

MULTISTAGE TRIVARIATE SURFACES

ROBERT E. BARNHILL AND SARAH E. STEAD

ABSTRACT. We construct four-dimensional surfaces that approximate arbitrarily placed information. The basic strategy is

(1) interpolate to the arbitrarily placed data with a general method,

(2) then evaluate this general method over a coarse rectilinear grid to provide data for a tensor product method, and

(3) render the surface by evaluating the tensor product over a fine grid.

Graphical illustrations are also included.

1. Multivariate interpolation.

A. Introduction. Four-dimensional "surfaces" occur often in science and engineering; for example, temperature in a furnace as a function of three spatial variables. For concreteness, let us start with data of the form $\{(x_i, y_i, z_i; F_i)\}_{i=1}^n$ where we assume a functional relationship of the form $F_i = F(x_i, y_i, z_i)$ (i.e., F is a trivariate function).

How do we interpolate to these data? There are two classes of methods for trivariate surface interpolation: 1) tetrahedral interpolation, and 2) distance-weighted interpolation. Tetrahedral interpolants (i.e., interpolants defined over tetrahedra) are discussed by Barnhill and Little and by Alfeld in this volume. Distance-weighted interpolants are discussed in this paper.

Surfaces need to be at least C^1 for most practical purposes. Since only positional information is usually available, the gradients must be created for most methods. This problem is addressed by Akima and by Stead, among others, in this volume.

Barnhill [1] gives a survey of three-dimensional surface methods. This survey is augmented by more recent developments in [2].

Keywords and phrases: Surfaces, Four-dimensional Surfaces, Interpolation, Boolean Sums, Delta Sums, Shepard's Formula, Hardy's Multiquadrics, Computer Graphics, Distance-weighted Interpolants.

This research was supported in part by the National Science Foundation with Grant MCS-8101854 and by the Department of Energy with Contract No. DE-AC02-82ER-12046.A000 to The University of Utah.

B. Shepard's formula. A convenient starting point for distance-weighted interpolants is Shepard's Method [14]. Shepard's Method was introduced for bivariate interpolation but, since its structure depends only on Euclidean distances, it can be generalized immediately to multivariate interpolation. Since our goal is to create C^1 four-dimensional surfaces, we state the Shepard's Formula which uses C^1 data; for point of evaluation P ,

$$(1.1) \quad S_1 F(P) = \sum_{i=1}^n w_i(P) G_i F(P)$$

where the weight functions w_i are defined by

$$w_i(P) = \frac{1}{[d_i(P)]^2} \bigg/ \sum_{j=1}^n \frac{1}{[d_j(P)]^2}$$

$$d_j(P) = \text{Euclidean distance } d(P, P_j)$$

and the Taylor operators G_i are defined by

$$G_i F(P) = \text{linear Taylor expansion of } F \text{ at } P \text{ about } P_i$$

$$= F(P_i) + (x - x_i)F_{1,0,0}(P_i) + (y - y_i)F_{0,1,0}(P_i) + (z - z_i)F_{0,0,1}(P_i),$$

where $P = (x, y, z)$ and $P_i = (x_i, y_i, z_i)$. The weight functions w_i can be rewritten in a more numerically stable form as

$$(1.2) \quad w_i(P) = \prod_{\substack{k=1 \\ k \neq i}}^n [d_k(P)]^2 \bigg/ \sum_{j=1}^n \prod_{\substack{l=1 \\ l \neq j}}^n [d_l(P)]^2.$$

The distance ("metric") $d_j(P)$ is defined by

$$d_j(P) = \{(x - x_j)^2 + (y - y_j)^2 + (z - z_j)^2\}^{1/2}.$$

(This notation for the distance function is standard for our subject, the alternative topological notation being $d(P, P_j)$.)

We notice that Shepard's Formula (1.1) is a convex combination of the data $G_i F$, i.e., the weight functions w_i are non-negative and sum to one. Convex combinations have many useful properties including numerical stability and max/min principles. (For more on convex combinations see [11].) We state a (new) max/min principle for $S_1 F$ after we recall the analogous max/min principle for the original Shepard's Formula

$$S_0 F = \sum_i w_i F_i;$$

$$\min_{1 \leq i \leq n} F_i \leq S_0 F \leq \max_{1 \leq i \leq n} F_i.$$

The max/min principle for $S_1 F$ is

$$(1.3) \quad \min_{1 \leq i \leq n} G_i F(P) \leq S_1 F(P) \leq \max_{1 \leq i \leq n} G_i F(P).$$

A property that is different between $S_0F(P)$ and $S_1F(P)$ is their asymptotic behavior as $P \rightarrow \infty$. Gordon and Wixom [9] showed that $\lim_{P \rightarrow \infty} w_i(P) = 1/n$, $i = 1, \dots, n$, which implies that

$$\lim_{P \rightarrow \infty} S_0F(P) = \frac{1}{n} \sum_{i=1}^n F_i \quad \text{and} \quad \lim_{P \rightarrow \infty} S_1F(P) = \infty$$

(unless $S_1 = S_0$) the latter following from the fact that each $G_i(P)$ is a polynomial.

The Shepard weight functions w_i have the property that

$$(1.4) \quad D^{p,q,r}w_i(P_j) = \begin{cases} \delta_{ij} & \text{for } p = q = r = 0 \\ 0 & \text{for } 0 < p + q + r < 2 \end{cases}$$

for $i, j = 1, \dots, n$. The properties of the w_i expressed by (1.4) imply that

$$D^{p,q,r}S_1F(P_j) = D^{p,q,r}G_jF(P_j) = D^{p,q,r}F(P_j),$$

the second equality following from the definition of G_jF at (1.1). Actually, equation (1.4) is true for all p, q, r such that $0 < p + q + r < \mu$ if d_i^2 is replaced by d_i^μ in (1.1). This opens up more possibilities; generalize G_iF to involve higher order derivatives, but satisfy the condition that the (total) order of G_iF 's derivatives is less than μ . A proof of (1.4) for bivariate w_i is given in [3]. The extension to trivariate w_i is immediate because the w_i only depend upon Euclidean distances.

C. Hardy's multiquadrics. The trivariate form of Stead's [15] generalization of Hardy's [10] "multiquadric" is

$$(1.5) \quad M_\mu F(P) = \sum_{i=1}^n c_i \{[d_i(P)]^2 + r\}^{\mu/2}$$

where μ is nonzero, $d_i(P)$ is the distance from P to P_i (as before) and r is positive. The c_i are chosen so that $M_\mu F$ interpolates;

$$(1.6) \quad M_\mu F(P_i) = F_i, \quad i = 1, \dots, n.$$

It is an open research question to determine conditions so that this interpolation problem has a solution.

The function precision of an interpolant is the span of its basis functions. So M_μ is precise for the functions $\{[d_i(P)]^2 + r\}^{\mu/2}, i = 1, \dots, n$. Thus M_μ is unusual in that M_μ has no polynomial precision, but M_μ is smooth, being in $C^\infty(\mathbf{R}^3)$.

D. Localization of global methods. Shepard's Formula and Hardy's Multiquadrics are both global methods, i.e., a change in one data value F_i affects the approximation everywhere. A global interpolant of the form $\sum_i \phi_i L_i F$ can be localized as follows.

(1) **Franke-Little weights.** We replace $\phi_i = \phi_i(P)$ by the function ϕ_i defined as

$$\phi_i \equiv \hat{\phi}_i / \sum_i \hat{\phi}_i$$

where

$$\hat{\phi}_i = \hat{\phi}_i(P) = \phi_i(P) \left(1 - \frac{d_i(P)}{R_i}\right)_+^m$$

and

$$x_+^m \equiv \begin{cases} x^m & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases},$$

and R_i is a “radius of influence” of P_i , e.g., the maximum distance from P_i to a specified number of nearest neighbors P_j . For many interpolants, including Shepard’s and Hardy’s, the ϕ_i are non-negative so that the $\hat{\phi}_i$ are non-negative and their sum is positive. Then the smoothness of ϕ_i is the smoothness of $\hat{\phi}_i$ intersected with C^{m-1} since the monospline $x_+^m \in C^{m-1}$. The ϕ_i are known as the Franke-Little weights.

(2) **Truncated weights.** We use only a specified number of the nearest points of interpolation, at a point of evaluation. This amounts to truncating the weight functions ϕ_i and yields a C^{-1} scheme. However, it is simple and sometimes effective.

2. Multistage methods.

A. Introduction. Schumaker [13] pointed out the possibility of “two-stage” methods. These are of the form PQ which is the composition of the operator P with Q . An example is Q is Shepard’s Method S and P is (tensor product) piecewise tricubic Hermite interpolant H . The compositions HS provides a C^1 approximation which can be evaluated efficiently, but which does not interpolate to the given arbitrarily placed data. (Actually, tensor products themselves are a simple example of a two-stage method, since a tensor product is the composition of linear operators P_x and P_y which operate on functions of x and y , respectively.)

Why would one bother with a two-stage method? That is, why not simply use a method such as S_1 or M_μ alone? The answer includes the following ingredients. Methods for arbitrarily spaced data are usually global, both S_1 and M_μ being examples. To become practical, they are localized as in §1D. However, their localized forms are expensive to compute, because of sorting the data points. Also, they may be discontinuous as the truncated weight functions in §1D(2). Finally, rendering a surface requires many evaluations and these evaluations are frequently on rectangular grids. So we use the following idea.

1. Interpolate to the arbitrary data with a general surface interpolant.
2. Then define a local tensor product method over a coarse grid, obtaining the grid values from the surface in step one.
3. Evaluate the tensor product surface on a fine grid for rendering the surface.

We call this procedure a "multistage" method.

We now develop some candidates for step one, the general surface method.

B. Boolean sums. Another example of a two-stage method is a Boolean sum $P \oplus Q \equiv P + Q - PQ$. Barnhill and Gregory [5] investigated the interpolation and precision properties of P and of Q . This search led to the Barnhill/Gregory Theorem which says, "the Boolean sum $P \oplus Q$ has (at least) the interpolation properties of P and (at least) the function precision of Q ." (They were constructing complicated triangular interpolants which led to the creation of additional useful theorems about Boolean sums. We are only using their simplest and most basic result here.)

This theorem can be used in (at least) two ways.

(1) Take the Boolean sum of an interpolation scheme P which does interpolate but has poor polynomial precision with a scheme Q which has the desired polynomial precision but need not interpolate.

(2) Build up a scheme from simpler schemes analogously to the Newton form of the (univariate) interpolating polynomial. If P_n is the interpolating polynomial operator to $n + 1$ data, then

$$\begin{aligned} P_n \oplus P_{n-1} &= P_n + P_{n-1} - P_n P_{n-1} \\ &= P_n + P_{n-1} - P_{n-1} \\ &= P_n \end{aligned}$$

because $P_{n-1}F$ is in the precision set of P_n . We may view $P_n = P_n \oplus P_{n-1} = P_{n-1} + (I - P_n)P_{n-1}$ as P_{n-1} plus a correction term. Using Davis' [6] language, this is a permanence principle.

Barnhill and Farin [4] used this idea to build up an explicit representation of a (bivariate) C^1 polynomial defined over a triangle.

Poeppelmeier [12] built up interpolants to arbitrary bivariate data by taking the Boolean sum $S \oplus B$ where S is Shepard's Formula and B is the piecewise defined Barnhill-Gregory nine-degrees-of-freedom C^1 triangular interpolant. Poeppelmeier's scheme is an instance of both (1) and (2) above. We state the general result as follows. Let $SF = \sum_{i \in \alpha} u_i L_i F$ and $BF = \sum_{j \in \beta} v_j M_j F$, where SF and BF are both in cardinal form, i.e., S interpolates to the linear functionals L_i with basis functions u_i , i in an index set α and M_j , v_j and β are analogous. Then the Boolean sum $(S \oplus B)F$ has the following properties.

- (1) $S \oplus B$ has at least the interpolation properties of S ,
- (2) $S \oplus B$ has at least the function precision of B , and
- (3) $S \oplus B$ is of the (simpler) form

$$(S \oplus B)F = BF + \sum_{i \in \alpha - (\alpha \cap \beta)} u_i [L_i F - L_i B F].$$

A simpler example is S equal to Shepard's Formula S_0 for four points and L equal to linear interpolation for the first three of the four points. Then $(S_0 \oplus L)F = LF + w_4[F_4 - (LF)_4]$ where w_4 is the fourth Shepard weight function and G_4 means evaluation of the function G at the fourth point. We observe that the Boolean sum $S_0 \oplus L$ interpolates at all four points, has linear polynomial precision, and its computation involves the addition of only one term to the (simpler) function LF , which is a kind of permanence principle.

A second example is Poeppelemeier's of S_0 and the Barnhill/Gregory scheme B defined over a subset of the points that define S_0 . The Boolean sum $(S_0 \oplus B)F$ is C^1 , in addition to having properties (1)–(3) above. This smoothness is an instance of the following general result.

SMOOTHNESS OF BOOLEAN SUMS. The smoothness of a general Boolean sum $P \oplus Q$ is as follows. Let $PF \in C^p$ and $QF \in C^q$ for all relevant F . Assume that PQF (or at least $P(I - Q)F$) can be formed. Then $(P \oplus Q)F \in C^{\min(p, q)}$.

Boolean sums are both powerful and interesting. However, both P and Q in $P \oplus Q$ operate on the same kind of data as can be seen from the basic definition of Boolean sum. The example of Shepard's Formula and Hermite Interpolation in §2A could not be created directly by a Boolean sum.

C. Delta sums. Foley in [7] and in this volume added a new ingredient to bivariate two-stage methods, which we extend to provide an answer to the non-comparable data problem of the preceding subsection. We recall the first example of §2 of Shepard's Method S and piecewise tricubic Hermite interpolation H . We cannot apply H to the arbitrary data because H 's data must be specified on a rectangular grid. Hence we cannot form the Boolean sums $S \oplus H$ or $H \oplus S$. Foley's idea, applied to this example, is the following scheme.

<i>Procedure:</i>	<i>Approximation so far:</i>
1. Interpolate with S .	S
2. Smooth with H .	HS
3. Add correction term $S(I - HS)$	$HS + S(I - HS)$

Foley calls the result a "delta sum" with the notation $S\Delta H \equiv HS + S(I - HS)$. Since $S\Delta H = S \oplus HS$, the Barnhill-Gregory Theorem immediately yields: $S\Delta H$ has the interpolation properties of S and the function precision of HS , which is the intersection of the function precisions of H and of S . Moreover, the smoothness of $S\Delta H$ is the minimum of the smoothness of H and of S . The sum $S\Delta H$ could be called a "three-stage" method since it contains the composition of three operators.

Foley goes on to define "delta iteration" which for this example is $\Delta_2 F = H\Delta_1 F + S(F - H\Delta_1 F)$ where $\Delta_1 F \equiv (S\Delta H)F$, etc.

D. Combinations. We have computed trivariate examples with various combinations of S_0 , S_1 , M_μ and local least squares L , Q of degree one and two, respectively. A sampling of these examples are given in the next section. First, we collect results on interpolation, precision, and smoothness of some of the combinations used.

DERIVATIVES OF BOOLEAN SUMS OF THE FORM $S \oplus Q$. If D is a first derivative operator, then

$$(i) D(S_0 \oplus Q)F|_i = DQF|_i, \text{ and}$$

$$(ii) D(S_1 \oplus Q)F|_i = DF|_i.$$

So, if the gradients need to be followed closely, then $S_1 \oplus Q$ should be used; otherwise $S_0 \oplus Q$ may suffice.

Our pictures are all rendered by methods of the form HT , where H is a restricted piecewise tricubic Hermite interpolation and T is some interpolant. H 's precise definition is the following: H is the (trivariate) tensor product of piecewise cubic Hermite interpolation with all derivatives of order greater than one set equal to zero. Thus HF has higher order "flat spots", is precise for linear functions and is C^1 , being the tensor product of a C^1 method. Properties of HT are

1. HTF interpolates to TF at the grid points,
2. HT 's precision is the intersection of H 's precision (linear) and T 's precision, and
3. HT 's smoothness is C^1 (the smoothness of H).

We display a table of the interpolation, polynomial precision, and smoothness of all the methods tested.

Property	Interpolation	Precision	smoothness
Operator			
S_0	$I_A(F)$	Constant	C^∞
S_1	$ID_A(F)$	Linear	C^∞
H	$ID_G(F)$	Linear	C^1
HS_0	$ID_G(S_0F)$	Constant	C^1
HS_1	$ID_G(S_1F)$	Linear	C^1
$S_0 \oplus L$	$I_A(F)$	Linear	C^{-1}
$S_0 \oplus Q$	$I_A(F)$	Quadratic	C^{-1}
$S_1 \oplus L = S_1$	$ID_A(F)$	Linear	C^∞
$S_1 \oplus Q$	$ID_A(F)$	Quadratic	C^{-1}
$H(S_0 \oplus L)$	$ID_G(S_0 \oplus L)F$	Linear	C^1
$H(S_0 \oplus Q)$	$ID_G(S_0 \oplus Q)F$	Linear	C^1
$H(S_1 \oplus L)$	$ID_G(S_1 \oplus L)F$	Linear	C^1
$H(S_1 \oplus Q)$	$ID_G(S_1 \oplus Q)F$	Linear	C^1
$S_1\Delta H = S_1 \oplus HS_1$	$ID_A(F)$	Linear	C^1
$H(S_1\Delta H)$	$ID_G(S_1\Delta H)F$	Linear	C^1
M_μ	$I_A(F)$	None	C^∞
$M_\mu \oplus L$	$I_A(F)$	Linear	C^{-1}
$M_\mu \oplus Q$	$I_A(F)$	Quadratic	C^{-1}
HM_μ	$I_G M_\mu F$	None	C^1
$H(M_\mu \oplus L)$	$I_G(M_\mu \oplus L)F$	Linear	C^1
$H(M_\mu \oplus Q)$	$I_G(M_\mu \oplus Q)F$	Linear	C^1

$I_A(F)$ means interpolation to F at arbitrary data points.

$ID_A(F)$ means interpolation to F and to ∇F at arbitrary data points.

$ID_G(F)$ means interpolation to F , ∇F at gridpoints.

3. Examples. A major part of the task of approximating trivariate data is to understand the resulting 4D surface. In this paper we display (3D) contours of the 4D surfaces. We display several contours together to create a visual feeling for the 4D surface and we distinguish the contours by color.

In the examples we specify “primitive” trivariate functions from which we use positions and gradients at 216 randomly spaced points in the unit cube. The 216 points in xyz are shown in Figure 0. For each example, we display the (3D) contours of the primitive functions and of the approximations to the discrete data.

EXAMPLE 1. TRICUBIC POLYNOMIAL. The primitive function is

$$F(x, y, z) = (x - 0.5)^3 + (y - 0.5)^3 + (z - 0.5)^3.$$

Figure 1 shows three contours of F , HM_1F (with $r = 1.0$) and $H(S_1\Delta H)F$,

respectively. These contours are the -0.1 , 0.1 , and 0.2 contours. Figure 2 shows the 0.0 contour of the same three functions. $H(M_1\Delta H)F$ was also computed but, visually, is identical with HM_1F . $H(S_1\nabla H)F$ is a substantial improvement over HS_1F so HS_1F is not displayed. The methods are localized by using the twenty closest neighbors for M_1 and the twelve closest neighbors for S_1 . A reason for exhibiting this example is that F 's contours are "layered" similarly to geological characteristics that are functions of the three spatial variables.

EXAMPLE 2. TRIQUADRATIC POLYNOMIAL. The primitive function is

$$G(x, y, z) = (x - 0.5)^2 + (y - 0.5)^2 + (z - 0.5)^2.$$

The contours of G are closed $3D$ surfaces, namely spheres. Figure 3 shows two contours of G and of HM_1G with M_1 defined as in Example 1. Surfaces HS_1G and $H(S_1\Delta H)G$ had unaesthetic "ripples" and so are not displayed in this paper. The inner contours are 0.07 and the outer contours are 0.17 .

EXAMPLE 3. TRIGONOMETRIC FUNCTION. The primitive function is

$$g(x, y, z) = \cos(3.14x)\cos(y - 0.5)\sin[3.14(z - 0.5)].$$

Figure 4 shows four contours (two periods each) of g , of HM_1g , and of $H(S_1\Delta H)g$. The contours are -0.6 (blue), -0.2 (green), 0.2 (black), and 0.6 (red).

NOTE. The computations for the examples were performed on a *DEC* system 20 at the University of Utah.

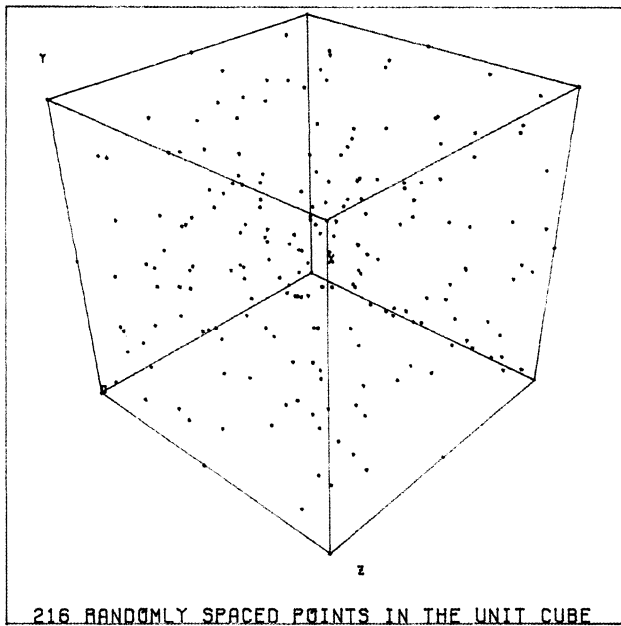
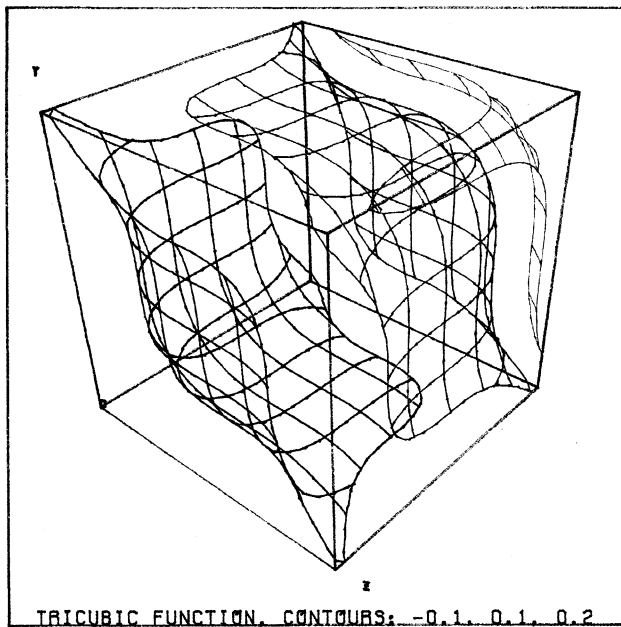
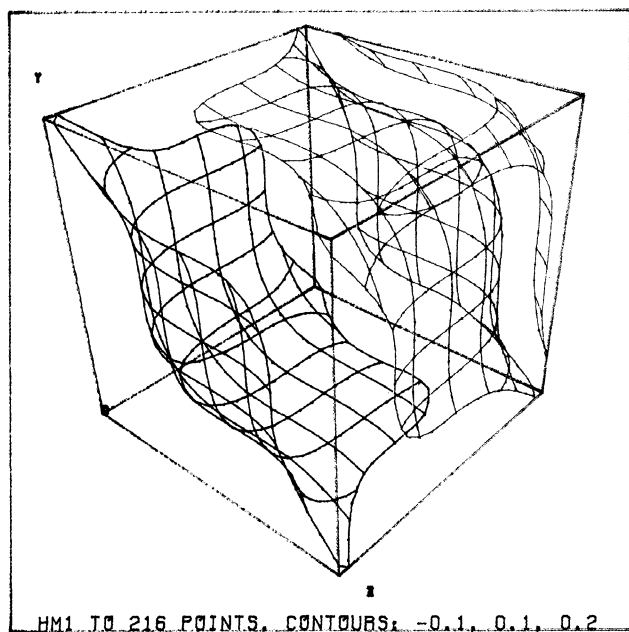
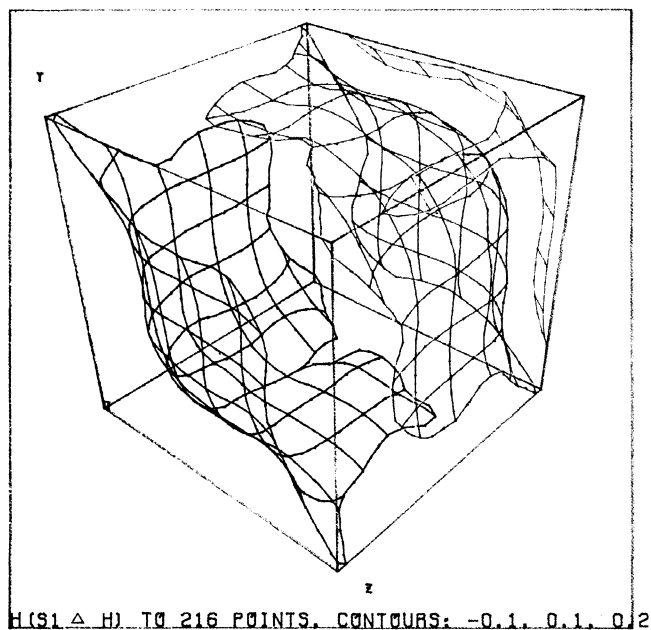


FIGURE 0. Points of Evaluation for Examples.

FIGURE 1.1 Three Contours of Tricubic Function F .

FIGURE 1.2 Three Contours of HM, F .FIGURE 1.3 Three Contours of $H(S_1 \Delta H)F$.

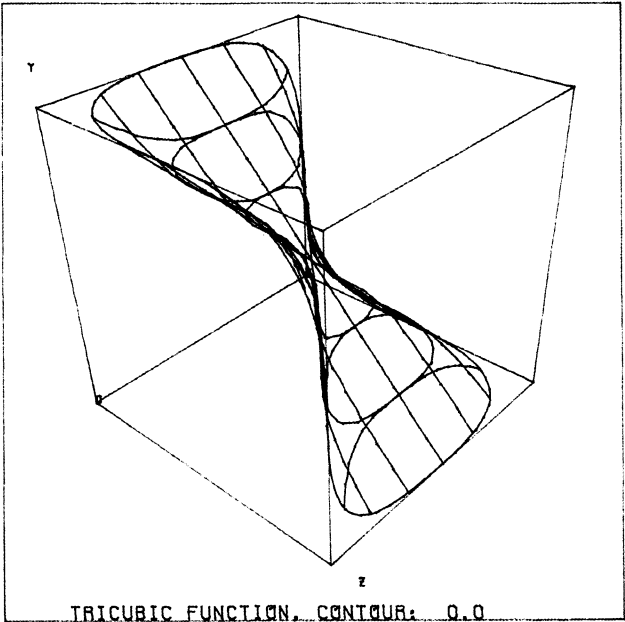


FIGURE 2.1 One Contour of Tricubic Function F .

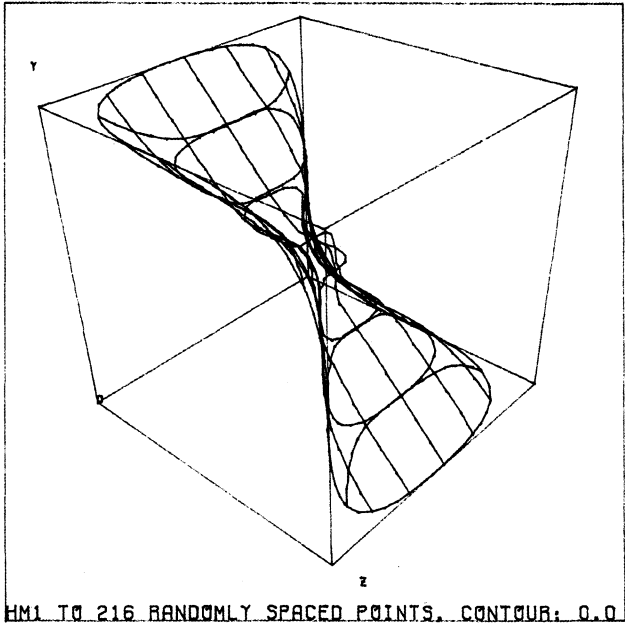
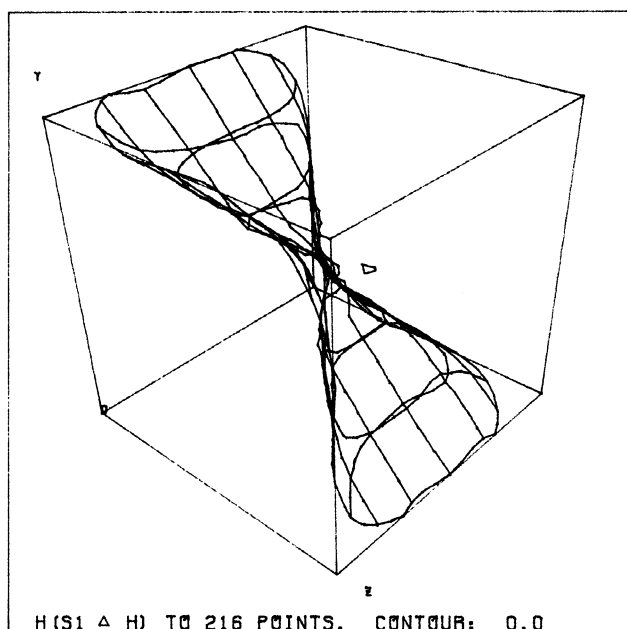
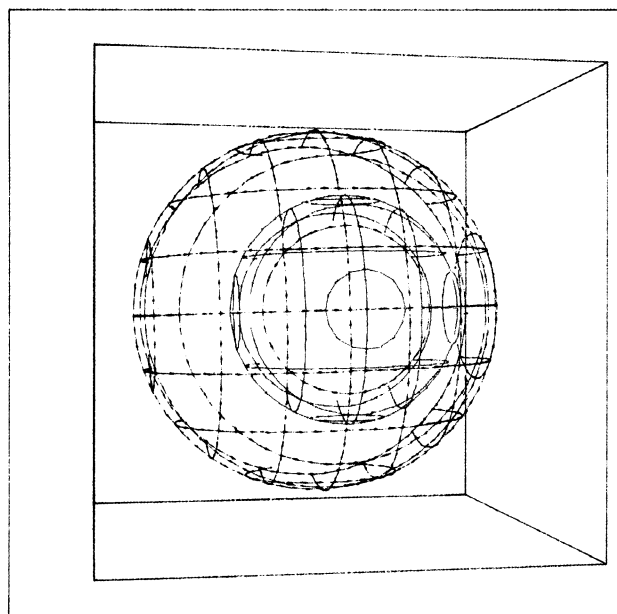


FIGURE 2.2 One Contour of HM, F .

FIGURE 2.3 One Contour of $H(S_1 \Delta H)F$.FIGURE 3.1 Two Contours of Triquadratic Function G .

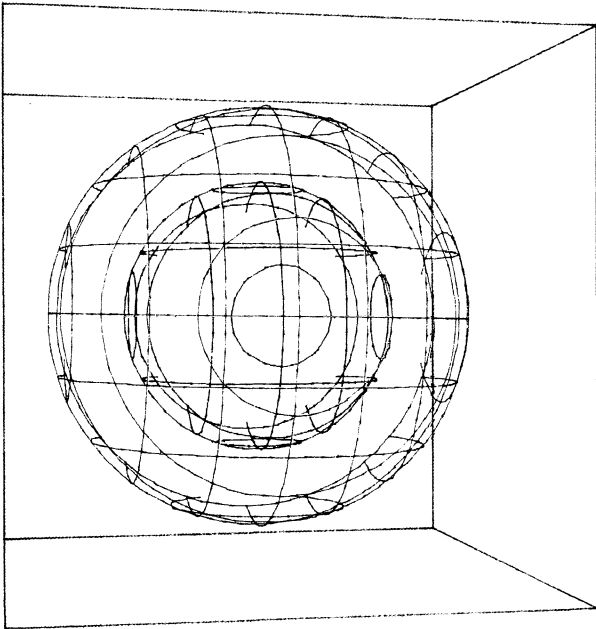


FIGURE 3.2 Two Contours of HM_1G .

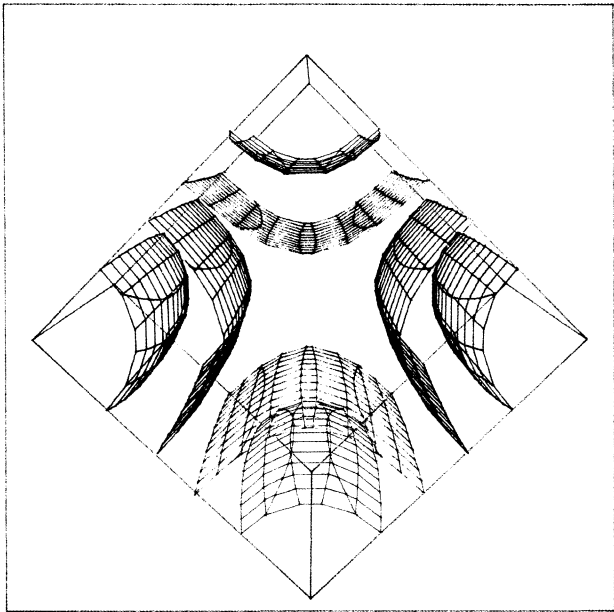
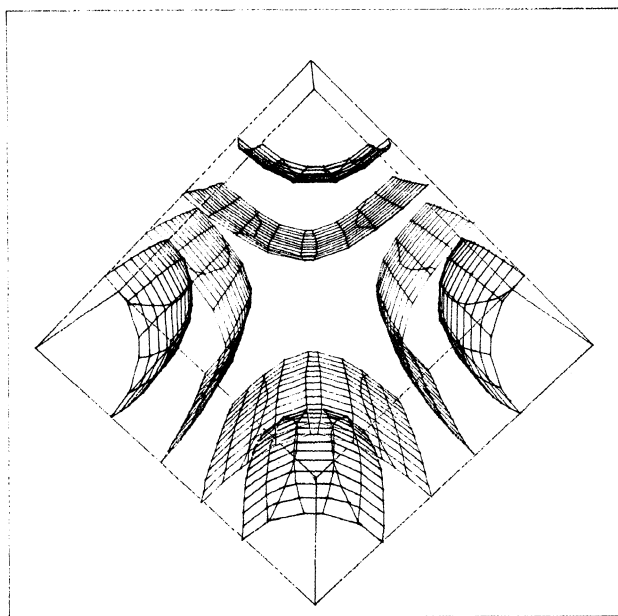
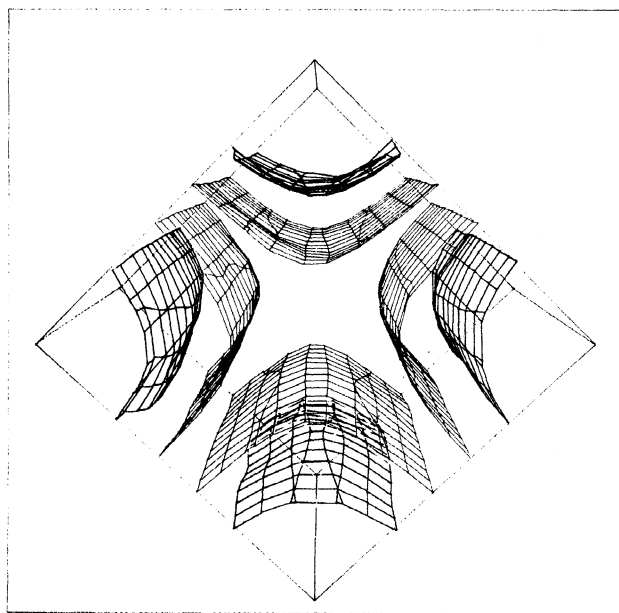


FIGURE 4.1 Four Contours of Trigonometric Function g .

FIGURE 4.2 Four Contours of HM_1g .FIGURE 4.3 Four Contours of $H(S_1\Delta H)g$.

ACKNOWLEDGMENTS. The research and the writing were both improved by friendly interactions with the Math CAGD Group at Utah, with particular thanks going to Paul Arner and Chip Petersen for help with the plots and Andrew Worsey for reading the manuscript.

REFERENCES

1. R. E. Barnhill, *Representation and approximation of surfaces* Mathematical Software III, J. R. Rice (ed.), Academic Press, NY, 1977, 69–120.
2. ———, *Computer Aided Surface Representation and Design Surfaces in Computer Aided Geometric Design*, R. E. Barnhill and W. Boehm, editors, North-Holland Publishing Company, 1983.
3. ———, R. P. Dube and F. F. Little, *Properties of Shepard's surfaces* Rocky Mtn. J. of Math. **13** (1983), 365–382.
4. R. E. Barnhill and G. Farin (1981), *C^1 quintic interpolation over triangles: two explicit representations*, International Journal for Numerical Methods in Engineering **17** (1981), 1763–1778.
5. R. E. Barnhill and J. A. Gregory, *Polynomial Interpolation to Boundary Data on Triangles*, Math. Comp. **29** (1975), 726–735.
6. P. J. Davis, *Interpolation and Approximation*, Dover Publishing Company, 1975.
7. T. A. Foley, *Smooth multivariate interpolation to scattered data*, Ph. D. Thesis, Department of Mathematics, Arizona State University, 1979.
8. R. H. Franke, *Scattered data interpolation: tests of some methods*, Math. Comp. **38** (1982), 181–200.
9. W. J. Gordon and J. A. Wixom, *Shepard's method of "metric interpolation" to bivariate and multivariate interpolation*, Math. Comp. **32** (1978), 253–264.
10. R. L. Hardy, *Multiquadric equations of topography and other irregular surfaces*, J. of Geophysical Res. **76** (1971), 1905–1915.
11. F. F. Little, *Convex combination surfaces*, Surfaces in Computer Aided Geometric Design, R. E. Barnhill and W. Boehm, editors, North-Holland Publishing Company, 1983.
12. C. C. Poepplmeier, *A Boolean sum interpolation scheme to random data for computer aided geometric design*, M. S. Thesis, Computer Science Department, University of Utah, 1975.
13. L. L. Schumaker, *Fitting surfaces to scattered data*, Approximation Theory II, G. G. Lorentz, C. K. Chui, and L. L. Schumaker (editors), Academic Press, N.Y., 1976, 203–268.
14. D. Shepard, *A two-dimensional interpolation function for irregularly-spaced data*, Proc. ACM Nat. Conf., 1968, 517–524.
15. S. E. Stead, *Smooth multistage multivariate approximation*, Ph.D. Thesis, Division of Applied Mathematics, Brown University, Providence, Rhode Island and CAGD Report, Department of Mathematics, University of Utah, Salt Lake City, Utah, 1983.

DEPARTMENT OF MATHEMATICS, UNIVERSITY OF UTAH, SALT LAKE CITY, UT 84112

DIVISION OF APPLIED MATHEMATICS, BROWN UNIVERSITY, PROVIDENCE, RI 02912