modeling and computer graphics are possible when both implicit and parametric representations are available, see for e.g. [6, 9]. For example for shaded displays, the parametric form yields a simple way of polygonalizing the surface, while the implicit form yields an efficient calculation of the exact normals of the surface at each of the vertex endpoints of the constructed polygonal mesh. We utilize both these advantages, and others, to yield an efficient method for displaying implicitly defined, rational quadric (degree two) and cubic (degree three) hypersurfaces.

The Problem: Given implicit representations of quadric and cubic hypersurfaces, in n dimensional space, $n \geq 3$, obtain realistic shaded displays of the surfaces in 3D and true 3D shaded displays of orthographic or perspective projections of 3D slices of the hypersurfaces in higher dimensional spaces.

Prior Work: Numerous facts on rational algebraic curves and surfaces can be gleaned from books and papers on analytic geometry, algebra and algebraic geometry, see for example [15, 17, 18, 19]. For the case of 3D space, all degree two algebraic surfaces (quadrics or conicoids), are rational. All degree three surfaces (cubic surfaces or cubicoids), except the cylinders of nonsingular cubic curves and the cubic cone, have a rational parameterization, with the exceptions again only having a parameterization of the type which allows a single square root of rational functions. Most algebraic surfaces of degree four and higher are not rational, although parameterizable subclasses can be identified. In general, for n dimensional space, all hypersurfaces (not cylinders or cones) of degree d with $d \leq n$, are rational.

Various algorithms have been given for constructing the rational parametric equations of implicitly defined

*Supported in part by ARO Contract DAAG29-85-C0018, NSF grant DMS 8816286 and ONR contract N00014-88-K-0402

algebraic curves and surfaces, (i.e., hypersurfaces in 2D and 3D). See for instance [2, 3, 11, 13, 16]. The parameterization algorithms presented in [4] and [5] are applicable for irreducible rational plane algebraic curves of arbitrary degree, and irreducible rational space curves arising from the intersection of two algebraic surfaces of arbitrary degree.

Several approaches are known for rendering parametric surfaces, see for e.g. [7, 10]. The algorithm in [7] is based on convex hull properties of Bezier surfaces and uses subdivision to polygonalize the surfaces. The polygons are of course then scan converted to produce the displayed image. On the other hand lcbw uses scan lining for direct scan conversions of the curved surface. The extension of these basic techniques for the wire-frame display of hypercubes and simplicies is given in [12], while [8] provide a hidden-line algorithm for such hyperobjects.

Results: Our main results are algorithms for polygonalizing implicitly defined, quadric and cubic hypersurfaces in $n \geq 3$ dimensional space and furthermore displaying their projections in 3D. The hypersurface display algorithm is in two steps. In section 3., step I of the algorithm constructs rational parametric equations of the implicitly defined hypersurfaces. In section 4., step II of the algorithm polygonalizes these hypersurfaces by an adaptive, generalized curvature dependent scheme. The number of hyperpolygons used are optimal, in that they are the order of the minimum number required for a smooth Gouraud like shading of the hypersurfaces[1]

2 Preliminaries

A point in complex projective space $\mathbf{CP}^n$ is given by a nonzero *homogeneous coordinate vector* $(X_0, X_1, \ldots, X_n)$ of $n+1$ complex numbers. A point in complex affine space $\mathbf{CA}^n$ is given by the *non-homogeneous coordinate vector* $(x_1, x_2, \ldots, x_n) = (\frac{X_1}{X_0}, \frac{X_2}{X_0}, \ldots, \frac{X_n}{X_0})$ of n complex numbers. The set of points $Z_d^n(f)$ of $\mathbf{CA}^n$ whose coordinates satisfy a single non-homogeneous polynomial equation $f(x_1, x_2, \ldots, x_n) = 0$ of degree d, is called an $n-1$-dimensional, affine hypersurface of degree d. The hypersurface $Z_1^n(f)$ is also known as a *flat* or a *hyperplane*, a $Z_2^n(f)$ is known as a *quadric* hypersurface, and a $Z_3^n(f)$ is known as a *cubic* hypersurface. The 1-dimensional hypersurface Z_d^2 is a *curve* of degree d, a two-dimensional hypersurface Z_d^3 is known as a *surface* of degree d, and three-dimensional hypersurface Z_d^4 is known as a *threefold* of degree d. A hypersurface Z_d^n is *reducible* or *irreducible* based upon whether $f(x_1, x_2, ..., x_n) = 0$ factors or not, over the field of complex numbers. A *rational* hypersurface $Z_d^n(f)$, can additionally be defined by rational parametric equations which are given as $(x_1 = G_1(u_1, u_2, \ldots, u_{n-1}), x_2 = G_2(u_1, u_2, \ldots, u_{n-1}), \ldots, x_n = G_n(u_1, u_2, \ldots, u_{n-1}))$, where $G_1, G_2, \ldots, G_n$ are rational functions of degree d in $\mathbf{u} = (u_1, u_2, \ldots, u_{n-1})$, i.e., each is a quotient of polynomials in $\mathbf{u}$ of maximum degree d.

[1] Current graphics workstations have built in hardware scan conversion chips, and accept polygons as their high level input for efficient rendering. We assume this scan conversion facility too, and therfore only concentrate on efficiently polygonalizing the curved surfaces.

3 Step I: Parameterizing the Implicit Hypersurfaces

QUADRICS: Consider the implicit representation of a quadric hypersurface (which is neither a cylinder nor a cone), in $n \geq 2$ space,

$$Z_2^n(f): \sum_{i_1+i_2+\ldots+i_n \leq 2} a_{i_1,i_2,\ldots,i_n} x_1^{i} \ldots, x_n^{i_n} = 0 \quad (1)$$

We assume that all quadratic terms of $Z_2^n(f)$ are present, for otherwise there exists a trivial parametric representation.

1. Choose a simple point $(\alpha_1, ..., \alpha_n)$ on $Z_2^n(f)$ and apply a linear coordinate transformation

$$y_j = x_j - \alpha_j, \qquad j = 1...n \quad (2)$$

   to make the hypersurface pass through the origin. Applying the linear transformation (2) to equation (1) yields

$$Z_2^n(f_1): \sum_{1 \leq i_1+i_2+\ldots+i_n \leq 2} b_{i_1 i_2 \ldots i_n} y_1^{i_1}, \ldots, y_n^{i_n} = 0 \quad (3)$$

   with the constant term $b_{00\ldots0} = 0$. That is

$$Z_2^n(f_1): b_{100\ldots0} y_1 + b_{010\ldots0} y_2 + \ldots + b_{000\ldots1} y_n + \sum_{i_1+i_2+\ldots+i_n=2} b_{i_1 i_2} y_1^{i_1} \ldots y_n^{i_n} = 0 \quad (4)$$

2. Now, there must be at least one nonzero coefficient amongst the linear terms in equation (4). Otherwise the origin is a singular point for the surface and this contradicts the earlier assumption. Without loss of generality, let $b_{100\ldots0} \neq 0$. Then apply the linear transformation

$$\begin{aligned} z_1 &= b_{100\ldots0} y_1 + b_{010\ldots0} y_2 + \ldots + b_{000\ldots1} y_n \\ z_j &= y_j, \qquad j = 2...n \end{aligned} \quad (5)$$

   which makes the $z_1 = 0$, the tangent hyperplane of $Z_2^n(f_1)$ at the origin. This yields

$$Z_2^n(f_2): z_1 + \frac{b_{200\ldots0}}{b_{100\ldots0}^2} z_1^2 + \left[b_{020\ldots0} + \frac{b_{010\ldots0}^2}{b_{100\ldots0}^2} \right] z_2^2$$

$$+\ldots+\left[b_{000\ldots2}+\frac{b_{000\ldots1}}{b_{100\ldots0}^2}\right]z_n^2$$

$$+\sum_{i_1+\ldots+i_n=2,i_k=i_\ell,\quad 1\le i\le k\ne\ell} c_{i_1i_2\ldots i_n}z_1^{i_1}\ldots z_n^{i_n}=0 \quad (6)$$

3. To equation (6), apply the linear coordinate transformation which maps the origin to infinity, along the z_1 axis. Namely,

$$\begin{aligned} z_1 &= \frac{1}{w_1} \\ z_j &= \frac{w_j}{w_1}, \quad j=2,\ldots,n \end{aligned} \quad (7)$$

This yields

$$Z_2^n(f_3): \frac{1}{w_1}+\frac{b_{200\ldots0}}{b_{100\ldots0}^2}\frac{1}{w_1^2}+\left[b_{020\ldots0}+\frac{b_{010\ldots0}^2}{b_{100\ldots0}^2}\right]\frac{w_2^2}{w_1^2}$$
$$+\ldots+\left[b_{000\ldots2}+\frac{b_{0000\ldots}^2}{b_{100\ldots0}^2}\right]\frac{w_n^2}{w_1^2}$$
$$+\frac{1}{w_1^2}\sum c_{i_2\ldots i_n}w_2^{i_2}w_3^{i_3}\ldots w_n^{i_n}=0 \quad (8)$$

4. Clearing the denominator of equation (8 and simplifying the expression for $Z_2^n(f_3)$ yields

$$\begin{aligned} w_1 &= -\frac{b_{200\ldots0}}{b_{100\ldots0}^2}-\sum d_{i_2\ldots i_n}w_2^{i_2}w_3^{i_3}\ldots w_n^{i_n} \\ &= g_2(w_2\ldots w_n) \end{aligned} \quad (9)$$

Hence from transformation (7) above, we obtain

$$\begin{aligned} z_1 &= \frac{1}{g_2(w_2,\ldots w_n)} \\ z_j &= \frac{w_j}{g_2(w_2\ldots w_n)}, \quad j=2,\ldots,n \end{aligned} \quad (10)$$

From transformation (5) we obtain,

$$\begin{aligned} y_1 &= \frac{1-b_{010\ldots0}w_2-b_{001\ldots0}w_3-\ldots-b_{000\ldots1}w_n}{b_{100\ldots0}g_2(w_2\ldots w_n)} \\ y_j &= z_j, \quad j=2,\ldots,n. \end{aligned} \quad (11)$$

Finally from transformation (2) we obtain,

$$x_j=y_j+\alpha_j, \quad j=1,\ldots,n. \quad (12)$$

as rational functions of the parameters $w_2,\ldots,w_n$, a rational parametric representation of the quadric hypersurface.

CUBICS: Consider the general implicit equation of a cubic hypersurface (which is neither a cylinder nor a cone), in $n\ge3$ space

$$Z_3^n(f): \sum_{i_1+i_2+i_n\le3} a_{i_1i_2\ldots i_n}x_1^{i_1}x_2^{i_2}\ldots x_n^{i_n}=0 \quad (13)$$

1. Choose a simple point $(\alpha_1,\alpha_2,\ldots,\alpha_n)$ on $Z_3^n(f)$ and apply the linear coordinate transformation

$$y_j=x_j-\alpha_j, \quad j=1,\ldots,n \quad (14)$$

which translates the hypersurface $Z_3^n(f)$ to pass through the origin. This yields

$$Z_3^n(f_1): \sum_{i_1+i_2+\ldots i_n=1} b_{i_1i_2\ldots i_n}y_1^{i_1}y_2^{i_2}\ldots y_n^{i_n}$$
$$+\sum_{i_1+i_2+\ldots+i_n=2} b_{i_1i_2\ldots i_n}y_1^{i_1}y_2^{i_2}\ldots y_n^{i_n}$$
$$+\sum_{i_1+i_2+\ldots+i_n=3} b_{i_1i_2\ldots i_n}y_1^{i_1}y_2^{i_n}\ldots y_n^{i_n}=0 \quad (15)$$

2. Apply the linear transformation

$$\begin{aligned} z_1 &= b_{100\ldots0}y_1+b_{010\ldots0}y_2+\ldots+b_{000\ldots1}y_n \\ z_j &= y_j, \quad j=1,\ldots,n \end{aligned} \quad (16)$$

which makes $z_1=0$ to be the new tangent hyperplane to the hypersurface at the origin. The hypersurface $Z_3^n(f_1)$ of equation (15) then becomes

$$Z_3^n(f_2): z_1+z_1\sum_{0<i_2+\ldots+i_n\le2} c_{i_2\ldots i_n}z_2^{i_2}\ldots z_n^{i_n}$$
$$+z_1^2\sum_{i_1+\ldots+i_n=1} d_{i_1\ldots i_n}z_1^{i_1}\ldots z_n^{i_n}$$
$$+\sum_{i_2+\ldots+i_n=2} s_{i_2\ldots i_n}z_2^{i_2}\ldots z_n^{i_n}$$
$$+\sum_{i_2+\ldots i_n=3} t_{i_2\ldots i_n}z_2^{i_2}\ldots z_n^{i_n} \quad (17)$$

3. Intersecting the hypersurface $Z_3^n(f_2)$ with the tangent hyperplane $z_1=0$ yields

$$Z_3^{n-1}(f_3): \sum_{i_2+\ldots+i_n=2} s_{i_2\ldots i_n}z_2^{i_2}\ldots z_n^{i_n}$$
$$+\sum_{i_2+\ldots+i_n=3} t_{i_2\ldots i_n}z_2^{i_2}\ldots z_n^{i_n}=0 \quad (18)$$

4. Consider a $\mathbf{u}=(u_1,\ldots,u_k)$, $k\le n-2$, parameter family of lines, passing through the origin and lying in the hyperplane $z_1=0$. These lines are given by

$$\begin{aligned} z_{i+2} &= u_iz_2, \quad 1\le i\le k \\ z_j &= z_2, \quad k<j\le n-2 \end{aligned} \quad (19)$$

5. Intersect these lines given by equation (19) with $Z_3^{n-1}(f_3)$ of equation (17) to yield

$$z_2 = \frac{-\sum_{i_2+\ldots+i_n=2} s_{i_2\ldots i_n} u_1^{i_3}\ldots u_k^{i_k+2}}{\sum_{i_1+\ldots i_n=3} t_{i_2\ldots i_n} u_1^{i_3}\ldots u_k^{i_k+2}} \qquad (20)$$

which together with (19) above yields a parametric representation of $Z_3^{n-1}(f_3)$ in terms of parameters $\mathbf{u} = (u_1, \ldots, u_k)$.

6. Using the linear transformation (14), (16), the parametric representation of $Z_3^{n-1}(f_3)$ and $Z_1 = 0$ we can straightforwardly construct a $\mathbf{u}$ parameterization of $Z_3^{n-1}(f_3)$ in the original space $(x_1, \ldots, x_n)$. Namely

$$x_i = M_i(\mathbf{u}) \qquad i \le i \le n \qquad (21)$$

7. Next choose another simple point $(\beta_1, \beta_2, \ldots, \beta_n)$ on $Z_3^n(f)$ and repeat steps 1., 2., 3. replacing $(\alpha_1, \alpha_2, \ldots \alpha_n)$ with $(\beta_1, \beta_2, \ldots, \beta_n)$. This would yield another $Z_3^{n-1}(\hat{f}_3)$ of similar structure as equation (17), viz.,the intersection of a corresponding hypersurface $Z_3^n(\hat{f}_2)$ with an appropriate tangent hyperplane $\hat{z}_1 = 0$.

8. Analogous to Step 4. above, consider then a $\mathbf{v} = (v_1, \ldots, v_l)$, $l = n - k - 1$, parameter family of lines, passing through the origin and lying in the hyperplane $\hat{z}_1 = 0$. These lines are again given by

$$\begin{aligned} \hat{z}_{j+2} &= v_j \hat{z}_2, \qquad 1 \le j \le l \\ \hat{z}_j &= \hat{z}_2, \qquad l < j \le n-2 \end{aligned} \qquad (22)$$

9. Similar to Steps 5. and 6. above, intersect these lines of equation (22) with $Z_3^{n-1}(\hat{f}_3)$ to derive a $\mathbf{v}$ parametric representation of $Z_3^{n-1}(\hat{f}_3)$ in the original space $(x_1, \ldots, x_n)$. Namely,

$$x_i = N_i(\mathbf{v}) \qquad 1 \le i \le n \qquad (23)$$

10. Finally consider the $(\mathbf{u}, \mathbf{v})$ parameter family of lines in $(x_1, \ldots, x_n)$ space joining points $(M_1(\mathbf{u}), M_2(\mathbf{u}), \ldots, M_n(\mathbf{u}))$ and $(N_1(\mathbf{v}), N_2(\mathbf{v}), \ldots, N_n(\mathbf{v}))$. Namely,

$$\begin{aligned} x_i &= N_i(\mathbf{v}) + \frac{(N_i(\mathbf{v}) - M_i(\mathbf{v}))}{N_1(\mathbf{u}) - M_1(\mathbf{u})}(x_1 - N_1(\mathbf{u})) \\ 1 &\le i \le n \end{aligned} \qquad (24)$$

11. Intersect these lines of equation (24) with the hypersurface $Z_3^n(f)$ to yield

$$f(x_1, \mathbf{u}, \mathbf{v}) = 0 \qquad (25)$$

with degree of x_1 to be at most three, i.e., the lines intersect the hypersurface in at most three distinct intersection points.

12. Two of the intersection points lying on the hypersurface $Z_3^n(f)$ have x_1 values $M_1(\mathbf{u})$, and $N_1(\mathbf{v})$, Hence $\frac{f(x_1,\mathbf{u},\mathbf{v})}{(x_1-M_1)(x_1-N_1)}$ yields an expression which is linear in x_1. Thus $x_1 = R(u, v)$ where R is a rational function in the $l + k = (n-1)$ parameters $\mathbf{u} = (u_1, \ldots, u_k)$, $\mathbf{v} = (v_1, \ldots, v_l)$. Using this together with equation (24) yields a parametric representation of the hypersurface $Z_3^n(f)$ in terms of the $n-1$ parameters $\mathbf{u}$, $\mathbf{v}$.

4 Step II: Polygonalization of Parametric Hypersurfaces

Knowing the parameterization, namely, $(x_1 = G_1(u_1, u_2, \ldots, u_{n-1}), x_2 = G_2(u_1, u_2, \ldots, u_{n-1}), \ldots, x_n = G_n(u_1, u_2, \ldots, u_{n-1}))$ of the $n-1$-dimensional hypersurface $Z_2^n(f)$ or $Z_3^n(f)$ in n space, points on the hypersurface can be straightforwardly generated by substituting parameter values, $\mathbf{u} = (u_1, \ldots, u_k) \epsilon \Re^n$. A generalized net of $n-1$ distinct, intersecting families of lower $n-2$-dimensional hypersurfaces can be obtained, all lying on the original hypersurface, by setting each parameter in turn to be fixed and having the $n-2$ remaining parameters varying. The $n-1$ different choices of the fixed parameter u_i, yields the $n-1$ different families of $n-2$-dimensional hypersurfaces. Recursing on the dimension of the hypersurface, one finally obtains 1-dimensional rational curves lying on the boundary of the hypersurface. By advancing values of each of the parameters u_i, starting from u_0, by small increments Δu_i, one obtains a a piecewise-linear approximation to these curves.

However advancing the parameters u_i naively, may yield quite unsuitable displays. In particular a parametrization of a curve is good if with a constant value Δu_i the points on the curve tend to bunch up in regions of high curvature and spread out in regions of low curvature. Such a reparameterization of the curve, can be obtained by applying the planar methods of [14], suitably generalized to curves in n dimensional space.

Considering all linear, curvature dependent, approximations of the intersecting families of curves, yields a wireframe polygonal complex $\Im$ of line segments. In 3D the hypersurfaces are ordinary two dimensional surfaces and the wireframe complex $\Im$ reduces to a curvature dependent polygonal surface mesh. Using Gouraud shading, available on most graphics workstations one is able to get excellent shaded displays. For four and higher dimensions, appropriate slices are computed of the complex $\Im$ and projected (orthographic or perspective) down to the 2D screen or in 3D for the stereographic 3D displays.

We first explain how the rational parameterization of a hypersurface in 3 space can be displayed smoothly using curvature (and torsion dependent) stepping of the parameters. Consider the rational surface S defined by the parametric equations

$$x = X(s,t), y = Y(s,t), z = Z(s,t)$$

where X, Y, Z are rational functions. A simple way of displaying S is to let s vary from s_i to s_f by a constant step of Δ_s and let t vary from t_i to t_f by a constant step of Δ_t. This creates a rectangular grid of (s,t) points. The surface can be directly polygonalized by evaluating it at each grid point and connecting the grid points together to form polygons. A better way of creating the grid is to step adaptively. Let $\mathbf{X}(t) = [X(s,t), Y(s,t), Z(s,t)]$. Then

$$\Delta_t = \frac{1}{(a\kappa_t + b\tau_t) \parallel \mathbf{X}' \parallel_2} \Delta\phi$$

$$\Delta_s = \frac{1}{(a\kappa_s + b\tau_s) \parallel \mathbf{X}' \parallel_2} \Delta\phi$$

Here κ and τ are the curvature and torsion of the surface, respectively, and are defined by

$$\kappa = \frac{\sqrt{(\mathbf{X}' \cdot \mathbf{X}')(\mathbf{X}''.\mathbf{X}'') - (\mathbf{X}'\mathbf{X}'')^2}}{(\mathbf{X}' \cdot \mathbf{X}')^{\frac{3}{2}}}$$

$$\tau = \frac{|[\mathbf{X}', \mathbf{X}'', \mathbf{X}''']^T|}{(\mathbf{X}' \cdot \mathbf{X}')(\mathbf{X}'' \cdot \mathbf{X}'') - (\mathbf{X}' \cdot \mathbf{X}'')^2}$$

To get Δ_t, all derivatives are performed with respect to t, and for Δ_s, with respect to s. Given some pair (s_0, t_0), to step along t, we compute Δ_t by evaluating the formula above at $s = s_0, t = t_0$, and likewise to step along s. One can use constant-stepping in one variable and adaptive stepping in the other, or adaptive stepping in both. The latter approach is more expensive, but we find it produces smoother-looking surfaces. We used the following stepping process. The algorithm below fills the given grid with n^2 (s,t) pairs. Stepping along s and t starts at s_0 and t_0 respectively.

```
makegrid(grid, s0, t0, n)
{
local s, t, i, k;
   /* initialize row and column 1 */
   grid(1,1) ← (s0, t0);
   for i := 2 to n do {
       s ← grid(1, i − 1).s;   t ← grid(1, i − 1).t;
       grid(1, i) ← (s, t + Δt(s, t));
       s ← grid(i − 1, 1).s;   t ← grid(i − 1, 1).t
       grid(i, 1) ← (s + Δs(s, t), t)
   }
   /* initialize rows and columns, diagonally*/
   for k := 2 to n do {
       /* row k */
       for i := k to n do {
           s ← grid(k, i − 1).s;   t ← grid(k, i − 1).t
           grid(k, i) ← (s, t + Δt(s, t))
       }
       /* col k */
       for i := k + 1 to n do {
           s ← grid(i − 1, k).s;   t ← grid(i − 1, k).t;
           grid(i, k) ← (s + Δs(s, t), t)
       }
   }
}
```

See appendix A and the figures 1. -4.. Figs 1. and 2. are for a hyperboloid of one sheet, with equal parameter ranges $-3 \leq s,t \leq 3$. However, fig 1. uses constant $s-t$ stepping of 0.1 requiring 3600 polygons to display, while fig 2. using adaptive stepping requires only 900 polygons for an equally good shaded display. Figs 3. and 4. are for a parabolic hyperboloid, with equal parameter ranges $-3 \leq s,t \leq 3$. However, fig 3. uses constant $s-t$ stepping of 0.1 requiring 3600 polygons to display, while fig 4. using adaptive stepping requires only 840 polygons for an equally good shaded display.

Hypersurfaces S in 4D space are three dimensional solids. Their displays are generated in the following manner. Fixing each of the three parameters, one at a time, in the rational parameterization of S, yields three distinct families of rational two-dimensional surfaces in 4D, which are polygonalized by the previous method. These surfaces in 4D are then projected (both orthographic and in perspective) down to 3D and then rendered as before. In the pictures at the end of the paper, the projected surfaces were not shaded, in order to be able to better visualize the combination of the three families of surfaces meshing together to yield the hypersurface. See appendix A and the figures. Figs. 5. , 6. , and 7. are the three distinct projected (perspective) families of surfaces (for s fixed, t fixed and u fixed, respectively), covering the parabolic hyperboloid hypersurface. Fig. 8. shows the composite projected display of the three surface family covering of the hypersurface. Fig. 9. shows the composite projected (perspective) display of the three surface family covering of the nodal cubic hypersurface. Fig. 10. shows a single projected (orthographic) family of surfaces lying on the hypersphere, together with the same family rotated by 30 degrees in the $z-w$ plane in 4D space. Fig. 11. shows a similar sequence of a single projected (orthographic) family of surfaces lying on the hyperboloid hypersurface, together with the same family rotated by 45 degrees in the $x-y$ plane in 4D space.

By similar methods, one can easily construct displays of projections (unfortunately) of hypersurfaces in higher dimensions. Of course, similar to the people of flatland [1], our visualization abilities of the boundary of the higher dimensional hypersurfaces, are severely lacking.

5 Conclusions and Future Research

We have presented algorithms for parametrizing and displaying quadric and cubic hypersurfaces for any dimension ≥ 3. The methods detailed in sections 3. and 4. were implemented in a combination of Common Lisp and C on a SUN 4-110. (Common Lisp for the polynomial symbolic manipulations and C for the numerical calculations). The graphics displays were generated on a Tektronix 4337 fitted with a stereoscopic 3D screen.

There exist other rendering alternatives and variations to these basic methods. Clearly, raytracing together with pattern and texture mapping, of bounded projected patches of these hypersurfaces are some of the unexplored alternatives. Algorithms for parametrizing and thereby displaying general quartics and higher degree hypersurfaces are as yet unknown. Deriving such parametrizations, for their use in scientific visualization and other manipulations, is also a keen area of future research with a number of open algorithmic problems.

Acknowledgements: I sincerely thank Insung Ihm and Andrew Royappa for the many hours they spent in front of the graphics workstation, helping me visualize in four and higher dimensional space.

References

[1] Abbot, E., (1952) *Flatland*, Dover Publications, New York.

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

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

[4] Abhyankar, S., and Bajaj, C., (1988), "Automatic Rational Parameterization of Curves and Surfaces III: Algebraic Plane Curves", *Computer Aided Geometric Design*, 5, 308 - 321.

[5] Abhyankar, S., and Bajaj, C., (1989), "Automatic Rational Parameterization of Curves and Surfaces IV: Algebraic Spavce Curves", To appear in *ACM Transactions on Graphics*, (1989).

[6] Bajaj, C., (1989), "Geometric Modeling with Algebraic Surfaces", To appear in *The Mathematics of Surfaces III*, Oxford University Press.

[7] Clark, J., (1979) "A Fast Algorithm for Rendering Parametric Surfaces" *Tutorial on Computer Graphics: Image Synthesis*, ed., K. Joy, C. Grant, N. Max, and L. Hatfield, 1988, 88 - 93.

[8] Burton, R.P. and D.R. Smith, (1982), "A Hidden Line Algorithm for Hyperspace", *SIAM Journal on Computing*, 11, 1, 71-80.

[9] Gordon,
W., and Riesenfeld, R., (1974) "Bernstein-Bezier Methods for Computer Aided Geometric Design of Free Form Curves and Surfaces", *Journal Of ACM*, 21, 2, 293 -310.

[10] Lane, J., Carpenter, L., whitted, T., and Blinn, J., (1980) "Scan line Methods for Parametrically Defined Surfaces", *Communications of the ACM*, 23, 1, 23 - 43.

[11] Levin, J., (1979), "Mathematical Models for Determining the Intersection of Quadric Surfaces", *Computer Graphics and Image Processing*, 11, 73 - 87.

[12] Noll, A.M., (1967), "A Computer Technique for Displaying n-Dimensional Hyper Objects", *Communications of the ACM*, 10, 8, 469-473.

[13] Ocken, S., Schwartz, J., and Sharir, M., (1983), "Precise Implementation of CAD Primitives using Rational Parameterizations of Standard Surfaces", *Planning, Geometry and Complexity of Robot Motion*, Ablex Publishing, Norwood, N.J., 245–266.

[14] Patterson, M., and Bajaj, C., (1989), "Curvature Adjusted Parameterization of Curves", Purdue University, Computer Science Technical Report, CER-89-18.

[15] Salmon, G., (1914), "*A Treatise on the Analytic Geometry of Three Dimensions*", vol I, II, ed., R. Rogers. Reprinted, Chelsea Publishing.

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

[17] Somerville, D., (1958) *An Introduction to the Geometry of N Dimensions*, Dover Publications, N.Y.

[18] Snyder, V., Coble, A., Emch, A., Lefschetz, Sharpe, F., and Sisam, C., (1928), *Selected Topics in Algebraic Geometry*, Reprinted, Chelsea Publishing 1970.

[19] Zariski, O., (1935), *Algebraic Surfaces*, Ergebnisse der Mathematik und ihre Grenzgebiete.

A Appendix: Examples

Fig	Hypersurface	Implicit Equation	Parametric Equation
1,2	Hyperboloid	$x^2 + y^2 - z^2 - 1 = 0$	$x = \frac{t^2 - 2*s*t-1}{t^2+1}$, $y = \frac{2*t+s(t^2-1)}{t^2+1}$, $z = s$
3,4	Parahyperboloid	$x^2 - y^2 - 2 * z = 0$	$x = s$, $y = t$, $z = \frac{1}{2} * (s^2 - t^2)$
5-8	Hyper-Parahyperboloid	$x^2 - y^2 - z^2 - 2 * w = 0$	$x = s$, $y = t$, $z = u$, $w = \frac{1}{2} * (s^2 - t^2 - u^2)$
9	Hyper-Nodal Cubic	$x^2 - y^2 - z^3 = 0$	$x = s * (s^2 - t^2)$, $y = t * (s^2 - t^2)$ $z = s^2 - t^2$, $w = u$
10	Hyper-sphere	$x^2 + y^2 + z^2 + w^2 - 1 = 0$	$x = \frac{s^2+t^2+u^2-1}{s^2+t^2+u^2+1}$, $y = \frac{2*s}{s^2+t^2+u^2+1}$ $z = \frac{2*t}{s^2+t^2+u^2+1}$, $w = \frac{2*u}{s^2+t^2+u^2+1}$
11	Hyper-hyperboloid	$x^2 + y^2 + z^2 - w^2 - 1 = 0$	$x = \frac{-s^2+t^2+u^2-1}{s^2-t^2-u^2-1}$, $y = \frac{-2u}{s^2-t^2-u^2-1}$ $z = \frac{-2t}{s^2-t^2-u^2-1}$, $w = \frac{-2s}{s^2-t^2-u^2-1}$

Color images for this paper can be found in the color plate section.

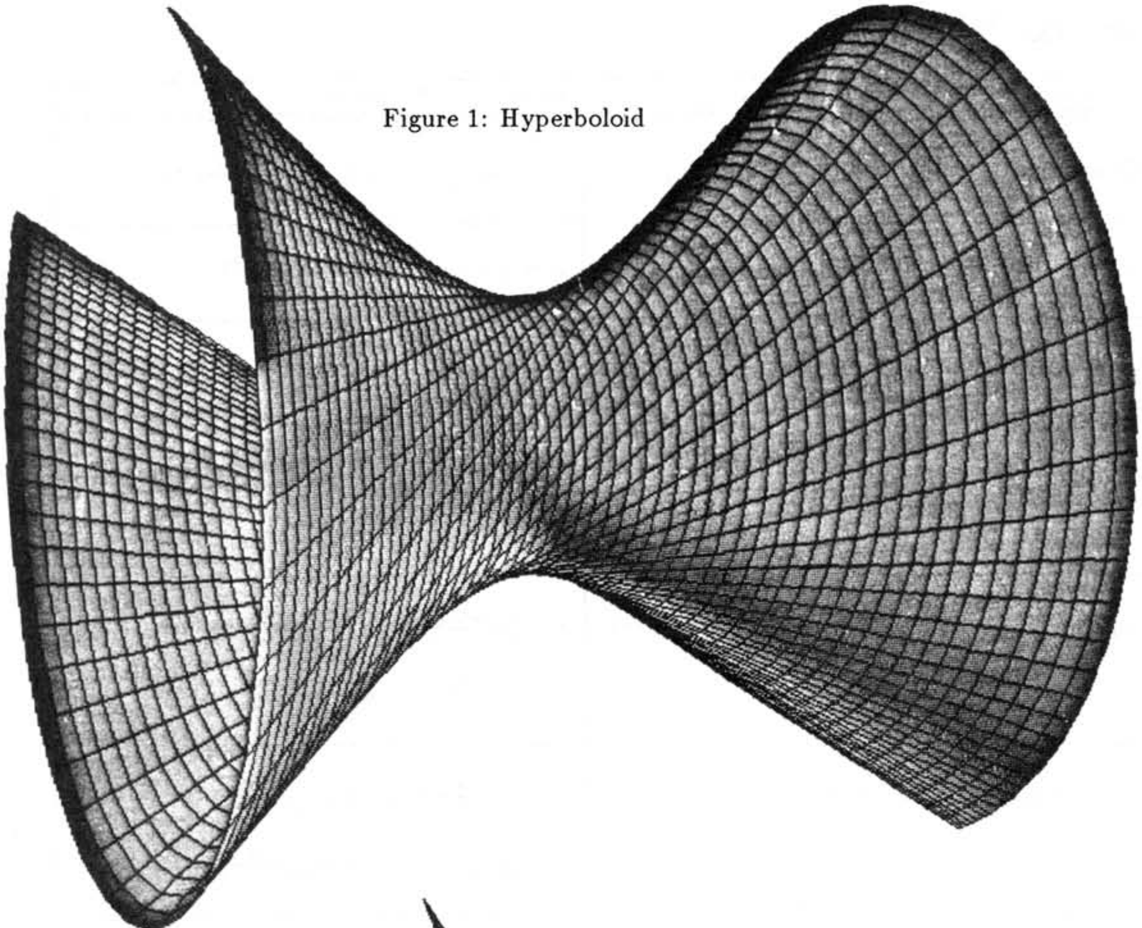


Figure 1: Hyperboloid

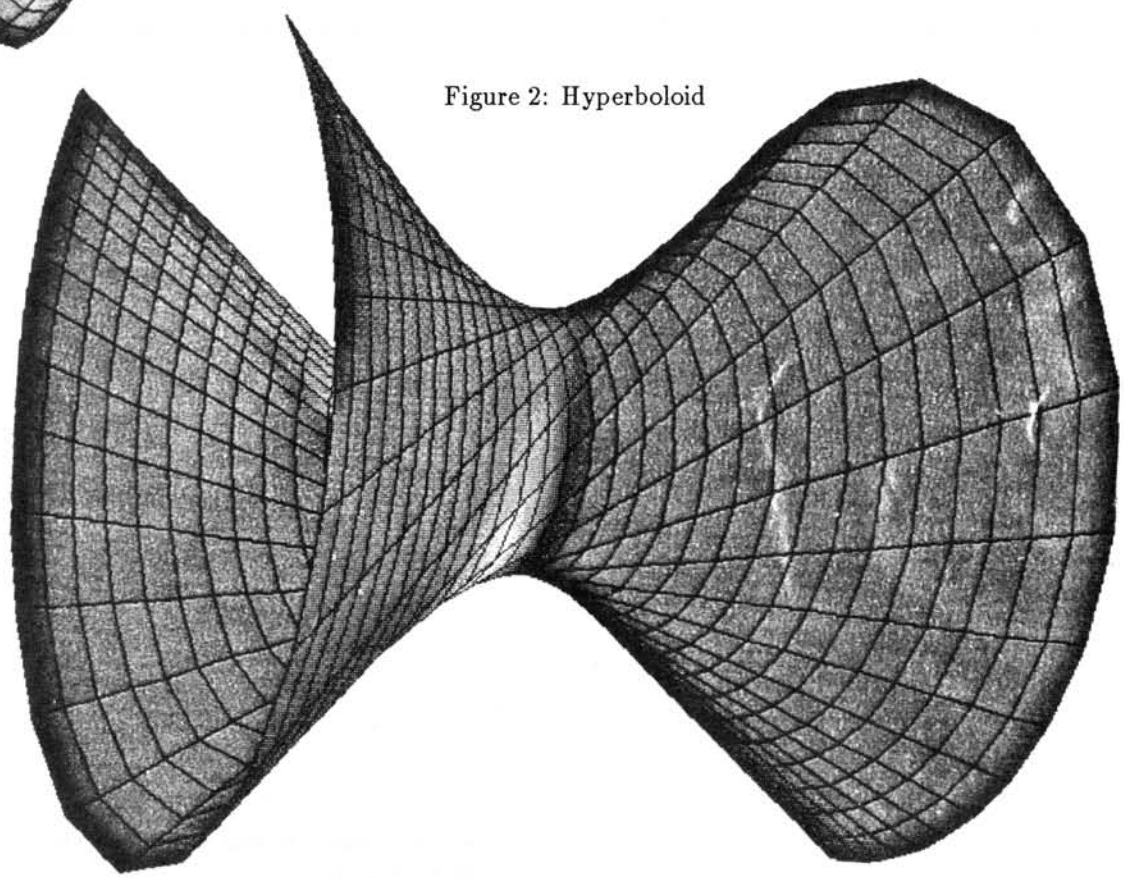


Figure 2: Hyperboloid

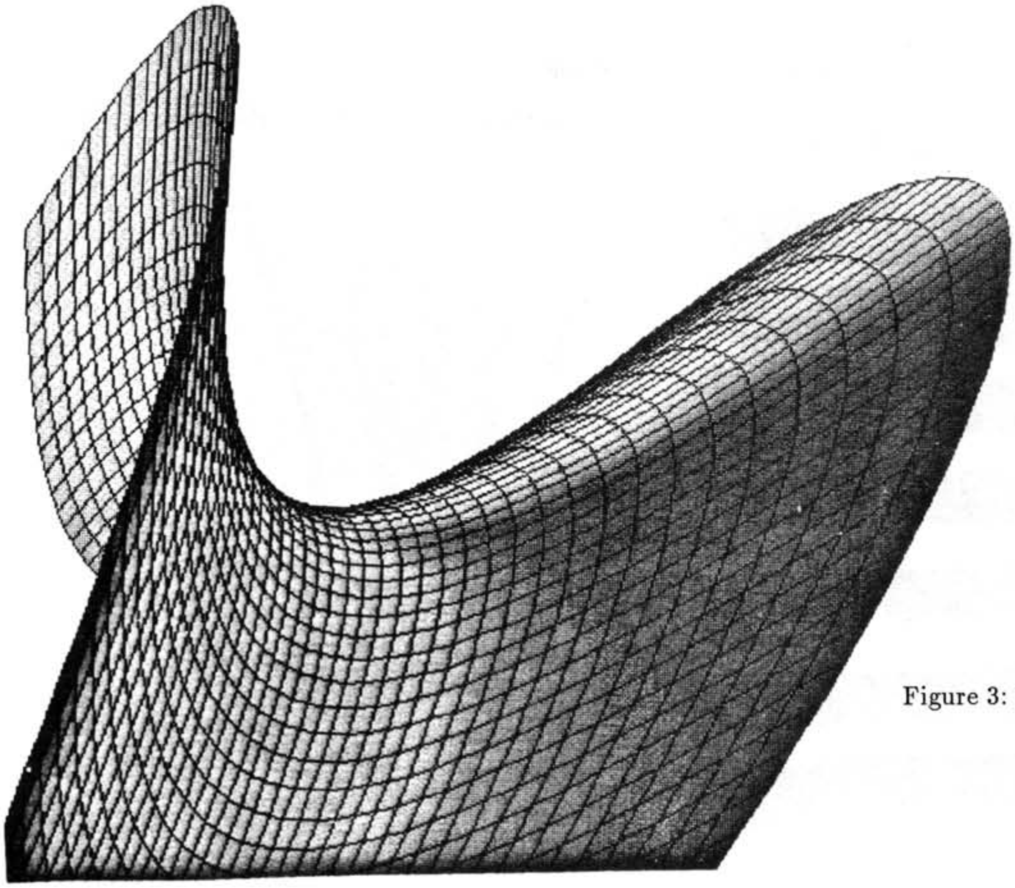


Figure 3: Parahyperboloid

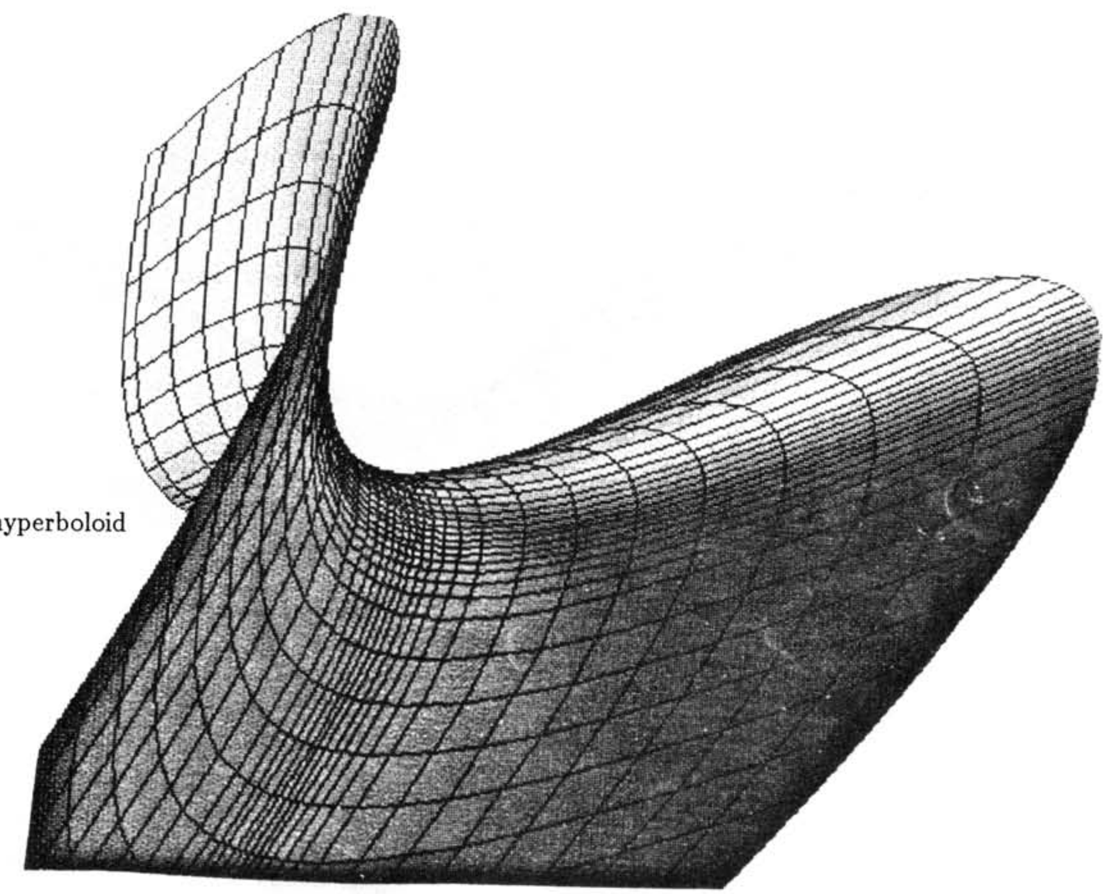


Figure 4: Parahyperboloid

Figure 5: Hyper-Parahyperboloid

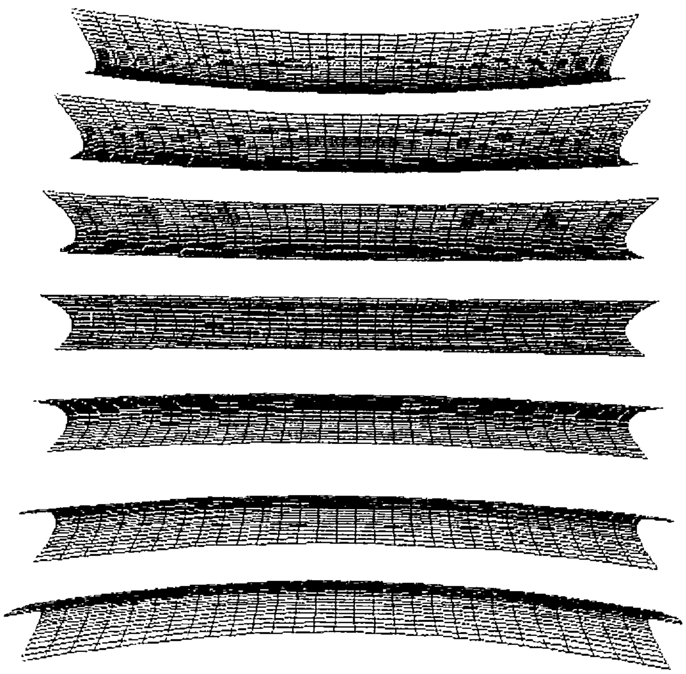


Figure 6: Hyper-Parahyperboloid

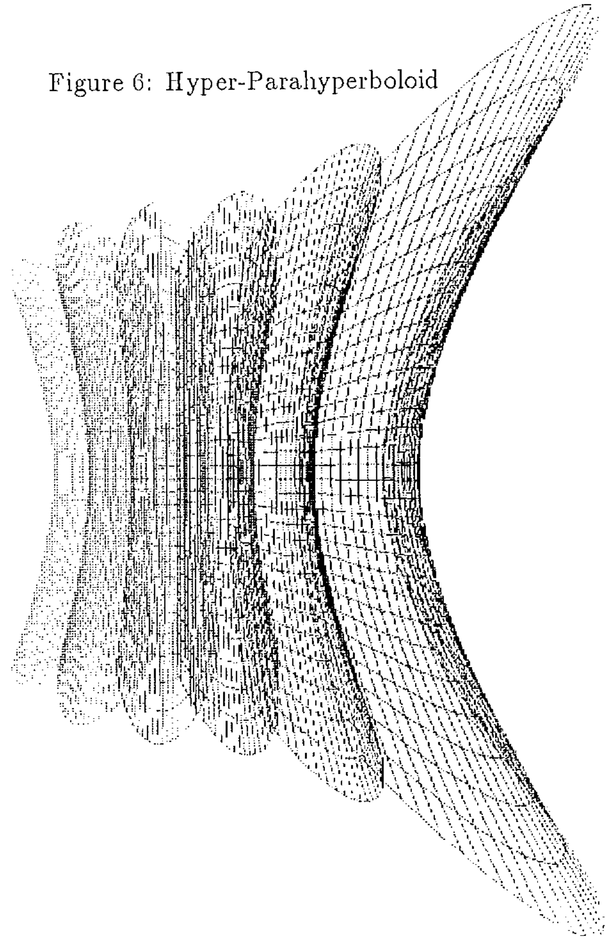
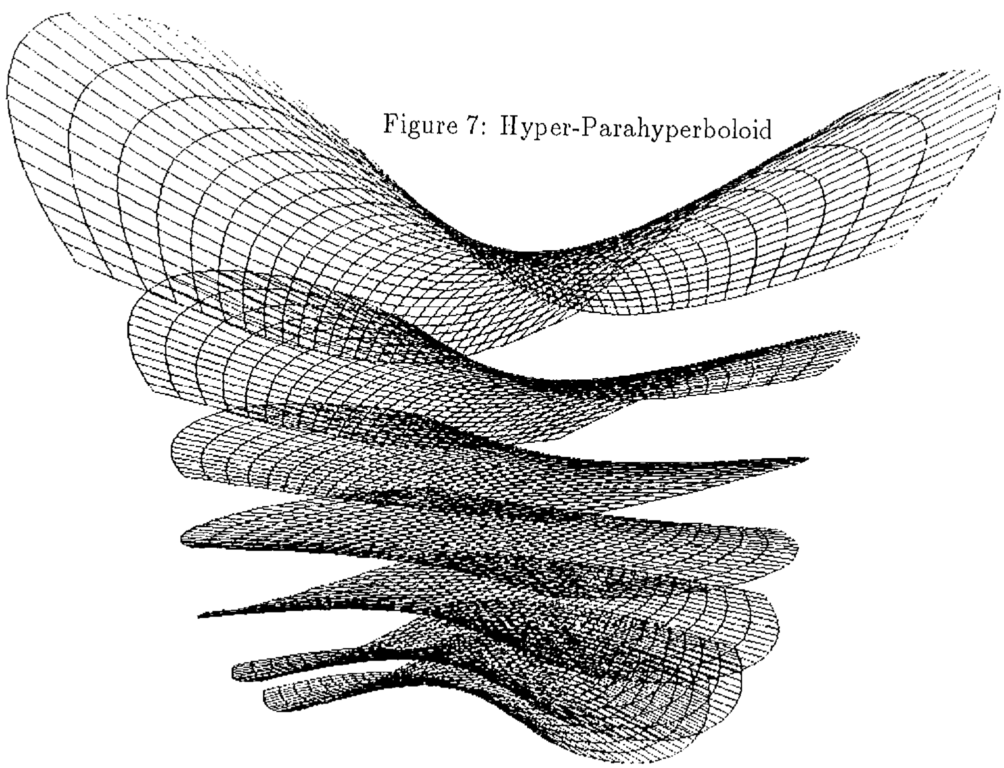


Figure 7: Hyper-Parahyperboloid



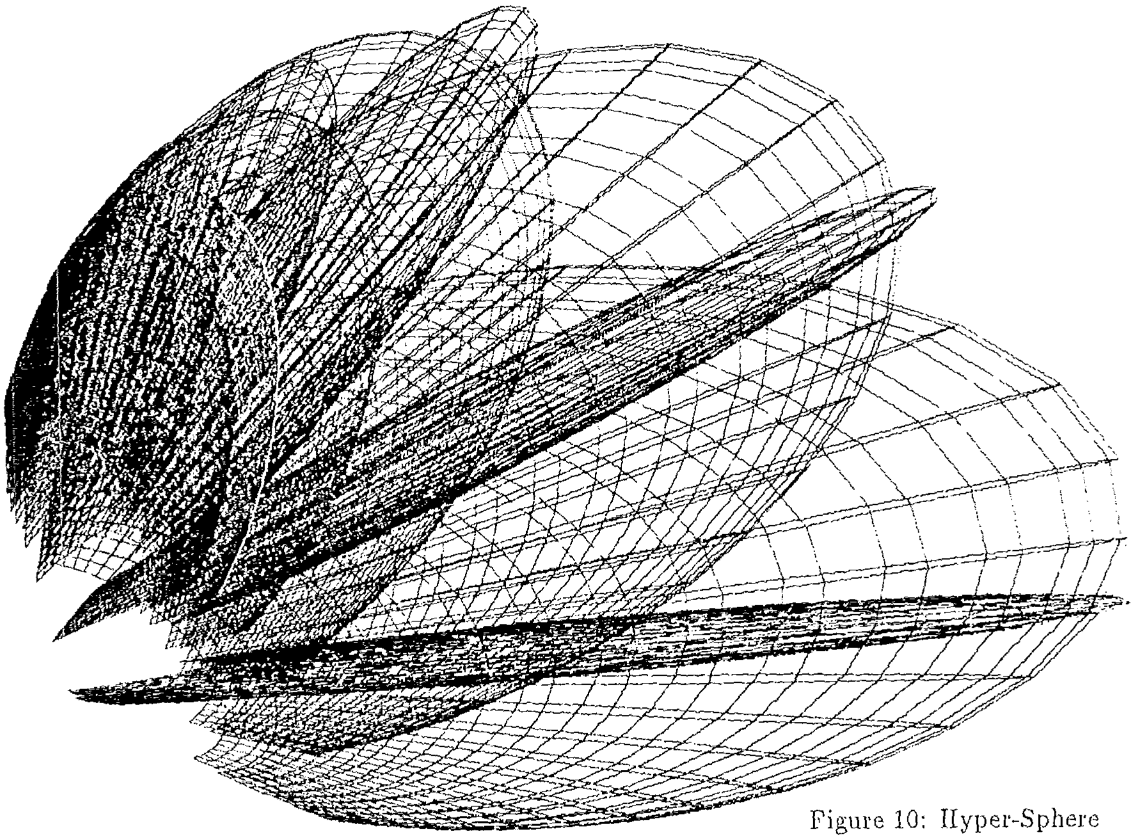


Figure 10: Hyper-Sphere

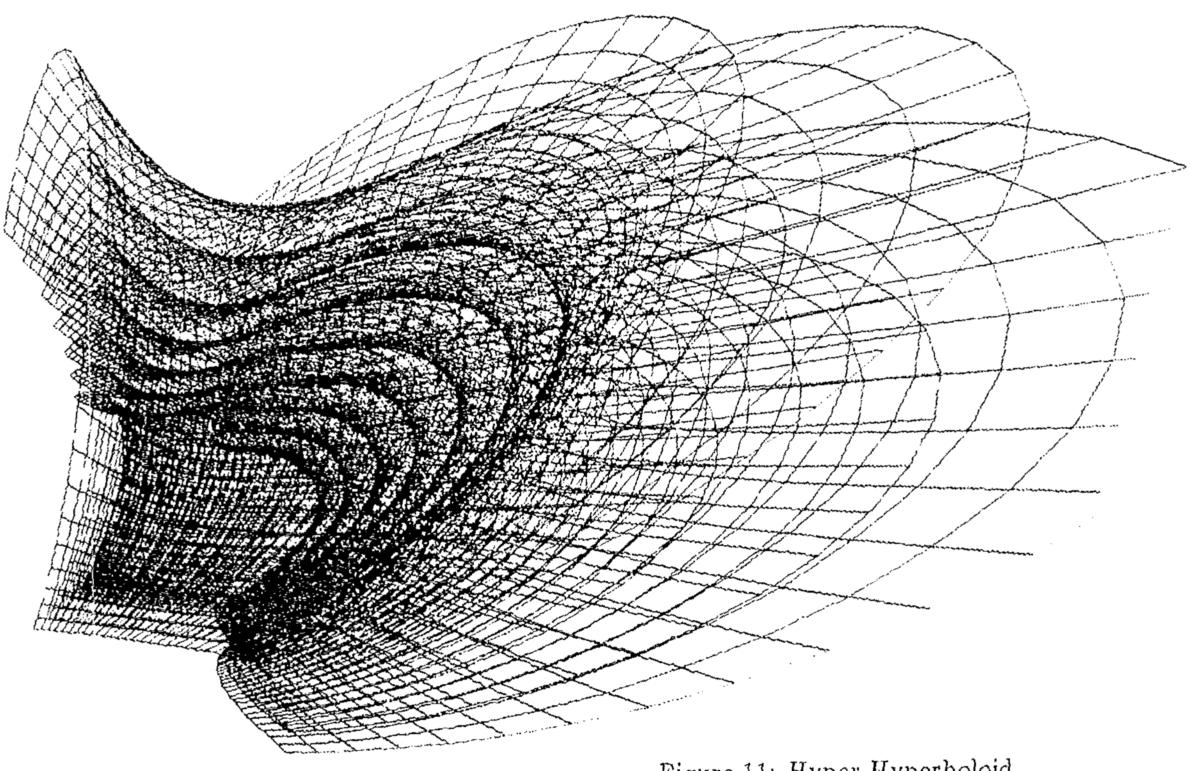


Figure 11: Hyper-Hyperboloid

Figure 8: Hyper-Parahyperboloid

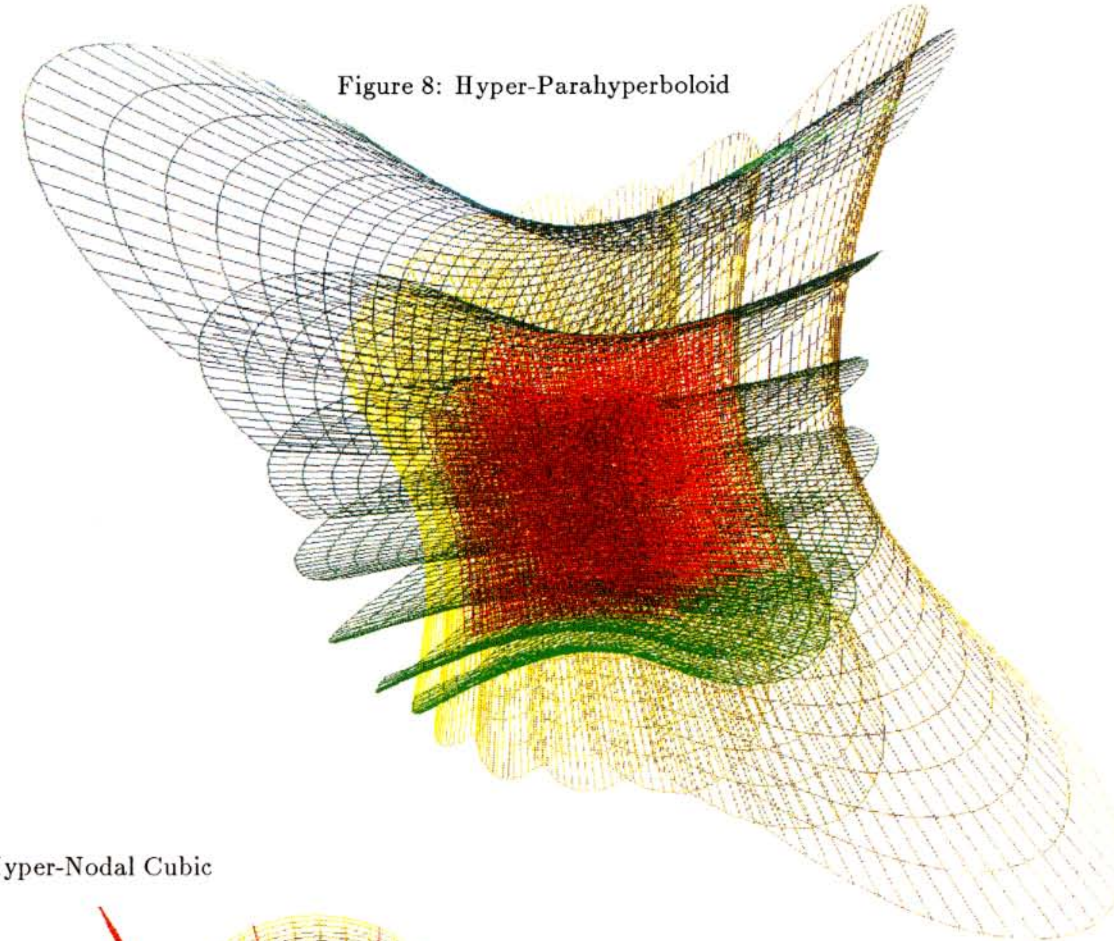
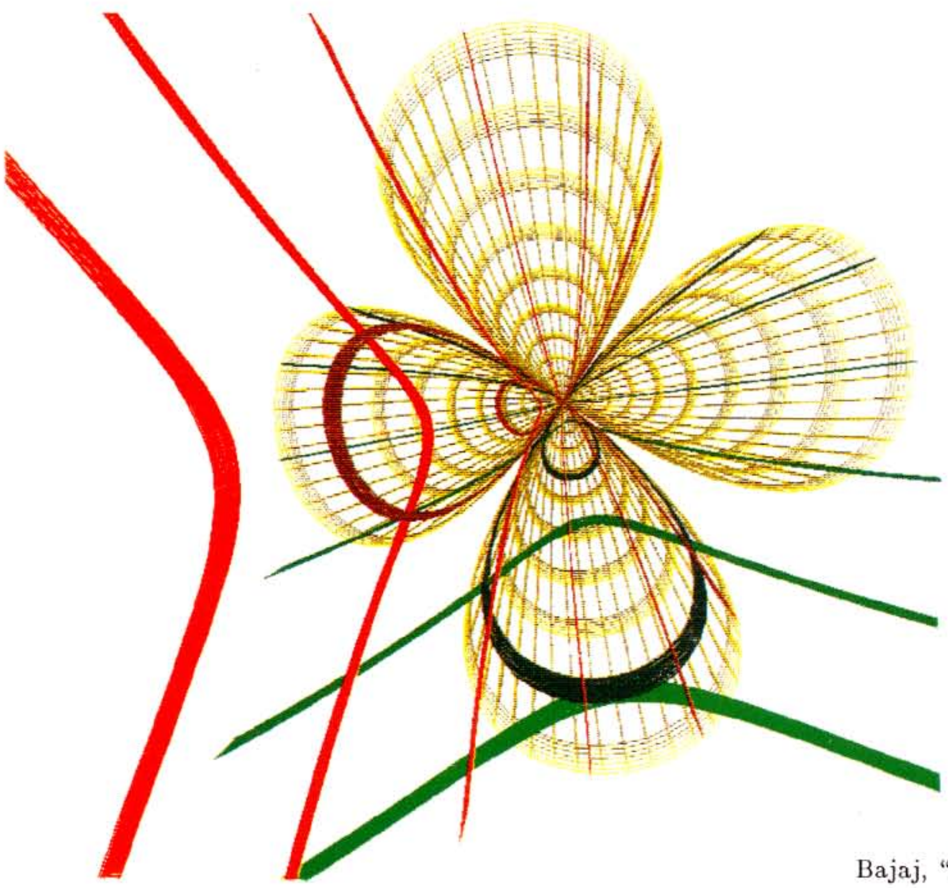


Figure 9: Hyper-Nodal Cubic



Bajaj, "Rational Hypersurface Display".