Scattered Data Techniques for Surfaces

Suresh K. Lodha
Computer Science
University of California
Santa Cruz, CA 95064
lodha@cse.ucsc.edu

Richard Franke
Department of Mathematics
Naval Postgraduate Scool
Monterey, CA 93943-5216
rfranke@nps.navy.mil

Abstract

This survey presents several techniques for solving variants of the following scattered data interpolation problem: given a finite set of N points in R^3, find a surface that interpolates the given set of points. Problems of this variety arise in numerous areas of applications such as geometric modeling and scientific visualization. A large class of solutions exists for these problems and many excellent surveys exist as well. The focus of this survey is on presenting techniques that are relatively recent. Some discussion of two popular variants of the scattered data interpolation problem – trivariate (or volumetric) case and surface-on-surface – is also included. Solutions are classified into one of the five categories: piecewise polynomial or rational parametric solutions, algebraic solutions, radial basis function methods, Shepard's methods and subdivision surfaces. Discussion on parametric solutions includes global interpolation by a single polynomial, interpolants based on data dependent triangulations, piecewise linear solutions such as alpha-shapes, and interpolants on irregular mesh. Algebraic interpolants based on cubic A-patches are described. Interpolants based on radial basis functions include Hardy's multiquadrics, inverse multiquadrics and thin plate splines. Techniques for blending local solutions and natural neighbor interpolants are described as variations of Shepard's methods. Subdivision techniques include Catmull-Clark subdivision technique and its variants and extensions. A brief discussion on surface interrogation techniques and visualization techniques is also included.

Keywords: algebraic, alpha shape, interpolation, interrogation, parametric, polynomial, radial, rational, scattered, splines, subdivision, surfaces, triangulation, visualization.

1 Introduction

This survey addresses the problem of interpolation of scattered data for surfaces. The basic problem is to find a surface that interpolates a finite set of N points $(x_1, y_1, z_1), \cdots, (x_N, y_N, z_N)$ in R^3. We shall refer to this problem as the *surface reconstruction* problem. There are numerous variants of this problem. We distinguish between at least three different basic variants of this problem. Besides the surface reconstruction problem described above, there is a simpler and more classical version of this problem when the data is prescribed over R^2; more precisely, one seeks a bivariate function $F(x, y)$,

which takes on certain prescribed values, that is, $F(x_k, y_k) = z_k$ for $k = 1, \cdots, N$. In other words, $(x_i, y_i) \neq (x_j, y_j)$ if $i \neq j$. We shall refer to this problem as the *function reconstruction* problem. In most cases, techniques for solving the surface reconstruction problem can be reduced to solving the function reconstruction problem but the converse is not true in general. A third variation of the scattered data interpolation problem is in fact an extension of the surface reconstruction problem. In this problem, often referred to as the *surface-on-surface* problem, the data is prescribed on some surface in R^3. More formally, the basic problem is to construct both a surface S that interpolates a finite set of N points (x_i, y_i, z_i), $i = 1, \cdots, N$ in R^3 and another surface that interpolates measured data values w_i on the surface points $(x_i, y_i, z_i) \in S$. This problem can be viewed as a problem of interpolating scattered data $(x_i, y_i, z_i, w_i) \in R^4$, where (x_i, y_i, z_i) are restricted to lie on a surface in R^3. Although this problem is referred to as the surface-on-surface problem, most of the research has investigated only the *function-on-surface* problem, where w_i is a scalar. The more general surface-on-surface problem occurs when w_i is a multi-valued function. Another popular special case of surface-on-surface problem arises when the scattered data points (x_i, y_i, z_i) belong to a volume in R^3; in other words, these data points are not restricted to lie on a surface. This case often arises in many scientific visualization applications and is referred to as the *trivariate* or *volumetric* scattered data problem.

Some important variations of all these three basic versions of the problem include (i) when the data is prescribed along with a triangulation or a mesh, (ii) when the data is prescribed on a regular or structured grid, typically a rectangular grid, (iii) when derivative or normal values are specified in addition to the data values, and (iv) when the given data is to be approximated rather than interpolated. These are very important problems as well; however to keep the contents of this survey manageable, the focus of this survey is to present *interpolation* techniques that work for truly *scattered* data when only *data values* are prescribed. Many techniques presented in this work will however assume that some kind of triangulation or connectivity information on the data is provided in the form of a mesh. We also make an attempt to provide a somewhat comprehensive bibliography within the scope of this survey particularly for more recent work on scattered data techniques. Detailed references to older works can then be found in the bibliography of several good surveys included in the bibliography at the end of this survey.

Clearly, without any additional restrictions, there are numerous solutions to the scattered data interpolation problem. For example, when the data is prescribed with a triangulation, the piecewise linear surface associated with the triangulation is a solution. Thus, different triangulations lead to different solutions. When a triangulation is provided, the focus of the majority of the solutions is to construct smooth solutions that are tangent or curvature continuous. Such solutions are aesthetically more pleasing and are compact representations of the original data. In addition, the coefficients of the smooth representations provide more intuitive handles for further handling or better understanding of the data. Smooth solutions are also often more faithful to the underlying phenomena of interest, and therefore, are in general more desirable. Often however the data is prescribed without any triangulation or connectivity information between the scattered data points. In such cases, one can either attempt to triangulate the given data set and construct a triangulation-based interpolant or simply use a point-based interpolant that does not require any triangulation of the data.

There are several ways to classify the solutions to scattered data interpolation problems, as is evident in the past surveys on this topic [Sch76, Fra82a, Alf89]. In this work, the class of solutions to the scattered data problem is classified into five categories based primarily on the form of the solution. The first category is the *polynomial or piecewise continuous polynomial parametric solutions*. Many well-known scattered data techniques including finite element methods

and multivariate splines fall in this category. In some applications, however, polynomial or piecewise polynomial solutions are not adequate. For example, polynomial or piecewise polynomial solutions cannot represent part of a sphere or a torus exactly, a desirable feature to have in some geometric modeling applications. Therefore, one considers a more generalized class of solutions, namely rational or piecewise continuous rational solutions, that is solutions which are ratios of polynomials. The second category consists of *algebraic solutions*, that can be expressed as a zero contour of some trivariate polynomial, that is as $F(x, y, z) = 0$, where $F(x, y, z)$ is a polynomial. More generally, piecewise tangent or curvature continuous algebraic solutions are constructed. The third category is the *radial basis function methods*, where the solutions are formed as linear combinations of functions that are radially symmetric. Hardy's multiquadrics, inverse multiquadrics and thin plate splines are important examples of this class of solutions. The fourth category of solutions include *Shepard's methods* and its variants. Depending upon the exact choices made, the interpolant can be radially symmetric, piecewise polynomial or a mixture of both. Therefore, based on the form of the solutions, this technique cuts across the classification of solutions into polynomial and radial categories. However, the key distinguishing feature of this technique is to construct interpolants by blending local solutions by using weight functions. We shall comment on this later again in Section 5. Finally, the fifth category of solutions use a procedure called *subdivision* to construct the interpolants. These techniques depend upon an underlying mesh that is prescribed along with the data. In general the solutions do not have closed form analytic expressions. However when the underlying mesh has some special structure, for example a rectilinear structure, then the solutions reduce to some well-known spline interpolants, for example tensor product B-splines.

The literature on scattered data interpolation is too vast to be covered in a single book, let alone a survey in a book. Several excellent surveys exist on selected subsets of solutions to scattered data interpolation problems. We refer the reader to [Pow90, Buh93, Har90] for radial basis function methods, to [Dyn92] for subdivision techniques, to [Baj92] for algebraic solutions, to [MLL$^+$92] for piecewise polynomial or rational parametric solutions to triangulated data and to [Sch76, Fra82a, BN84, Alf89, Nie93d, NF94, FH94] for polynomial and rational solutions as well as Shepard's methods and radial basis techniques. Nevertheless, we are not aware of any survey on scattered data interpolation techniques that attempts to cover all the five categories of solutions described in the previous paragraph. By bringing all these techniques together in one place, it is hoped that this presentation will expose the practitioners in the field of geometric modeling, computer graphics and scientific visualization to a wide variety of options available for scattered data interpolation and help facilitate a critical evaluation of merits and demerits of employing one method over the other. We also refer the reader to [Nie93a, Nie94] for some research issues in scattered data modeling and analysis.

In order to attempt this ambitious task of presenting such a large class of solutions in a limited space, we have made several compromises. First, discussion of scattered data interpolation techniques in one dimension or for curves is mostly omitted. However one must take note of the fact that every univariate technique for scattered data interpolation extends in a straightforward manner to tensor product data. However the straightforward extension is not always useful. For example, the tensor product nu-splines tightens up all along the parameter curves undesirably. In such cases, additional techniques need to be invented in order to tackle these problems [Nie86]. Nevertheless, many techniques for scattered data interpolation derive their motivation from similar schemes for curves and therefore, we will mention them wherever appropriate. Second, the focus of this presentation is on scattered data interpolation for *surfaces*, although many techniques do extend to higher dimensions, in particular to volumetric data and function-on-surface problems. A brief discussion on these two popular variations of the scattered data interpolation problem is included for

the sake of completeness. We refer the reader to [Alf89, Nie93d, ND91, NT94a, NT94b, Tve91] for scattered volumetric data modeling. Third, an attempt has been made to focus on relatively recent techniques of scattered data interpolation, although for the sake of completeness, brief descriptions or appropriate references for older techniques are presented. Finally, mathematical details of interpolation techniques are kept to a minimum level for which the reader is referred to the original work. Rather a broad overview with intuitive understanding of the interpolation techniques is emphasized.

This survey is organized as follows: Section 2 surveys piecewise polynomial and piecewise rational parametric solutions. Section 3 describes algebraic solutions. Section 4 gives an overview of radial basis function methods. Section 5 deals with Shepard's methods. Section 6 discusses subdivision techniques. Section 7 presents a discussion on surface visualization and surface interrogation techniques. Finally Section 8 concludes with some final remarks and future directions.

2 Polynomial and Rational Parametric Solutions

Perhaps the most popular representation in both academics and industry for solving scattered data interpolation problems is the piecewise continuous polynomial or rational parametric solutions. The polynomial solutions are represented as $x = x(u, v)$, $y = y(u, v)$ and $z = z(u, v)$, where $x(u, v)$, $y(u, v)$ and $z(u, v)$ are bivariate polynomial functions in the parameters u and v. The degree of these polynomials is then referred to as the parametric degree of the solutions. A slight generalization is to consider rational solutions that can be expressed in the form $x = \frac{x(u,v)}{w(u,v)}$, $y = \frac{y(u,v)}{w(u,v)}$, $z = \frac{z(u,v)}{w(u,v)}$, where $w(u, v)$ is also a polynomial function. The tensor product B-splines and NURBS are examples of solutions that belong to this category.

2.1 Function Reconstruction

The literature on solutions to function reconstruction problem by piecewise polynomial and rational parametric functions is rich and many good surveys exist [Sch76, Fra82a, FN91, FHN93, Alf89, Far86, BFK84, Nie93d, Bar83]. In the category of polynomial and rational interpolants, most of these surveys emphasize construction of smooth interpolants. The presentation here is intended to be somewhat complementary to these surveys. In particular, global interpolation by a single polynomial and the dependency of piecewise linear solutions on underlying triangulations are emphasized. Both these methods are specific to solving the function reconstruction problem. That is, there is no straightforward extension of these techniques for solving the surface reconstruction problem. Nevertheless these techniques are important because the global interpolation by a single polynomial highlights the inherent difficulties in solving scattered data interpolation problem for more than one variable, while piecewise linear solutions highlight the dependency of all finite element spline-based solutions on the underlying triangulations.

Polynomial Interpolation Method: Polynomial interpolation in one variable is a very basic tool in numerical analysis. A detailed account is easily available in many standard numerical analysis textbooks. Many different forms of interpolating polynomial have been studied by Lagrange, Cauchy, Hermite, Newton and Chebyshev. It is well-known that univariate interpolation using polynomials such as Lagrange interpolation can be badly behaved. Nevertheless the existence and uniqueness of a univariate polynomial of degree n for scattered data specified at $n + 1$ points along the real line is a very satisfying result. It is therefore natural to attempt to formulate some analogous result for surfaces or higher dimensions.

However, major difficulties arise for bivariate or multivariate data. We now discuss the bivariate case. The dimension of the space of bivariate polynomials of total degree n is $\binom{n+2}{2}$. Therefore, to guarantee existence and uniqueness of a polynomial interpolant of total degree n in 2 variables, a necessary condition is to have the number of scattered data points to be exactly $\binom{n+2}{2}$, not a very natural restriction. Even if there are exactly $\binom{n+2}{2}$ points, there may not exist a polynomial of degree n that interpolates all the points, as is clear by considering the example of all the points lying on a straight line. Therefore, in general, for arbitrary number of scattered data points there is the problem of choosing the right polynomial subspace from which to pick the interpolant. In particular, this choice of the polynomial subspace depends upon the geometric configuration of points. For example, given $\binom{n+2}{2}$ points on a straight line, one must choose a polynomial of degree $\binom{n+2}{2} - 1$, while if the points are in general position, a bivariate polynomial of degree n would suffice.

Given a set of scattered data points, de Boor and Ron [dBR92] have proposed a scheme for choosing a polynomial subspace S of the same dimension as the the number of scattered data points in such a way that the existence and the uniqueness of the interpolant is guaranteed. Moreover, the elements of this polynomial subspace and therefore the resulting interpolant has as small degree as possible, a property referred to as *minimal degree*. These interpolants have several other nice properties including translation invariance, scale invariance and coordinate system independence, although these interpolants are not affine-invariant in general.

We present a brief intuitive description of the construction of the polynomial subspace S. The construction proceeds incrementally as follows. To begin with, S consists of constants only. At the k-th step (starting from $k = 1$), consider the space P_k of all the bivariate homogeneous polynomials of degree k. The dimension of the space of these polynomials is $k+1$. Find if any of the polynomials from $P_k + S$ and not completely in S vanish at all the given scattered data points. Let l be the dimension of the space Q_k of these polynomials of degree k. Note that since these polynomials vanish at all the scattered data points, addition of these polynomials to any interpolant will still be an interpolant. Therefore, one should stay away from these polynomials. This is achieved by considering the orthogonal complement R_k of Q_k in $P_k + S$. The inner product chosen for finding this orthogonal complement is $< f, g >= \frac{1}{\alpha!} \sum_\alpha D^\alpha f(0) D^\alpha g(0)$, where $D^\alpha = \frac{\partial^{\alpha_1+\alpha_2}}{\partial x^{\alpha_1} \partial y^{\alpha_2}}$. This choice of inner product helps achieve the many nice properties of the interpolant mentioned above. The subspace R_k with dimension $k+1-l$ is then added to the polynomial subspace S. If the dimension of the resulting polynomial subspace equals the number of scattered data points, then the process terminates, otherwise k is incremented by 1 and the process is repeated. A proof that this method always works is provided in [dBR90]. A constructive procedure for building these interpolants using Gaussian elimination as well as a description of a MATLAB-like program is also presented [dBR92].

We discuss the following example when the data is prescribed over the 6 points on a regular hexagon. First, the constant function is added to the polynomial subspace S. Since no three points are collinear, there is no linear polynomial that vanishes at all the data points and therefore, all linear polynomials are added. Thus far the dimension of the polynomial subspace is 3. Next, all bivariate polynomials of degree 2 are considered. Of these, exactly one, namely the circle vanishes at all the points. The dimension of the orthogonal complement of the circle is 2, which is then added to S. The dimension of S is still short by 1. Therefore, one considers all the bivariate polynomials of degree 3 now. To determine the polynomials that vanish at the scattered data points, let us label the vertices of the hexagon from P_1 to P_6. Consider the cubic polynomial M_1 obtained by forming the product of linear polynomials that vanish at the opposite pair of vertices of the hexagon, that is the product of the lines P_1P_4, P_2P_5 and P_3P_6. M_1 clearly vanishes at all the six points. Similarly consider the cubic polynomial M_2 formed by taking the product of

three linear polynomials P_1P_2, P_3P_4 and P_5P_6 and the cubic polynomial M_3 formed by taking the product of three linear polynomials P_1P_2, P_3P_6 and P_4P_5. It is clear that these three cubic polynomials M_1, M_2 and M_3 vanish at all the six points. It is also not too difficult to observe that these three polynomials are linearly independent. The dimension of the orthogonal complement of these polynomials is 1. This subspace is then added to S to complete the construction of S. The dimension of S is now equal to 6, same as the number of data points. A symmetry argument is presented in [dBR92] to conclude that a cubic generator of S is $x^3 - 3x^2y$. This completes the construction of S. Existence and uniqueness of the interpolant to any specified values over the six points on a regular hexagon is now guaranteed when the interpolant is chosen from the subspace S constructed above.

Triangulation-Based Methods: The construction of a unique global polynomial interpolant to scattered data is satisfying. However these multivariate Lagrange interpolants will have high degree for large data sets and will suffer from similar defects as in the univariate Lagrange case. Therefore, piecewise polynomial or rational interpolants of low degree are needed for scattered data interpolation in general. Tensor product bicubic B-splines and NURBS are very popular and successful interpolants for data on a rectilinear grid. For general scattered data points, a typical method is to triangulate the domain data points in the plane and then construct piecewise continuous interpolants on each triangle. These methods are referred to as *triangulation-based methods*. Either a triangulation is assumed to be given or is constructed before fitting polynomial or rational patches to the data sets.

A collection $\Delta = \{T_i\}_{i=1}^{M}$ of triangles in the plane is called a *triangulation* of a region Ω provided that (i) any pair of triangles from Δ intersect at most at one common vertex or along a common edge, and (ii) the union of the triangles $\{T_i\}$ is Ω. Note that the set Ω need not be convex in general and may have holes. A simple solution to the scattered data interpolation problem is to construct a piecewise linear interpolant, on the triangulation of the given data set. More sophisticated solutions then attempt to construct smoother interpolants.

Piecewise Linear Solutions: A popular solution is to construct a Delaunay triangulation of the scattered data points in R^2 and construct the piecewise linear interpolant on this triangulation. Delaunay triangulation is a triangulation with many nice properties and a good source of reference is [PS85]. Delaunay triangulation can be computed in $O(N \log N)$ time. Recent studies have however shown that depending upon the nature of the data, even though the triangulation may exhibit nice properties, the corresponding interpolant may not. To remedy the situation, piecewise linear interpolants have been suggested that are built on triangulations that depend upon data [DLR90, Bro91, QS90].

Different kinds of triangulations have been studied that use different optimality criteria [Sch93b]: (i) *Max-min Angle Criterion:* A triangulation Δ is said to be optimal with respect to the max-min angle criterion provided that the maximum of the minimum angle of any triangle in Δ is the maximum over all possible triangulations of the same region Ω with the same vertices. This triangulation is also referred to as the *Delaunay triangulation*. There is another well-known equivalent criterion to characterize a Delaunay triangulation. This criterion is known as the *circle criterion*. A triangulation is said to satisfy the circle criterion if a circle circumscribing any triangle of the triangulation does not contain any other vertices of the triangulation in its interior. If the scattered data points are in the general position, that is no four points are co-circular, then the Delaunay triangulation is unique. (ii) *Min-max Angle Criterion:* A triangulation Δ is said to be optimal with respect to the min-max angle criterion provided that the minimum of the maximum angle of any triangle in Δ is the minimum over all possible triangulations of the same region Ω with the same vertices. (iii) Other criteria include maximization of minimum height of the triangles, min-

imization of the maximum slope of the triangles, maximization of minimum area of the triangles, maximization of minimum radius of the inscribed circles, minimization of maximum radius of the circumscribed circles.

The standard approach to finding best triangulations is to apply an appropriate edge swapping algorithm as described below. Given an interior edge e of a triangulation, there is a natural associated quadrilateral Q_e formed from the two triangles of δ which share the edge e. Such an edge is swappable provided that Q_e is convex and no three of its vertices are collinear. If an edge e is swappable, then we can create a new triangulation by actually swapping the edge. An edge swapping algorithm swaps an edge if swapping improved the triangulation with respect to the optimal criterion. However, except for Delaunay triangulation, this local edge swapping criteria leads to only locally optimal and not to globally optimal triangulation. Schumaker proposed a *simulated annealing* technique to search for globally optimal triangulations for a wide class of optimality criteria [Sch93a]. Although this technique is not guaranteed to find the globally optimal triangulation, the overall probability of finding the globally optimal triangulation is increased. Each step of the algorithm selects a random edge for a local swap. If it improves the triangulation with respect to the optimality criterion, then the algorithm swaps the edge. Depending upon an annealing schedule, the algorithm swaps the edge sometimes even if the swap deteriorates the triangulation. This flexibility allows the algorithm to get out of locally optimized triangulations that are not globally optimum. Each stage of the algorithm thus takes only a constant amount of time after an initial triangulation is found. The top diagram of Figure 1 shows the wireframe drawing of a steep hill, that is defined analytically. This function is sampled at some scattered data points and then an interpolant is constructed and compared with the original function. The upper left diagram of Figure 1 shows the Delaunay triangulation of the data points. The upper right diagram of Figure 1 shows the piecewise linear interpolant built on the Delaunay triangulation. The middle left diagram of Figure 1 shows a locally optimized triangulation of the data points using edge swapping algorithm and starting with the Delaunay triangulation. The optimality criterion used for these figures is to minimize the sum of the angle between the normals of the adjacent linear patches. The middle right diagram of Figure 1 shows the piecewise linear interpolant built on the locally optimized triangulation on the left. The lower left diagram of Figure 1 shows a triangulation of the data points obtained after 60 iterations of using the simulated annealing technique to construct a globally optimized triangulation. The lower right diagram of Figure 1 shows the piecewise linear interpolant built on this triangulation. Even though this globally optimized triangulation has thin and skinny triangles, the interpolant based on this triangulation has smaller error than the interpolant based on the Delaunay triangulation or the locally optimized triangulation.

An algorithm for computing the globally optimum triangulation for the minimization of the maximum angle problem is derived in [ETW92] with computational complexity $O(N^2 \log N)$. Algorithms for computing the globally optimum triangulation for the maximization of the minimum height problem and the minimization of the maximum slope problem are derived in [BEE$^+$93] with computational complexity $O(N^2 \log N)$ and $O(N^3)$ respectively. These algorithms are examples of a general paradigm described in [BEE$^+$93].

Spline Solutions: As stated in the introduction, piecewise linear solutions are not adequate for many applications that demand smoother interpolants. The focus of spline methods is to construct piecewise C^1 or C^2 continuous interpolants over *triangulated* scattered data using as few patches as possible of low degree. The classical results include (i) fitting piecewise C^1 continuous quadratic polynomial patches by splitting each triangle into six or twelve subtriangles, known as Powell-Sabin split [PS77], (ii) fitting piecewise C^1 continuous cubic polynomial patches by splitting each triangle into three triangles, known as Clough-Tocher split [CT65], and (iii) fitting piecewise C^1

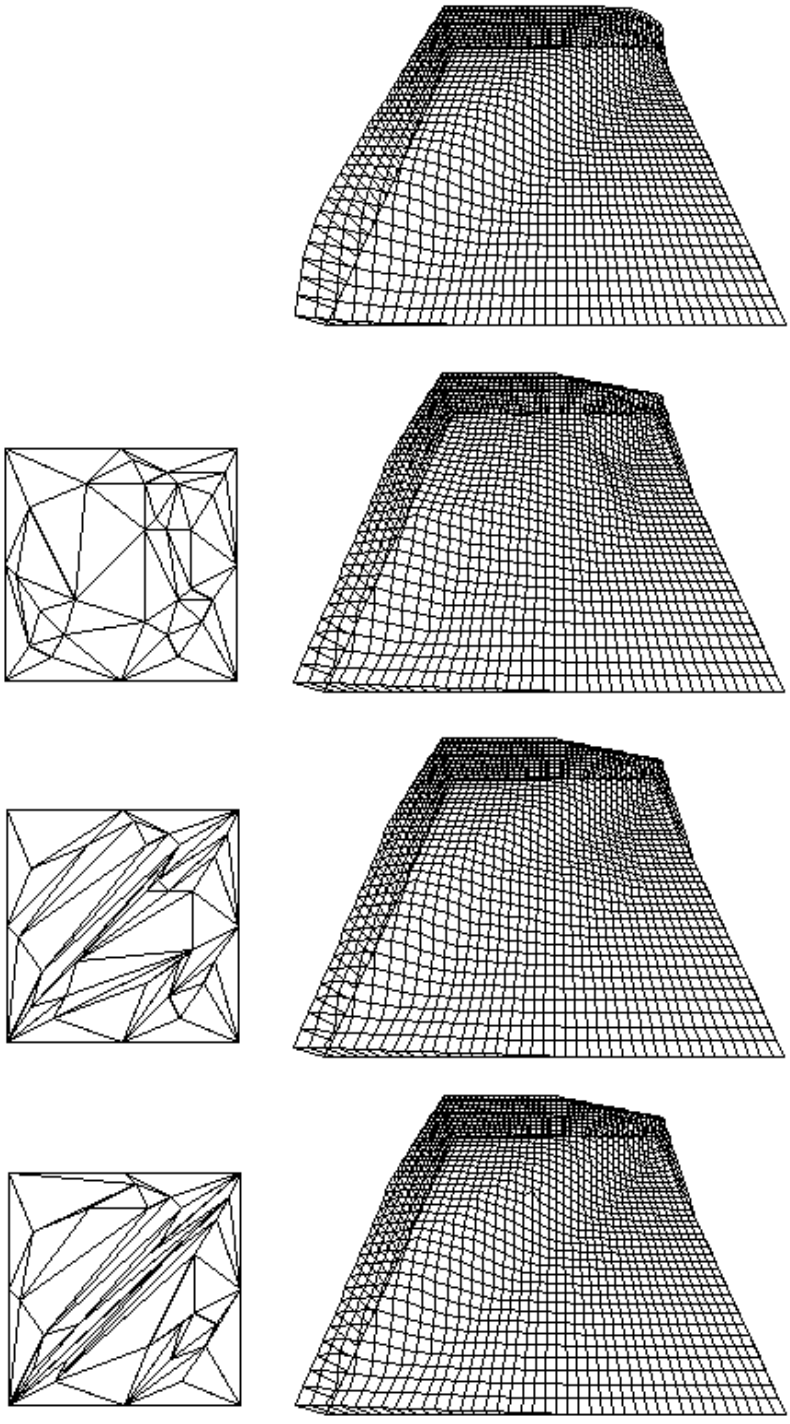

Figure 1: Simulated annealing: (a) wireframe drawing of a steep hill, (b) left: Delaunay triangulation of scattered data points; right: piecewise linear interpolant on this triangulation, (c) left: locally optimized triangulation; right: piecewise linear interpolant on this triangulation, (d) left: triangulation obtained by simulated annealing technique; right: piecewise linear interpolant on this triangulation.

continuous quintic polynomial patches without splitting the triangle, that is one patch per triangle [Far86]. Some early C^1 continuous interpolants were proposed by Akima [Aki78b, Aki78a], Lawson [Law86, Law77], and Renka and Cline [RC84, Ren84a]. These interpolants are local in the sense that any change in a data value affects the interpolant only in a neighborhood of that data point. These and similar solutions have been used in finite element methods and are known as *finite element solutions*. Most of these interpolants have additional degrees of freedom. The quality of interpolants as measured by simple characteristics such as wiggles in Gaussian curvature can depend heavily on how these degrees of freedom are chosen. In some cases, default values for these additional degrees of freedom are chosen according to some simple heuristics and the user is allowed to manipulate these "shape" parameters interactively. In other applications, these degrees of freedom are chosen as solutions of some variational principles, which minimize certain energy functionals globally.

Minimum-Norm Network Method [Nie80, Nie83] is an example of a global interpolant based on a variational principle. This method was found to be of one of the most effective triangulation-based interpolant tested by Franke [Fra82a]. First, the convex hull of the data points (x_i, y_i) in the plane is triangulated. Each edge of the triangulation is then replaced by an interpolating cubic polynomial curve such that the network of piecewise cubic polynomials minimizes the integral of the square of the second derivative over all edges in the triangulation. The curve network is then filled in by piecewise C^1 continuous rational surface patches. This technique has been widely used in constructing a large class of scattered data interpolants with variations in all the three steps above: in constructing the triangulation, in choosing the interpolating curve network and in filling in by smooth surface patches. Variations of these three generic steps have also been used in solving the surface reconstruction problem described in Section 2.2 as well as in constructing algebraic solutions described in Section 3. Other types of piecewise continuous polynomial and rational solutions based on constraints and based on the discretization of a transfinite interpolant have also been used [Nie74, Bar83, Nie79, NF84, Alf89].

The finite element spline solutions can also be viewed as solving the scattered data interpolation problem from the space of piecewise polynomials of a given degree with certain smoothness defined over the original triangulation or a refined triangulation. Elements of these piecewise polynomial spaces are referred to as multivariate splines. A good source of references for studies related to the dimension of these multivariate splines and their applications to scattered data interpolation problem can be found in the survey article by Alfeld [Alf89]. Finite element solutions are examples of multivariate splines, where the splines have minimally supported basis functions. The property of minimal support in turn makes the interpolants local and of relatively higher degree than is possible by global interpolants in general. An intermediate set of subspaces between finite element solutions and the full space of piecewise polynomials is *vertex splines*. These splines are minimally supported as well, but in contrast to finite elements, they have specified smoothness property at the vertices of the triangulation. Use of vertex splines in solving scattered data interpolation problems is presented in [Chu88].

Multivariate B-splines are splines obtained by projections of simplices and are also referred to as simplex splines [DM83]. When these splines are obtained as projections of a cube, they are referred to as box splines. Box splines have been used to solve scattered data interpolation on a three-direction mesh. An excellent description and further references can be found in [DL91]. More recently, a multivariate B-spline scheme that uses a combination of B-patches and simplex splines has also been used to solve the scattered data interpolation problem over arbitrary triangulations [FS93]. The piecewise interpolant is defined on a refined triangulation that depends upon the choice of knots needed to define the interpolants. All these multivariate spline schemes and finite element

schemes are restricted to solving the function reconstruction problem and cannot handle the more difficult surface reconstruction problem.

2.2 Surface Reconstruction

In analogy to the function reconstruction problem, a first step or the simplest solution to the surface reconstruction problem for scattered data may seem a piecewise linear solution to the problem. However ' to find a triangulation of scattered data in R^3 that respects the underlying topology of the data is a much more difficult problem. To observe the inherent difficulty of the problem, note that same set of scattered data points can correspond to several topologically different surfaces. Therefore, if the intent is to reconstruct a surface with a particular topology, some additional information about the topology of the surface must be provided. If this additional information is provided in the form of a triangulation, then of course, the piecewise linear interpolant is trivially constructed. On the other hand it is not clear what is a good or appropriate way of specifying the underlying topology of the data other than an underlying mesh. However, important advances have been made in constructing a piecewise linear *approximation* to scattered data sets in R^3 [HDD$^+$92, HDD$^+$93, HDD$^+$94, BBX95]. The closest to a piecewise linear interpolant to scattered data sets is the concept of alpha-shape, which we discuss next.

Alpha-Shapes: Alpha-shapes are a generalization of the convex hull of the scattered data points [EM94]. They are based on Delaunay tetrahedralization of the scattered data points in R^3. One of the major strengths of this method is that in contrast to most other methods for scattered data interpolation for surface reconstruction, this method does not assume any underlying triangulation of the given data set.

Given a set of points S in R^3, one can build a tetrahedralization of the convex hull of S, that is a partition of the convex hull of S into tetrahedra, in such a way that the circumscribing sphere of each tetrahedron T does not contain any point of S in its interior. Such a tetrahedralization is called a 3D Delaunay tetrahedralization of the given set of points. Under certain non-degeneracy assumptions on the data set such as no more than four points are co-spherical, the Delaunay tetrahedralization is unique. The expected running time of the algorithm for computing Delaunay tetrahedralization is roughly $\log N$ times the number of tetrahedra using a randomized incremental flipping-based algorithm [ES92]. In the worst case, the number of tetrahedra can be about $O(N^2)$. In practice, however, most point sets have many fewer Delaunay tetrahedra and therefore, the running time of the algorithm is much better than this theoretic bound.

For $\alpha = \infty$, the alpha-shape is identical to the the 3D Delaunay tetrahedralization of the convex hull of the scattered data points. This tetrahedralization consists of tetrahedra, the triangular faces of these tetrahedra, the edges of these triangular faces and the vertices which are same as the given scattered data points. For $\alpha = 0$, the alpha-shape is identical to the given scattered data points. Intermediate alpha-shapes for $0 < \alpha < \infty$ are obtained by shrinking the convex hull as follows. Consider a spherical eraser of radius α that passes through the convex hull erasing any tetrahedra, triangles or edges such that the eraser can pass through them without touching any of the vertices that bound these elements. In other words, the minimum radius of the circumscribing sphere of the elements that are erased is strictly greater than α. This procedure can erase triangular faces of some tetrahedra that are not erased or erase some edges of triangular faces that are not erased. In such a situation, retain all the lower components of an element that was not erased. For example, if a tetrahedron is not erased, keep all its edges and triangles. Notice that the original vertices are always kept throughout this procedure. The space covered by the resulting set of tetrahedra, triangles, edges and vertices is referred to as the alpha-shape of the given scattered data set. Of

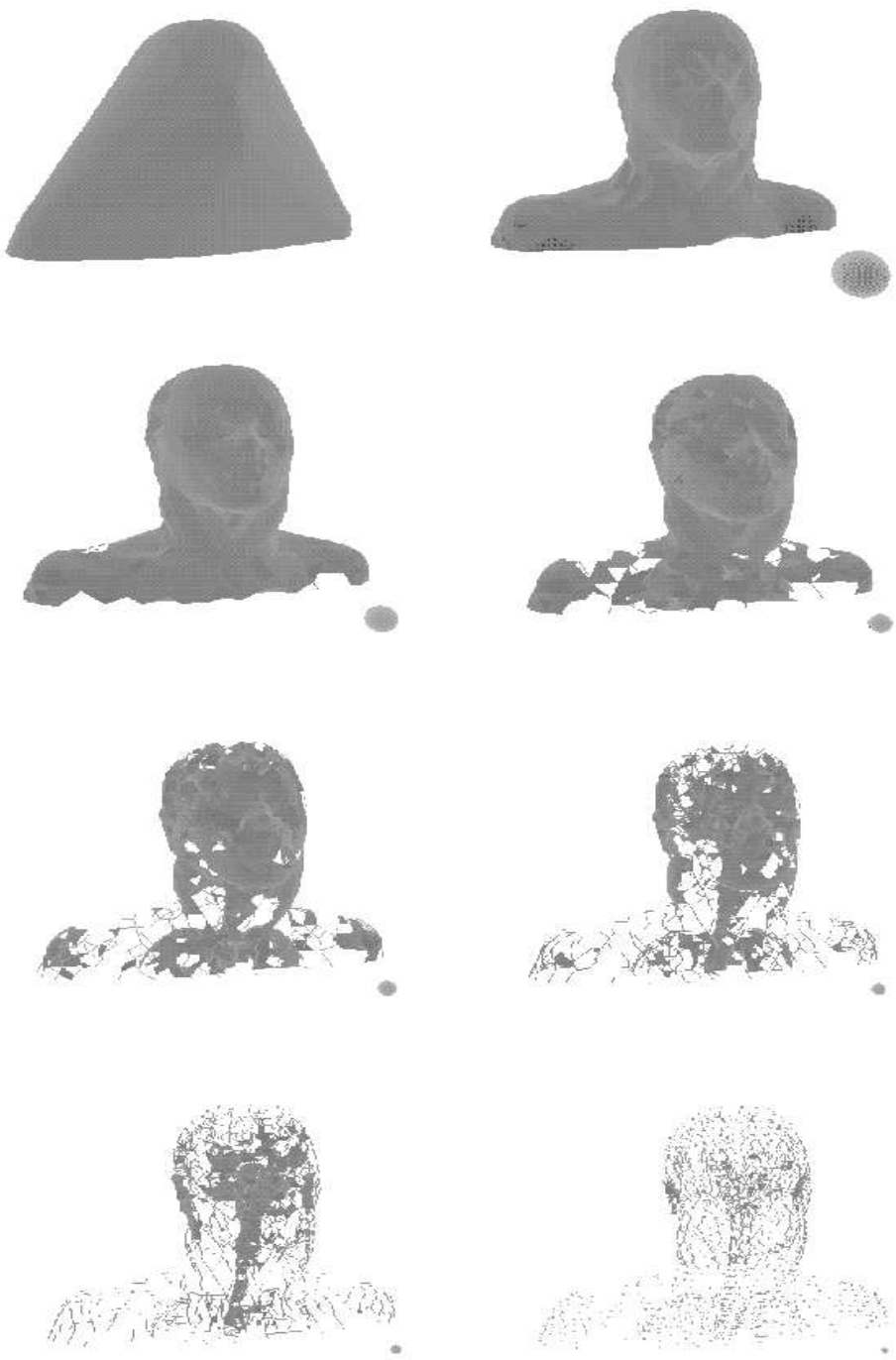

Figure 2: Alpha-Shapes of a human figure for decreasing values of alpha; the value of alpha is shown as the radius of the sphere next to the alpha-shapes

course, it is possible to have cavities and holes in the alpha-shape.

This intuitive description of the concept of alpha-shape is described very elegantly using the notion of alpha complexes and nerves of the Voronoi decomposition. An algorithm for computing and displaying the alpha shapes has been presented and implemented [EM94]. Figure 2 represents 8 alpha-shapes obtained from the scattered data points sampled from a person's bust by gradually reducing the value of α from ∞ to close to zero.

It is tempting to adopt the alpha-shape approach to create piecewise linear triangulation of the given scattered data set. This would be possible if an alpha could be found when all the tetrahedra are erased and exactly the triangles defining the triangulation are left. It is easy to see that this is not possible to achieve if the density of scattered data points is sparse so that the points across the surface are equally close or closer than the points on the surface. For example, this would be a problem for a poorly sampled thick torus where the difference between inner and outer radius is small. Even with additional assumptions on the density of the scattered data points it is not clear whether this approach can be adapted to produce piecewise linear triangulation of the scattered data points that respects the underlying topology of the data set that is prescribed by the user in some form. Due to these difficulties, the rest of this section will assume that the underlying topology of the data set is specified by a triangulation or a mesh. The focus now is on constructing smooth interpolants.

Interpolants on Triangulated Data: A large number of piecewise polynomial and rational interpolants have been suggested for surface reconstruction to fit a triangulated scattered data. Fortunately, there is an excellent survey of these interpolants [MLL$^+$92]. Some more interpolants of this type are presented in [Pet90]. The overall idea for constructing these interpolants consists of two key steps: first, edges of the triangulation are replaced by an interpolating curve network usually with cross-boundary derivatives. Farin [Far82] and Piper [Pip87] give sufficient conditions for two polynomial patches to meet with tangent plane continuity. Computation of cross-boundary derivatives along the interpolating curve network is described in [CK83, Her85, Nie87b, Jen87].

In the second and the final step, each triangular hole bounded by three curves is filled in by one or more polynomial or rational surface patches in such a way that the overall surface is tangent or curvature continuous. The challenge in these schemes is to construct a smooth interpolant of low degree with few pieces. There are two major approaches to filling in these triangular holes: *split domain schemes* and *convex combination schemes*. In split domain schemes, the triangular domain of each hole is split into several subdomains and then a patch is fitted into each subdomain. One of the most popular splitting technique employs Clough-Tocher split, where a triangle is split into three triangles by inserting a point in the interior of the triangle and connecting it to the three vertices of the original triangle. This splitting technique therefore fits three patches per triangular hole. Splitting allows the data along each boundary to be matched independently of the data on the other two boundaries. The remaining degrees of freedom are used to make the internal boundaries of the patches to meet with tangent plane continuity. This technique has been used to construct three polynomial patches of degree four per hole [SS87, Jen87, Pip87]. Convex combination schemes create one single patch per hole. First, three patches are constructed, each of which interpolates part of the boundary data. Then a convex combination of the three patches is formed in such a way that the resulting patch interpolates all of the data. The convex combination typically utilizes rational weight functions and therefore produces rational interpolants. This technique has been used to create piecewise tangent and curvature continuous rational scattered data interpolants [Nie88, Nie87b, Her85, HP89, Hag86].

An early survey on surface construction based upon triangulations is available in [NF83]. One of the major findings of the more recent survey [MLL$^+$92] is that most of the local interpolants suffer

from similar shape defects. The primary cause of the shape defect seems to be in the construction of boundary curves. The shape defects in the boundary curves, such as flatness seems to be propagated inward resulting in the flat spots of the surfaces. One way to overcome these defects could be to use some form of global optimization on the boundary curves as in the minimum norm network method described in Section 2.1.

Interpolants on Irregular Mesh: The techniques described above are applicable when the underlying data has been triangulated. However, there are a number of important practical situations when the topology of the underlying data is prescribed not in terms of triangles, but with some connectivity information between the data points in terms of edges and faces, that need not be triangles. In fact, the faces need not be planar. We shall refer to a mesh of vertices, edges and (possibly non-planar) faces as an irregular mesh if there are no restrictions on the number of edges a face can have or on the number of edges that meet at a vertex. This section describes some of the recent techniques to build interpolants on irregular meshes.

Although attempts have been made to fill n-sided holes for more than a decade, S-patches [LD89] are perhaps one of the most promising representation for filling n-sided holes because they are multi-sided generalization of Bernstein-Bézier surface patches, and thus have much better potential of integration with existing geometric modeling systems than previously proposed systems. S-patches are generalizations of both tensor product Bernstein-Bézier patches and triangular Bernstein-Bézier patches. S-patches build on the theory of Bézier simplexes and generalized barycentric coordinates. An S-patch with n sides admits a rational representation with degree $d(n-2)$, where d is referred to as the depth of the S-patch. Thus a 5-sided patch of depth 3 will have a rational representation of degree 9. This can result in somewhat excessive storage and slow algorithms for computing with S-patches.

We now describe a scheme for interpolating scattered data points over an irregular mesh, that consists of piecewise tangent continuous triangular quadratic and cubic polynomial patches [Pet93, Pet95]. The algorithm consists of a corner-cutting type procedure that is often used in subdivision surfaces (Section 6). In the first step of the mesh refinement, every s-sided face is subdivided into s quadrilaterals by inserting the centroid of the surface and connecting it to the midpoints of all the edges. Now let us say, an original vertex V_1 is surrounded by m quadrilaterals. In the second step of the mesh refinement, a preliminary quadrilateral center $\hat{C}_i$ is introduced for each of the m quadrilaterals for $i = 1, \cdots, m$. If V_1, V_{i2}, V_{i3} and V_{i4} are the four vertices of the i-th quadrilateral and a_{i1}, a_{i2} are arbitrary blend ratios, where $0 < a_{i1}, a_{i2} < 1$, then

$$\hat{C}_i = (1-a_{i1})(1-a_{i2})V_1 + (1-a_{i1})a_{i2}V_2 + a_{i1}(1-a_{i2})V_3 + a_{i1}a_{i2}V_4.$$

By default, the blend ratios can be chosen to be half so that the preliminary cell centers are the centroids of the quadrilaterals. These preliminary centers $\hat{C}_i$ are then perturbed a little to new positions C_i in order to satisfy a planarity condition. The planarity condition requires that all points A_i, $i = 1, \cdots, m$, lie in the same plane, where $A_i = \frac{C_i + C_{i+1}}{2}$ and $(i+1)$ is taken modulo m. This condition is trivially satisfied by $\hat{C}_i$ if $m = 3$. In fact, this condition is also satisfied by $\hat{C}_i$ if $m = 4$ since $A_1 + A_3 = A_2 + A_4$. Therefore for $m = 3, 4$, $C_i = \hat{C}_i$. For $m \geq 5$,

$$C_i = V_1 + \frac{2\omega_m}{m} \sum_{j=1}^{m} \cos\left(\frac{2\pi}{m} j\right) \hat{C}_{i+j}$$

where $0 < \omega_m < 1$ with default values $\omega_m^{-1} = 1 + \cos\frac{2\pi}{m}$ if m is even and $\omega_m^{-1} = 2\cos\frac{\pi}{m}$ if m is odd. The proof that these new C_i's satisfy the planarity condition is given in [Pet95, Loo94]. This process of computing the cell centers C_i is repeated for all the quadrilaterals surrounding any

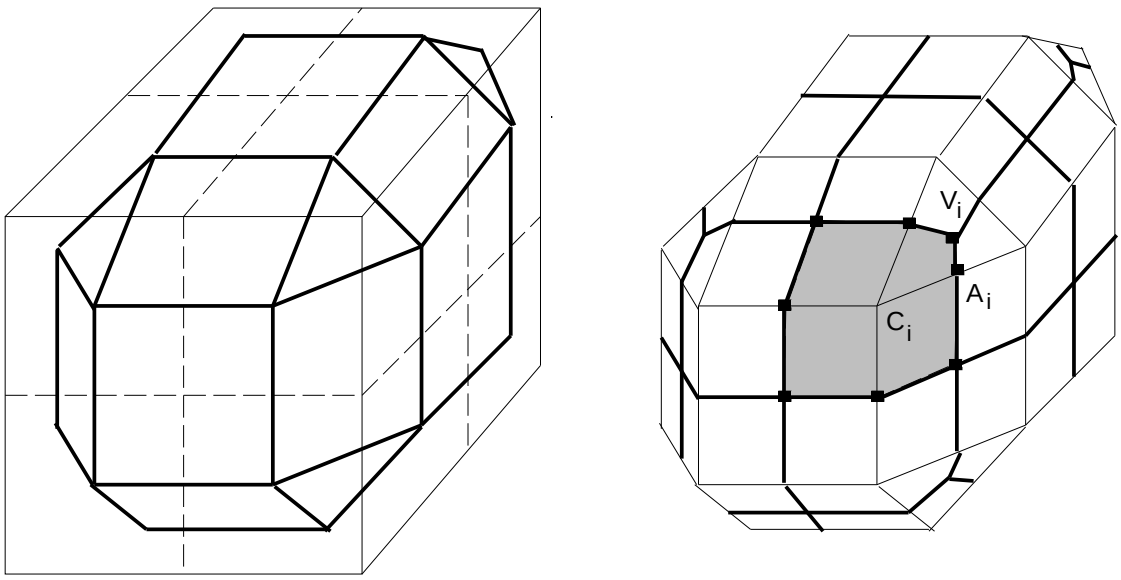

Figure 3: Mesh refinement procedure for irregular meshes

vertex of the original mesh. The cell centers C_i of the adjacent cells are now connected to each other. If the vertices of the input mesh are to be interpolated, then the cell centers $C_i, i = 1, \cdots, m$, of the refined mesh constructed around V are moved by $V - \sum_{i=1}^{m} \frac{C_i}{m}$. This process makes V the centroid of C_i, $i = 1, \cdots, m$ and will be interpolated by the construction to be described further. This first step of the algorithm is illustrated in the left diagram of Figure 3. The user starts with a cube. A cube has six faces and each face has four edges. Therefore each face of the cube is subdivided into four rectangles or cells. These four rectangles or cells are shown as dotted lines in the left diagram of Figure 3. In the next step, the cell centers C_i are computed for each of these rectangles. In this case, the cell centers are simply the centroids of each cell. These centroids are now connected to the centroids of the adjacent cells. Part of the the new configuration is shown in dark in the left diagram of Figure 3. The interpolation step where these cell centers are now moved to $V - \sum_{i=1}^{m} \frac{C_i}{m}$ is not shown in the figure. In the third step of the mesh refinement, the original vertices (which are the centroids of C_i) are introduced back into the mesh to ensure interpolation and they are connected with A_i. This process subdivides each s-sided face into s quadrilaterals as shown in the right diagram of Figure 3. Moreover, the output of this step is a refined mesh such that three points are associated with each cell edge. These vertices of the refined mesh can be construed as the control points of a quadratic box spline. The resulting box-spline surface will then be the solution to the scattered data interpolation problem. Alternative solutions consisting of quadratic and cubic triangular patches or biquadratic rectangular patches or a combination of these have also been described [Pet95]. For example, each of the quadrilateral subcells is covered with four triangular quadratic or cubic patches by inserting a control point S near the centroid of the quadrilateral subcell and connecting S with the vertices of the quadrilateral. In the right diagram of Figure 3, four such quadrilateral cells are shown in dark. Each of these cells are then further subdivided into four cubic patches (not shown in the diagram). The exact formula for S and the coefficients that define the cubic triangular patches are based on continuity conditions and described in [Pet95]. Figure 4 shows a twisted cubic consisting of quadratic and cubic polynomial patches. A similar approach is taken in constructing approximation to scattered data by using a refinement step and a patch-filling step using polynomial patches of degree 4 [Loo94].

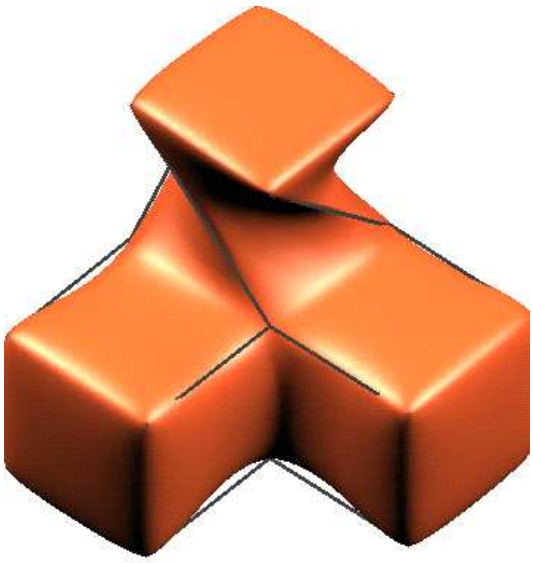

Figure 4: Scattered data interpolation over an irregular mesh

2.3 Surfaces on Surfaces

Trivariate Scattered Data: The triangle-based methods generalize to the trivariate case by considering tetrahedra instead of triangles, although additional complexity is possibly a problem. Assume the data points are given and a Delaunay tetrahedralization of the convex hull (or other convenient region) is computed. For smooth interpolation it is then necessary to have a function defined on each tetrahedron (a patch) that matches value and first derivatives across boundaries to adjacent tetrahedra. This can be achieved by various methods. Rescorla [Res87] uses a ninth degree patch due to Ženíšek[Zen73], with 220 degrees of freedom and requiring derivatives as high as fourth order to be estimated. Clearly a simpler patch is required, and there are several alternatives. The Clough- Tocher patch can be extended to tetrahedra [Alf84] using piecewise polynomials of degree five, with four tetrahedra in the split. There is an extension of the Clough-Tocher patch to n-dimensions, by Worsey and Farin [WF87]. There is also a generalization of the Powell-Sabin split [PS77] using piecewise quadratic polynomials, by Worsey and Piper [WP88], and while this offers easy exact contouring, the patch subdivides the tetrahedron into 24 subtetrahedra. Nielson and Opitz [NO92] generalized the side-vertex scheme [Nie79] to a face-vertex scheme for tetrahedra. This method yields a rational patch. Because the face data is not known from derivative estimates, it is necessary to estimate the value and normal derivative for each face. Nielson and Tvedt [NT94b] discuss more of the details of the required scheme. Several methods for estimating derivatives are possible. Based on experience with the bivariate case, this is a crucial step in the process and the quality of the approximation depends heavily on reasonable estimates. Most likely to give good results is a generalization of the minimum norm network method [Nie83, Pot92], generalized to tetrahedra by Nielson and Tvedt [NT94b], or generalizations of methods used by Renka and Cline [RC84] and Franke and Nielson [FN80] using local least squares quadratic approximations.

Function-on-Surface Problem: Several good surveys exist on constructing scattered data interpolants over surfaces [BF91, FN91, Nie93c, HL93]. These surveys include polynomial, rational, radial and shepard-type interpolants as well. There are two major approaches to solving the surface-on-surface problem: (i) trivariate approach, and (ii) bivariate approach.

Trivariate Approach: In this approach, the problem of constructing interpolants on surfaces is construed as a trivariate scattered data interpolation problem, that is, the problem of interpolating scattered data points $(x_i, y_i, z_i, w_i) \in R^4$. Although, the domain points (x_i, y_i, z_i) are restricted to lie on a surface Fin R^3, one can simply ignore this fact and build an interpolant on a volumetric domain that contains the surface by using any method for solving the trivariate or volumetric problem as discussed above. This approach was adopted in [Bar85]. This approach may yield acceptable results for surfaces where the geometry of the surface and the geometry of the Euclidean space R^3 is not too different. However, this approach does poorly on surfaces where the data points that are close in R^3 are far apart on the surface, for example in case of an airplane wing [HL93].

Bivariate Approach: The bivariate approach involves finding a bivariate function $f(x, y, z)$ (where (x, y, z) are restricted to lie on a surface $f \in R^3$) that interpolates the given data, that is, $f(x_i, y_i, z_i) = w_i$. To construct the interpolant, one usually has to take into account the geometry and topology of the surface F. This approach is typically much more complicated than the trivariate approach.

In addition to these two approaches, there are other hybrid approaches as well. For example, an interpolant can be constructed inside a volumetric tube surrounding the surface [BOP92]. Moreover, one can use a *domain mapping method* to extend trivariate or bivariate approaches to a much larger class of surfaces as explained below. Typically, one can solve the surface-on-surface problem on a specific surface such as a sphere using either trivariate or bivariate approach. In the domain mapping method [FLN$^+$90b], the idea is to map the surface F to a new surface G, say a sphere, using a mapping Φ in such a way that the geodesic distances on F are not too badly distorted so that the intrinsic geometry is more or less preserved. The new surface G is chosen in such a way that one can apply a direct trivariate or bivariate interpolation method to construct an interpolant $\hat{I}$ which interpolates the values w_i on the scattered data points $\Phi(x_i, y_i, z_i) \in G$. The desired interpolant is then given by $I(x_i, y_i, z_i) = \hat{I}(\Phi(x_i, y_i, z_i))$. Thus the domain mapping method can be used to solve the surface-on-surface problem for surfaces that are topologically equivalent to sphere, but need not be convex.

All these approaches have been applied to construct scattered data interpolants on surfaces. Algebraic, radial and Shepard-type interpolants on surfaces are described in Sections 3, 4, and 5 respectively. Here we discuss polynomial and rational interpolants on surfaces, that are based on triangulations (described in Sections 2.1 and 2.2) and those that have been extended from the planar case to spheres and convex surfaces. The first step is to construct a triangulation of scattered data points over spheres. As in the planar case, different criteria can be used to generate different triangulations. It is important to generate triangulations that cover the surface uniformly. The *spherical circumscribed circle criterion* ensures that any given data point lies outside the spherical circle that passes through three points defining a triangle [Law84, Ren84b]. The *spherical max-min angle criterion* ensures that for every quadrilateral in a triangulation defined by two adjacent triangles, the minimum of the six angles associated with the quadrilateral is greater than the minimum of the six angles associated with the quadrilateral obtained by swapping the diagonal [NR87]. These triangulation methods have been extended to *convex surfaces* using *one-sided property criterion* [BO90]. A triangle has the one-sided property provided that all other scattered data points lie on the same side of the plane passing through the three points defining the triangle. A triangulation possesses the one-sided property provided every triangle in the triangulation has this property. To build such a triangulation, one can start with some initial triangle and then add one point at a time ensuring that this property remains satisfied [BO90]. This algorithm leads to the circumscribed-circle criterion triangulation on a sphere. In the second step of the algorithm, the gradient of the function is estimated at every vertex essentially in the same way as in the planar

case by constructing a local least squares quadratic approximation. In the third and final step of the algorithm, every triangle is replaced by a polynomial or a rational surface patch that meets the adjacent patch with tangent plane continuity using the minimum norm network method [NR87]. This technique has been further extended to the case of closed surfaces that are topologically equivalent to a sphere by using domain mapping method [FLN+90b]. Minimum norm network method has also been extended to surfaces that are at least C^2 differentiable [Pot92].

Recently, scattered data interpolants have been suggested on spheres and sphere-like surfaces based on spherical Bernstein-Bézier polynomials and their generalizations to spherical triangular B-splines [ANS96, PS95].

3 Algebraic Solutions

An alternative to the parametric solutions discussed in the previous section is to construct functions or surfaces in *implicit* form, that is, the function or the surface can be represented as the zero set of some function in R^3. More formally, the solution is represented as some suitable subset of $f(x, y, z) = 0$. Moreover, the defining function f is often chosen to be a polynomial. The solutions are then referred to as *algebraic* solutions. The degree of the polynomial f is then referred to as the algebraic degree of the solution. Observe that an implicit form can be achieved trivially for functions defined as $z = g(x, y)$ by rewriting them as $f(x, y, z) = z - g(x, y) = 0$. Therefore, most of this section focuses on the problem of surface reconstruction.

Relationship between Parametric and Algebraic Surfaces: The relationship between parametric and algebraic solutions is very interesting and rich. A rational parametric surface of parametric degree n can always be represented as an algebraic surface of algebraic degree at most n^2 [SAG84]. This process of converting from rational parametric surfaces to algebraic surfaces, referred to as implicitization, remains computationally very expensive in general. In practice, therefore, this conversion is rarely attempted. In this section, therefore, an algebraic surface refers to a surface that has been *defined* using an implicit form, that is, as $f(x, y, z) = 0$.

One interesting situation, when implicitization becomes a little more tractable and therefore practical, arises when there are base points in the rational parametric representation. A base point of a rational parametric representation is a point (u, v) for which $x(u, v) = y(u, v) = z(u, v) = w(u, v) = 0$, that is both the numerator and the denominator vanishes and therefore, the surface is undefined at those points in the rational parametric form. It has been proved that presence of k base points (counted with appropriate multiplicities) in the rational parametric representation of a surface with parametric degree n reduces its algebraic degree from n^2 in general to $n^2 - k$ [Chi90]. Therefore, presence of base points is advantageous in constructing algebraic surfaces of low degree. Tensor product surfaces of parametric degree $m \times n$ always have at least $2mn$ base points. S-patches (which have base points) and another rational multisided generalization of Bernstein-Bézier surfaces with base points have been used to fill holes [LD90, War92].

In the reverse direction, the procedure for converting algebraic surfaces into rational parametric form is known as *parametrization*. For $n > 3$, there always exists an algebraic surface of degree n that does not have a rational parametric representation of any degree at all [SAG84]. Fortunately, the scenario for low degree algebraic surfaces is rather promising. An algebraic surface of degree 1 is simply a plane, and therefore can easily be represented as a polynomial parametric surface of degree 1 and vice-versa. An algebraic surface of degree 2 referred to as *quadrics* include such well-known shapes as spheres, cylinders, cones, ellipsoids etc. It is also well-known that any quadric can be represented as a rational parametric surface of degree 2. Algebraic surfaces of degree 3 referred

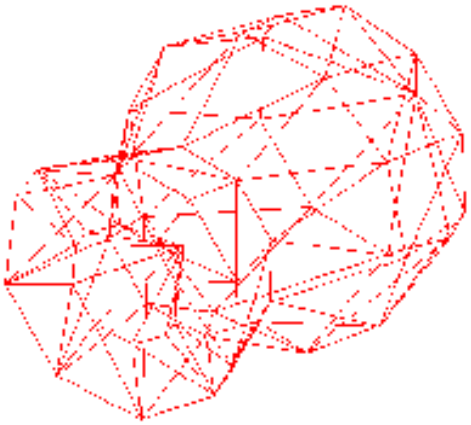

Figure 5: Triangulated scattered data points

Figure 6: C^1 interpolation using cubic algebraic A-patches

to as *cubics* do admit a rational parametric representation of degree 3 at most with the exception of cones and cylinders generated by non-rational cubic curves [Sed90a]. Sederberg has carried out a detailed study of cubic algebraic surfaces in the context of computer aided geometric design [Sed90a, Sed90b]; still an automatic method of constructing a rational parametrization for a cubic algebraic surface needs to be explored. Algebraic surfaces of degree 4 referred to as *quartics* include surfaces such as the torus and cyclides. Quartic surfaces have been studied by classical geometers and classified into several categories [Sal14]; still a constructive algorithm for parametrization of those quartics that are parametrizable has not yet been developed. We shall soon see that most scattered data interpolation problems can now be solved using low degree algebraic surfaces, in most cases by surfaces with algebraic degree up to 3.

Algebraic Interpolants: Algebraic surfaces have been used to solve both the surface reconstruction as well as surfaces-on-surfaces problem. Sederberg [Sed85] pioneered the use of algebraic surfaces in computer aided geometric design. Patrikalakis and Kreizis [PK89] considered algebraic surfaces in a tensor product B-spline form for regular grids. Algebraic surfaces have been used to interpolate a given set of points, normals and algebraic space curves by choosing the degree of the algebraic surface sufficiently high in order to contain the given data [BI92a]. This approach typically constructs one algebraic surface to fit the given data and works well for surface blending applications, such as the smooth join of four cylinders with a quartic surface.

Bajaj and Ihm [BI92b] constructed a piecewise C^1 continuous smooth algebraic interpolant to a triangulated set of scattered data by adopting an approach that is quite similar to the approach adopted in fitting piecewise polynomial interpolants to the scattered data. The three steps in this approach are: (i) Each edge of the triangulation is replaced by a quadric or cubic algebraic space curve; (ii) A set of normals are prescribed along these curves; (iii) Each triangular facet is replaced by one algebraic surface of degree at most seven that interpolates the constructed curves and the normals.

More recently, the construction of piecewise smooth algebraic interpolants to scattered data

utilizes the following steps: (i) The triangulated set of scattered data points is surrounded by a network of tetrahedra, within which the algebraic interpolant is built. In order to make the interpolant continuous across the tetrahedra, the tetrahedra is sometimes split into several tetrahedra; (ii) Within each tetrahedron, the interpolant is expressed in terms of Bernstein-Bézier form, and the degree of the interpolant is chosen sufficiently high to ensure the desired continuity across the tetrahedra. Continuity conditions impose certain restrictions on a subset of control points. Remaining control points are chosen by user or by some default procedure in order to manipulate the shape of the interpolant.

This approach was followed by Dahmen [Dah89] to construct C^1 piecewise smooth quadric interpolants using 6 patches per triangle. Dahmen [DTS93] and Guo [Guo91] constructed C^1 piecewise continuous cubic algebraic surfaces using 3 patches per triangle. However there are three major difficulties in these solutions. First, it is a non-trivial task to construct a surrounding network of tetrahedron within which the trivariate polynomial is built. In particular, this network of tetrahedra can run into the problem of self-intersections. The second difficulty is that these solutions can have multiple sheets within the bounding tetrahedron. To determine which of these sheets are extraneous and which sheet represents the solution is a non-trivial task. The third difficulty is that the solutions can be singular, for example, the solution may intersect itself within the bounding tetrahedron. In addition, the added challenge is to construct an interpolant with low degree and with as few patches as possible.

To overcome these difficulties several heuristics were suggested in the solutions proposed by Dahmen and Guo. More recently Bajaj and Xu [BCX94] have constructed algebraic surface patches referred to as *A-patches* that are guaranteed to be non-singular and single sheeted within each tetrahedron [BCX95]. Moreover they enumerate the exceptional situations encountered in building the network of tetrahedra surrounding the triangulation in R^3 and provide strategies for rectifying them. Moreover, they use one cubic A-patch per face of the triangulation, except for two special cases where 3 patches per triangle are used. Therefore the solution consists of much fewer number of patches than in earlier approaches. This technique has also been extended to provide solutions by C^2 smooth mesh of A-patches of degree 5. Figure 5 shows the triangulated scattered data set. Figure 6 shows the interpolatory cubic A-patches to fit the scattered data. More recently, various approaches for constructing functions on surfaces described in Section 2.3 have been extended to construct algebraic surfaces [BX94]. These algebraic models use C^2 quintic A-patches without splits, and construct approximate solutions to the the problem of surfaces-on-surfaces.

4 Radial Basis Function Methods

One of the main advantages of radial basis function techniques is that unlike most other methods this approach does not require any information about the connectivity of the scattered data points. In addition, the radial interpolants are translation and rotation invariant. Moreover, some of the radial basis functions, in particular multiquadrics and thin plate splines have performed very well in practical applications [Fra82a, Har90]. Radial basis function techniques have mostly been used to solve the function reconstruction problem. Variations of these techniques have also been used to solve the surface-on-surface problem.

A function $\phi(r_k)$, where $r_k = \sqrt{(x - x_k)^2 + (y - y_k)^2}$ is referred to as a *radial function*, because it depends only upon the Euclidean distance between the points (x, y) and (x_k, y_k). The points (x_k, y_k) are referred to as *centers* or *knots*. In particular, the function $\phi(r_k)$ is radially symmetric around the center (x_k, y_k). The solution to the scattered data interpolation problem is obtained by considering a linear combination of the translates of a suitably chosen radial basis function.

Sometimes a polynomial term is added to the solution when ϕ_k is conditionally positive definite (defined later in this section), or in order to achieve polynomial precision. More formally, the solution to the interpolation problem is sought in the following form:

$$F(x,y) = \sum_{k=1}^{N} A_k \phi(r_k) + \sum_{l=1}^{M} B_l q_l(x,y), \tag{1}$$

where $q_l(x,y), l = 1, \cdots, M$ is any basis for the space P_m of bivariate polynomials of degree less than m, and therefore $M = \frac{m(m+1)}{2}$. Notice that $m = 0, 1, 2$ corresponds to the case when no polynomial, constant function or a linear polynomial is added to the interpolant respectively. In order to satisfy the interpolation conditions, one poses the following system of N linear equations in N unknowns A_k, $k = 1, \cdots, N$, when $m = 0$:

$$\sum_{k=1}^{N} A_k \phi(r_{ik}) = f_i, i = 1 \cdots, N, \tag{2}$$

where $r_{ik} = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2}$ is the Euclidean distance between the points (x_i, y_i) and (x_k, y_k). When $m \neq 0$, a slightly modified system of $N + M$ linear equations in $N + M$ unknowns $A_k, k = 1, \cdots, N$ and $B_l, l = 1, \cdots, M$ is formulated as follows:

$$\begin{aligned} \sum_{k=1}^{N} A_k \phi(r_{ik}) + \sum_{l=1}^{M} B_l q_l(x_i, y_i) &= f_i, i = 1, \cdots, N, \\ \sum_{k=1}^{N} A_k q_l(x_k, y_k) &= 0, l = 1, \cdots, M. \end{aligned} \tag{3}$$

In addition, throughout this section we shall assume the following mild geometric condition on the location of scattered data points:

$$p(x,y) \in P_m, p(x_i, y_i) = 0, i = 1, \cdots, N \Rightarrow p \equiv 0. \tag{4}$$

Notice that this geometric condition is vacuous for $m = 0, 1$. For $m = 2$, this condition states that all the scattered data points do not lie on a straight line. Assuming that a solution to the system of equations (2) or (3) exist, the radial basis interpolant is then given by Equation (1).

One of the most popular radial basis functions was introduced by Hardy more than 25 years ago [Har71]. These basis functions are referred to as *Hardy's multiquadrics*. Multiquadrics are defined as $\phi(r_k) = \sqrt{r_k^2 + h^2}$, where h is a suitably chosen parameter. Observe that the multiquadrics *grow* as the distance from the centers increase. This appears as a counter-intuitive choice of radial basis functions to many, and therefore *inverse multiquadrics* was invented and was also applied very successfully [Har90]. Inverse multiquadrics are defined as the inverse of multiquadrics, that is, $\phi(r_k) = \frac{1}{\sqrt{r_k^2 + h^2}}$. Inverse multiquadrics decay as the distance from the centers increase. It is amazing that it was not until 1985 that it was established that the linear system of equations (2) and (3) is solvable for multiquadrics as well as inverse multiquadrics [Mic86]. This however did not distract the early practitioners of this method and the evidence of the success of Hardy's multiquadrics mounted over several years [Har90]. At the time when Hardy's multiquadrics and inverse multiquadrics were being applied very successfully to several applications without much theoretical justification, *thin plate splines* were introduced by Harder and Desmarais [HD72]. Duchon [Duc76] developed a complete theory of thin plate splines with theoretical justification based on

variational principles. Thin plate splines were further studied by Meinguet, Madych and Nelson [Mei79, MN88]. Thin plate spline basis functions are defined as: $\phi(r_k) = r_k^2 \log r_k$. Observe that the thin plate splines also grow as the distance from the centers increase. These basis functions were also found to be successful by Franke [Fra82a]. For the thin plate splines, although the system of equations (2) may become singular for certain locations of data points, the system of equations (3) is always solvable for $m \geq 2$. In other words, addition of at least a linear polynomial is required to guarantee the existence of solution in the case of thin plate splines.

In addition to the three radial basis functions, multiquadrics, inverse multiquadrics and thin plate splines, there are a number of other radial basis functions that have been considered by various researchers including odd powers of distance function, that is, the linear distance function $\phi(r_k) = r_k$, and the cubic distance function $\phi(r_k) = r_k^3$, the Gaussian radial basis function $\phi(r_k) = e^{-h^2 r_k^2}$, and the shifted thin plate splines $\phi(r_k) = (r_k^2 + h^2)\log(r_k^2 + h^2)^{\frac{1}{2}}$. There are several good surveys on theoretical developments of radial basis functions [Dyn87, Dyn89] including recent in-depth surveys by Powell [Pow90], Light [Lig92], Buhmann [Buh93], and Schaback [Sch95a] that address these and other radial basis functions. These mathematical surveys describe the results in great generality. These surveys also contain an extensive bibliography on radial basis functions for the interested reader. Also, a variety of radial basis functions based on trigonometric and exponential functions have been extensively used in mining, geology and meteorology from a statistical point of view to study kriging [Cre91, FBG88, Fra86]. In particular, Franke [Fra86] lists 24 explicit formulas for these radial basis functions and is an excellent starting point for those interested in this literature. Here we describe the conditions that guarantee the existence of radial basis interpolants. Then, in order to make this presentation more accessible to the practitioners of this field, we report on some practical and useful properties of specific radial basis functions such as multiquadrics, inverse multiquadrics and thin plate splines.

Existence: To describe the existence of radial basis interpolants, we begin with a few definitions. Given a radial function $\phi(r_k)$ and N scattered data points, consider the $N \times N$ square symmetric matrix $A = (a_{ij})$, where $a_{ij} = \phi(r_{ij})$. The radial function $\phi(r_k)$ is said to be positive definite iff $v^t A v \geq 0$ for all $v \in R^n$. The radial function $\phi(r_k)$ is said to be strictly positive definite if in addition, $v^t A v > 0$ whenever $v \neq 0$. If a radial basis function is strictly positive definite, then the matrix A is invertible. This is exactly what is needed in order to solve the system of linear equations (2) and guarantee the existence of an interpolant.

However, as mentioned before, for some radial basis functions such as thin plate splines, the matrix A is not always invertible and addition of at least a linear polynomial term to the interpolant is required. To describe these existence results in somewhat greater generality, the notion of conditionally positive definiteness is introduced. Let P_m denote the space of polynomials of degree less than m. Consider the collection V of vectors $v = (v_1, \cdots, v_N)$ in R^N that satisfy $\sum_{i=1}^{N} v_i q(x_i) = 0$ for any $q \in P_m$. The radial function $\phi(r)$ is said to be conditionally positive definite (cpd) of order m iff $v^t A v \geq 0$ for all $v \in V$. The radial function $\phi(r)$ is said to be conditionally strictly positive definite (cspd) of order m if in addition, $v^t A v > 0$ whenever $v \neq 0$. It can be proved with a little effort that the the system of equations 3 is uniquely solvable if the radial basis function is conditionally strictly positive definite of order m and the scattered data points satisfy the geometric condition (4).

Micchelli provided the following characterization for conditionally positive definite functions and derived the following important result:

Theorem [Mic86]: A function $f(t)$ is conditionally positive definite of order m in R^d for $d \geq 1$, if and only if $(-1)^j \frac{d^j}{dt^j} f(\sqrt{t}) \geq 0$, $t > 0$, $j \geq m$. If in addition $\frac{d^m}{dt^m} f(\sqrt{t}) \neq constant$, then $f(t)$ is

conditionally strictly positive definite of order m.

It is now easy to verify that multiquadrics ($m \geq 1$), inverse multiquadrics ($m \geq 0$), thin plate splines ($m \geq 2$), linear distance function ($m \geq 1$), cubic power of the distance function ($m \geq 2$), shifted thin plate splines ($m \geq 2$), and Gaussians ($m \geq 0$) are cspd of order m up to a constant multiple, that is, either these functions or their negatives are cspd of order m. Some of these results derived from the theorem above can be strengthened further. In particular, the scattered data interpolation problem is solvable with $m = 0$ and $f(t)$ cspd of order 1, whenever $f(t) < 0$ for $t > 0$. This result guarantees the solvability of interpolation problem for multiquadrics without any addition of a constant or a polynomial term. Since so many choices of m are available, what is an appropriate m to choose while using these interpolants? We address this question later in this section under the discussion on the reproduction of polynomials.

Cardinal Interpolation: We first include a brief discussion on cardinal interpolation because it provides some insights why radial basis functions that *grow* with distances may work well.

A cardinal interpolant is obtained by considering the following scattered data interpolation problem: choose exactly one of the data values to be 1 and all the rest of the data values to be 0. Let us denote such an interpolant to be $\psi_i(x, y)$ where i denotes the location of the data point, where the data value is 1. The importance of the cardinal interpolation arises from the hope that the general solution to any scattered data interpolation problem can be obtained as a linear combination of cardinal interpolants, namely, $\sum_{i=1}^{N} f_i \psi_i(x, y)$. This certainly would be true, if ψ_i were *finite* combinations of radial basis functions. In general, however, ψ_i are obtained by infinite combinations of radial basis functions as $\psi_i(x, y) = \sum \mu_k \phi(r_k)$. The construction of the final solution in terms of linear combinations of cardinal interpolants then depends upon the convergence of certain infinite series.

Unfortunately, cardinal interpolation by radial basis functions for scattered data locations is in its infancy [BDL95, DR95]. However cardinal interpolation by radial basis functions when the data is prescribed on an infinite regular integer lattice is well understood. For this infinite gridded data it can be proved that the the coefficients μ_k decay so rapidly that the cardinal interpolant ψ_i exists for linear, cubic, thin plate splines, multiquadrics, inverse multiquadrics and Gaussian radial basis functions [Pow90]. Intuitively, appropriate cancellation takes place for fast growing radial basis functions at locations far away from the data center where it achieves the value 1.

Choice of Parameters: Some of the radial basis functions such as multiquadrics, inverse multiquadrics, Gaussians and shifted thin plate splines depend upon a parameter h. Currently there is little theory to guide a user towards an appropriate choice of this parameter. Apart from a few isolated theoretical results, most of the results are experimental. These results indicate that the performance of the radial interpolants depends critically on the choice of this parameter [CF91]. In the initial experiments, this parameter was chosen as the average distance between data points and performed quite well [Fra82a, Har90]. However this choice of the parameter did not perform very well on track data. Track data is obtained when the data points are close together along a path or a track, but the distance between the tracks is one or more orders of magnitude higher. This type of data arises in important practical applications, for example, oceanographic and meteorological data collected on a ship. The accuracy of the multiquadric and inverse multiquadric interpolant on track data was improved by using different choices of this parameter [CF91]. Subsequently the accuracy of multiquadric and inverse multiquadric interpolants were further improved by choosing a parameter that minimized the difference between multiquadric and inverse multiquadric interpolants [Fol94]. Different parameter values at different data points have also been used to improve the accuracy of these interpolants [KC92]. However, note that it is easy to construct data sets for

which the matrix for Equation 3 is singular when different parameter values are used at different points. Improvements of multiquadric interpolant by using a linear combination of basis functions centered at the same point but with different values of the parameter have also been achieved [BM92]. Floater and Iske have also also constructed interpolants for track data in several steps using compactly supported radial basis functions that we discuss later in this section [FI96a].

Selection of Centers: So far we have considered only the case when the centers for radial basis interpolant are located at the data points. However this need not be true. The question is what happens if the centers are located elsewhere? There is a theoretical reason that the interpolation is optimal if the centers are located at the data points [Sch95a]. Experiments with location of centers have mostly focused on approximating methods [MF92]. Although least squares approximation is most popular, other objective functions have also been considered for minimization [Sal92, Gir92]. The attraction is to achieve a good approximation with fewer centers. This can reduce the computation time in constructing and evaluating the interpolant or the approximant. Results indicate that significant savings can result at the cost of introducing very small errors [CN94]. However, the behavior of the condition number and therefore the problems of the stability of the computation (depending on the spacing between the data points relative to the multiquadric parameter) for interpolation by multiquadric functions carries over to least squares approximation. Sivakumar and Ward [SW93] have given upper bounds for the norms of the inverses of the normal equations associated with the least squares approximation problem.

We now describe recent results on least squares approximation to scattered data using multiquadric functions. Franke, Hagen, and Nielson [FHN94] described a greedy method for selection of centers (knots) for least squares approximation. Still better results were obtained by allowing both the multiquadric parameter and the knot locations to be determined by a nonlinear least squares process. A possible problem is the coalescence of knots, leading to poor conditioning of the least squares problem. In a subsequent paper [FHN95] the combined knots were replaced by a single repeated knot (the multiquadric basis function and its two partial derivatives as basis functions at the knot). This generally has a favorable impact on the condition number of the problem. Recent work has used an idea suggested by Sampson and Guttorp [SG92] in the context of fitting covariance functions to empirical covariance data. The basic idea is to simultaneously determine a transformation of the domain region and the parameters of the fitting function which are calculated in the transformed domain. The transformation used by Sampson and Guttorp included no guarantee of being one-to-one. However, if the transformation is one-to-one, then the important property of positive definiteness of covariance functions follows to the new domain. This idea has been applied by Franke and Hagen [FH97] to construct a biquadratic transformation of the domain in terms of Bézier control points that is one-to- one, and thus the property of conditional positive definiteness of the multiquadrics is carried over to the new domain. We now briefly explain how this biquadratic transformation takes up degrees of freedom equivalent to 5 knots. A biquadratic transformation in two dimensions can be described by 9 control vertices and has 18 degrees of freedom in two dimensions to begin with. However, one control point is fixed at the origin, taking up two degrees of freedom. Another control point is constrained to lie on the (positive) x-axis, taking up another degree of freedom. Four control points are constrained to be on the straight line segments defined by the corners, taking up additional four degrees of freedom. The remaining eleven parameters determine the locations of the control points in such a way that the region is a convex quadrilateral in the upper half plane with the center control vertex inside the region. Because there is no constraint on the size of the transformed region, including the multiquadric parameter as a parameter would be redundant. Hence the value of the multiquadric parameter is taken to be one on the transformed domain, leaving ten additional parameters as compared to the

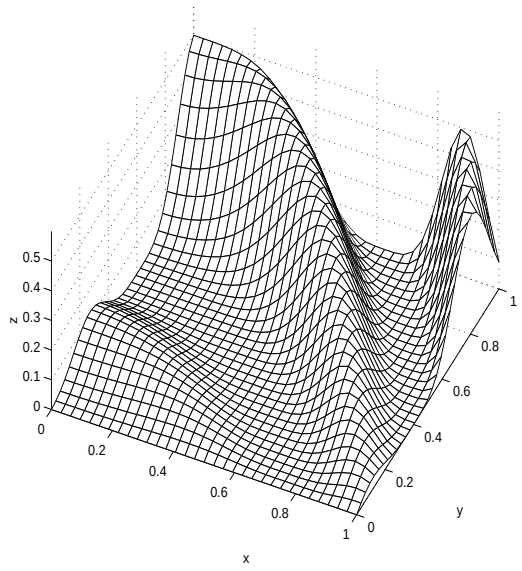

Figure 7: Original function

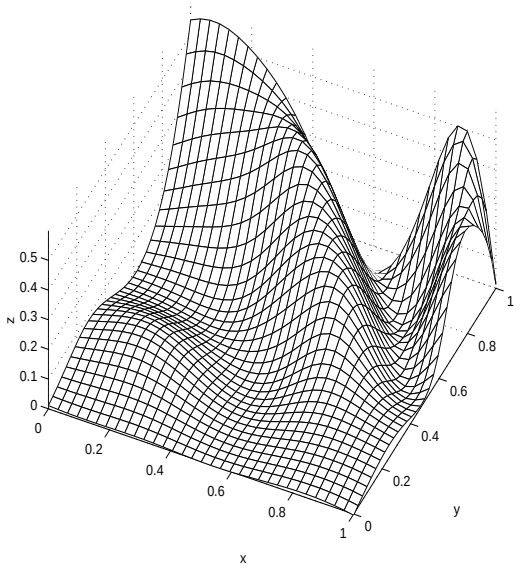

Figure 8: Approximation [FHN94]

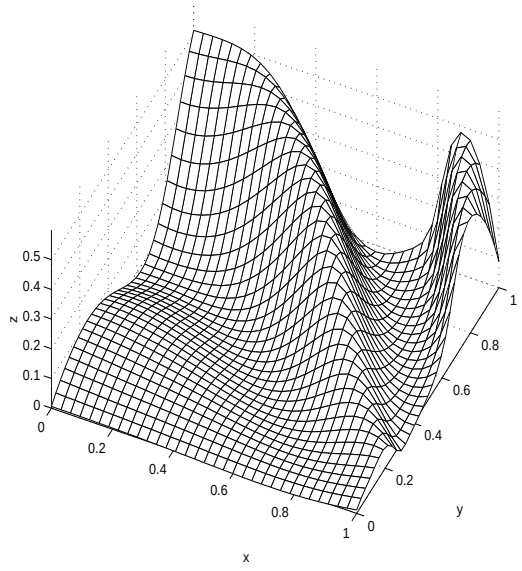

Figure 9: Approximation obtained by combining nonlinear least squares technique with domain transformation described in [FH97]

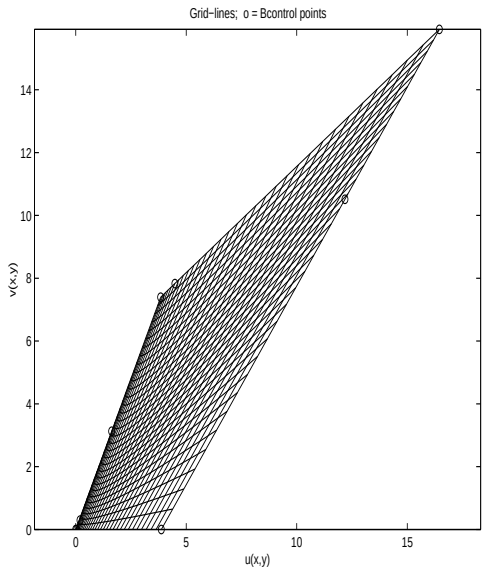

Figure 10: The transformed domain from the unit square used in obtaining the approximation shown in Figure 9

"old" multiquadric approximation. Ten additional parameters is equivalent to five free knots in two dimensions.

The potential for improved fits to scattered data with a few knots is shown by the following sequence of figures. The parent function from which the 200 data points were taken is shown in Figure 7. Figure 8 shows the approximation to the surface using the techniques of [FHN94] to obtain the knot locations and a nonlinear least squares method with 17 knots. Figure 9 shows the results obtained by using the techniques of [FH97] combining the domain transformation with the approximation being computed in the transformed domain, using 12 knots, since degrees of freedom equivalent to 5 knots are taken up by the biquadratic transformation as described in the previous paragraph. Figure 10 shows the transformed domain with the transformed equally spaced grid lines from the unit square, emphasizing the relatively benign behavior of the function near the x=0 edge (note that the grid lines are compressed there) compared to the x=1 edge. The root mean square error .0122 of Figure 9 compares favorably to the root mean square error .0210 of Figure 8.

Reproduction of Polynomials: We now address the question of what is an appropriate m to choose in Equation (3) while using a radial basis function. By adding a polynomial of degree m to the interpolant, observe that if the values at the scattered data points are obtained by sampling any polynomial $p(x, y)$ of degree m or less, then the system of equations (3) always has a solution by choosing $A_k \equiv 0$ so that the interpolant reduces to the polynomial $p(x, y)$. This property of the interpolant is referred to as the property of reproduction of polynomials or *polynomial precision*. To reemphasize this point, consider addition of a constant or a linear polynomial to the multiquadrics or inverse multiquadrics. Such a modified interpolant with the associated conditions (3) would reproduce the constant or the linear polynomial if the data values were constants or were obtained by sampling a linear polynomial respectively. In practice, multiquadrics and inverse multiquadrics are implemented without any polynomial term at all or at most with addition of a constant [FHN94], while the thin plate splines require and are usually implemented with addition of a linear polynomial term. More generally, the minimum value of m that guarantees the existence of the interpolant is often used. Experiments seem to indicate that addition of polynomial terms do not seem to improve the accuracy of the interpolant for non-polynomial functions [CF91].

There is another deeper reason for considering reproduction of polynomials. In numerical analysis and approximation theory, the property of reproduction of polynomials is often crucial in establishing good approximation properties of the interpolant. This belief originates in the work by Fix and Strang [FS69] and has been very influential in constructing interpolants including construction of subdivision-based interpolants that we discuss later in Section 6. Fix and Strang linked the good convergence behavior of an interpolant (not necessarily a radial basis function) with its polynomial reproduction properties by deriving certain necessary and sufficient conditions on the Fourier transform of a finite set of functions. Later Jia pointed out an error in the statement by Fix and Strang and provided a counterexample [Jia86]. However, de Boor and Jia were able to establish this link again under slightly different conditions for functions of compact support. This result was slightly improved by Jackson [dBJ85]. A complete link between good convergence behavior and the polynomial reproduction properties was however reestablished by Cheney and Light by again deriving necessary and sufficient conditions on the Fourier transform of a finite set of functions with rapid decay [CL94]. A most readable account of these results is available in [Lig92].

One of the consequences of the Cheney-Light theory discussed in the previous paragraph is that the radial basis functions that have bounded integral (and therefore necessarily decay away from the centers) such as Gaussians or radial basis functions with compact support may *not* work well for an infinite gridded data on an integer lattice when the parameter h or the support radius involved in these radial basis functions scales inversely with the data density [Jac88, Pow90]. For example,

Gaussians $e^{-h^2 r_k^3}$ do not make good interpolants on an infinite integer grid when h is proportional to the minimum distance between the data points. In such cases, Gaussians do not reproduce even constants or any polynomials in any dimension and are poor choice for interpolants. In contrast, for such gridded data, if the data values are sampled from a polynomial of degree m, and if cardinal interpolation is used to construct an interpolant, then the radial interpolant actually reproduces the given polynomial in case of linear, cubic, thin plate splines, multiquadrics and inverse multiquadrics with $m = d, d+2, d+1, d-2$ and d respectively where d is the dimension of the underlying space except for the case $d = 1$ for inverse multiquadrics [Pow90]. In particular, for bivariate polynomials, polynomials of degree up to 2, 4, 3, 0 and 2 are reproduced by linear, cubic, thin plate splines, multiquadrics and inverse multiquadrics. Another important observation is that this property of reproduction of polynomials improves with increasing dimensions.

Based on these observations on infinite grids, Gaussians and other integrable radial basis functions were said to have "a severe disadvantage", because they are not exact on constant functions [Buh89]. The other choices, especially multiquadrics, were said to be "far superior". However, as stated above, this theory is applicable only to the setting where the parameter h scales inversely with the data density. However, if the parameter h is fixed, it has been shown that Gaussians have an even better convergence behavior than multiquadrics for scattered data [MN92]. In fact, in other circumstances as well, even when h is allowed to vary discreetly, so that Gaussians do not satisfy the necessary and sufficient conditions linking good convergence behavior with polynomial reproduction property, Gaussians also have a good convergence behavior [BL92]. Therefore, the most recent research seems to break down the "classical wisdom" that the advantage of a particular radial basis function be tied to its polynomial reproduction properties. We shall comment about this later again in this section when disussing radial basis functions with compact support.

Computation and Evaluation of the Interpolant: Even though the theory guarantees the existence of the radial basis interpolant, the work to compute and evaluate the interpolant still remains. Direct or simple iterative methods for solving a system of N linear equations requires $O(N^2)$ storage and $O(N^3)$ computations. This is very expensive for large data sets. To make matters worse, the entries in the matrix A grow away from the diagonal for radial basis functions that grow away from centers such as multiquadrics and thin plate splines. This leads to very unstable computations while inverting the matrix A unless "preconditioning" is applied as described in the next paragraph below. Narcowich and Ward have done extensive work on studying the stability of solving these equations [NW91, NW92]. Both lower and upper bounds on the norm of the inverse matrix are discussed in [Buh93].

To overcome this difficulty, Dyn, Levin and Rippa proposed a scheme based on finite differences that enhances the stability of the computation of radial basis interpolants based on thin plate splines and shifted thin plate splines for scattered data sets [DLR86]. This technique, referred to as *preconditioning*, also enhances accuracy of multiquadric methods, but to a very limited extent. The proposed technique works for all conditionally positive definite functions of positive order. More recently, another technique has been proposed for efficient computation of multiquadric interpolants, although only for gridded data [Bax92].

A direct technique for evaluating a radial basis interpolant at one point is $O(N)$. Beatson, Newsam and Powell [BN92, Pow92a, Pow92b] have proposed techniques for fast evaluation of thin plate spline interpolants. The key idea is to lump many terms together for centers that are far away from a center, and approximate the resulting partial sum by a single truncated Laurent expansion. This technique has an initial set up cost of $O(N \log N)$ to identify a hierarchical set of indices that define the centers that are suitable for grouping. However, after this initial set up cost, this leads to an evaluation algorithm which is $O(1)$ per evaluation. These techniques also enable

the computation of the thin plate spline interpolants in $O(N)$ storage and far fewer than $O(N^3)$ computations [BL97].

4.1 Radial Basis Functions with Compact Support

Instability and computational cost associated with the computation of the radial interpolants that grow with distances away from the centers makes it unlikely that these radial basis functions can be applied to large scattered data sets that consist of more than tens of thousands of points, that typically arise in many scientific data sets. This limitation of growing radial basis functions has prompted researchers to work with radial basis functions which decay with distances or even better, radial basis functions with compact support [Sch95a, GC98]. Observe that all the radial basis functions mentioned so far do not have compact support. The decaying radial basis functions, such as Gaussians and inverse multiquadrics have often been employed in practice by truncating them. Another popular radial basis function with compact support is rotated cubic B-splines. However Wu has recently established that rotated B-splines of order $2m$ are not positive definite on any R^d for $d > 1$ [Wu94a]. Therefore, interpolants based on rotated B-splines could lead to a singular problem. Schaback and Wendland [SW94] established that Euclidean hats and radialized tensor-product B-splines are positive definite radial basis functions with compact support and therefore guarantees the solvability of the interpolation problem. Euclidean hats are polynomial functions only in odd dimensional spaces. Radialized tensor-product B-splines are not polynomials and are somewhat difficult to compute.

Recently, Wu has constructed compactly supported positive definite radial basis functions that are piecewise polynomials with breakpoints only at zero and at the boundary of the support [Wu94b]. The construction starts with functions $f_k(x) = (1 - x^2)_+^k$, $x \in R$, $l \geq 0$. Univariate convolutions of these functions are taken as follows: $\phi_{k,0}(x) = (f_k * f_k)(x)$. Univariate derivatives of these convolutions are formed as follows for $0 \leq k \leq l$: $\phi_{k,l} = D^l \phi_{k,0}$, where $D = -\frac{1}{x}\frac{d}{dx}$. These derivatives are positive definite of order $2l+1$. Moreover, these functions are piecewise polynomials of degree at most $4k - 2l + 1$ with breakpoints only at zero and at the boundary. The order of continuity is C^{2k-l} at the boundary but C^{2k-2l} at zero. Therefore, for $l > 0$, singularities at the boundary are less severe than the singularities at the zero. This property has a very positive effect on the visual appearance of the solutions. Explicit formulas for many of these radial basis functions appear in [Wu94b], for example, up to constant multiples, $\phi_{2,1} = (1 - r)_+^4(4 + 16r + 12r^3 + 3r^3)$ and $\phi_{3,1} = (1-r)_+^6(6+36r+82r^2+72r^3+30r^4+5r^5)$. $\phi_{3,1}$ is a C^4 function at the zero, C^5 at the boundary and is of degree 11. Initial results on using these radial basis functions for scattered data interpolation seem to be promising [Sch95a]. Figure 11 shows the well-known Franke's analytic function with two hills, a valley and a saddle. Figure 12 shows the thin plate spline interpolant for the Franke's function on a 5×5 grid. Figure 13 shows the function $\phi_{2,1}$. Figure 14 shows Wu's C^0 interpolant to the Franke's function by using $\phi_{1,1}$ with support radius of .499 on a 5×5 grid. This figure highlights the pitfalls associated with interpolating by functions with compact support in general. Figure 15 shows Wu's C^2 interpolant to the Franke's function by using $\phi_{2,1}$ with support radius of .499 on a 5×5 grid. This means that there are only 9 interpolation points at most in each support, and even fewer points in supports near the boundary. This lack of points near the boundary introduces wiggles near the boundary as shown in Figure 15. These wiggles are ironed out by using a support radius of 1.5. Figure 16 shows the interpolant obtained by using $\phi_{3,1}$ with support radius of 1.5 on a 5×5 grid. The choice of the support radius is critical in these applications. Increasing the support radius improves the quality of the results at the cost of more contributions in each support.

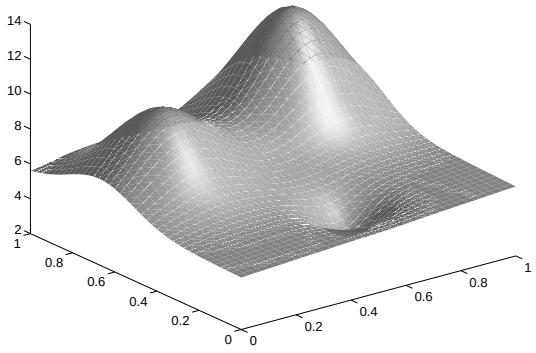

Figure 11: Franke's function

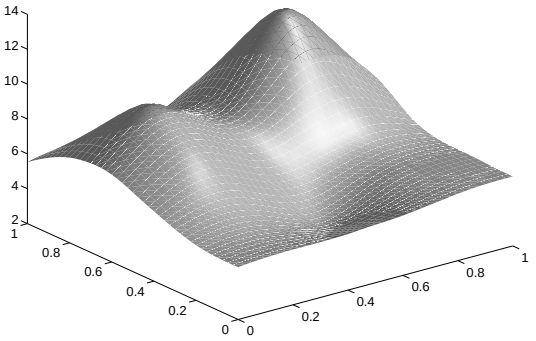

Figure 12: Thin plate spline interpolant

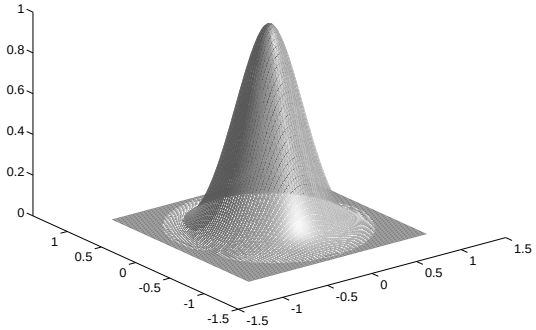

Figure 13: Wu's function $\phi_{2,1}$

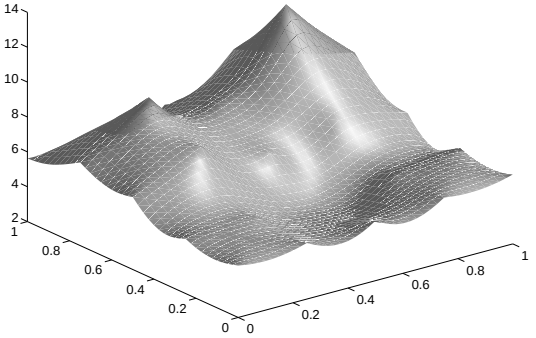

Figure 14: Wu's C^0 interpolant

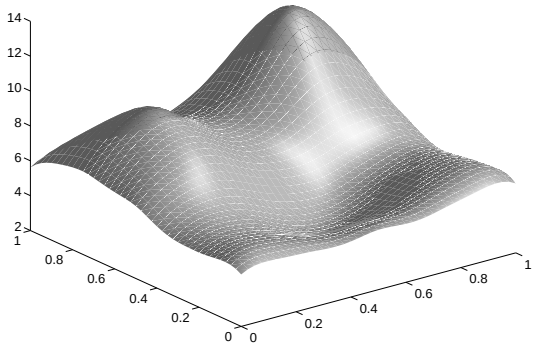

Figure 15: Wu's C^2 interpolant

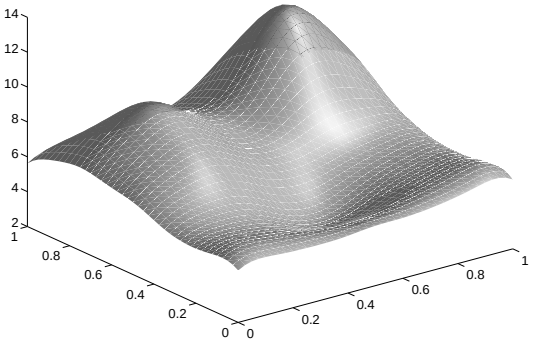

Figure 16: Wu's C^4 interpolant with support 1.5

Wendland [Wen95] has constructed further instances of smooth compactly supported positive definite radial basis functions. Moreover, for a specific dimension d, these functions possess the lowest possible degree among all piecewise polynomial compactly supported radial functions, which are positive definite on R^d and of a given degree of smoothness. These functions $\psi_{k,l}$ are of degree $k+2l$, and are positive definite of order d, where $\lfloor \frac{d}{2} \rfloor = k-l-1$. These functions are C^{2l} continuous at zero, and C^{l+k-1} continuous at the boundary. Examples include $\psi_{3,1} = (1-r)_+^4(4r+1)$, $\psi_{4,2} = (1-r)_+^6(35r^2+18r+3)$, and $\psi_{5,2} = (1-r)_+^7(16r^2+7r+1)$. $\psi_{3,1}$ is of degree 5 and is positive definite of order 3. Moreover, it is C^2 continuous at the zero, and C^3 continuous at the boundary. Similarly, $\psi_{4,2}$ is of degree 8 and is positive definite of order 3. Moreover, it is C^4 continuous at the zero, and C^5 continuous at the boundary.

Floater and Iske [FI96b, FI96a] have used Wendland's radial basis functions with compact support described in the previous paragraph for modeling scattered data in two major steps – thinning algorithm [FI98] and hierarchical residual interpolation. The construction is of great interest and therefore, we describe it in somewhat greater detail. The first step, referred to as the *thinning algorithm*, creates a hierarchy from the given scattered data points by removing one point at a time until there are only a small number K left. The points are removed in such a way that the points are distributed as evenly as possible. This step thus creates a hierarchy

$$Y_K \subset Y_{K+1} \subset \cdots \subset Y_{N-1} \subset Y_N = X,$$

where N is the total number of given data points. The criteria for removal or thinning the data set uses two metrics. The first metric is the *separation distance* of a set S defined as half of the minimum distance between any pair of distinct points belonging to the set, that is,

$$q(S) = \min_{\substack{\mathbf{x}_i, \mathbf{x}_j \in S \\ \mathbf{x}_i \neq \mathbf{x}_j}} \frac{||\mathbf{x}_i - \mathbf{x}_j||}{2}.$$

The second metric is the *radius of the largest inner empty sphere*, that is

$$Q(S) = \max_{\mathbf{x} \in \Omega} \min_{\mathbf{x}_j \in S} ||\mathbf{x} - \mathbf{x}_j||,$$

where Ω is the closed interior of some polygon surrounding the given scattered data set X. Uniformity of a set S is then defined as

$$\rho(S) = \frac{q(S)}{Q(S)}.$$

Observe that $0 < \rho(S) \leq 1$. An example of a data set with high uniformity is the set of nodes in a triangular grid made of equilateral triangles. In this case, $\rho = \frac{\sqrt{3}}{2}$. During the thinning algorithm, the objective is to increase the uniformity of the reduced set at every successive removal of a point. Moreover, the boundary nodes are treated separately in order to prevent boundary erosion. In practice, the Delaunay triangulation is computed at each step, and $q(S)$ and an approximation to $Q(S)$ is computed. The computational complexity of this step of the algorithm in their current implementation is $O(N^2)$, and can be improved $O(N \log N)$ using known techniques [FI96a].

The second step of the algorithm, to which I refer to as the *hierarchical residual interpolation*, chooses a suitable subsequence, say of length m, of the resulting hierarchy. For example, a geometric sequence $Y_{31} \subset Y_{125} \subset Y_{500} \subset Y_{2000}$ of length 4 consisting of 31, 125, 500 and 2000 points are chosen in one example. The interpolation is achieved in m stages. At the first stage, all the points in Y_{31}

are interpolated using a radial basis function with compact support. Positive definiteness of this function guarantees a unique solution. In all the examples described in [FI96a, FI96b], Wendland's $\psi_{3,1}$ function has been chosen. The support of this basis function is chosen based on a combination of metrics $q(Y_{31})$ and $Q(Y_{31})$. At the second stage, the *residual* is interpolated at all the points in Y_{125} using a *smaller* support radius that is chosen based on a combination of metrics $q(Y_{125})$ and $Q(Y_{125})$. Support radii at each stage are chosen to obtain an appropriate trade-off between good approximation behavior governed by $Q(S)$ and the stability of computation governed by $q(S)$. The interpolation of the successive residuals is obtained by using smaller and smaller radii of the compact radial basis functions. This hierarchical multi-stage interpolation technique has been applied to several data sets including track data, feature data, contour data and analytic data sets. The sparse linear system of equations needed to solve the interpolation problem is solved using the conjugate gradient method on sparse matrices. The computational complexity of this step is asymptotically only $O(N)$, which is a significant improvement over traditional radial basis functions with growing support. The possibility that this technique can be applied to much larger data sets than the traditional radial basis functions such as multiquadrics makes this technique extremely attractive.

Uncertainty Relationship: We now add certain remarks on the theoretical results on stability concerning the use of radial basis functions. Numerical observations and theoretical results indicate that there is an uncertainty relationship between the error and the stability of the computation while using radial basis functions with parameter h that is proportional to the minimum distance between data points [Sch95b]. Roughly speaking, in this situation, the product of the error and the instability of the computation is always greater than some constant. Therefore, it is not possible to achieve both small error and stability. This is one of the greatest disadvantages of radial basis approach. However, again we emphasize that this situation is valid when the parameter h is proportional to the minimum distance between the data points.

In order to understand how the uncertainty relationship can be used to improve both the error and the stability while using radial basis functions of compact support, let us introduce some notation. Given a Wendland's radial basis function $\psi_{k,l}$ that are +ve definite of order d, where $\lfloor \frac{d}{2} \rfloor = k - l - 1$, let $\beta = \frac{d}{2} + l + \frac{1}{2}$. Let, the bandwidths β_1 and β_2 be $\beta_1 = (\frac{h}{Q})^d$ and $\beta_2 = (\frac{h}{q})^d$. The results on error and stability can now be stated as follows: (i) the approximation error E is roughly equal to $\beta_1^{-\frac{\beta}{d}}$; (ii) the condition number S or the norm of the inverse matrix, that measures the stability of the computation, is roughly equal to $\beta_2^{\frac{2\beta}{d}}$. Observe that the bandwidths are fixed when the parameter h varies in proportion to the minimum distance between data points. In such a situation, since q and Q are roughly proportional, one can multiply the two results together to obtain the uncertainty relationship:

$$E^2 S = constant.$$

In cases where the bandwidths are fixed, the uncertainty relationship therefore implies that improving error necessarily degrades stability of computation and vice-versa. Even when h is fixed and data is dense so that the bandwidth B_1 is large by including enough centers in and around the domain, error may become smaller but at the cost of increasing unstable computation.

One way to get around the uncertainty principal is to perhaps use the hierarchical approach as used by Floater and Iske described above [FI96a]. In this case, first, since the support radii are growing smaller with increasing density, ther bandwidth B_2 remains roughly fixed so that the stability of the computation remains about the same. In usual cases, since B_1 remains the same as well, the error does not improve. However, the error is perhaps made smaller by using *residuals* in each successive step. A theoretical justification of this technique is still being investigated.

Some promising theoretical results have been reported that indicate good convergence behavior by using hierarchical residual interpolation technique, when the functions used in successive steps are convolutions of functions used in the previous step [NSW97]. Therefore, good interpolation is possible by using hat function, quadratic B-splines, quartic B-splines and so on in the successive steps of a residual hierarchical interpolation scheme.

4.2 Surfaces on Surfaces

Trivariate Scattered Data: Radial basis function methods such as the multiquadric method and thin plate splines are easily extendible to higher dimensions. Important results concerning the theory of such methods was given by Micchelli [Mic86], and these results also have a bearing on whether the given method is proper, that is, whether the matrix of the system of equations for the coefficients of the basis functions is (conditionally) positive definite and therefore, the existence of a solution is guaranteed. In particular, the multiquadric method is a proper method for three dimensions. The analogue of thin plate splines in three dimensions is the basis function $\phi(r_k) = r_k$, although in various publications it has been assumed to be either $\phi(r_k) = r_k^2 \log r_k$ or $\phi(r_k) = r_k^3$. While these are both proper, neither minimizes the "thin volume functional", the integral of the sum of the squares of the second derivatives of the interpolating function. Interpolation functions in s-dimensions that minimize a functional that is the integral of sums of squares of mth order derivatives are detailed in Wahba and Wendelberger [WW80]. This reference discusses a method including smoothing and the use of Generalized Cross Validation (GCV) to choose the smoothing parameter. A related resource of importance for practical applications is RKPACK [Gu91]. RKPACK (for Reproducing Kernel Package) is a general purpose program for solving interpolation and smoothing problems in s-dimensions using reproducing kernel methods (such as radial basis function methods) with a choice of GCV or Generalized Maximum Likelihood (GML) methods to choose the smoothing parameter. The principal problem with using radial basis function methods in higher dimensions is the large systems of equations that must be solved. For even moderate sized data sets consisting of more than 300 to 500 points, local methods such as the ones discussed later in Section 5 are needed. For example, overlapping cubes with product Hermite cubic weight functions, as generalized from [Fra82b], could be used.

Function-on-Surface Problem: We now briefly describe how various approaches described in Section 2.3 have been extended to solve the problem of interpolating data on a surface using radial basis functions. First using the trivariate approach as described above, one can simply use the trivariate radial basis function methods. For example, one can use the *trivariate multiquadric method* by employing the basis functions $\sqrt{r_k^2 + h^2}$, where r_k is now the Euclidean distance between points in R^3 instead of R^2. This method has been used to solve the scattered data interpolation problem on a sphere and a cylinder. Using the domain mapping approach described in Section 2.3, this method has also been extended to solve the scattered data interpolation problem on surfaces that are topologically equivalent to spheres.

The bivariate approach has also been applied to construct radial interpolants on surfaces. The key idea is to replace the Euclidean distance by geodesic distance on the surface. In general, however, it is not easy to compute the geodesic distance on an arbitrary surface. Fortunately, for a sphere, the geodesic distance $s_k(\mathbf{x})$ between points $\mathbf{x}$ and $\mathbf{x}_k$ is given by $cos^{-1}(\mathbf{x} \cdot \mathbf{x}_k)$ and measures the great circle distance between the two points. However, this geodesic distance function is singular both at $\mathbf{x}_k$ and at the antipodal point $\overline{\mathbf{x}_k}$. Therefore, the straightforward extension of Hardy's multiquadrics by choosing the functions $R(s_k(\mathbf{x})) = (s_k(\mathbf{x})^2 + h^2)^{\frac{1}{2}}$ runs into the undesirable problem that $R(s_k(\mathbf{x}))$ has discontinuous derivatives. This difficulty can be overcome by considering

spherical inverse multiquadrics by using the functions $R_k(\mathbf{x}) = (1+l^2-2l\cos(s_k(\mathbf{x})))^{-\frac{1}{2}}$ as suggested in [HG75] or by considering *spherical multiquadrics* by using the functions $R_k(\mathbf{x}) = (1 + l^2 - 2l\cos(s_k(\mathbf{x})))^{\frac{1}{2}}$ as suggested in [PE90].

Interestingly, the trivariate multiquadric method turns out to be equivalent to the spherical multiquadrics for spheres [PE96]. This fact has not appeared in the literature, and is not well-known. The proof is however straightforward and simple. First, observe that the trivariate multiquadrics $\sqrt{r_k^2 + h^2}$ can be rewritten for a sphere of radius 1 as

$$\sqrt{(\mathbf{x} - \mathbf{x}_k)^2 + h^2} = \sqrt{2 + h^2 - 2\mathbf{x} \cdot \mathbf{x}_k}.$$

The spherical multiquadrics $R_k(\mathbf{x})$ can be rewritten as

$$R_k(\mathbf{x}) = \sqrt{l}\sqrt{\frac{1}{l} + l - 2\mathbf{x} \cdot \mathbf{x}_k}.$$

Comparing the two equations above, one observes that the trivariate multiquadric and the spherical multiquadrics differ only by a constant $\sqrt{l}$ by choosing $h = \frac{|l-1|}{\sqrt{l}}$. Since they differ only by a constant, they yield the same solution to the system of linear equations posed for the interpolation problem.

Analogous to the planar case, Foley also suggested the use of a C^2 *continuous modified* form of the multiquadrics [Fol90], where the basis functions are rounded off in a small neighborhood of the antipodal point. All these radial basis function methods can however lead to a poorly conditioned system of equations for large data sets as in the planar case involving merely more than 200 points [PE90].

For the case of sphere, pseudosplines proposed by Wahba [Wah81] seem very well behaved. We also refer the reader to a very nice survey on spherical splines that use zonal functions with small and scalable support [FSF97]. Zonal functions on spheres are of the type $f(x \cdot y)$ and are rotationally invariant.

5 Shepard's Methods and Variants

A scattered data interpolant using Shepard's method or its variant can be purely radially symmetric or a purely polynomial or rational function or a mixture of both. This method of solving scattered data interpolation problem depicts an approach rather then a prechosen form of the solution. This approach has been used to solve both the function reconstruction problem and the surfaces-on-surfaces problem. The key idea is to define the scattered data interpolant $f(x, y)$ as a weighted mean of the values $z_i = f_i$ by choosing some blending functions or weight functions. More precisely, $f(x) = \sum_{i=1}^{N} w_i(x, y) f_i$, where the weight functions $w_i(x, y)$ satisfy the following properties: (i) $w_i(x_j, y_j) = \delta_{ij}$, and (ii) *partition of unity* property, that is $\sum_{i=1}^{N} w_i(x, y) = 1$. In addition, the weight functions $w_i(x, y)$ are at least C^0 continuous and non-negative.

Shepard suggested the weight functions $w_i(x, y)$ to be $\frac{\sigma_i(x,y)}{\sum_{i=1}^{N} \sigma_i(x,y)}$, where $\sigma_i(x, y) = \frac{1}{r_i^\mu}$. In other words, weight functions are obtained by normalizing *inverse distance weight functions*. These weight functions are radially symmetric, are positive and C^0 continuous. In contrast to the radial basis function approach discussed in the previous section, observe that this approach does not require solving any system of equations.

The original Shepard's method as outlined above exhibited several difficulties. First, the method is global because the weight functions do not have compact support. The global nature of the

Shepard's method can be made local by multiplying the $\sigma_i(x,y)$ by positive damping functions $\lambda_i(x,y)$ with $\lambda_i(x_i,y_i) = 1$, that vanish outside some suitable neighborhood of the point (x_i,y_i). One such damping function suggested by Franke-Little is for example $\lambda_i(x,y) = (1 - \frac{r_i}{h_i})_+^\mu$ [Bar77]. The damping function therefore vanishes outside a disk of radius h_i centered around the point x_i. The interpolant can, however, be sensitive to the choice of the radius h_i. One such choice for h_i is $\frac{D}{2}\sqrt{\frac{N_w}{N}}$, where D is the maximum distance between any two data points and N_w is an arbitrarily chosen constant [FN80].

The other significant problem with the original Shepard's method is that the the interpolating function has cusps at the data points (x_i,y_i) for $0 < \mu < 1$, corners at the data points for $\mu = 1$ and flat spots at the data points for $\mu > 1$. To remove the discontinuities in the derivatives of the interpolant and the flat spots, one can use other weight functions $w_i(x,y)$ built from $\sigma_i(x,y) = \frac{e^{-hr_i^2}}{r_i^2}$. This weight function is found to be extremely time consuming [Fra82a]. An alternative is to consider the interpolant $f(x,y) = \sum_{i=1}^N w_i(x,y)L_i(x,y)$, where $L_i(x,y)$ is a local interpolant or approximant to data values with the property $L_i(x,y) = f_i$. Observe that since $w_i(x_j,y_j) = \delta_{ij}$, the interpolant $f(x,y)$ satisfies the interpolation property $f(x_j,y_j) = f_j$. A typical $L_i(x,y)$ can be a linear function that interpolates the function values and its first partial derivatives that are given or estimated from the given data. Another choice for $L_i(x,y)$ is a quadratic polynomial function that interpolates the given point and achieves good approximation properties in the neighborhood of the point. The *modified quadratic Shepard's method*, introduced by Franke and Nielson [FN80], is obtained by choosing (i) a quadratic $L_i(x,y)$, (ii) the inverse distance weight functions, as suggested originally by Shepard, along with (iii) the Franke-Little damping functions described in the previous paragraph. Although this version of Shepard's method performed better than other versions implemented and tested by Franke [Fra82a], it did not perform as well as the purely radial basis function methods.

In another generalization of Shepard's method, the weight functions $w_i(x,y)$ need not have the property $w_i(x_j,y_j) = \delta_{ij}$. The interpolant is still taken to be $f(x,y) = \sum_{i=1}^N w_i(x,y)L_i(x,y)$. In this generalization, one can choose any local interpolant $L_i(x,y)$ and any local blending function $w_i(x_j,y_j)$ such that $L_i(x,y)$ interpolates all the points in the support of $w_i(x,y)$. Since the weight functions $w_i(x,y)$ still have the partition of unity property, the interpolation is achieved. The support of the weight functions is usually selected in such a way that the number of data points in the support of each weight function is roughly equal. As an example, the weight functions are built with $\sigma_i(x,y) = 1 - 3\frac{r_i^2}{h_i^2} + 2\frac{r_i^3}{h_i^3}$ for $r_i \le h_i$ and 0 otherwise. The parameter h_i, the radius of the disk of support is chosen to be the distance between the point (x_i,y_i) and its fifth closest neighbor. To achieve interpolation, $L_i(x,y)$ is now chosen to be a quadratic polynomial that interpolates all the six points in the support of $w_i(x,y)$. The reconstructed surface is a mixture of polynomial and radial functions (due to presence of r_i^3). As another example, the weight functions $w_i(x,y)$ are formed from the tensor product polynomial Hermite functions and $L_i(x,y)$ is chosen to be a polynomial function that interpolates all the points in the support of $w_i(x,y)$. In this variation of Shepard's method, the overall interpolant is therefore piecewise polynomial. As a third example, one can choose local interpolants to be one of the radial basis functions such as thin plate splines, Hardy's multiquadrics or inverse multiquadrics and blend these local interpolants together using radial blending functions such as the inverse distance weight functions (as suggested originally by Shepard) modified with Franke-little damping functions. Franke [Fra82b] implemented a method that blends local thin plate spline interpolants. The performance of this interpolant was poorer than the interpolant using the modified quadratic Shepard's method, but can be improved by using local interpolants of larger support than suggested in that paper.. Foley [Fol87] used the

flexibility in constructing these interpolants to build a system for constructing several interpolants for noisy, rapidly varying or nonuniformly distributed data points using these and multistage ideas described below. Extensive experimentation is required to control the enormous flexibility in these interpolants to achieve the desired result.

Multistage Methods: Global radial basis interpolants are extremely expensive to compute for very large data sets. If interpolants are computed by truncating the global radial basis functions or by summing the contribution from a prechosen number of nearest points of interpolation, the interpolant can be discontinuous. Although Franke and Salkauskas [FS95] have shown how to achieve continuity by a convolution process for such methods, the proposed scheme is not computationally viable. Localized Shepard's interpolants or localized radial basis interpolants with appropriate blending functions can also be expensive due to the need for sorting the data points. Also, rendering a surface requires many evaluations and these evaluations are usually on rectangular grids. Multistage methods are invented to overcome these problems.

The general idea of a multistage interpolant is to (i) use some local least square approximation or some local interpolant to estimate the values of the interpolant on a tensor product grid, (ii) use some interpolation scheme to generate a tensor product surface that interpolates the estimated values at the grid points, and (iii) apply a correction based on Shepard-type method to obtain a global interpolant.

Another type of multistage interpolant was proposed by Barnhill and Gregory [BG75]. Let P and Q be two linear operators such that the composition PQ is defined. The Boolean sum $P \oplus Q$ is defined as $P + Q - PQ$. The Boolean sum interpolant has (at least) the interpolation properties of P and the precision properties of Q. A common application of this scheme has been to choose P to be a Shepard-type interpolant and Q to be a polynomial (of chosen degree) least square approximation to the given data. Thus one can obtain an interpolation scheme with an arbitrarily high degree of polynomial precision.

Yet another type of multistage interpolant was proposed by Foley [Fol84] to overcome the difficulty of the Boolean sum that it cannot be applied to tensor product based operators. Nielson and Foley [NF80] proposed the *delta sum* of two operators, $P\Delta Q$ as $P \oplus QP$, that is, $QP + P - PQP$. We refer the reader to [BS84, Fol87] for a discussion on these interpolants.

Natural Neighbor Interpolants: Natural neighbor interpolants [Sib81] can also be viewed as a variant of Shepard's method. In this interpolant, the choice of the damping function $\lambda_i(x, y)$ at a data point depends upon the neighboring geometry around that data point. A given set of scattered data points (x_i, y_i) in the plane determines a collection of polygons known as *Dirichlet*, *Voronoi* or *Thiessen tessellation* [PS85]. This tessellation is dual to the Delaunay triangulation [PS85]. A *Dirichlet tile* τ_i for a point (x_i, y_i) is a region consisting of all those points which are closer to (x_i, y_i) than any other given scattered data point. More precisely, $\tau(x_i, y_i) = \{(x, y) \in R^2 : r_i \leq r_j, \forall j = 1, 2, \cdots, N\}$. Two existing points are called neighbors if their tiles share a common edge. Now let (x, y) be an arbitrary point in the convex hull of the the given points. If this point is inserted in the tessellation, it then acquires its own tile $\tau(x, y)$, assembled from parts of tiles of existing points, called neighbors of (x, y). More precisely, $\tau(x, y) = \{(p, q) \in R^2 : ||(p, q) - (x, y)|| \leq ||(p, q) - (x_j, y_j)||, \forall j = 1, 2, \cdots, N\}$. This situation is illustrated in Figure 17. Assume that (x, y) has m neighbors $(x_1, y_1), \cdots, (x_m, y_m)$. Now let $u(x, y)$ be the area of the Dirichlet tile $\tau(x, y)$. Moreover, let $u_i(x, y)$ be the area of that portion of the Dirichlet tile $\tau(x, y)$ that intersects the tile $\tau(x_i, y_i)$. The damping function $\lambda_i(x, y)$ is then defined to be the ratio of these two areas, namely $\frac{u_i(x,y)}{u(x,y)}$ for $i = 1, \cdots, m$.

The simplest natural neighbor interpolant is then given by $f(x, y) = \sum_{i=1}^{m} \lambda_i(x, y) f_i$. The

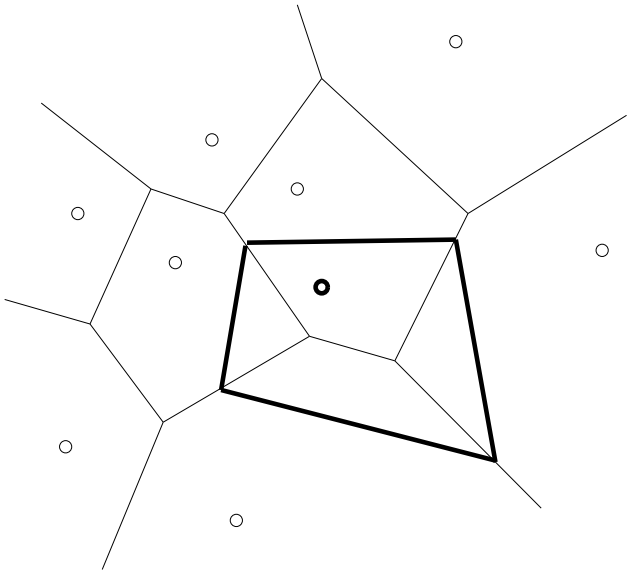

Figure 17: Dirichlet tile of an inserted point is highlighted

natural neighbor interpolant is local, has linear precision and is continuously differentiable. The interpolant is, in general, a piecewise rational quartic function [Far90]. As another example, $\sigma_i(x,y)$ is chosen to be $\frac{\lambda_i(x,y)}{r_i(x,y)}$, and the weight functions are chosen to be $\frac{\sigma_i(x,y)}{\sum_{i=1}^{N}\sigma_i(x,y)}$. The natural neighbor interpolant is then given by $f(x,y) = \sum_{i=1}^{N} w_i(x,y)L_i(x,y)$, where $L_i(x,y)$ are chosen to be linear polynomials. Properties of natural neighbor interpolants and generalizations of Sibson's natural neighbor interpolant are discussed in [Far90, Wat92, Wat94].

5.1 Surfaces-on-Surfaces

Trivariate Scattered Data:

Shepard's Method: This method generalizes very readily to arbitrary dimensions, with the same shortcomings it has in two dimensions. The same techniques to overcome some of them apply in higher dimensions. In particular, modification of the "inverse distance" function for finite support and local approximation by a polynomial greatly improves its behavior. A version of this method is available as an ACM algorithm [Ren88].

Multistage methods: Multistage methods, either with or without the third step to obtain interpolation, are readily applied in higher dimensions. Foley [Fol87] discusses the trivariate case using the multiquadric method and his program TRIHASH (trivariate Hermite interpolation and Shepard). This is a very general program that uses a local interpolation or approximation method (multiquadric, or polynomials of various degrees) to estimate values for the function on a cubical grid. The derivative values may be estimated from this same function, or they may be estimated from the cuberille data. The Hermite interpolant is then constructed for the gridded data, but this in general does not interpolate the original data. It is then an option to add a (hopefully small) correction term based on a local Shepard's method to the tricubic Hermite function to achieve interpolation.

Function-on-Surface Problem: We now briefly describe the extension of various approaches described in Section 2.3 to Shepard-type techniques for solving the problem of scattered data

interpolation on surfaces [BDL83, BPR87]. We present two different solutions using the bivariate approach described in Section 2.3.

The first solution can be used to solve the problem on convex surfaces [BPR87]. The interpolant is defined as before by $f(\mathbf{x}) = \sum_{i=1}^{N} w_i(\mathbf{x}) L_i(\mathbf{x})$. The weight function $w_i(\mathbf{x})$ are defined in terms of $\sigma_i(\mathbf{x})$ as before, where $\sigma_i(\mathbf{x}) = \frac{\lambda_i(\mathbf{x})}{g_i(\mathbf{x})^2}$. The distance function $g_i(x)$ is now the geodesic distance between the points $\mathbf{x}$ and $\mathbf{x}_i$ along the surface. If the surface is not given explicitly, $g_i(\mathbf{x})$ can be found approximately using the geodesics on the sphere that approximates the surface at the point $\mathbf{x}$. This means that one can only expect to get good results when the surface is convex. The $\lambda_i(x)$ are damping functions that now take into account the geometry of the surface. The idea is to choose $\lambda_i(\mathbf{x})$ in a way that $\lambda_i(\mathbf{x}_i) = 1$, $\lambda_i(\mathbf{x}_j) = 0$ and that these functions gradually reduce to zero in the neighborhood of $\mathbf{x}_i$. To determine the neighborhood of points on a surface, first a *barrier* is created that separates those points on the surface that are close together in R^3 but far apart on the surface. The barrier naturally depends on the shape of the surface. For example, in case of an aeroplane wing, a barrier could consist of a planar segment between the two sides of the wing. Once the barrier is created, scattered data points are projected (usually using some simple projection such as a perpendicular projection) onto the barrier to create *barrier points*. The given data points along with the barrier points are then tessellated using 3D Delaunay tetrahedralization. Given any data point $\mathbf{x}_i$ its neighbors can then be classified into barrier points and non-barrier points. Both barrier and non-barrier points are used in constructing $\lambda_i(\mathbf{x})$. More precisely, $\lambda_i(\mathbf{x}) = \prod_j l_{ij}(\mathbf{x})$, where j varies over all the neighbors of $\mathbf{x}_i$. $l_{ij}(\mathbf{x})$ is a cubic Hermite function that takes the value 1 at $\mathbf{x}_i$, 0 at $\mathbf{x}_j$ and whose first derivative vanishes at both $\mathbf{x}_i$ and $\mathbf{x}_j$. The local interpolants $L_i(\mathbf{x})$ are biquadratic functions that interpolate the five nearest non-barrier points in the neighborhood of $\mathbf{x}_i$.

The second solution uses localized versions of radial basis function methods and blends these using weight functions to build an interpolant on a sphere S [PE90]. The interpolant is defined as before by $f(\mathbf{x}) = \sum_{i=1}^{M} w_i(\mathbf{x}) L_i(\mathbf{x})$. $L_i(\mathbf{x})$ are now local interpolants defined on spherical caps $K_i = \mathbf{x} \in S : s_i(\mathbf{x}) \leq \rho_i$ with spherical midpoints at M_i and geodesic radii ρ_i. M_i and ρ_i can be chosen in a data dependent way so that the caps contain nearly the same (sufficiently large) number of points. However, by default, one can choose a regular covering of the sphere by congruent caps so that M_i are vertices of a regular polyhedron, that is a tetrahedron (M=4), octahedron (M=6), cube (M=8), icosahedron (M=12) or dodecahedron (M=20) [PE90]. The radius ρ_i has to be greater than the supremum of the radii of circumcircles of the triangles in the triangulation. The weight function $w_i(\mathbf{x})$ are defined in terms of $\sigma_i(\mathbf{x})$ as before, where $\sigma_i(\mathbf{x})$ are now Franke-Little type weights. More precisely, $\sigma_i(\mathbf{x}) = \left(1 - \frac{s_i(\mathbf{x})}{\rho_i}\right)_+^{\mu}$. The choices $\mu = 2$ and $\mu = 3$ usually lead to sufficiently smooth surfaces.

6 Subdivision Methods

Subdivision methods apply the process of cutting the corners of a polyhedron-like configuration or mesh of vertices, edges and faces. In addition to the scattered data points or vertices, the connectivity relations between the vertices are assumed to be provided in terms of edges and faces. The faces need not be planar and the vertices do not have to lie on a topologically regular mesh. Therefore, subdivision techniques are generally well suited for solving the surface reconstruction problem.

Perhaps the best known subdivision technique was introduced by Chaikin in 1974 for curves [Cha74]. He generated a smooth curve from a polygon by successively refining the polygon by

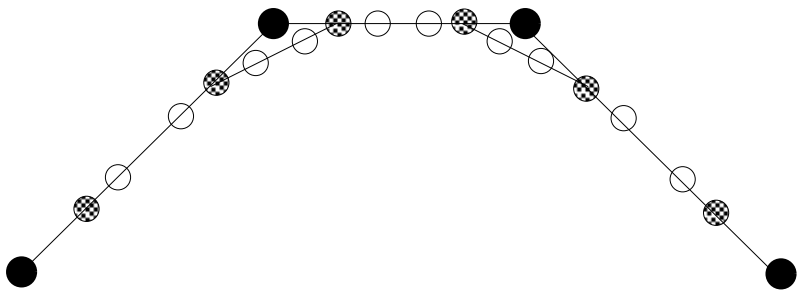

Figure 18: Chaikin's corner cutting algorithm

cutting its corners. The smooth curve obtained by this scheme turns out to be a quadratic B-spline curve obtained by treating the original vertices as control points. This idea was generalized to surfaces by Catmull-Clark [CC78] and Doo and Sabin [DS78] in 1978. In these schemes, an initial control mesh, that is, a polyhedron-like configuration of vertices, edges and faces is given in 3D. Although faces of this configuration need not be planar, popular special cases arise when each face is a triangle or a rectangle. Corners of the polyhedron-like configuration are "cut" or in other words, the polyhedron is refined or subdivided by adding new vertices, edges and faces by using a deterministic rule. In the limit as the number of subdivision steps goes to infinity, the polyhedron converges to a surface. With careful choice of the deterministic corner cutting rules, it is possible to show that the limiting surface exists, is continuous and possesses a continuous tangent plane. Except for some special cases, the limiting surface does *not* have an explicit analytic expression to represent the surface. If each face of the polyhedron is a rectangle, the Doo-Sabin subdivision rules generate biquadratic tensor product B-splines [DS78] and the Catmull-Clark subdivision rules generate bicubic tensor product B-splines [CC78]. Another subdivision technique by Loop generates triangular box splines if each face of the polyhedron is a triangle [Loo87].

All the subdivision techniques mentioned above are approximating schemes. There are two key approaches to constructing interpolating subdivision surfaces. One approach is to first compute a new configuration of vertices, edges and faces with the same topology such that the vertices of the new configuration converge to the given vertices in the limit. The subdivision technique is then applied to this new configuration. This method was first suggested by Nasri [Nas87]. The second approach is to modify the deterministic subdivision rules so that the limiting surface does interpolate the vertices. This method was first suggested by Dyn, Levin and Gregory as a 4-point [DLG87] scheme for curves and later extended to a butterfly subdivision scheme for surfaces [DLG90]. An extension to a 6-point interpolatory scheme for curves is also presented in [Dyn92], which is also an excellent reference for an in-depth discussion of this approach.

To appreciate interpolatory subdivision techniques, we first discuss the following approximating schemes for constructing subdivision surfaces: (i) Chaikin's corner cutting algorithm for the quadratic B-spline curve, (ii) its generalization to the cubic B-spline curve, and (iii) the Catmull-Clark subdivision technique that generalizes the construction of tensor product bicubic B-spline surfaces. We then present the interpolatory generalization of Catmull-Clark subdivision technique and the butterfly subdivision scheme for surfaces.

Chaikin's Algorithm:
Suppose we are given vertices or control points c_i, $i = 0, \cdots, n$, where $c_i \in R^3$. Let us relabel the given control points as $c_i^0 = c_i$. Chaikin's algorithm uses a binary subdivision process that computes recursively from the given set of control points c_i^0, $i = 0, \cdots, n$ new sets of control points

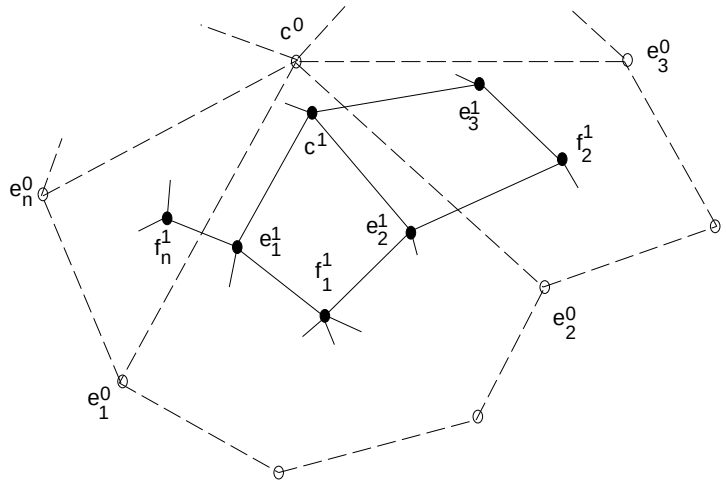

Figure 19: Catmull-Clark subdivision algorithm

c_i^k, $i = 0, \cdots, 2^k n - 2^{k+1} + 2$ as follows: c_i^{k+1} as follows:

$$c_{2i}^{k+1} = \frac{3}{4} c_i^k + \frac{1}{4} c_{i+1}^k,$$

$$c_{2i+1}^{k+1} = \frac{1}{4} c_i^k + \frac{3}{4} c_{i+1}^k.$$

The control points at level k define a new control polygon. This process is illustrated in Figure 18. The initial control points are shown as black circles. The control points obtained after the first iteration are shown as cross-hatched circles. The control points obtained after two iterations ate shown as empty circles. The initial control polygon and the control polygon obtained after the first iteration are shown by straight lines. The control polygon obtained after the second subdivision step is obtained by joining all the empty circles. In the limit, this control polygon converges to a quadratic B-spline curve and is C^1 continuous. Observe that at every step of this algorithm, a corner of the control polygon is chopped off. Therefore this class of algorithms is referred to as corner cutting algorithms. Moreover, at each step of the subdivision, the polygon is subdivided into a new control polygon, which has roughly twice the number of original control points. Therefore this class of algorithms is also referred to as subdivision algorithms.

Cubic B-spline Subdivision Algorithm:
Chaikin's subdivision algorithm for quadratic B-spline curves can be generalized to a subdivision algorithm for a B-spline curve of any degree n. The subdivision rules that generate cubic B-spline curves are as follows:

$$c_{2i}^{k+1} = \frac{1}{2} c_i^k + \frac{1}{2} c_{i+1}^k,$$

$$c_{2i+1}^{k+1} = \frac{1}{8} c_i^k + \frac{3}{4} c_{i+1}^k + \frac{1}{8} c_{i+2}^k.$$

In this case, the control polygon converges to a cubic B-spline curve and is C^2 continuous.

Catmull-Clark Subdivision Technique:
Catmull and Clark generalized the subdivision scheme from curves to surfaces [CC78]. The scheme starts with an initial given mesh M^0 and subdivides the mesh many times using the same set of deterministic rules at every step. Let M^i be the mesh after i steps. To describe the subdivision

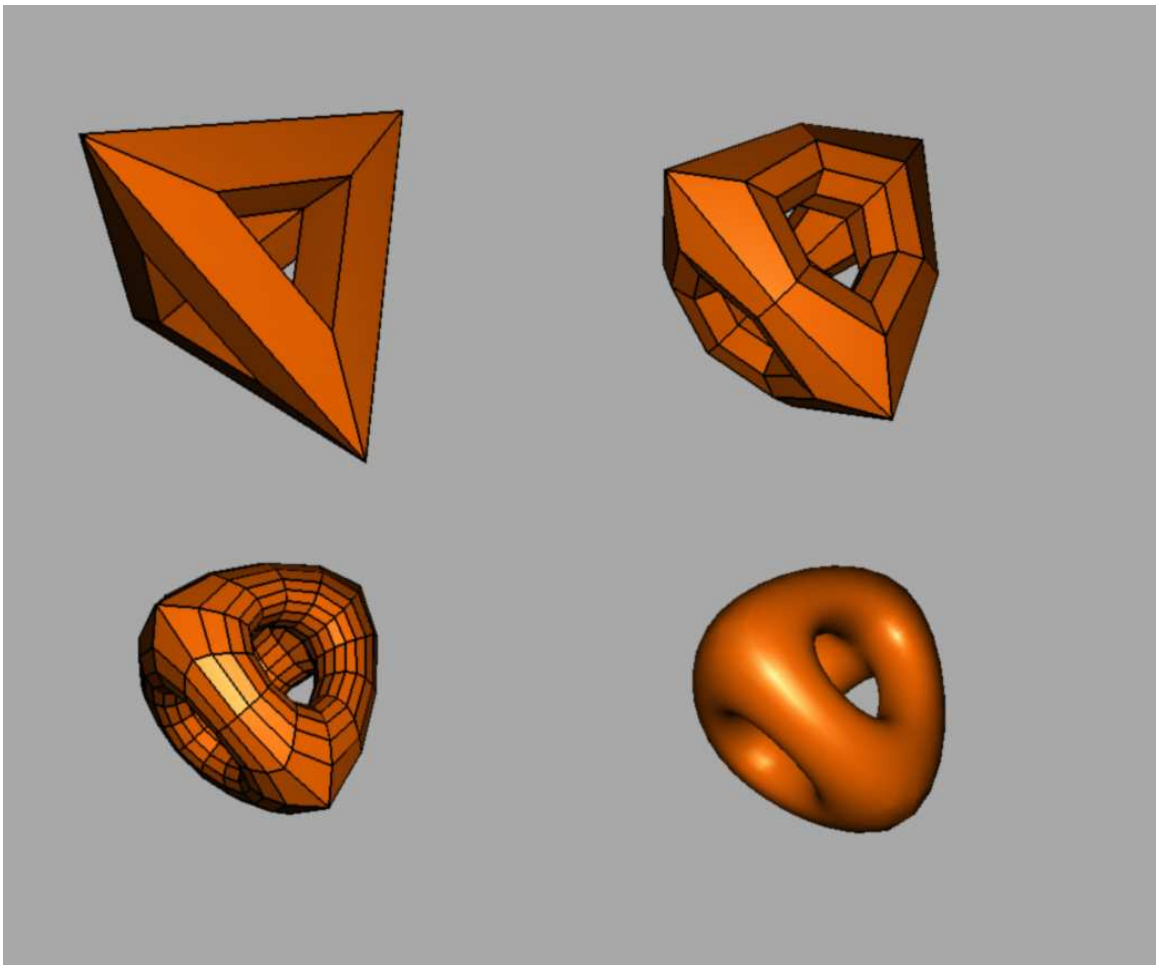

Figure 20: Catmull-Clark subdivision surface

rules at the $i+1$-th step, consider the neighborhood of a vertex c^i surrounded by n edge points $e_1^i, \cdots, e_n^i$ and n faces, as shown in Figure 19. Such a vertex is said to be of order n. At the next subdivision step,

1. Create new face points $f_1^{i+1}, f_2^{i+1}, \cdots, f_n^{i+1}$ at the centroid of each face of the mesh M^i.
2. Each new edge point $e_1^{i+1}, \cdots, e_n^{i+1}$ is computed by taking an average of surrounding points. Specifically,
$$e_j^{i+1} = \frac{c^i + e_j^i + f_{j-1}^{i+1} + f_j^{i+1}}{4},$$
where subscripts are to be taken modulo n.
3. A new vertex point c^{i+1} is computed as follows:
$$c^{i+1} = \frac{n-2}{n} c^i + \frac{1}{n^2} \sum_{j=1}^{n} e_j^i + \frac{1}{n^2} \sum_{j=1}^{n} f_j^{i+1}.$$

Notice that after first subdivision step, all faces are quadrilateral. For this scheme vertices of degree 4 are *regular* vertices. All other vertices, that is vertices of degree other than 4 are

referred to as *extraordinary* points. It can be shown that except at extraordinary vertices, the mesh converges to a tensor product bicubic B-spline surface and is therefore curvature continuous everywhere except perhaps at the extraordinary points. Near an extraordinary point, the surface does not possess a closed form parametrization. However the surface has a well-defined tangent plane at the limit point of an extraordinary point [BS88]. The upper left diagram of Figure 20 shows the initial control mesh. The upper right diagram and the lower left diagrams of Figure 20 show the mesh obtained after one and two iterations of the subdivision algorithm respectively. The lower right diagram of Figure 20 shows the Catmull-Clark subdivision surface obtained in the limit.

Interpolatory Catmull-Clark Subdivision Technique:
Catmull-Clark subdivision technique presented above does not interpolate the given control points. To generate subdivision surfaces that interpolate the given vertices, one has to compute a new configuration of vertices, edges and faces with the same topology such that the vertices of the new configuration converge to the given vertices in the limit. Nasri [Nas87] suggested this approach by applying the Doo-Sabin scheme to a modified set of control points with the same topology. To compute the new or modified configuration, a system of interpolation constraints that relate the new set of vertices to the given set of vertices is developed. Halstead et al [HKD93] suggested an interpolatory subdivision scheme by applying the Catmull-Clark subdivision scheme to a modified set of control points. They derived closed form expressions for the interpolation constraints even though the limiting surface may not have a closed form expression. This approach results in a global sparse system of linear equations. In general, the system may be singular. In such cases, one can use a least square approximation to the solution. Alternatively, one can subdivide the original configuration twice and then attempt to compute a modified configuration such that the vertices of the modified configuration converges to the given vertices in the limit. This approach guarantees the existence of an interpolating solution and introduces much more flexibility in choosing the additional points of the modified configuration, which can then be constrained by using some global variational principle to obtain even smoother surfaces [HKD93].

We now discuss these interpolation conditions briefly. For details the reader is referred to [HKD93]. Consider the column vector of vertices $V_n^i = (c^i, e_1^i, \cdots, e_n^i, f_1^i, \cdots, f_n^i)$ in the neighborhood of the vertex c^i. Let V_n^{i+1} be the corresponding column vector of vertices after the next subdivision step. Since the points in V_n^{i+1} are computed by the linear combination of the points in V_n^i, the subdivision can be expressed as $V_n^{i+1} = S_n V_n^i$, where S_n is a square matrix of order $(2n+1) \times (2n+1)$. The properties of the limiting surface will be governed by the properties of V_n^{i+1} as i approaches infinity. Since V_n^{i+1} is the image of V_n^1 under S_n^i, the eigenstructure of S_n plays a key role.

Let λ_1 be the largest eigenvalue of S_n and let l_1 be the corresponding left eigenvector. It can then be shown that a point c^1 having a neighborhood V_n^1 converges to the point v^∞ where $v^\infty = l_1 V_n^1$. For Catmull-Clark surfaces it can be shown that

$$l_1 = \frac{1}{n(n+5)}(n^2, 4, \cdots, 4, 1, \cdots, 1),$$

where 4 and 1 are repeated n times each. Therefore,

$$v^\infty = \frac{n^2 c^1 + 4\sum_{j=1}^n e_j^1 + \sum_{j=1}^n f_j^1}{n(n+5)}. \tag{5}$$

This equation can be used as an interpolation condition on the points of a new mesh M^1 by setting v^∞ to a point to be interpolated. This approach leads to a square linear system of equations of the

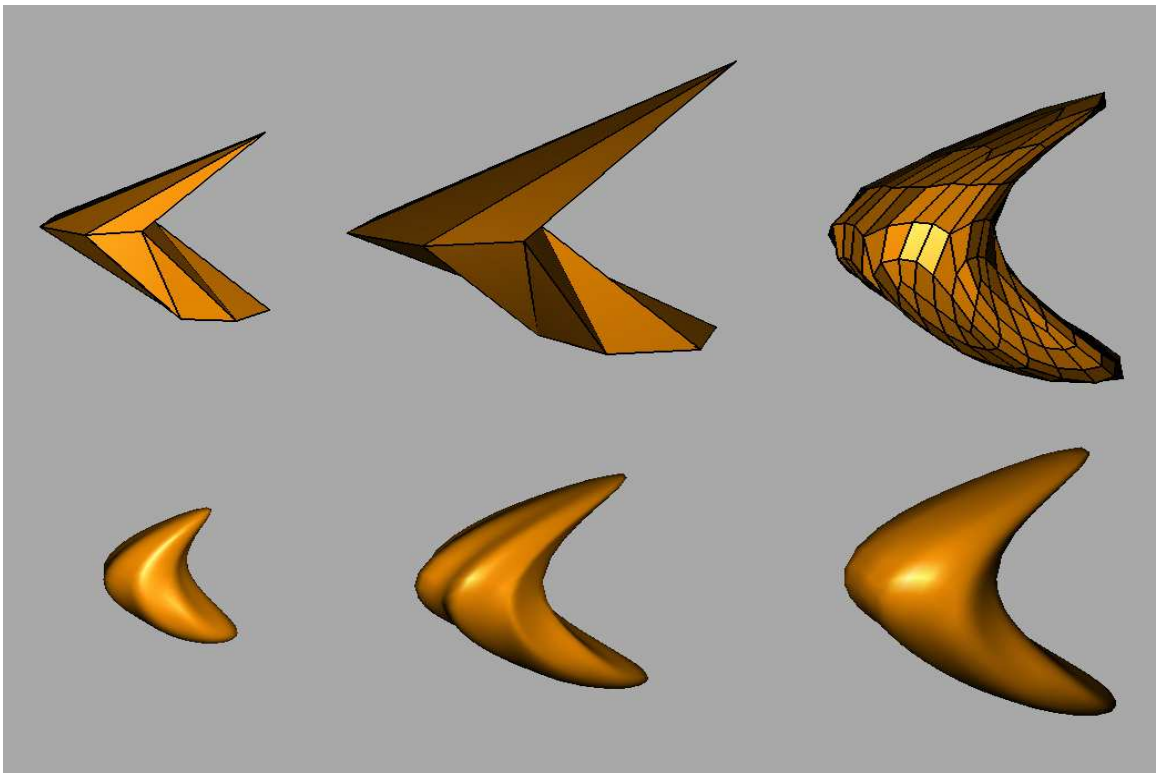

Figure 21: Interpolatory Catmull-Clark subdivision surface

form $Ax = b$, where x is the column vector of the unknown vertex coordinates of the modified mesh, b is the corresponding column vector of vertex coordinates of the given mesh and the entries of the square matrix A is obtained from Equation 5 above. The upper left diagram of Figure 21 shows the initial control mesh. The lower left diagram of Figure 21 shows the corresponding Catmull-Clark surface. This surface does not interpolate the original vertices. The upper middle diagram of Figure 21 shows the interpolating control mesh obtained by solving the system of equations discussed above. The lower middle diagram of Figure 21 shows the corresponding Catmull-Clark surface. This surface does interpolate the original vertices, but has some wiggles. The upper right diagram of Figure 21 shows the control mesh obtained by subdividing the original control mesh twice, and minimizing an objective function globally subject to the interpolation constraints. The lower right diagram of Figure 21 shows the corresponding Catmull-Clark surface that interpolates the original vertices and is much smoother.

Butterfly Subdivision Technique:
The second key technique for constructing subdivision interpolants is to modify the set of deterministic rules for subdivision. This approach was adopted by by Dyn, Levin and Gregory [DLG87] in extending the approximating subdivision techniques for curves to interpolatory subdivision techniques. In a *four-point interpolatory scheme*, given control points $c_i^0 \in R^3$, $i = 0, \cdots, n$, new set of control points c_i^k, $i = 0, \cdots, 2^k n$ are computed as follows:

$$c_{2i}^{k+1} = c_i^k, 0 \leq i \leq 2^{k-1} n,$$

$$c_{2i+1}^{k+1} = (\frac{1}{2} + w)(c_i^k + c_{i+1}^k) - w(c_{i-1}^k + c_{i+2}^k), 0 \leq i \leq 2^{k-1} n - 1.$$

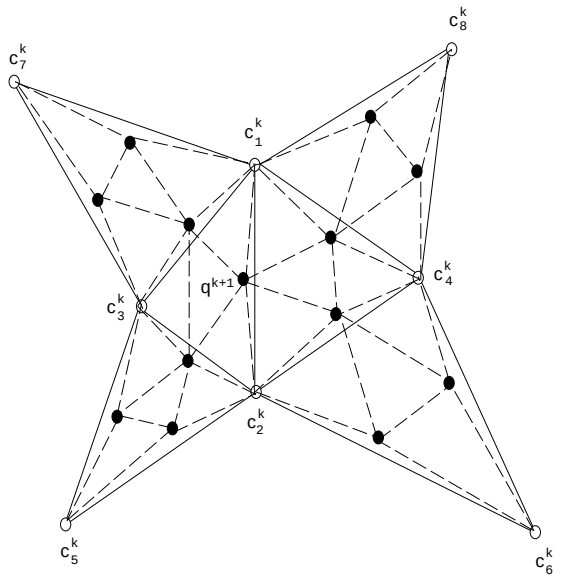

Figure 22: Butterfly interpolatory subdivision algorithm

At each stage all the old points are kept and new points are inserted in between. Therefore this scheme is clearly interpolatory. It was established that for any $|w| < \frac{1}{2}$, the limiting curve is continuous. Moreover, for any $0 < w < \frac{(\sqrt{5}-1)}{8}$, the limiting curve is C^1 continuous. However, for a general set of control points, there is no w for which the limiting curve is C^2 continuous.

To obtain a C^2 continuous curve, one can use a six-point interpolatory scheme, where the new control points are defined as follows:

$$c_{2i}^{k+1} = c_i^k, -1 \leq i \leq 2^k n + 1,$$

$$c_{2i+1}^{k+1} = (\frac{9}{16} + 2\theta)(c_i^k + c_{i+1}^k) - (\frac{1}{16} + 3\theta)(c_{i-1}^k + c_{i+2}^k) + \theta(c_{i-2}^k + c_{i+3}^k), -1 \leq i \leq 2^k n.$$

This scheme reduces to the four-point scheme with $w = \frac{1}{16}$ for $\theta = 0$. The limiting curve produced by this scheme is however curvature continuous for $0 < \theta < 0.02$.

The 4-point interpolatory subdivision scheme for curves was generalized by Dyn, Levin and Gregory to surfaces [DLG90]. This scheme however requires the initial control mesh to be a triangulation. The scheme subdivides the given triangulation into a refined triangulation by retaining the old control points and introducing the new control points. The rules for inserting new control points is an eight-point rule, as shown in Figure 22. The configuration in Figure 22 justifies the terminology "butterfly" for this scheme. The new point q^{k+1} corresponding to the edge $c_1^k c_2^k$ is introduced as follows:

$$q^{k+1} = \frac{1}{2}(c_1^k + c_2^k) + 2w(c_3^k + c_4^k) - w(c_5^k + c_6^k + c_7^k + c_8^k).$$

After inserting the new points, a refined triangulation is formed. The refined triangulation consists of all the edges connecting each new point q^{k+1} with the old points c_1^k and c_2^k and the four new points corresponding to the edges (c_i^k, c_j^k), $i = 1, 2, j = 3, 4$. Thus each old triangle is subdivided or refined into four triangles. Also observe that all the new vertices generated by this refinement process have degree six, that is, there are exactly six edges meeting at the new vertices. For the

purposes of this scheme, the vertices with degree six are *regular* vertices. If all vertices are regular, then this subdivision scheme generates a tangent continuous surface provided $0 < w < \frac{1}{16}$. Even a stronger statement can be made. If the initial triangulation does not have any vertex of degree three, then it can be proved that with $0 < w < w_0$, $w_0 > \frac{1}{16}$, the limiting surface is C^1.

Remarks: One of the greatest challenges with the subdivision techniques is to establish the exact criteria on the set of subdivision rules that yield nice properties for the limiting subdivision surface [Rei95, War95]. The proofs to establish that the limiting surface exists, and is continuous or tangent plane continuous are often very long, laborious and subtle. Systematic attempts have been made by several researchers to both simplify and improve upon the existing results. It seems quite likely that new set of rules can be designed that will guarantee higher order continuity for subdivision surfaces including curvature continuity. The other major disadvantage of the subdivision surfaces is the lack of a closed form expression for the final surface. This difficulty may also be overcome by providing a large class of algorithms that support operations on subdivision surfaces.

On the plus side, the concept of subdivision is fairly easy to understand and implement. The limiting subdivision surface is curvature continuous except at a finite number of "extraordinary points" where the topology of the given control mesh deviates from the regular topology. Higher order continuity is obtained with very few control points. Finally subdivision surfaces generalize traditional B-splines and therefore are likely to be incorporated into the existing geometric modeling systems with relative ease.

7 Surface Visualization and Interrogation

So far we have focused mainly on the construction of scattered data interpolants without discussing the quality of these interpolants. It is a non-trivial task to determine the quality of these interpolants. We have discussed smoothness of these interpolants, that is, whether they are tangent continuous or curvature continuous. But what about other properties of these interpolants? We now discuss a few techniques for visualizing these interpolants and interrogating them.

7.1 Surface Visualization

A common technique to understand the interpolants better is to render the surface and try to inspect the surface visually by rotating and zooming in and out of the surface. There are several standard techniques for rendering a surface [FvDFH90]. However rendering a surface-on-surface in order to provide the viewer a convenient and unobstructed view of these surfaces is not an easy task. Therefore, we now discuss a few techniques for rendering surfaces-on-surfaces [FL91, FL90, FHN93, Nie93c], [PO94, PHD91, FLN90a, FLN$^+$90b].

Bivariate Function over Plane: This method can be used to visualize a function over a surface when the surface can be parametrized by using some planar region as a domain. For example a sphere can be parametrized using latitudes and longitudes on a rectangular domain [FLN90a]. The function is then simply viewed over this rectangular domain. Clearly such a mapping is distorted because small regions near the poles are shown as large regions on the rectangular domain. Moreover the boundary curves on the left and right are equal and the boundary curves on the front and back degenerate to one point each, namely the north and the south pole.

Contour Plots: A contour plot for a function over a surface consists of contour lines on the surface where the function assumes a given value. Since in most applications it is difficult to determine these curves exactly, the surface is typically subdivided into subdomains, the function is evaluated at the corners of these subdomains and approximated inside the subdomains using some simple

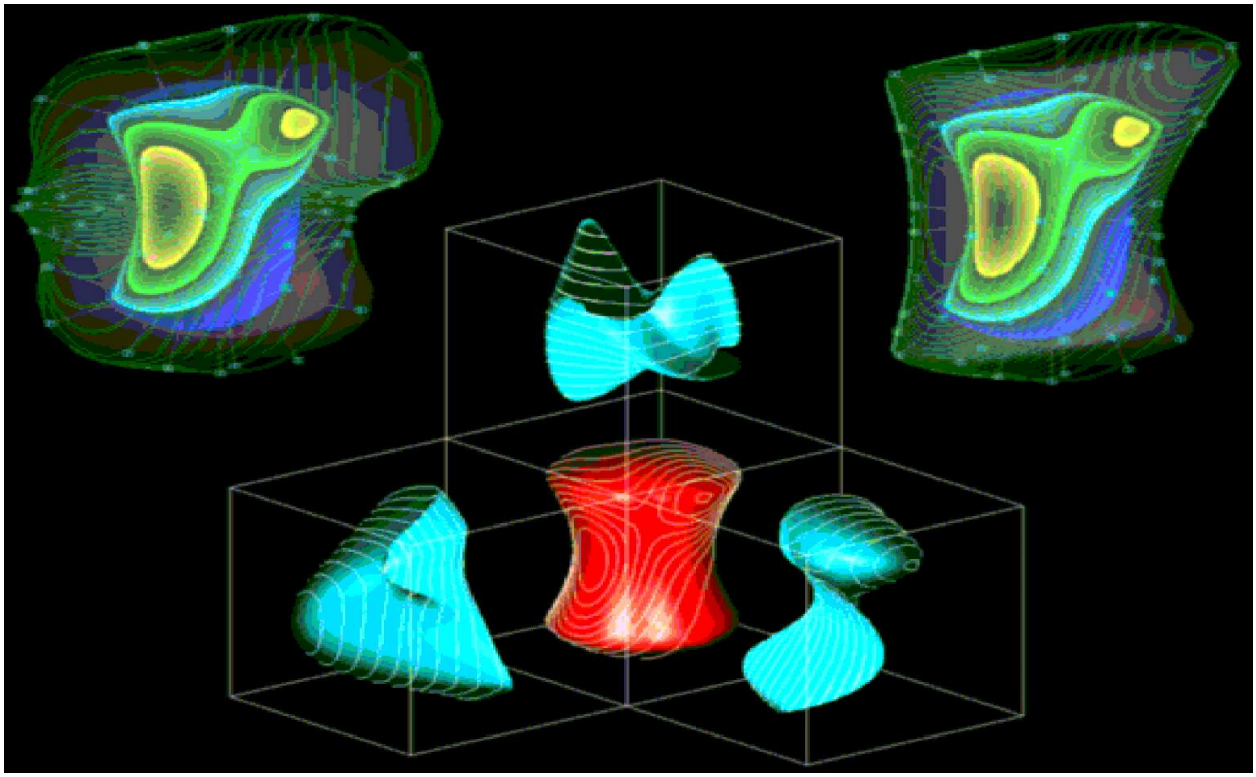

Figure 23: Hypersurface projection graph

technique such as piecewise linear interpolation. Therefore it is important to subdivide the surface into uniform regions. For example, a subdivision of the sphere using latitudes and longitudes of equal differences yield a non-uniform partitioning of the sphere. A much better triangulation of the sphere consists of the decomposition of the sphere into a collection of spherical triangles [FLN90a]. Using this triangulation, one can draw contour plots for a function defined on the sphere. There are several variations of this technique including color-blended contours that allow the user to observe both the contour curves and the behavior of the function between the contour curves [FLN90a]. Contour plots were also used extensively by Tvedt [Tve91] in his evaluation of trivariate schemes for interpolation. His computer program (called Slice Viewer) gives the user a thumbnail display of the contours of 20 slices parallel to a chosen coordinate plane for each of the parent function, the interpolation function, and the error. By clicking on one of the slices, the user can simultaneously display the three contours for that slice in much larger sizes. By cycling through the contours the user can attempt to obtain some idea of the qualitative behavior of the functions. These techniques although useful do not show the geometric shape or the smoothness of the functions.

Transparent Surface Graphs: This method extends the idea of a surface graph over a plane. Given a point P on the surface let the value of the function on the surface be v. The value v is then shown as a point which is at a distance v from the surface along the normal direction at the point P. However there are several difficulties with this method. First, if the values are negative at some points and positive at others, the graph of surface-on-surface may intersect the domain surface making the comprehension of the function very difficult. To avoid this problem, the function can be translated to be positive everywhere so that the minimum value is mapped onto zero, that is on the domain surface itself. Second, such a graph of surface-over-surface will cover the domain surface

itself. To overcome this difficulty, the graph of surface-on-surface is drawn as a transparent surface so that one can see the domain surface through this surface. Third, it is still difficult to visualize the correspondence between the points on the domain surface and the points on surface-on-surface in three dimensions. This difficulty is overcome by connecting a suitably chosen subsample of points on the domain surface and connecting them with the points on the surface-on-surface by line segments. The upper left diagram of Figure 23 shows this technique where the data is sampled on an apple-core type domain. The surface-on-surface is shown as a transparent surface projected in a direction normal to the surface. However, this method may still be very difficult to visualize on non-convex domains due to occlusion and possibly self-intersections. An alternative is to plot the point not along the normal direction but along the radial direction from some suitably chosen point. The upper right diagram of Figure 23 shows this technique where the surface-on-surface is shown as a transparent surface projected in a radial direction. This may alleviate the problem on some non-convex domains but not all. There are other variations of this approach including an approach that combines the transparent surface with the color-blended contour approach [FLN90a, FL90].

Hypersurface Projection Graph: The surface-on-surface consists of points $(x, y, z, f(x, y, z))$ in 4D, where (x, y, z) are restricted to lie on some surface. To visualize this 4-dimensional graph, the idea is to use projections of this 4D graph in three dimensions. A simple approach is to use three orthogonal projections, $(x, y, f(x, y, z))$, $(x, f(x, y, z), z)$, and $(f(x, y, z), y, z)$ to display the three 3D surfaces in three windows simultaneously with the domain surface in the center. This technique is illustrated in Figure 23 [FHN93]. Several variants of this approach including parallel and central projections are discussed in [PHD91].

7.2 Surface Interrogation

Although surface visualization may succeed in detecting large unwanted undulations in the surface interpolant, this approach may fail to detect many important characteristics of the interpolant or fail to distinguish between different interpolants that may be differentiated using other techniques for interrogating them. A popular characteristic of the surface is its Gaussian curvature. Lounsbery et al [LMD92, MLL$^+$92] assigned a color to every point on the surface according to the value of the Gaussian curvature at that point. This technique of *pseudo-coloring* can be used to color the surface according to any characteristic of interest and helps to bring out the uneven distribution of the characteristic, particularly in the interpolants that are designed to fit smooth symmetrical shapes such as spheres or tori. There are numerous other characteristics of surfaces, such as normals to the surface, principal or normal curvatures or mean curvature that can be examined to reveal important information about the quality of these interpolants. We now discuss a few techniques for interrogating surfaces to determine the aesthetic quality, convexity, curvature and continuity properties of the surfaces [HHS$^+$92, HHS$^+$93, HH92].

Reflection Line Method: A reflection line is the projection on the surface of the reflection of a light line when viewed from some eye point. Reflection lines are therefore dependent both upon the viewing angle as well as the direction of the light line. In order to evaluate the surface one uses a set of parallel light lines while maintaining the same view point. This procedure is then repeated from a different viewing angle. Computation of reflection lines require solving a system of non-linear equations by numerical means. The existence and unambiguity of solutions has to be ensured by an appropriate choice of the eye position [Kla80].

This method determines unwanted dents by emphasizing irregularities in the reflection line pattern of parallel straight lines. This method simulates the light cage used in automobile industry and is considered an effective tool to evaluate the aesthetic quality of a surface.

Isophotes: Given a light source, isophotes are lines of equal intensity on the surface. Since intensity at any point is determined by the angle between the light source and the normal to the surface, isophotes are defined by $< \vec{N}(u,v), \vec{L} >= constant$, where $\vec{N}(u,v)$ is the normal vector to the parametrized surface and $\vec{L}$ is the direction of the light source. This equation is solved numerically to compute the isophotes. Observe that the silhouettes are special isophotes (constant = 0) with respect to the light source. If the surface is G^k continuous, the isophotes are G^{k-1} continuous. Therefore, isophotes can be used to test geometric continuity between surface patches. However, this technique depends upon appropriate choice of lighting direction as well. Pottmann [Pot88] describes a generalized isophote method for automatic testing of continuity across the boundaries of a patchwork of surfaces.

Orthotomics and Polarity Methods: Given a surface $X(u,v)$ and a point P, which is neither on the surface nor on any of the tangent planes of the surface, a $k-$orthotomic of the surface is a new surface obtained by reflecting P across tangent planes of X and multiplying the lengths of the reflection by a factor of k. More formally, the $k-$ orthotomic of a surface $X(u,v)$ with respect to a point P is

$$Y_k(u,v) = P + k < X(u,v) - P, N(u,v) > N(u,v).$$

The $k-$orthotomic surface has a singularity at a point iff the Gaussian curvature of the surface at the corresponding point vanishes or changes its sign at that point [Hos85]. Therefore orthotomics help in visualizing the convexity of surfaces.

In polarity method with respect to a unit sphere the surface is mapped onto a new polarity surface. The polarity surface is defined as the envelope of the planes obtained by mapping the points $x(u_0,v_0)$, $y(u_0,v_0)$, $z(u_0,v_0)$ of the surface onto the planes defined by the equations $x(u_0,v_0)x + y(u_0,v_0)y + z(u_0,v_0)z + 1 = 0$. The equation of the polarity surface can be written down as:

$$P(u,v) = \vec{N}(u,v).\frac{det(\vec{N}, X_u, X_v)}{det(X, X_u, X_v)}.$$

If the Gaussian curvature of the surface vanishes or changes its sign at a point, the polarity surface has a singularity at the corresponding point [Hos84]. Therefore polarity methods also help in visualizing the convexity of surfaces.

Focal Surfaces: A focal surface is defined as

$$F_i(u,v) = X(u,v) + \kappa_i^{-1}(u,v)N(u,v),$$

where κ_i is one of the two principal curvatures. The two points $F_1(u,v)$ and $F_2(u,v)$ are called the focal points and define the centers of curvature of two principal directions. A generalized focal surface is defined as

$$G(u,v) = X(u,v) + af(\kappa_1,\kappa_2)N(u,v),$$

where the scalar function f now depends upon the principal curvatures. By choosing the function f appropriately, it is possible to interrogate the surface effectively for curvature behavior and discontinuities. For example, by choosing $f = \kappa_1\kappa_2$ or $f = \kappa_1^2 + \kappa_2^2$, one can test for points of vanishing curvature or for flat points [HH92]. Thus focal surfaces can be used to detect regions of unwanted curvature situations.

7.3 Affine Invariance

In many applications in computer aided geometric design and scientific visualization, it is desirable that the scattered data interpolant satisfies some invariance property such as translation invariance.

A compelling example occurs in scientific disciplines, where the interpolant ought not to depend upon the unit of measurements for the data. In other words, the interpolant should not depend upon whether the data was measured in terms of inches and seconds or in terms of feet and minutes. Yet many interpolants fall short of desirable invariance properties [Nie87a, Nie93b, NF89].

A scattered data interpolant S is said to be invariant with respect to a transformation T of the input data X if $S \circ T(x) = T \circ S(x)$. A method S is said to be translation (respectively rotation, affine) invariant if the interpolant S satisfies the above property for any translation T (respectively rotation, affine transformation). Nielson and Foley [NF89] further distinguish between *scale-invariant* methods when the interpolant is invariant with respect to any scale transformation T and the *scalar-invariant* methods when the interpolant is invariant with respect to those scale transformations, where the scaling is identical along all coordinate axes. This definition clearly implies that any scale-invariant interpolant is also scalar-invariant. However, the converse is not necessarily true. In fact, this distinction is important because as we will soon see, many methods are scalar-invariant but not scale-invariant.

Most radial basis functions, that do not have free parameters, such as the thin plate splines $r^2 \log r$, are translation, rotation and scalar invariant, but not scale-invariant. This is also true of the original Shepard method, that is purely based on the inverse distance weight function $\frac{1}{r^\mu}$. Interpolants based on other radial basis functions, that have free parameters h, such as multiquadrics $\sqrt{r_k^2 + h^2}$, are translation and rotation invariant but not scale-invariant. Scalar invariance property of these interpolants is slightly subtle. Given a fixed parameter, these interpolants are *not* scalar invariant. However, if the parameter depends upon the distance between data points in such a way so that it scales accordingly, then these interpolants are scale-invariant. An example of this occurs, when the parameter h is chosen as a constant times D, the maximum distance between any two data points – a popular choice as described in Section 5.

Nielson and Foley [Nie87a, Nie93b, NF89] have described a method for modifying many scattered data methods in order to make them affine invariant by replacing Euclidean metric by an *affinely invariant metric*. The metric is a weighted norm which effectively removes the cross covariances from the data, resulting in equidistant curves in the metric. Polynomial and rational interpolants based on triangulations can also be made affinely invariant by using an *affine invariant triangulation* if the interpolant itself is in an affine invariant form such as the Bernstein-Bézier form. An affine invariant Delaunay triangulation can also be found by using an affine invariant metric.

7.4 Visual Interrogation

In view of a large variety of characteristics that can be used to interrogate a surface interpolant, it seems worthwhile to create a flexible system for visually comparing and contrasting the quality of the interpolants. The quality of interpolants can mean any combination of different characteristics that the user may consider important. An initial step was taken in this direction by by Tvedt [Tve91] who used contour plots in his evaluation of trivariate schemes for interpolation [NT94a, NT94b]. More recently, several new techniques for visual comparison of surface and volumetric interpolants have been presented [LSPW96, LJZU97]. The viewer can control an interactive query-driven toolbox to create a wide variety of graphics that allow probing of geometric information in useful and convenient ways. These techniques combine the strengths of traditional techniques such as pseudo-coloring, differencing, overlay and transparency with new glyph-based probing techniques. We describe some of these techniques briefly. In *pseudo-coloring*, the difference between two interpolants is mapped to some color ramp. In *differencing*, pointwise difference between the two interpolants is computed and the difference surface thus obtained is rendered. In *overlay* the

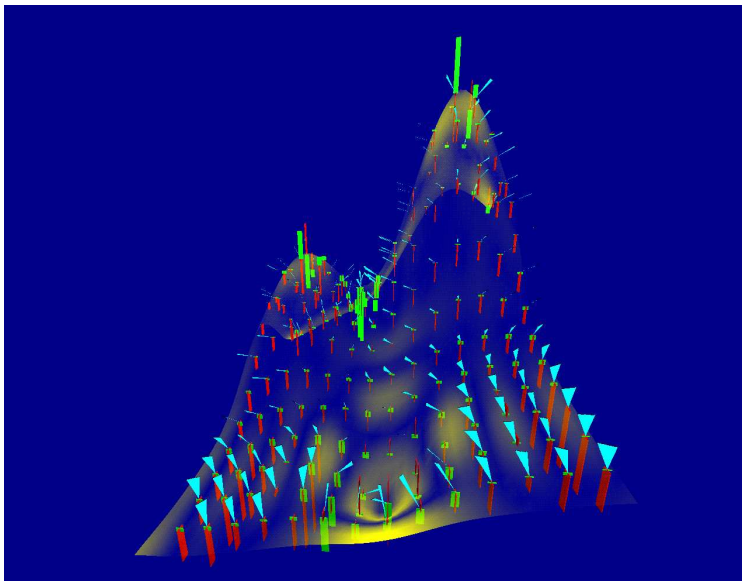

Figure 24: Comparison of geometric features of two interpolants

two interpolants are shown superimposed over each other. In *transparency*, the difference between the two interpolants is mapped to the transparency, thereby making the surface transparent in the regions where the differences between the two interpolants are high. In *glyphs*, differences between different geometrical characteristics of the two interpolants are mapped to some properties of certain geometrical objects such as ellipsoids, triangular strips or cross-hairs. In *probes*, certain subregions of the interpolants are selected where these characteristics or the differences between these characteristics are viewed. Figure 24 shows an example of a glyph-based technique where some geometric features of the multiquadric and thin plate spline interpolants are compared on Franke's data set that is shown in Figure 11. The normals of two interpolants are joined with a triangular strip in Figure 24. These triangular strips thus indicate the deviation between the normals of the two interpolants. The differences between the mean and Gaussian curvatures of the two interpolants are mapped to the widths of the cross-hairs and also shown in Figure 24.

8 Conclusions

In spite of a large number of techniques available for scattered data interpolation, there is a great need for developing new techniques for scattered data interpolation and gaining better understanding of existing techniques. This situation arises because there are serious gaps or shortcomings in many existing techniques.

Tensor product interpolants such as NURBS are examples of very successful techniques on gridded data, but they do not work on scattered data. Radial basis function methods such as Hardy's multiquadrics, inverse multiquadrics and thin plate spline interpolants are very successful in a large variety of applications, but cannot effectively handle typically data sets consisting of more than 300 data points. The generalized version of Shepard's method for constructing global interpolants by blending local interpolants using locally-supported weight functions can create a large collection of solutions. However these solutions depend crucially on a number of parameters such as the support of weight functions, the choice of the weight functions and the choice of

the local interpolant. Moreover the results indicate that these methods vary from being very poor to good, although even the best methods in this category cannot produce results as good as the global radial basis function methods. Interpolants for surface reconstruction problem also suggest that even the best local interpolant does not produce as effective shapes as the global techniques. The dependence of the triangulation-based methods such as finite element solutions or multivariate splines or minimum-norm network methods on the underlying grid is not well understood. Subdivision techniques, particularly those using global variational principles, seem to produce satisfactory results, but do not have any closed form analytic expressions.

The scattered data interpolation and approximation of functions of three or more independent variables is in its infancy. There is much that remains untested even for the two popular variations of the surfaces-on-surfaces problems – trivariate (or volumetric) scattered data problem and the function-on-surface problem. For example, although the study conducted by Nielson and Tvedt [Tve91, NT94a, Nie93d] compares nine trivariate schemes (Shepard's method, modified quadratic Shepard method, volume splines, multiquadric method without linear precision, multiquadric with linear precision, local volume splines, piecewise linear on tetrahedra, and a C^1 method on tetrahedra) to 54 sets of data (6 configurations of points and 9 functions sampled at those points), this experimentation is confined to mostly analytic functions on very small data sets (less than thousand points). The surface-on-surface problem in its full generality has hardly been investigated.

Finally, not enough research has gone into effective comparison of different properties of interpolants. The criteria for judging the quality of an interpolant are not uniform. In computer aided geometric design applications, geometric properties of interpolants can perhaps be used as guidelines for comparing interpolants. Also in these applications, analytic test data are relatively easy to generate. In contrast, there are no clear criteria for evaluating the quality of an interpolant in scientific visualization applications. Depending upon the application, past experiences or judgements of experts are invariably used in evaluating the results. It is not clear whether these criteria can be translated into objective measures. Often times, however, additional information about data points, such as the underlying topology of data points, data distribution, gradient information at the data points, accuracy of the data can guide towards an appropriate choice of the interpolation or approximation method to be used.

In conclusion, the research seems to suggest that global techniques (such as using variational principles or solving a large system of equations), which are more expensive, often produce better results than local techniques. Second, there is a serious gap between theoretical and practical results for several categories of interpolants. In spite of active research and recent advances on radial basis interpolants and subdivision interpolants, they are not well understood. On the other hand, in spite of active experimentation of Shepard-type interpolants, theoretical properties and parameters that can guarantee good solutions is not well understood either. There seems to be little doubt, however, that there is no one best method and that the appropriate choice of an interpolation method very much depends upon the practical problem at hand.

Acknowledgments: The first author would like to thank Indranil Chakraborty for encouraging him to write this survey. We are grateful to Janice Tarrant for implementing the simulated annealing program and helping us to create Figure 1; to Herbert Edelsbrunner for providing the bust data and the alpha-shape generation program, that helped us to create Figure 2; to Bob Sheehan for creating Figure 2; to Jörg Peters for providing Figure 4; to Chandrajit Bajaj for providing Figures 5 and 6; to Robert Schaback for providing Figures 11, 12, 13, 14, 15 and 16; to Tony DeRose for providing Figures 20 and 21; to Greg Nielson for providing Figure 23; and to Clarke Steinback for helping us to convert images between different formats. We are also thankful to Indranil Chakraborty, Chandrajit Bajaj, Carl de Boor, Tony DeRose, Herbert Edelsbrunner, Hans Hagen, Will Light,

Greg Nielson, Jörg Peters, Helmut Pottmann, and Robert Schaback for valuable feedback on some of the topics presented in this chapter, which helped us to improve the presentation of this work. This research was partially supported by the National Science Foundation grants CCR-9309738, IRI-9423881, and CDA-9115268, ONR grant N00014-92-J-1807, and by the faculty research funds granted by the University of California, Santa Cruz.

References

[Aki78a] H. Akima. Algorithm 526: Bivariate interpolation and smooth surface fitting for irregularly distributed data points. *ACM Transcations on Mathematical Software*, 4:160–164, 1978.

[Aki78b] H. Akima. A method for bivariate interpolation and smooth surface fitting for irregularly distributed data points. *ACM Transcations on Mathematical Software*, 4:148–159, 1978.

[Alf84] P. Alfeld. A trivariate Clough-Tocher scheme for tetrahedral data. *Computer Aided Geometric Design*, 1:169–181, 1984.

[Alf89] P. Alfeld. Scattered data interpolation in three or more variables. In T. Lyche and L. L. Schumaker, editors, *Mathematical Methods in CAGD*, pages 1–33. Academic Press, 1989.

[ANS96] P. Alfeld, M. Neamtu, and L. Schumaker. Fitting scattered data on sphere-like surfaces using spherical splines. *J. Comp. Appl. Math.*, 73:5–43, 1996.

[Baj92] C. L. Bajaj. Surface fitting with implicit algebraic surface patches. In H. Hagen, editor, *Topics in Surface Modeling*, pages 23–52. SIAM, Philadelphia, 1992.

[Bar77] R. E. Barnhill. Representation and approximation of surfaces. In J. R. Rice, editor, *Mathematical Software III*, pages 69–120. Academic Press, New York, 1977.

[Bar83] R. E. Barnhill. Computer aided surface representation and design. In R. E. Barnhill and W. Boehm, editors, *Surfaces in CAGD*, pages 1–24. North-Holland, 1983.

[Bar85] R. E. Barnhill. Surfaces in computer-aided geometric design: a survey with new results. *Computer Aided Geometric Design*, 2:1–17, 1985.

[Bax92] B. J. C. Baxter. *The interpolation theory of radial basis functions.* PhD thesis, University of Cambridge, 1992.

[BBX95] C. L. Bajaj, F. Bernardini, and G. Xu. Automatic reconstruction of surfaces and scalar fields from 3D scans. *SIGGRAPH Proceedings*, pages 109–118, 1995.

[BCX94] C. Bajaj, J. Chen, and G. Xu. Free form surface design with A-patches. *Proceedings of Graphics Interface*, pages 174–181, 1994.

[BCX95] C. Bajaj, J. Chen, and G. Xu. Modeling with cubic A-patches. *ACM Transactions on Graphics*, 14(2):103–133, April 1995.

[BDL83] R. E. Barnhill, R. P. Dube, and F. F. Little. Properties of Shepard's surfaces. *Rocky Mountain Journal of Mathematics*, 13(2):365–382, 1983.

[BDL95] M. D. Buhmann, N. Dyn, and D. Levin. On quasi-interpolation by radial basis functions with scattered centers. *Constructive Approximation*, 11:239–254, 1995.

[BEE+93] M. Bern, H. Edelsbrunner, D. Eppstein, S. Mitchell, and T. S. Tan. Edge insertion for optimal triangulations. *Discrete Computational Geometry*, 10:47–65, 1993.

[BF91] R. E. Barnhill and T. A. Foley. Methods for constructing surfaces on surfaces. In H. Hagen and D. Roller, editors, *Geometric Modeling: Methods and their Applications*, pages 1–15. Springer, Heidelberg, 1991.

[BFK84] W. Boehm, G. Farin, and J. Kahmann. A survey of curve and surface methods in CAGD. *Computer Aided Geometric Design*, 1(1):1–60, 1984.

[BG75] R. E. Barnhill and J. A. Gregory. Polynomial interpolation to boundary data on triangles. *Mathematics of Computation*, 29:726–735, 1975.

[BI92a] C. L. Bajaj and I. Ihm. Algebraic surface design using Hermite interpolation. *ACM Transactions on Graphics*, 11(1):61–91, January 1992.

[BI92b] C. L. Bajaj and I. Ihm. Smoothing polyhedra using implicit algebraic splines. *SIGGRAPH Proceedings*, 26(2):79–88, 1992.

[BL92] R. K. Beatson and W. A. Light. Quasi-interpolation in the absence of polynomial reproduction. In D. Braess and L. L. Schumaker, editors, *Numerical Methods of Approximation Theory*, pages 21–39. Birkhauser-Verlag, Basel, 1992.

[BL97] R. K. Beatson and W. A. Light. Fast evaluation of 2-dimensional polyharmonic splines. *IMA Journal of Numerical Analysis*, 17:343–372, 1997.

[BM92] M. D. Buhmann and C. A. Micchelli. Multiquadric interpolation improved. *Computers and Mathematics with Applications*, 24(12):21–25, 1992.

[BN84] R. E. Barnhill and G. M. Nielson. Introduction to surfaces. *Rocky Mountain Journal of Mathematics*, 14:1–4, 1984.

[BN92] R. K. Beatson and G. N. Newsam. Fast evaluation of radial basis functions. *Computers and Mathematics with Applications*, 24(12):7–19, 1992.

[BO90] R. E. Barnhill and H. S. Ou. Surfaces defined on surfaces. *Computer Aided Geometric Design*, 7:323–336, 1990.

[BOP92] R. E. Barnhill, K. Opitz, and H. Pottmann. Fat surfaces: a trivariate approach to triangle-bases interpolation on surfaces. *Computer Aided Geometric Design*, 9:365–378, 1992.

[BPR87] R. E. Barnhill, B. R. Piper, and K. L. Rescorla. Interpolation to arbitrary data on a surface. In G. E. Farin, editor, *Geometric Modeling: Algorithms and New Trends*, pages 281–289. SIAM, 1987.

[Bro91] J. L. Brown. Vertex based data dependent triangulations. *Computer Aided Geometric Design*, 8:239–251, 1991.

[BS84] R. E. Barnhill and S. E. Stead. Multistage trivariate surfaces. *Rocky Mountain Journal of Mathematics*, 14:103–118, 1984.

[BS88] A. A. Ball and J. T. Storry. Conditions for tangent plane continuity over recursively defined b-spline surfaces. *ACM Transactions on Graphics*, 7(2):83–102, 1988.

[Buh89] M. D. Buhmann. *Multivariate interpolation using radial basis functions.* PhD thesis, University of Cambridge, 1989. Ph.D. Dissertation.

[Buh93] M. D. Buhmann. New developments in the theory of radial basis function interpolation. In K. Jetter and F. I. Utreras, editors, *Multivariate approximation: CAGD to Wavelets*, pages 35–76. World-Scientific, Singapore, 1993.

[BX94] C. Bajaj and G. Xu. Modeling scattered function data on curved surface. In J. Chen, N. Thalmann, Z. tang, and D. Thalmann, editors, *Fundamentals of Computer Graphics*, pages 19–29. Beijing, China, 1994.

[CC78] E. E. Catmull and C. H. Clark. Recursively generated B-spline patches on arbitrary topological meshes. *Computer-Aided Design*, 10:123–146, 1978.

[CF91] R. E. Carlson and T. A. Foley. The parameter r^2 in multiquadric interpolation. *Computers and Mathematics with Applications*, 21(9):29–42, 1991.

[Cha74] G. Chaikin. An algorithm for high speed curve generation. *Computer Graphics and Image Processing*, 3:346–349, 1974.

[Chi90] Eng-Wee Chionh. *Base points, resultants, and the implicit representation of rational surfaces.* PhD thesis, University of Waterloo, Canada, 1990.

[Chu88] C. K. Chui. *Multivariate Splines.* SIAM, Philadelphia, 1988.

[CK83] H. Chiyokura and F. Kimura. Design of solids with free-form surfaces. *Computer Graphics*, 17(3):289–298, 1983.

[CL94] E. W. Cheney and W. A. Light. Quasi-interpolation with base functions having non-compact support. *Constructive Approximation*, 1994.

[CN94] R. E. Carlson and B. K. Natarajan. Sparse approximate multiquadric interpolation. *Computers and Mathematics with Applications*, 26:99–108, 1994.

[Cre91] N. A. C. Cressie. *Statistics for Spatial Data.* John Wiley & Sons, 1991.

[CT65] R. Clough and J. Tocher. Finite element stiffness matrices for analysis of plates in bending. In *Proceedings of Conference on Matrix Methods in Structural Mechanics*, Wright-Patterson A. F. B., Ohio, 1965. Air Force Institute of Technology.

[Dah89] W. A. Dahmen. Smooth piecewise quadric surfaces. In T. Lyche and Larry Schumaker, editors, *Mathematical Methods in CAGD*, pages 181–193. Academic Press, Boston, 1989.

[dBJ85] C. de Boor and R. Q. Jia. Controlled approximation and a characterization of the local approximation order. *Proceedings of American Mathematical Society*, 95:547–553, 1985.

[dBR90] C. de Boor and Amos Ron. On multivariate polynomial interpolation. *Constructive Approximation*, 6:287–302, 1990.

[dBR92] C. de Boor and Amos Ron. Computational aspects of polynomial interpolation in several variables. *Mathematics of Computation*, 48:705–727, 1992.

[DL91] M. Daehlen and T. Lyche. Box splines and applications. In H. Hagen and D. Roller, editors, *Geometric Modeling: Methods and Applications*, pages 35–94. Springer-Verlag, 1991.

[DLG87] N. Dyn, D. Levin, and J. A. Gregory. A 4-point interpolatory subdivision scheme for curve design. *Computer Aided Geometric Design*, 4:257–268, 1987.

[DLG90] N. Dyn, D. Levin, and J. A. Gregory. A butterfly subdivision scheme for surface interpolation with tension control. *ACM Transactions on Graphics*, 9:160–169, April 1990.

[DLR86] N. Dyn, D. Levin, and S. Rappa. Numerical procedures for surface fitting of scattered data by radial functions. *SiAM Journal of Scientific and Statistical Computing*, 7:639–659, April 1986.

[DLR90] N. Dyn, D. Levin, and S. Rippa. Data dependent triangulations for piecewise linear interpolant. *IMA Journal of Numerical Analysis*, 10:137–154, 1990.

[DM83] W. Dahmen and C. A. Micchelli. Recent progress in multivariate splines. In C. Chui, L. L. Schumaker, and J. D. Ward, editors, *Approximation Theory IV*, pages 27–121. Academic Press, 1983.

[DR95] N. Dyn and A. Ron. Radial basis function approximation: from gridded centers to scattered centers. *Proceedings of the London Mathematical Society*, 71(3):76–108, 1995.

[DS78] D. Doo and M. Sabin. Behaviour of recursive division surfaces near extraordinary points. *Computer-Aided Design*, 10(6):356–360, 1978.

[DTS93] W. Dahmen and T-M. Thamm-Scharr. Cubicoids: modeling and visualization. *Computer Aided Geometric Design*, 10:93–108, 1993.

[Duc76] J. Duchon. Interpolation des fonctions de deux variables suivant le principe de la flexion des plaques minces. *Analyse Numeriques*, 10:5–12, 1976.

[Dyn87] N. Dyn. Interpolation of scattered data by radial functions. In C. K. Chui, L. L. Schumaker, and F. I. Utreras, editors, *Topics in multivariate approximation*, pages 47–61. Academic Press, New York, 1987.

[Dyn89] N. Dyn. Interpolation and approximation by radial and related functions. In C. K. Chui, L. L. Schumaker, and J. D. Ward, editors, *Approximation theory VI: Volume 1*, pages 211–234. Academic Press, New York, 1989.

[Dyn92] N. Dyn. Subdivision schemes in computer-aided geometric design. In W. Light, editor, *Advances in Numerical Analysis (Volume II): Wavelets, Subdivision Algorithms, and Radial Basis Functions*, pages 36–104. Clarendon Press, Oxford, 1992.

[EM94] H. Edelsbrunner and E. P. Mucke. Three-dimensional alpha shapes. *ACM Transactions on Graphics*, 13:43–72, 1994.

[ES92] H. Edelsbrunner and N. R. Shah. Incremental topological flipping works for regular triangulations. *Proceedings of the 8th Annual ACM Symposium on Computational Geometry*, pages 43–52, 1992.

[ETW92] H. Edelsbrunner, T. S. Tan, and R. Waupotitsch. An $o(n^2 \log n)$ time algorithm for the minmax angle triangulation. *SIAM Journal of Scientific and Statistical Computing*, 13:994–1008, 1992.

[Far82] G. Farin. A construction for visual C^1 continuity of polynomial surface patches. *Computer Graphics and Image Processing*, 20:272–282, 1982.

[Far86] G. Farin. Triangular Bernstein-Bézier patches. *Computer Aided Geometric Design*, 3(2):83–128, 1986.

[Far90] G. Farin. Surfaces over Dirichlet tessellations. *Computer Aided Geometric Design*, 7:281–292, 1990.

[FBG88] R. Franke, E. Barker, and J. Goerss. The use of observed data for the initial-value problem in numerical weather prediction. *Computers and Mathematics in Applications*, 16:169–184, 1988.

[FH94] T. A. Foley and H. Hagen. Advances in scattered data interpolation. *Surveys on mathematics for industry*, 4:71–84, 1994.

[FH97] R. Franke and H. Hagen. Least square surface approximation using multiquadrics and parametric domain distortion. Submitted to CAGD, 1997.

[FHN93] T. A. Foley, H. Hagen, and G. M. Nielson. Visualizing and modeling unstructured data. *Visual Computer*, 9:439–449, 1993.

[FHN94] R. Franke, H. Hagen, and G. M. Nielson. Least square surface approximation to scattered data using multiquadric functions. *Advances in Computational Mathematics*, 2:81–99, 1994.

[FHN95] R. Franke, H. Hagen, and G. M. Nielson. Repeated knots in least squares multiquadric functions. *Computing Supplem.*, 10:177–185, 1995.

[FI96a] M. S. Floater and A. Iske. Multistep scattered data interpolation using compactly supported radial basis functions. *Journal of Comp. Appl. Math.*, 73:65–78, 1996.

[FI96b] M. S. Floater and A. Iske. Thinning and approximation of large sets of scattered data. In F. Fontanella, K. Jetter, and P. J. Laurent, editors, *Advanced Topics in Multivariate Approximation*. World Scientific Publishing Co., Inc., 1996. to appear.

[FI98] M. S. Floater and A. Iske. Thinning algorithms for scattered data interpolation. *BIT*, 1998. to appear.

[FL90] T. A. Foley and D. A. Lane. Visualization of irregular multivariate data. In G. M. Nielson and L. Rosenblum, editors, *Proceedings of Visualization '90*, pages 247–254. IEEE, 1990.

[FL91] T. A. Foley and D. A. Lane. Multi-valued volumetric visualization. In G. M. Nielson and L. Rosenblum, editors, *Proceedings of Visualization '91*, pages 218–225. IEEE, 1991.

[FLN90a] T. A. Foley, D. Lane, and G. M. Nielson. Visualizing functions over a sphere. *IEEE Computer Graphics and Applications*, 10(1):32–40, January 1990.

[FLN+90b] T. A. Foley, D. Lane, G. M. Nielson, R. Franke, and H. Hagen. Interpolation of scattered data on closed surfaces. *Computer Aided Geometric Design*, 7:303–312, 1990.

[FN80] R. Franke and G. M. Nielson. Smooth interpolation of large sets of scattered data. *International Journal for Numerical Methods in Engineering*, 15:1691–1704, 1980.

[FN91] R. Franke and G. M. Nielson. Scattered data interpolation: A tutorial and survey. In H. Hagen and D. Roller, editors, *Geometric Modeling: Methods and Applications*, pages 131–160. Springer, NY, 1991.

[Fol84] T. A. Foley. Three-stage interpolation to scattered data. *Rocky Mountain Journal of Mathematics*, 14:141–150, 1984.

[Fol87] T. A. Foley. Interpolation and approximation of 3-D and 4-D scattered data. *Comput. Math. Applic.*, 13(8):711–740, 1987.

[Fol90] T. A. Foley. Interpolation to scattered data on a spherical domain. In J. Mason and M. Cox, editors, *Algorithms for Approximation II*, pages 303–312. Chapman and Hall, 1990.

[Fol94] T. A. Foley. Near optimal parameter selection for multiquadric interpolation. *Journal of Applied Science and Computation*, 1:54–69, 1994.

[Fra82a] R. Franke. Scattered data interpolation: Tests of some methods. *Mathematics of Computation*, 38:181–200, 1982.

[Fra82b] R. Franke. Smooth interpolation of scattered data by local thin plate splines. *Computers and Mathematics with Applications*, 8:273–281, 1982.

[Fra86] R. Franke. Covariance functions for statistical interpolation. Technical report, Naval Postgraduate School, Monterey, California, 1986. NPS-53-86-007.

[FS69] G. Fix and G. Strang. Fourier analysis of the finite element method in Ritz-Galerkin theory. *Stud. Appl. Math.*, 48:265–273, 1969.

[FS93] P. Fong and H. P. Seidel. An implementation of triangular B-spline surfaces over arbitrary triangulations. *Computer Aided Geometric Design*, 10:267–275, 1993.

[FS95] R. Franke and K. Salkauskas. Localization of multivariate interpolation and smoothing methods. *Journal of Computational and Applied Mathematics*, 73:79–94, 1995.

[FSF97] W. Freeden, M. Schreiner, and R. Franke. A survey on spherical spline approximation. *Surveys on Mathematics for Industry*, 7:29–85, 1997.

[FvDFH90] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes. *Computer Graphics: Principles and Practice*. Addison-Wesley, New York, 1990.

[GC98] G. Gaspari and S. E. Cohn. Construction of correlation functions in two and three dimensions. *Quarterly Journal of the Royal Meteorological Society*, 1998. to appear.

[Gir92] F. Girosi. Some extensions of radial basis functions and their applications in artificial intelligence. *Computers and Mathematics with Applications*, 24(12):61–80, 1992.

[Gu91] C. Gu. Smoothing splines by generalized cross validation or generalized maximum likelihood. *Siam Journal on Scientific and Statistical Computing*, 12:383–398, 1991.

[Guo91] B. Guo. Surface generation using implicit cubics. In N. M. Patrikalakis, editor, *Scientific Visualization of Physical Phenomena*, pages 485–530. Springer-Verlag, Tokyo, 1991.

[Hag86] H. Hagen. Geometric surface patches without twist constraints. *Computer Aided Geometric Design*, 3:179–184, 1986.

[Har71] R. L. Hardy. Multiquadric equations of topography and other irregular surfaces. *Journal of Geophysical Research*, 76:1905–1915, 1971.

[Har90] R.L. Hardy. Theory and applications of the multiquadric-biharmonic method. *Computers and Mathematics with Applications*, 19:163–208, 1990.

[HD72] R. L. Harder and R. N. Desmarais. Interpolation using surface splines. *Journ. Aircraft*, 9:189–191, 1972.

[HDD+92] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. *SIGGRAPH Proceedings*, pages 71–78, 1992.

[HDD+93] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Mesh optimization. *SIGGRAPH Proceedings*, pages 19–26, 1993.

[HDD+94] H. Hoppe, T. DeRose, T. Duchamp, H. Jin, J. McDonald, and W. Stuetzle. Piecewise smooth surface reconstruction. *SIGGRAPH Proceedings*, pages 295–302, 1994.

[Her85] G. Herron. Smooth closed surfaces with discrete triangular interpolants. *Computer Aided Geometric Design*, 2:297–306, 1985.

[HG75] R. L. Hardy and W. M. Gopfert. Least squares prediction of gravity anomalies, geoidal undulations, and deflections of the vertical with multiquadric harmonic functions. *Geophysical Research Letters*, 2:423–426, 1975.

[HH92] H. Hagen and S. Hahmann. Generalized focal surfaces: A new method for surface interrogation. *Proceedings of IEEE Visualization '92*, pages 70–76, 1992.

[HHS+92] H. Hagen, S. Hahmann, T. Schreiber, Y. Nakajima, B. Wordenweber, and P. H. Grundstedt. Surface interrogation algorithms. *IEEE Computer Graphics and Applications*, 12(5):53–60, September 1992.

[HHS+93] H. Hagen, S. Hahmann, T. Schreiber, E. Gschwind, B. Wordenweber, and Y. Nakajima. Curve and surface interrogation. In H. Hagen, H. Mueller, and G. Nielson, editors, *Focus on Scientific Visualization*, pages 243–258. Springer Verlag, 1993.

[HKD93] M. Halstead, M. Kass, and T. DeRose. Efficient fair interpolation using Catmull-Clark surfaces. *SIGGRAPH Proceedings*, pages 35–44, 1993.

[HL93] J. Hoschek and D. Lasser. *Computer Aided Geometric Design*. A. K. Peters, Wellesley, Massachusetts, 1993.

[Hos84] J. Hoschek. Detecting regions with undesirable curvature. *Computer Aided Geometric Design*, 1:183–192, 1984.

[Hos85] J. Hoschek. Smoothing of curves and surfaces. *Computer Aided Geometric Design*, 2:97–105, 1985.

[HP89] H. Hagen and H. Pottmann. Curvature continuous triangular interpolants. In T. Lyche and L. L. Schumaker, editors, *Mathematical Methods in Computer Aided Geometric Design*, pages 373–384. Academic Press, Boston, 1989.

[Jac88] I. R. H. Jackson. *Radial basis function methods for multivariable approximation*. PhD thesis, University of Cambridge, 1988.

[Jen87] T. Jensen. Assembling triangular and rectangular patches and multivariate splines. In G. Farin, editor, *Geometric Modeling: Algorithms and New Trends*, pages 203–220. SIAM, Philadelphia, 1987.

[Jia86] R. Q. Jia. A counterexample to a result concerning controlled approximation. *Proceedings of American Mathematical Society*, 97:647–654, 1986.

[KC92] E. J. Kansa and R. E. Carlson. Improved accuracy of multiquadric interpolation using variable shape parameters. *Computers and Mathematics with Applications*, 24(12):99–120, 1992.

[Kla80] R. Klass. Correction of local surface irregularities using reflection lines. *Computer-Aided Design*, 12:73–77, 1980.

[Law77] C. L. Lawson. Software for C^1 surface interpolation. In J. R. Rice, editor, *Mathematical Software III*, pages 161–194. Academic Press, New York, 1977.

[Law84] C. L. Lawson. C^1 surface interpolation for scattered data on a sphere. *Rocky Mountain Journal of Mathematics*, 14:177–202, 1984.

[Law86] C. L. Lawson. Properties of n-dimensional triangulations. *Computer Aided Geometric Design*, 3:231–247, 1986.

[LD89] C. Loop and T. DeRose. A multisided generalization of Bézier surfaces. *ACM Transactions on Graphics*, 8(3):204–234, 1989.

[LD90] C. Loop and T. DeRose. Generalized B-spline surfaces of arbitrary topology. *Computer Graphics*, 24(4):347–356, 1990.

[Lig92] W. Light. Some aspects of radial basis function approximation. In S. P. Singh, editor, *Approximation Theory, Spline Functions and Applications*, volume 356 of *NATO ASI Series C*, pages 163–190. Kluwer, 1992.

[LJZU97] S. K. Lodha, A. Joseph, and B. Zane-Ulman. Comparing volumetric radial interpolants. University of california, Sanra Cruz, In Preparation, 1997.

[LMD92] M. Lounsbery, S. Mann, and T. DeRose. Parametric surface interpolation. *IEEE Computer Graphics and Applications*, 12(5):45–52, September 1992.

[Loo87] C. Loop. *Smooth subdivision surfaces based on triangles.* PhD thesis, University of Utah, 1987. M. S. Thesis, Department of Mathematics.

[Loo94] C. Loop. Smooth spline surfaces over irregular meshes. *SIGGRAPH Proceedings '94*, 1994.

[LSPW96] S. K. Lodha, B. Sheehan, A. Pang, and C. Wittenbrink. Visualizing geometric uncertainty of surface interpolants. In *Proceedings of Graphics Interface '96, Toronto, Canada*, pages 238–245, May 1996.

[Mei79] J. Meinguet. Multivariate interpolation at arbitrary points made simple. *Z. Angew. Math. Phys.*, 30:292–304, 1979.

[MF92] J. R. McMahon and R. Franke. Knot selection for least squares thin plate splines. *SIAM Journal on Scientific and Statistical Computing*, 13(2):484–498, 1992.

[Mic86] C. A. Micchelli. Interpolation of scattered data: distance matrices and conditionally positive definite functions. *Constructive Approximation*, 2:11–22, 1986.

[MLL+92] S. Mann, C. Loop, M. Lounsbery, D. Meyers, J. Painter, T. DeRose, and K. Sloan. A survey of parametric scattered data fitting using triangular interpolants. In Hans Hagen, editor, *Curve and Surface Design*, pages 145–172. SIAM, 1992.

[MN88] W. R. Madych and S. A. Nelson. Multivariate inerpolation and conditionally positive definite functions. *Approximation Theory and Applications*, 4:77–89, 1988.

[MN92] W. R. Madych and S. A. Nelson. Bounds on multivariate polynomials and exponential error estimates for multiquadric interpolation. *Journal of Approximation Theory*, 70:94–113, 1992.

[Nas87] A. H. Nasri. Polyhedral subdivision methods for free-form surfaces. *ACM Transactions on Graphics*, 6(1):29–73, 1987.

[ND91] G. M. Nielson and T. Dierks. Modelling and visualization of scattered volumetric data. In E. J. Farrell, editor, *Extracting Meaning from Complex Data: Processing, Display, Interaction II*, pages 22–33. Springer-Verlag, 1991. volume 1459 of Proceedings of SPIE Conference, February 26–28 1991.

[NF80] G. M. Nielson and T. Foley. Multivariate interpolation to scattered data using delta iteration. In E. W. Cheney, editor, *Approximation Theory III*, pages 419–424. Academic Press, New York, 1980.

[NF83] G. M. Nielson and R. Franke. Surface construction based upon triangulations. In R. E. Barnhill and W. Boehm, editors, *Surfaces in CAGD*, pages 163–179. North-Holland, 1983.

[NF84] G. M. Nielson and R. Franke. A method for construction of surfaces under tension. *Rocky Mountain Journal of Mathematics*, 14:203–222, 1984.

[NF89] Gregory M. Nielson and Thomas A. Foley. A survey of applications of an affine invariant norm. In T. Lyche and L.L. Schumaker, editors, *Mathematical Methods in CAGD*, pages 445–467. Academic Press, 1989.

[NF94] G. M. Nielson and T. Foley. Modeling of scattered multivariate data. In C. Giertsen and P. Fevang, editors, *Eurographics '94 State of the Art Reports*, pages 38–59. Norsied Press, Oslo, Norway, 1994.

[Nie74] G. M. Nielson. Multivariate smoothing and interpolating splines. *SIAM Journal on Numerical Analysis*, 11(2):435–446, 1974.

[Nie79] G. M. Nielson. The side-vertex method for interpolation in triangles. *Journal of Approximation Theory*, 25:318–336, 1979.

[Nie80] G. M. Nielson. Minimum norm interpolation in triangles. *SIAM Journal on Numerical Analysis*, 17(1):46–62, February 1980.

[Nie83] G. M. Nielson. A method for interpolating scattered data based upon a minimum norm network. *Mathematics of Computation*, 40:253–271, 1983.

[Nie86] G. M. Nielson. A rectangular nu-spline for interactive surface design. *IEEE Computer Graphics and Applications*, 6(2):35–41, 1986.

[Nie87a] G. M. Nielson. Coordinate free scattered data interpolation. In C. Chui, F. Utreras, and L. Schumaker, editors, *Topics in Multivariate Approximation*, pages 175–184. Academic Press, New York, 1987.

[Nie87b] G. M. Nielson. A transfinite visually continuous triangular interpolant. In G. Farin, editor, *Geometric Modeling: Algorithms and New Trends*, pages 235–246. SIAM, 1987.

[Nie88] G. M. Nielson. Interactive surface design using triangular network splines. In S. Slaby and H. Stachel, editors, *Engineering Graphics and Descriptive Geometry*, volume 2, pages 70–77. 1988.

[Nie93a] G. M. Nielson. CAGD's top ten: what to watch. *IEEE Computer Graphics and Applications*, 13(1):35–37, 1993.

[Nie93b] G. M. Nielson. A characterization of an affine invariant triangulation. In G. Farin, H. Hagen, H. Noltemeier, and W. Knoedel, editors, *Geometric Modelling: Computing Supplementum 8*, pages 191–210. Springer-Verlag, 1993.

[Nie93c] G. M. Nielson. Modeling and visualizing volumetric and surface-on-surface data. In H. Hagen, H. Mueller, and G. Nielson, editors, *Focus on Scientific Visualization*, pages 235–246. Springer Verlag, 1993.

[Nie93d] G.M. Nielson. Scattered data modeling. *IEEE Computer Graphics and Applications*, 13(1):60–70, 1993.

[Nie94] G. M. Nielson. Research issues in modeling for the analysis and visualization of large data sets. In L. Rosenblum et al. editor, *Scientific Visualization: Advances and Challenges*, pages 143–157. Academic Press, 1994.

[NO92] G. M. Nielson and K. Opitz. The face-vertex method for smooth interpolation in tetrahedron. Computer Science Tech Report TR-92-013, 1992.

[NR87] G. M. Nielson and R. Ramraj. Interpolation over a sphere based upon a minimum norm network. *Computer Aided Geometric Design*, 4:41–57, 1987.

[NSW97] F. J. Narcowich, R. Schaback, and J. D. Ward. The multilevel method: Rates of approximation. Manuscript, 1997.

[NT94a] G. M. Nielson and J. Tvedt. Comparing methods of interpolation for scattered volumetric data. In D. Rogers and R. A. Earnshaw, editors, *State of the Art in Computer Graphics – Aspects of Visualization*, pages 67–86. Springer-Verlag, 1994.

[NT94b] G. M. Nielson and J. Tvedt. Modeling of scattered multivariate data. In C. Giertson and P. Fevang, editors, *Eurographics '94 State of the Art Reports*, pages 38–59. Norsied Press, Oslo, Norway, 1994.

[NW91] F. J. Narcowich and J. D. Ward. Norm of inverses and condition number for matrices associated with scattered data. *Journal of Approximation Theory*, 64:69–94, 1991.

[NW92] F. J. Narcowich and J. D. Ward. Norm estimates for the inverses of a general class of scattered data radial function interpolation matrices. *Journal of Approximation Theory*, 69:84–109, 1992.

[PE90] H. Pottmann and M. Eck. Modified multiquadric methods for scattered data interpolation over a sphere. *Computer Aided Geometric Design*, 7:313–321, 1990.

[PE96] H. Pottmann and M. Eck. Equivalence of trivariate multiquadrics and spherical multiquadrics. Private Communication, 1996.

[Pet90] J. Peters. *Fitting smooth parametric surfaces to 3D data*. PhD thesis, University of Wisconsin, Madison, Center for the Mathematical Sciences, 1990.

[Pet93] J. Peters. Smooth free-form surfaces over irregular meshes generalizing quadratic splines. *Computer Aided Geometric Design*, 10:347–361, 1993.

[Pet95] J. Peters. C^1 surface splines. *SIAM Journal on Numerical Analysis*, 32(2):645–666, 1995.

[PHD91] H. Pottmann, H. Hagen, and A. Divivier. Visualizing functions on a surface. *The Journal of Visualization and Computer Animation*, 2:52–58, 1991.

[Pip87] B. Piper. Visually smooth interpolation with triangular Bézier patches. In G. Farin, editor, *Geometric Modeling: Algorithms and New Trends*, pages 221–234. SIAM, 1987.

[PK89] N. M. Patrikalakis and G. A. Kriezis. Representation of piecewise continuous algebraic surfaces in terms of B-splines. *The Visual Computer*, 5:360–374, 1989.

[PO94] H. Pottmann and K. Opitz. Curvature analysis and visualization for functions defined on Euclidean spaces and surfaces. *Computer Aided Geometric Design*, 11:655–674, 1994.

[Pot88] H. Pottmann. Eine verfeinerung der isophotenmethode zur qualitatsanalyse von freiformflachen. *CAD and Computergraphik*, 4:99–109, 1988.

[Pot92] H. Pottmann. Interpolation on surfaces using minimum norm networks. *Computer Aided Geometric Design*, 9:51–67, 1992.

[Pow90] M. J. D. Powell. The theory of radial basis function approximation in 1990. In W. Light, editor, *Advances in Numerical Analysis (Volume II): Wavelets, Subdivision Algorithms, and Radial Basis Functions*, pages 105–210. Clarendon Press, Oxford, 1990.

[Pow92a] M. J. D. Powell. Tabulation of thin plate splines on a very fine two-dimensional grid. In D. Braess and L. L. Schumaker, editors, *Numerical Methods of Approximation Theory*, pages 221–244. Birkhauser-Verlag, Basel, 1992.

[Pow92b] M. J. D. Powell. Truncated Laurent expansions for the fast evaluation of thin plate splines. In J. C. Mason and M. G. Cox, editors, *Algorithms for Appoximation III*. Chapman and Hall, London, 1992.

[PS77] M. J. D. Powell and M. A. Sabin. Piecewise quadratic approximations on triangles. *ACM Transactions on Mathematical Software*, 3(4):316–325, 1977.

[PS85] Franco Preparata and Michael Ian Shamos. *Computational Geometry: An Introduction*. Springer Verlag, 1985.

[PS95] R. Pfeifle and H. P. Seidel. Spherical triangular B-splines with applications to data fitting. In Frits Post and M. Gobel, editors, *Eurographics '95*. Blackwell Publisher, 1995.

[QS90] E. Quak and L. L. Schumaker. Cubic spline fitting using data dependent triangulations. *Computer Aided Geometric Design*, 7:293–301, 1990.

[RC84] R. J. Renka and A. K. Cline. A triangle-based C^1 interpolation method. *Rocky Mountain Journal of Mathematics*, 14:223–237, 1984.

[Rei95] U. Reif. A unified approach to subdivision algorithms near extraordinary points. *Computer Aided Geometric Design*, 12:153–174, 1995.

[Ren84a] R. J. Renka. Algorithm 624: triangulation and interpolation of arbitrary distributed points in the plane. *ACM Transactions on Mathematical Software*, 10:440–442, 1984.

[Ren84b] R. J. Renka. Interpolation of data on the surface of a sphere. *ACM Transactions on Mathematical Software*, 10:417–436, 1984.

[Ren88] R. J. Renka. Algorithm 661: QSHEP3D: quadratic Shepard method for trivariate interpolation of scattered data. *ACM Transactions on Mathematical Software*, 14:151–152, 1988.

[Res87] K. L. Rescorla. C^1 trivariate polynomial interpolation. *Computer Aided Geometric Design*, 4:237–244, 1987.

[SAG84] T. W. Sederberg, D. C. Anderson, and R. Goldman. Implicit representation of parametric curves and surfaces. *Computer Vision Graphics and Image Processing*, 28:72–74, 1984.

[Sal14] G. Salmon. *A Treatise on the Analytic Geometry of Three Dimensions*. Longmans, Green, London, 1914.

[Sal92] K. Salkauskas. Moving least squares interpolation with thin-plate splines and radial basis functions. *Computers and Mathematics with Applications*, 24(12):177–185, 1992.

[Sch76] L. L. Schumaker. Fitting surfaces to scattered data. In *Approximation Theory II*, pages 203–268. Academic Press, New York, 1976.

[Sch93a] L. L. Schumaker. Computing optimal triangulations using simulated annealing. *Computer Aided Geometric Design*, 10:329–345, 1993.

[Sch93b] L. L. Schumaker. Triangulations in CAGD. *IEEE Computer Graphics and Applications*, 12(1):47–54, 1993.

[Sch95a] R. Schaback. Creating surfaces from scattered data using radial basis functions. In M. Daehlen, T. Lyche, and L. L. Schumaker, editors, *Mathematical Methods in CAGD III*. Vanderbilt Press, 1995.

[Sch95b] R. Schaback. Error estimates and condition numbers for radial basis function interpolation. *Advances in Computational Mathematics*, 3:251–264, 1995.

[Sed85] T. W. Sederberg. Piecewise algebraic surface patches. *Computer Aided Geometric Design*, 2:53–59, 1985.

[Sed90a] T. W. Sederberg. Techniques for cubic algebraic surfaces I. *IEEE Computer Graphics and Applications*, 10(3):14–25, July 1990.

[Sed90b] T. W. Sederberg. Techniques for cubic algebraic surfaces II. *IEEE Computer Graphics and Applications*, 10(5):12–21, September 1990.

[SG92] P. D. Sampson and P. Guttorp. Nonparametric estimation of nonstationary spatial covariance structure. *Journal of the American Statistical Association*, 87:108–119, 1992.

[Sib81] R. Sibson. A brief description of natural neighbor interpolation. In D. V. Barnett, editor, *Interpreting Multivariate Data*, pages 21–36. Wiley, New York, 1981.

[SS87] L. A. Shirman and C. H. Séquin. Local surface interpolation with Bézier patches. *Computer Aided Geometric Design*, 4(4):279–295, December 1987.

[SW93] N. Sivakumar and J. D. Ward. On the least squares fit by radial functions to multidimensional scattered data. *Numerische Mathematik*, 65:219–243, 1993.

[SW94] R. Schaback and H. Wendland. Special cases of compactly supported radial basis functions. Manuscript, Gottingen, Germany, 1994.

[Tve91] J. Tvedt. A software system for comparison of scattered data interpolation methods. M. S. Thesis, Arizona State University, Tempe, 1991.

[Wah81] G. Wahba. Spline interpolation and smoothing on the sphere. *Siam Journal on Scientific and Statistical Computing*, 2:5–16, 1981. also errata, SJSSC, 3:385–386, 1982.

[War92] J. Warren. Creating rational multi-sided Bézier surfaces using base points. *ACM Transactions on Graphics*, 11(2):127–139, 1992.

[War95] J. Warren. Subdivision methods for geometric design. Manuscript, Rice University, 1995.

[Wat92] D. F. Watson. *Contouring: A guide to the analysis and display of spatial data*. Pergamon Press, 1992.

[Wat94] D. F. Watson. *nngridr: an implementation of natural neighbor interpolation*. David Watson, Claremont, Australia, 1994.

[Wen95] H. Wendland. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in Computational Mathematics*, 4:389–396, 1995.

[WF87] A. J. Worsey and G. Farin. An n-dimensional Clough-Tocher interpolant. *Constructive Approximation*, 3:99–110, 1987.

[WP88] A. J. Worsey and B. Piper. A trivariate Powell-Sabin interpolant. *Computer Aided Geometric Design*, 5:177–186, 1988.

[Wu94a] Z. Wu. Characterization of positive definite radial functions. Manuscript, Department of Mathematics, University of Gottinger, Germany, 1994.

[Wu94b] Z. Wu. Multivariate compactly supported positive definite radial functions. Manuscript, Department of Mathematics, Fudan University, China, 1994.

[WW80] G. Wahba and G. Wendelberger. Some new mathematical methods for variational objective analysis using splines and cross-validation. *Monthly Weather Review*, 108:36–57, 1980.

[Zen73] A. Ženíšek. Polynomial approximation on tetrahedrons in the finite element method. *Journal of Approximation Theory*, 7:334–351, 1973.