APPLIED COMPUTING, MATHEMATICS AND STATISTICS GROUP
Division of Applied Management and Computing

Constrained Visualization using the Shepard Interpolation Family

K.W. Brodlie, M.R. Asim and K. Unsworth

Research Report 09/2003
December 2003

ISSN 1174-6696

RESEARCH REPORT

LINCOLN UNIVERSITY
Te Whare Wānaka O Aoraki

Lincoln University | Applied Computing, Mathematics and Statistics Group

Applied Computing, Mathematics and Statistics Group

The Applied Computing, Mathematics and Statistics Group (ACMS) comprises staff of the Applied Management and Computing Division at Lincoln University whose research and teaching interests are computing and quantitative disciplines. Previously this group was the academic section of the Centre for Computing and Biometrics at Lincoln University.

The group teaches subjects leading to a Bachelor of Applied Computing degree and a computing major in the Bachelor of Commerce and Management. In addition, it contributes computing, statistics and mathematics subjects to a wide range of other Lincoln University degrees. In particular students can take a computing and mathematics major in the BSc.

The ACMS group is strongly involved in postgraduate teaching leading to honours, masters and PhD degrees. Research interests are in modelling and simulation, applied statistics, end user computing, computer assisted learning, aspects of computer networking, geometric modelling and visualisation.

Research Reports

Every paper appearing in this series has undergone editorial review within the ACMS group. The editorial panel is selected by an editor who is appointed by the Chair of the Applied Management and Computing Division Research Committee.

The views expressed in this paper are not necessarily the same as those held by members of the editorial panel. The accuracy of the information presented in this paper is the sole responsibility of the authors.

This series is a continuation of the series "Centre for Computing and Biometrics Research Report", ISSN 1173-8405.

Copyright

Correspondence

This paper represents work to date and may not necessarily form the basis for the authors' final conclusions relating to this topic. It is likely, however, that the paper will appear in some form in a journal or in conference proceedings in the near future. The authors would be pleased to receive correspondence in connection with any of the issues raised in this paper. Please contact the authors either by email or by writing to the address below.

Any correspondence concerning the series should be sent to:

The Editor
Applied Computing, Mathematics and Statistics Group
Applied Management and Computing Division
PO Box 84
Lincoln University
Canterbury
NEW ZEALAND

Email: computing@lincoln.ac.nz

Constrained Visualization using the Shepard Interpolation Family

K.W. Brodlie,[1] M.R. Asim[2] and K. Unsworth[3]

[1] School of Computing, University of Leeds, Leeds LS2 9JT, UK
[2] Department of Computer Science,University of Engineering and Technology, Lahore, Pakistan
[3] Applied Computing, Mathematics and Statistics Group, Division of Applied Management and Computing, P.O.Box 84, Lincoln University, Canterbury, New Zealand

Abstract

This paper discusses the problem of visualizing data where there are underlying constraints that must be preserved. For example, we may know that the data is inherently positive. We show how the Modified Quadratic Shepard method, which interpolates scattered data of any dimensionality, can be constrained to preserve positivity. We do this by forcing the quadratic basis functions to be positive. The method can be extended to handle other types of constraints, including lower bound of 0 and upper bound of 1 - as occurs with fractional data. A further extension allows general range restrictions, creating an interpolant that lies between any two specified functions as the lower and upper bounds.

Categories and Subject Descriptors (according to ACM CCS):

1. Introduction

Visualization can be seen as a process of visual reconstruction. We create a representation of the overall behaviour of the entity we are interested in, from a limited set of sampled information. This reconstruction is achieved by interpolation. However we often have some additional information that we wish to build into the reconstruction: the entity may be subject to certain physical laws which constrain its behaviour - for example, we know densities are always positive and any credible visualization must honour this. Another common constraint occurs with data values that are specified as fractions of a whole - here the reconstruction must lie within the range [0,1] to be realistic. In this paper we examine one particular interpolation approach - the Shepard family of interpolants - and show how this can be adapted to handle constraints on the range of the interpolant.

The problem we are addressing is the interpolation of scattered data. This problem occurs in very many practical situations where data is gathered experimentally (for example, we shall look later at rainfall measurements gathered at a set of recording stations) or is computed in a simulation process using an unstructured grid. There are many approaches to this general problem - a good review of the whole area is given by Lodha and Franke [LF00].

Some methods are based on an initial triangulation of the data points (or equivalent in higher dimensions), followed by a piecewise construction of the interpolant - one piece per triangle. A very simple technique of this type is piecewise linear interpolation. This has a nice property of remaining within the bounds of the data, and thus preserving for example positivity in the data. However it is only C^0 continuous. Smoother interpolants over triangulations will typically fail to remain within the bounds of the data, but several have been modified to incorporate constraints. For example, Asim [Asi00] modifies the Barnhill et al [BBG73] blending method (constrained cubics as suggested by Asim and Brodlie [AB03] are created along triangle edges and blended in the interior); Ong and Wong [OW96] create a C^1 interpolant by blending constrained rational cubics along triangle edges using the Nielson [Nie79] side-vertex method. Mulansky and Schmidt [MS94] construct a constrained C^1 interpolant using quadratic splines on a Powell-Sabin refinement of the original triangulation. Chan and Ong [CO01] create a constrained C^1 interpolant as a combination of cubic Bezier triangles.

All these approaches, however, require the points to be triangulated. Another major class of methods for scattered data interpolation do not involve any prior triangulation step, and can be thought of as 'meshless'. The two main types are radial basis functions (RBFs), which include multiquadrics and thin-plate splines, and Shepard-type methods, which include the modified quadratic Shepard approach and also the moving least squares technique. Both types, RBFs and Shepard, are widely used in practice.

However there has been relatively little work done on the imposition of constraints for these meshless methods. For RBFs, in the special case of thin-plate splines for 2D data, Utreras [Utr85] has shown how positivity can be imposed as a constraint, but the computational cost is rather high, requiring a global optimization problem to be solved at each step of an iteration. Xiao and Woodbury [XW99] look at a number of meshless methods for constrained scattered data interpolation for 3D data. In areas where the entity is known to have a particular value, say, zero, extra data points are added in order to 'encourage' the interpolant to take values close to zero in these areas. If a physical constraint additionally tells us that the entity is non-negative, then the interpolant is simply clamped at zero. A difficulty with this approach is that the resulting interpolant will have derivative discontinuity where the clamping is applied. Our aim is to generate a constrained interpolant which is computationally efficient, and which incorporates the constraint as part of the interpolation process, rather than as an *a posteriori* process such as clamping.

We shall adapt the Shepard family of interpolants. These are described in detail in section 2. Essentially they provide an estimate of the value of the underlying function as a positive combination of basis functions, each basis function exactly interpolating one data point. The influence of each basis function in the combination decreases with distance from its associated data point. We shall derive the constrained family by considering the simple constraint of positivity: that is, we assume the data points are positive, and require the resulting interpolant to be also positive. The Shepard methods will not generally provide this property since any of the basis functions may go negative, and cause the interpolant - as a weighted combination of the basis functions at any point - to similarly go negative. We shall solve the problem by constraining each basis function to be positive within a certain region around its associated data point. This is described in section 3 - showing first by example how positivity is lost when a basis function goes negative, and then discussing a general approach that preserves positivity by applying a linear transformation to any basis function which goes negative, this transformation being such that positivity is achieved. The 1D case is used as a simple illustration, and then we look at the practical application of the method in 2D and 3D. Crucial to the success of the method is an efficient way of deciding if there is a problem to be solved at all, that is do any of the basis functions go negative? The method is illustrated by applying it to rainfall measurements.

In section 4, we extend the work to cover a more general lower bound on the value of the interpolant than simply zero; and also look at upper bounds, and lower and upper bounds. A particularly important case occurs for lower bound of zero, and upper bound of one: this '[0,1] constraint' applies to all data expressed as fractions. Section 5 concludes and suggests further work.

There is one important word to add on terminology. We shall use the term 'positivity' to refer to 'greater than or equal to zero', rather than the more rigorous but somewhat awkward 'non-negativity'. When we mean 'greater than zero', we use the term 'strictly positive'.

2. Shepard Family of Interpolants

The general problem we are addressing is the following. Given a set of N data points $\mathbf{x}_i, i = 1, 2, \ldots, N$, where $\mathbf{x} = (x, y, z, \ldots)$, with associated data values f_i, we seek an interpolating function $F(\mathbf{x})$ such that $F(\mathbf{x}_i) = f_i$. Later we shall be concerned with imposing constraints on the behaviour of $F(\mathbf{x})$, but to start with we consider the unconstrained case.

A popular approach to the problem emerged in the 1960s amongst the contour plotting community, and is now associated with the name of one of its proponents, Shepard [She68]. In its basic form, it involves an inverse distance-weighted average of the data values, constructing $F(\mathbf{x})$ as:

$$F(\mathbf{x}) = \sum_{i=1}^{N} w_i(\mathbf{x}) f_i \tag{1}$$

where the normalised weight function $w_i(\mathbf{x})$ has the form:

$$w_i(\mathbf{x}) = \frac{\sigma_i(\mathbf{x})}{\sum_{j=1}^{N} \sigma_j(\mathbf{x})} \tag{2}$$

where $\sigma_i(\mathbf{x}) = \frac{1}{d_i^2(\mathbf{x})}$ with $d_i = \|\mathbf{x} - \mathbf{x}_i\|_2$

The weights $w_i(\mathbf{x})$ satisfy:

1. $\sum_{i=1}^{N} w_i(\mathbf{x}) = 1$
2. $w_i(\mathbf{x}) \geq 0$
3. $w_i(\mathbf{x}_j) = \delta_{ij}$.

Notice that we have been able to specify the method for arbitrary dimension of the space; we have not had to specify any connectivity between data points; and we have not had to solve any linear system of equations (as is needed in the radial basis function approach).

In practice however this method does not work particularly well, for two reasons:

1. The function $F(\mathbf{x})$ has zero derivative at the data points,

exhibiting as 'flat spots' in the interpolating curve or surface in 1D or 2D respectively. This led to the suggestion by Franke and Nielson [FN80] of replacing the constant f_i in the averaging process by a local best-fit quadratic approximation $Q_i(\mathbf{x})$.

2. The method is global in the sense that any interpolation involves a computation involving all data points. This is computationally inefficient. This led to the suggestion by Franke and Little (reported in [Bar77]) that the weighting functions σ_i be subjected to a damping factor λ to reduce them to zero outside a certain radius of the data point.

When these two modifications are taken together, we have the modified quadratic Shepard method (as proposed by Franke and Nielson [FN80] and further discussed by Nielson [Nie93]). We create an interpolant $F(\mathbf{x})$ as:

$$F(\mathbf{x}) = \sum_{i=1}^{N} w_i(\mathbf{x}) Q_i(\mathbf{x}) \tag{3}$$

where the normalised weight function $w_i(\mathbf{x})$ has the form:

$$w_i(\mathbf{x}) = \frac{\sigma_i(\mathbf{x})}{\sum_{j=1}^{N} \sigma_j(\mathbf{x})} \tag{4}$$

where $\sigma_i(\mathbf{x}) = \frac{1}{d_i^2(\mathbf{x})}(1 - \frac{d_i(\mathbf{x})}{r_w})_+^2$

with $d_i = \|\mathbf{x} - \mathbf{x}_i\|$, and where r_w is a constant defining an area of interest around the interpolation point $\mathbf{x}$, outside of which basis functions have zero weight.

$Q_i(\mathbf{x})$ is the best inverse distance weighted least squares approximation by a quadratic function to the data points. The least-squares calculation is again restricted to those data points within a specified radius, say r_q, of $\mathbf{x}_i$, in order that the method is local. We write Q_i as:

$$Q_i(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{x}_i)^T A(\mathbf{x} - \mathbf{x}_i) + \mathbf{g}^T(\mathbf{x} - \mathbf{x}_i) + f_i \tag{5}$$

(For simplicity of notation, the subscript i is omitted from the terms $A, \mathbf{g}$ on the RHS.)

This method is widely used in practice. An ACM algorithm was published by Renka in 1988 [Ren88a],[Ren88b] for 2D and 3D, and this is available in the NAG Library [NAG03]. More recently, Renka has developed two variations, one with cubic rather than quadratic basis functions [Ren99a], and the other with 10 parameter cosine series [Ren99b]. On average, there is some improvement gained by using cubics rather than quadratics, but the differences are not dramatic - see the detailed comparison by Renka and Brown [RB99]. Berry and Minser [BM99] have presented an ACM algorithm for 5D modified quadratic Shepard interpolation, and describe its application to the response of forests to changing climate.

A somewhat unsatisfactory feature of the approach is the sensitivity of the interpolant to the choice of the parameters r_w and r_q. Franke and Nielson [FN80] suggested a choice, for $N = 2$, of:

$$r_w = \frac{D}{2}\sqrt{\frac{N_w}{N}},\ r_q = \frac{D}{2}\sqrt{\frac{N_q}{N}} \tag{6}$$

where D is the greatest distance between two data points, and N_w and N_q are the average number of points to be used in the weighted average, and weighted fitting of Q_i, respectively. This choice presumes a fairly even distribution of the data points. Suggested values are $N_w = 9, N_q = 18$, while for 3D, Nielson [Nie93] suggests $N_w = 27, N_q = 54$, with (presumably) cube root being required in equation (6).

Renka [Ren88a],[Ren88b] took a slightly different approach to the choice of radii. He allows r_w, r_q to vary, choosing r_w large enough so that the weighted average process includes at least N_w basis functions, and r_q large enough so that the weighted fitting of the quadratic Q_i includes at least N_q data points. With this interpretation, the recommended values (after many experiments) are, for 2D, $N_w = 19, N_q = 13$; and for 3D, $N_w = 32, N_q = 17$ or 18 (although a larger N_q for some problems was better). This approach is intended to allow for unevenly distributed data points.

The modified quadratic Shepard method is now widely used, and the original Shepard method of equation (1) is rarely, if ever, seen in practice. However the original method does have one useful property which we lose in the modified quadratic version. As explained by Gordon and Wixom [GW78], the original Shepard method satisfies the following Maximum Principle:

Theorem 1 (Maximum Principle for Shepard's Method) Let $M = max\{f_i\}$ and $m = min\{f_i\}$. Then $F(\mathbf{x})$ satisfies:

$$m \leq F(\mathbf{x}) \leq M \tag{7}$$

Thus the interpolant lies within the range of the data, and one consequence for example is that a positive interpolant is guaranteed if the data values are positive.

We illustrate this with a very simple example, in 1D. The data set in Table 1 shows the percentage of oxygen in the flue gas, as coal burns in a furnace. The oxygen percentage is inherently positive, and we therefore require the interpolant to preserve this property (we have used this data set previously to demonstrate positive curve drawing by piecewise cubics - see [AB03]). Figure 1 shows the original Shepard interpolant - the 'flat spots' are very evident, and indeed the appearance is in general unsatisfactory. However note that the curve does remain positive. As we extrapolate to infinity, the value of the curve tends to the average of the data values.

Figure 2 shows the modified quadratic Shepard interpolant, applied to the same data set, with $N_w = 9$ and $N_q = 18$ (in Franke and Nielson style). Generally the behaviour is far superior, but the curve now goes beyond the range of the data values and indeed the positivity constraint is violated. We see the same behaviour when the method is applied to

x (time in mins)	0	2	4	10	28	30	32
y (% of oxygen)	20.8	8.8	4.2	0.5	3.9	6.2	9.6

Table 1: *Percentage of oxygen in flue gas*

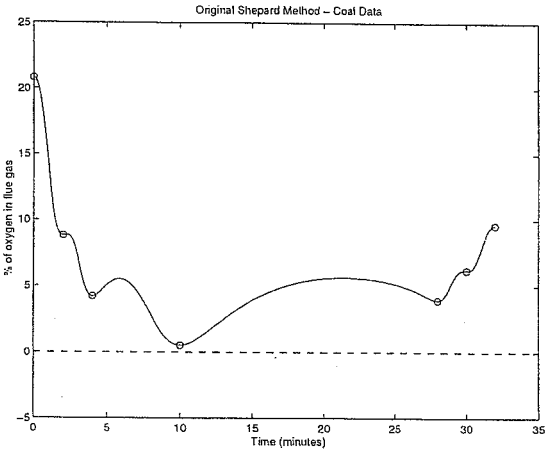

Figure 1: *One-dimensional coal burning data - Flat-spots from the Original Shepard Method*

surface interpolation in 2D, or volumetric interpolation in 3D, or indeed higher dimensions.

This motivates our work. We would like to retain the improved interpolation behaviour of the modified quadratic approach, but we would like to be able to impose constraints.

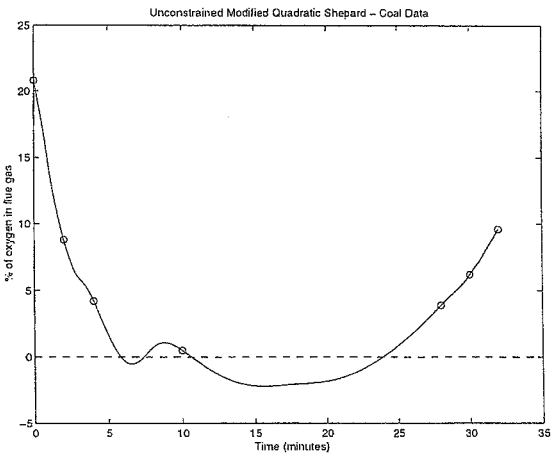

Figure 2: *One-dimensional coal burning data - Modified Quadratic Shepard Method loses positivity*

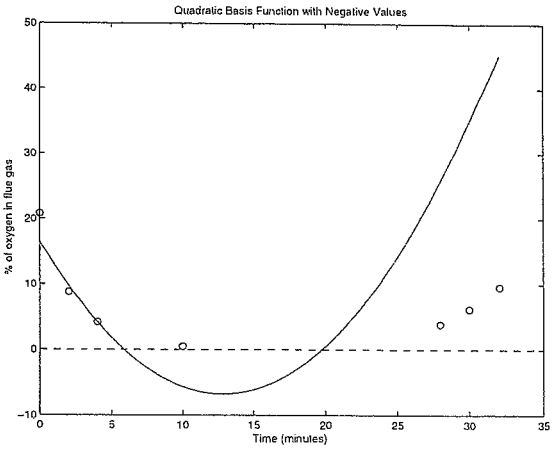

Figure 3: *Quadratic basis function Q_3 has negative values*

Rather than the Maximum Principle of Theorem 1 and the original Shepard method, we would like to express the constraint in a way which is detached from the data. In the first instance we shall consider positivity, and so we seek an interpolant $F(\mathbf{x})$ which will satisfy the constraint

$$F(\mathbf{x}) \geq 0 \tag{8}$$

3. Constrained Modified Quadratic Shepard Method

3.1. How Positivity is Lost

We can gain useful insight into the problem through examining the 1D example of the previous section. In Figure 3, we show the quadratic basis function $Q_3(x)$ that is generated, interpolating at $(x_3, y_3) = (4.0, 4.2)$ and approximating the other data in a weighted least-squares sense. It clearly goes negative within the range of interpolation, and contributes to the loss of positivity exhibited in the overall modified quadratic Shepard, or MQS, interpolant shown in Figure 2.

Remember that F is a positive linear combination of the Q_i values at x. Therefore, we can ensure positivity if we can constrain each basis function to be positive within the range of the interpolation. This sufficient condition is a key point of our approach.

3.2. Positive Quadratic Basis Functions

Our objective then is to constrain the quadratic basis functions to be positive within the region where they are active. For the modified quadratic version, this means that the ith basis function must be positive within a region:

$$\|\mathbf{x} - \mathbf{x}_i\|_2 \leq r_w \tag{9}$$

where r_w is fixed for the Franke-Nielson approach, and varies with $\mathbf{x}$ in the case of the Renka approach.

We are therefore interested in solving the problem: minimize

$$Q_i(\mathbf{x}) = \frac{1}{2}(\mathbf{x}-\mathbf{x}_i)^T A(\mathbf{x}-\mathbf{x}_i) + \mathbf{g}^T(\mathbf{x}-\mathbf{x}_i) + f_i \qquad (10)$$

subject to the constraint (9).

If the minimum is positive, then obviously $Q_i(\mathbf{x})$ is positive everywhere it is active in the interpolation calculation and no action need be taken. If the minimum is negative, however, then it is possible that the basis function could contribute a negative component in the evaluation of $F(\mathbf{x})$ in equation (3). In this case we modify Q_i.

Inspection of Figure 3 lets us motivate the modification. The range of Q_3 is too great, and thus we are led to apply a positive scaling factor, α say, where $\alpha < 1$. The factor α must compress the range of data values $[Q_3^{min}, f_3]$ (where Q_3^{min} is the minimum of Q_3) to the range $[0, f_3]$. This scaling will destroy the interpolation condition, $Q_3 = f_3$, and so we also apply a shift, $\beta = (1-\alpha)f_3$, in order to retain interpolation. In this way we construct a constrained quadratic basis function, R_3, which is a scaled and shifted transform of Q_3, compressing the range of Q_3, while making sure it still passes through the data point.

In general, then, we construct for any Q_i which goes negative, a revised basis function R_i, which is a linear transformation of Q_i:

$$R_i(\mathbf{x}) = \alpha Q_i(\mathbf{x}) + \beta \qquad (11)$$

where we apply a scale factor $\alpha \in [0, 1]$ to reduce the range of Q_i and a shift factor β to maintain interpolation. Specifically,

$$\alpha = \frac{f_i}{f_i - Q_i^{min}}, \quad \beta = (1-\alpha)f_i \qquad (12)$$

where Q_i^{min} is the minimum of Q_i within the region it is active.

There are two points to note at this stage:

- If $f_i = 0$ for any i, that is, the data value equals the constraint, then we have $\alpha = 0$ and $\beta = f_i$. Thus $R_i(\mathbf{x}) = f_i$, and the basis function reverts to the constant value used in the original Shepard method 1.
- If we want to construct an interpolant that is strictly positive, then we need to choose α such that:

$$0 < \alpha < \frac{f_i}{f_i - Q_i^{min}}$$

The smaller the value of α, the less is the range of R_i.

This simple 'scale-then-shift' operation has some nice properties in addition to raising the minimum to zero, and preserving the interpolation condition. First we rewrite equation (10) in terms of the unique stationary point of Q_i, say $\mathbf{x}_s$, as:

$$Q_i(\mathbf{x}) = \frac{1}{2}(\mathbf{x}-\mathbf{x}_s)^T A(\mathbf{x}-\mathbf{x}_s) + Q_s \qquad (13)$$

where Q_s is the value of Q_i at $\mathbf{x}_s$. Then we have:

$$\begin{aligned} R_i(\mathbf{x}) &= \alpha Q_i(\mathbf{x}) + \beta \\ &= \frac{1}{2}(\mathbf{x}-\mathbf{x}_s)^T(\alpha A)(\mathbf{x}-\mathbf{x}_s) + \gamma \end{aligned} \qquad (14)$$

where $\gamma = \beta + \alpha Q_s$. From equation (14), it is clear that R_i has the same minimum point, $\mathbf{x}_s$, as Q_i and moreover, since the Hessian matrix A is scaled by $\alpha \in [0, 1]$, the eigenvectors of the new Hessian are unchanged, and the eigenvalues are scaled by a uniform positive constant α. This implies that the essential nature of the function, in terms of convex and concave regions, is unchanged by the linear transformation to R_i.

Specifically, we have the following property:

Property 1 Suppose $\mathbf{x}_A$ and $\mathbf{x}_B$ are any two points such that

$$Q_i(\mathbf{x}_A) \leq Q_i(\mathbf{x}_B) \qquad (15)$$

Then it follows from equation (11) that

$$R_i(\mathbf{x}_A) \leq R_i(\mathbf{x}_B) \qquad (16)$$

In the next subsection, we look at how this works for the one-dimensional case, as a simple illustration of the method. For higher dimensions, the solution of the constrained minimization problem (given by (10)) requires some discussion, as the approach will only be feasible if this can be solved efficiently - so this is described in the following subsection. We then show how the method works in practice on 2D and 3D interpolation problems.

3.3. One-dimensional Positive MQS

In the one-dimensional case, the problem (10) reduces to: minimize

$$Q_i(x) = \frac{1}{2}a(x-x_i)^2 + g(x-x_i) + f_i \qquad (17)$$

subject to the constraint $|x - x_i| \leq r_w$.

If $a \leq 0$, that is, Q_i is concave, then the minimum, x_{min}, will lie at an end-point of the interval; if $a > 0$, then Q_i is convex and x_{min} may lie in the interior (if $x_{min} = x_i - \frac{g}{a} \in [x_i - r_w, x_i + r_w]$) or at an end-point otherwise. Whatever the case, it is straightforward to calculate the linear transformation:

$$R_i(x) = \alpha Q_i(x) + \beta \qquad (18)$$

with α and β chosen according to equation (12).

In Figure 4, we show the revised quadratic basis function R_3 for our simple example, with the original Q_3 alongside. Notice how the new basis function is positive, and retains the same shape except for being 'squashed'. In Figure 5, we show the resulting 'constrained MQS' interpolant when the

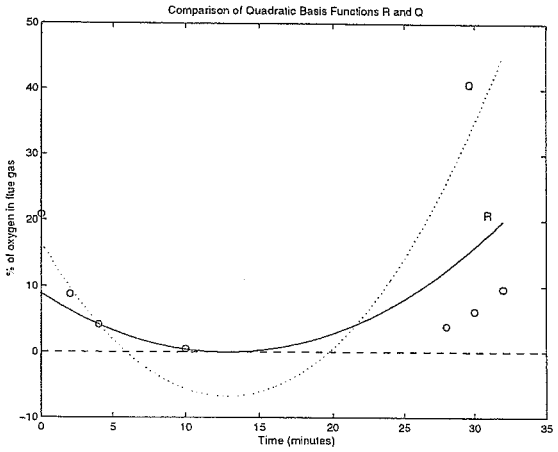

Figure 4: *Quadratic basis function R_3 is positive while Q_3 has negative values*

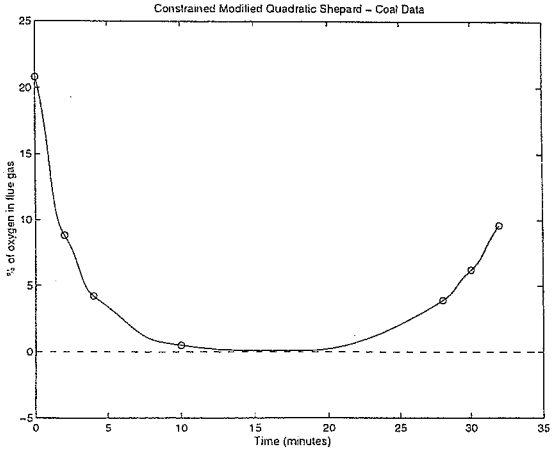

Figure 5: *One-dimensional coal burning data - Constrained Modified Quadratic Shepard Method*

revised basis functions are combined in the style of equation (1).

3.4. Solving the Constrained Minimization Problem

In the one-dimensional case, the constrained minimization problem (10) was easy to solve. For higher dimensions the situation is less trivial, and the success of the constrained interpolant depends on being able to solve this efficiently. Fortunately, just such an efficient solution is provided by the Levenberg-Marquardt algorithm.

Recall that the problem to be solved is the following: minimize

$$Q_i(\mathbf{x}) = \frac{1}{2}(\mathbf{x}-\mathbf{x}_i)^T A(\mathbf{x}-\mathbf{x}_i) + \mathbf{g}^T(\mathbf{x}-\mathbf{x}_i) + f_i \qquad (19)$$

subject to the constraint (9). The following theorem (see Theorem 5.2.1 of Fletcher [Fle87] for proof) gives the solution to minimizing a quadratic function within a sphere of given radius, r_w, about a given point $\mathbf{x}_i$:

Theorem 2 (Levenberg-Marquardt) The point

$$\mathbf{x}(\nu) = \mathbf{x}_i - (A + \nu I)^{-1}(-\mathbf{g}) \qquad (20)$$

is the solution of the problem (19), if and only if there exists $\nu \geq 0$ such that

- $A + \nu I$ is positive semi-definite
- if $\nu > 0$, then $\|\mathbf{x} - \mathbf{x}_i\|_2 = r_w$

If such a ν exists, it is unique.

Levenberg-Marquardt algorithms typically proceed as follows. Equation (20) defines a trajectory $\mathbf{x}(\nu)$, and we seek the value of ν such that

$$\|\mathbf{x}(\nu) - \mathbf{x}_i\|_2 = r_w \qquad (21)$$

This is a nonlinear equation in one variable, ν, and is relatively straightforward to solve.

Insight into the calculation is provided in Figure 6. Here we see the contours of a two-dimensional quadratic, Q_i, associated with data point $\mathbf{x}_i$. The unconstrained minimum of Q_i is shown, and this is the solution of the problem (20), for sufficiently large r_w. However as we reduce r_w, the trajectory $\mathbf{x}(\nu)$ follows the path shown in the dotted line towards $\mathbf{x}_i$, which it reaches as r_w tends to zero. For any given r_w, the point where the trajectory intersects the circle $\|\mathbf{x}-\mathbf{x}_i\|_2 = r_w$ is the required minimum point.

This example also gives us insight into what happens when we replace Q_i by its modified form R_i. Figure 7 shows the contours of R_i corresponding to the Q_i of Figure 6. Notice that the contours are identical in shape, in fact it is only the values attached to contours that change. The contour line through $\mathbf{x}_i$ is unique in being unchanged, those below are increased, those above are decreased. The zero contour goes through the intersection between the trajectory and the constraining circle.

Although the figures describe the two-dimensional case, note that the Levenberg-Marquardt algorithm applies to any dimensionality.

3.5. Practical Examples in 2D and 3D

As noted earlier, there are many examples where positive interpolants are important. The case we use here to illustrate the method is rainfall data from sites in New Zealand, supplied by the New Zealand National Institute of Water and Atmospheric Research [NIW03]. The data was collected at some 133 stations throughout New Zealand, and represents

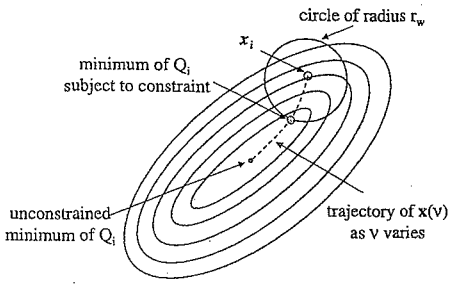

Figure 6: *Levenberg-Marquardt algorithm: The dotted line shows the trajectory of* $\mathbf{x}(\nu)$ *as* ν *varies. Each point on this trajectory is the solution of a constrained minimization of* Q_i *for some value of* r_w*. One instance of* r_w *is shown in the diagram. For* $r_w = 0$*, the minimum is at* $\mathbf{x}_i$ *and as* r_w *increases the minimum follows the trajectory shown, until eventually the unconstrained minimum is reached. Increasing* r_w *further does not alter the minimum.*

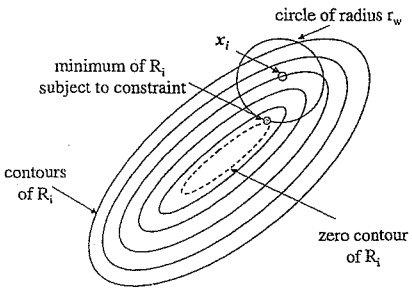

Figure 7: *Transformation from* Q_i *to* R_i*: The diagram shows the contours for* R_i *which is a scaled and shifted transform of the* Q_i *of Figure 6, the factors* α, β *chosen to achieve positivity using equation (12). Notice that the contours are unchanged in shape. Their values however are transformed: those above* f_i *are reduced in value; those below* f_i *are increased in value; the* f_i *contour line is unchanged in value. The dotted line shows the zero value contour of* R_i *: this line, which passes through the constrained minimum point of* R_i*, has had its value increased from* Q_i^{min} *to zero.*

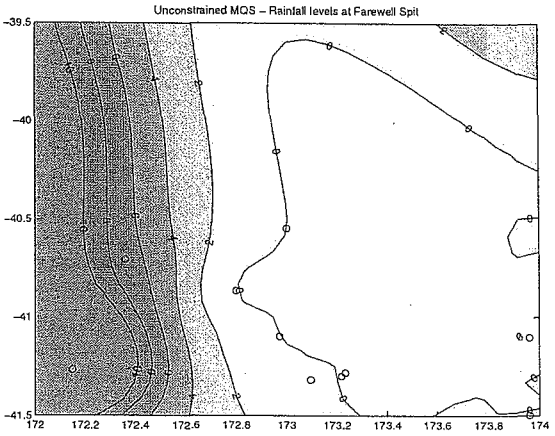

Figure 8: *Unconstrained MQS Interpolation of Rainfall Levels near Farewell Spit, New Zealand*

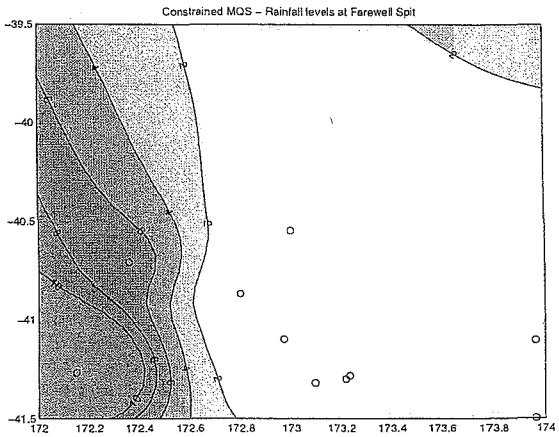

Figure 9: *Constrained MQS Interpolation of Rainfall Levels near Farewell Spit, New Zealand*

the measurement of total rainfall in millimetres, for 2nd May 2002. Figure 8 shows interpolation using the normal, unconstrained MQS method, using all the data for the interpolation but zooming in on a region at the north part of New Zealand, Farewell Spit. The contour map shows the interpolant generating negative values, which are clearly unreal. By contrast, Figure 9 shows the constrained method, with all areas showing positive rainfall values. The small circles indicate data points.

Figures 10 and 11 show similar contrasting behaviour in a region near Arthur's Pass on South Island.

These examples are both two-dimensional. In order to illustrate the method in 3D, we included the heights of the

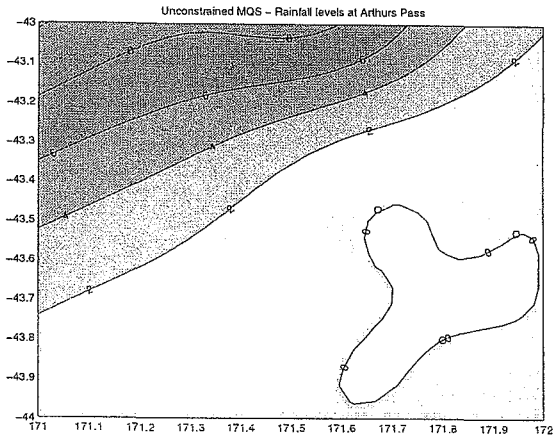

Figure 10: *Unconstrained MQS Interpolation of Rainfall Levels near Arthur's Pass, New Zealand.*

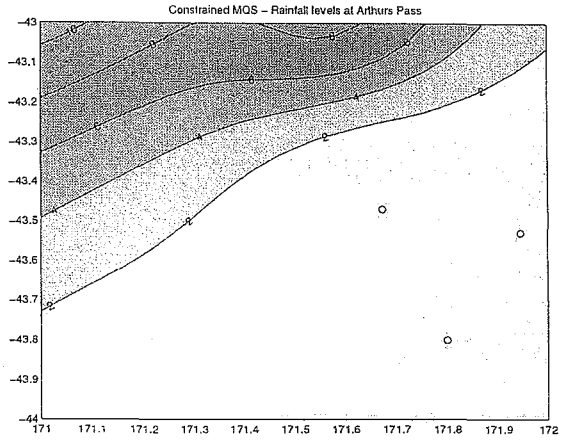

Figure 11: *Constrained MQS Interpolation of Rainfall Levels near Arthur's Pass, New Zealand*

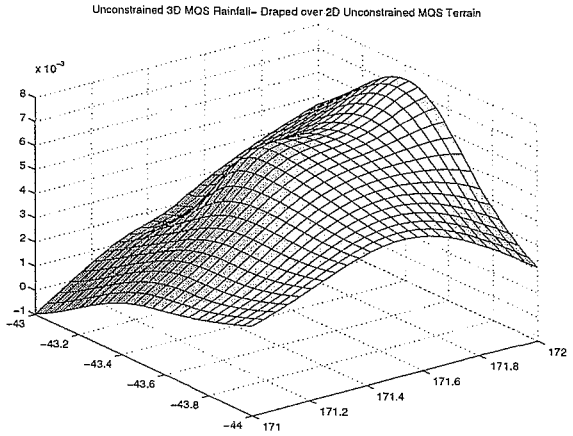

Figure 12: *Unconstrained MQS Rainfall Draped over Unconstrained MQS Terrain. Note the white area indicates negative values of rainfall.*

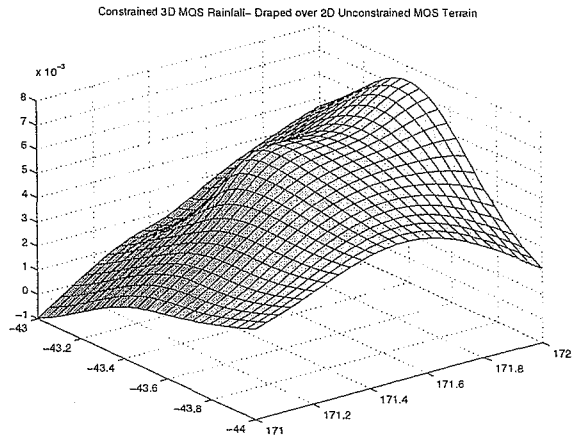

Figure 13: *Constrained MQS Rainfall Draped over Unconstrained MQS Terrain*

weather stations as well as the latitude and longitude, and created a 3D Shepard interpolant. To display the rainfall in the Arthur's Pass region, we used the unconstrained MQS to create a surface approximating the terrain, and then evaluated the 3D rainfall interpolant over this surface - first using the 3D unconstrained interpolant (shown in Figure 12) and second using the 3D constrained interpolant (shown in Figure 13). As expected, negative rainfall values occur with the unconstrained, but not the constrained version.

4. General Constraints

Once we know how to achieve a positivity constraint, we are able to apply **any** constraint on the value of the interpolant, provided the constraint is satisfied by the data. We consider a variety of cases in turn, and then illustrate with an example.

4.1. Arbitrary Lower Bound

Suppose we wish to construct an interpolant $F(\mathbf{x})$ such that

$$F(\mathbf{x}) \geq B(\mathbf{x}) \tag{22}$$

where $B(\mathbf{x})$ is any function of $\mathbf{x}$. Suppose the data values f_i satisfy the constraints

$$f_i \geq B(\mathbf{x}_i), i = 1, 2, \ldots N. \tag{23}$$

A simple approach (as suggested by Chan and Ong

[CO01] and Asim and Brodlie [AB03]) is to convert this into an equivalent positivity problem. Using the techniques just described, we construct a positive interpolant $T(\mathbf{x})$ to the positive data values $t_i = f_i - B(\mathbf{x}_i)$, and form the required interpolant $F(\mathbf{x})$ as:

$$F(\mathbf{x}) = T(\mathbf{x}) + B(\mathbf{x}) \tag{24}$$

This has important applications where one wants to construct one surface above another. This would occur for example with borehole measurements, where one wanted to show one strata of rock above another.

4.2. Arbitrary Upper Bound

A negativity condition can be achieved by changing the sign of the data values (to be positive), then interpolating with positivity constraint, and finally changing the sign of the resultant interpolant. This then extends easily to the case of an arbitrary upper bound using the approach in section 4.1.

4.3. Upper and Lower Bounds - the $[0,1]$-constraint problem

Suppose we want to constrain the interpolant to the range $[0,1]$ - for example, the data might be expressed as fractions. We apply exactly the same approach as for positivity, scaling Q_i to make its range fit within $[0,1]$, and then shifting so that interpolation is preserved. Specifically, we construct a new basis function R_i as:

$$R_i(\mathbf{x}) = \alpha Q_i(\mathbf{x}) + \beta \tag{25}$$

where we apply a scale factor $\alpha \in [0,1]$ to reduce the range of Q_i and a shift factor β to maintain interpolation. The required scale factor α is the smaller of the scale factors required to achieve lower bound of 0 and upper bound of 1, namely

$$\alpha = min\{\alpha_{lower}, \alpha_{upper}\}$$

where

$$\alpha_{lower} = \frac{f_i}{f_i - Q_i^{min}}, \alpha_{upper} = \frac{1 - f_i}{Q_i^{max} - f_i}$$

where Q_i^{max}, Q_i^{min} are the maximum and minimum of Q_i within the region it is active. Again the Levenberg-Marquardt method is used to identify the maximum and minimum values. Note that using the smaller of the two α values means that *both* constraints are satisfied: choosing a lower α than required to satisfy a constraint will simply 'flatten' the function R_i more than is actually required, and R_i will lie well within the corresponding bound. As in section 3.2, if any f_i equals 0 or 1, the corresponding $R_i(\mathbf{x})$ will be a constant (0 or 1 respectively). As before, $\beta = (1 - \alpha) f_i$.

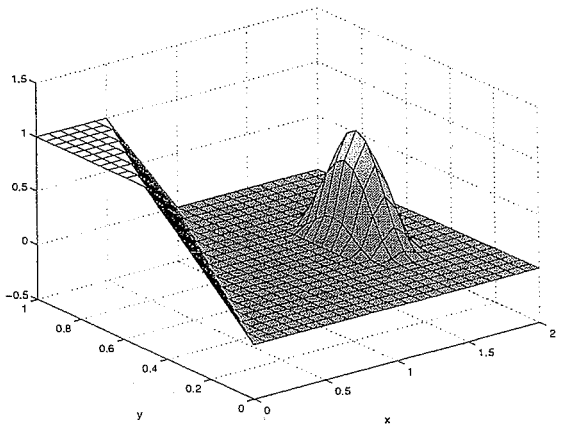

Figure 14: *Lancaster and Salkauskas function $S(x,y)$ - Surface View*

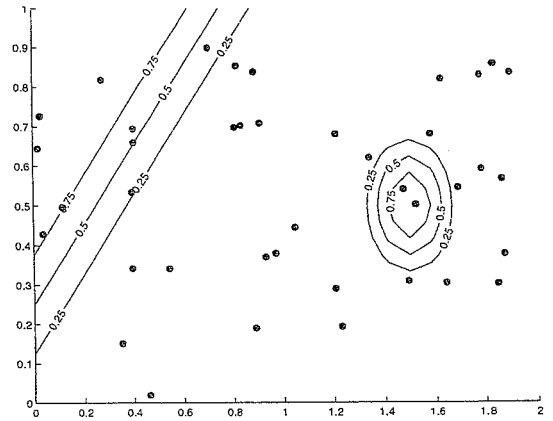

Figure 15: *Lancaster and Salkauskas function $S(x,y)$ - Contour Map*

We illustrate this technique on a test example from Lancaster and Salkauskas [LS86]. A function $S(x,y)$ is defined as:

$$S(x,y) = \begin{cases} 1.0 & \text{if } (y-x) \geq 0.5 \\ 2(y-x) & \text{if } 0.5 \geq (y-x) \geq 0.0 \\ \frac{cos(4\pi\sqrt{(x-1.5)^2+(y-0.5)^2})+1}{2} & \text{if } (x-1.5)^2 + (y-0.5)^2 \leq \frac{1}{16} \\ 0 & \text{otherwise} \end{cases}$$

The function $S(x,y)$ is shown in Figure 14. Notice that it has areas where it is exactly zero, and a peak and upper shelf where it has a value of 1.0. A contour representation is also shown, as Figure 15, where we additionally show the 40 data points that were used to construct the test data.

We construct a $[0,1]$-constraint test problem by evaluating

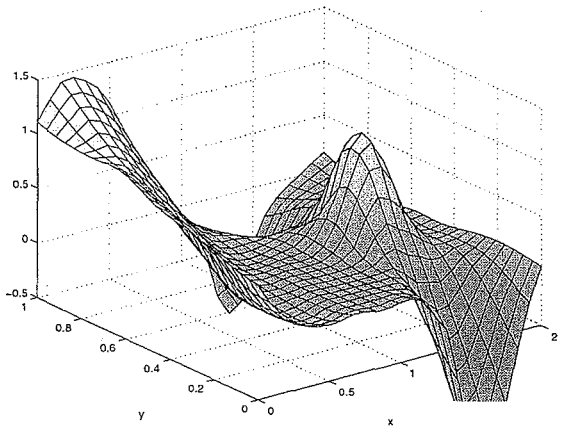

Figure 16: *Unconstrained reconstruction of Lancaster and Salkauskas function $S(x, y)$ - Surface View*

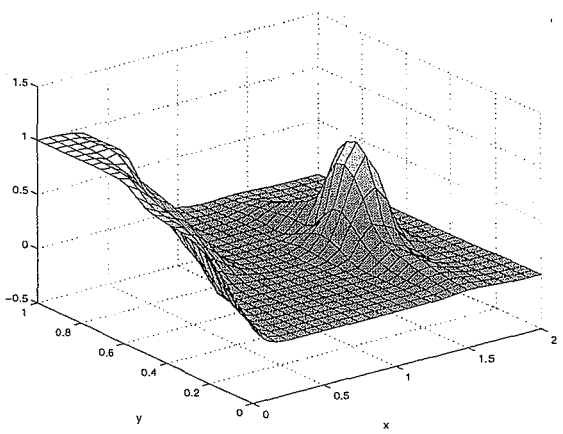

Figure 18: *Constrained reconstruction of Lancaster and Salkauskas function $S(x, y)$ - Surface View*

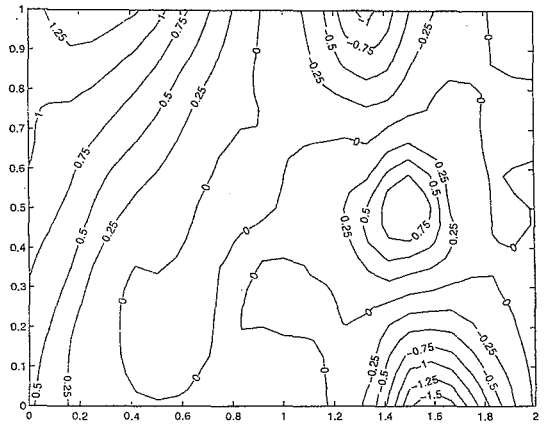

Figure 17: *Unconstrained reconstruction of Lancaster and Salkauskas function $S(x, y)$ - Contour Map*

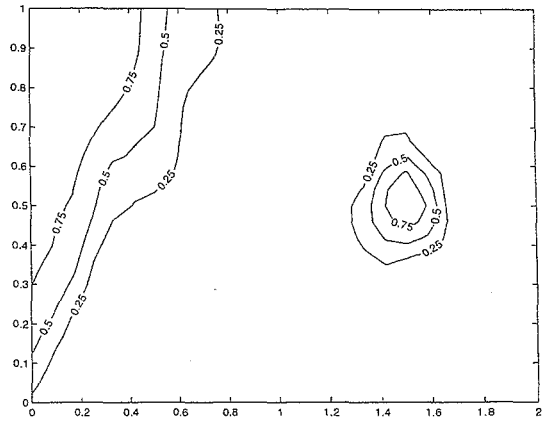

Figure 19: *Constrained reconstruction of Lancaster and Salkauskas function $S(x, y)$ - Contour Map*

$S(x, y)$ at a random set of 40 points, and requiring the interpolation scheme to reconstruct the function in such a way that it remains within the $[0, 1]$ limits. A sequence of figures shows how the new method performs. In Figure 16, we show the surface recreated by the unconstrained MQS technique. It is clear that it goes below zero and above one, and this is confirmed very clearly in the contour representation, shown in Figure 17.

By contrast, in Figure 18, we show the surface generated by the constrained method. It lies within the $[0, 1]$ limits, as is confirmed by the contour representation shown in Figure 19. Notice that the flat plane with zero height, where many of the data values equal the constraint, is reproduced quite well by the algorithm. In this area, many of the R_i functions will be constant, equal to zero, and this enables a good reconstruction of the base plane.

4.4. Arbitrary Upper and Lower Bounds

Having solved the $[0, 1]$ constraint problem, it is then easy to solve the general problem of constructing an interpolant $F(\mathbf{x})$ subject to upper and lower bounds, that is,

$$A(\mathbf{x}) \geq F(\mathbf{x}) \geq B(\mathbf{x}). \tag{26}$$

To achieve this we create a new set of data values

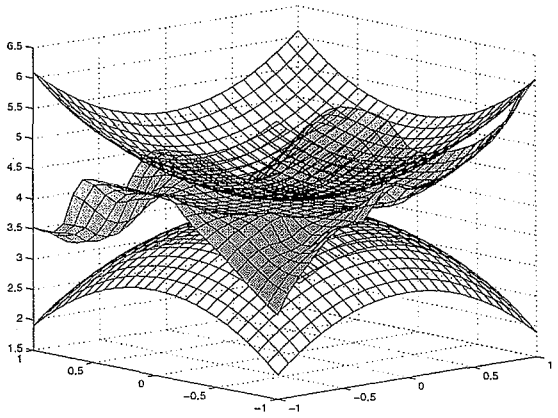

Figure 20: *Surface between Surfaces. In this figure, we show lower and upper bound quadratic surfaces as a wireframe mesh, and the constructed interpolant as a shaded surface. The interpolant is constructed from data that is randomly located, with random values in the range between the lower and upper bounds.*

$$t_i = \frac{f_i - B(\mathbf{x}_i)}{A(\mathbf{x}_i) - B(\mathbf{x}_i)} \tag{27}$$

We construct a $[0,1]$-constrained interpolant $T(\mathbf{x})$ to the data points $(\mathbf{x}_i, t_i)$, and then construct $F(\mathbf{x})$ as:

$$F(\mathbf{x}) = T(\mathbf{x})(A(\mathbf{x}) - B(\mathbf{x})) + B(\mathbf{x}) \tag{28}$$

Note that everything in this constrained section would apply to any interpolation method for which positivity, or $[0,1]$ constraint, can be proved.

To illustrate the method we show a rather contrived example where we have defined data values randomly between upper and lower quadratic 'bowls', and required an interpolant to be created that keeps within the limits imposed by the bowls. The result is shown in Figure 20.

5. Conclusions and Future Work

We have shown how the modified quadratic Shepard method for interpolation of scattered data of any dimension, can be constrained to preserve positivity of the data. This has been demonstrated in examples in 1D, 2D and 3D. The method has also been adapted in order to constrain the interpolant within $[0,1]$ limits, so that it can be used to interpolate fractional data. We have shown how the positivity and the $[0,1]$ constraints can be generalised to any arbitrary functions as lower and upper bounds.

We would like to be able to extend this work to cover other approaches to data interpolation, such as RBFs, and to data approximation where we do not require the function to pass through the data points.

An important application of the work is to the visualization of data subject to error, where we wish to show upper and lower bound visualizations that lie entirely above and entirely below the predicted 'surface'. We hope to pursue this is in a separate study.

Acknowledgements

Ken Brodlie carried out much of this work during two sabbatical visits: one to Lincoln University, New Zealand; and the other to Caltech, California, USA. He is extremely grateful to both institutions for the warmth of their hospitality and the stimulating environment that allowed this work to be carried out.

Rafiq Asim acknowledges the Government of Pakistan for providing funds to carry out his part of this research, at the University of Leeds.

References

[AB03] ASIM M., BRODLIE K.: Curve drawing subject to positivity and more general constraints. *Computers and Graphics 27* (2003), 469–485.

[Asi00] ASIM M.: *Visualization of data subject to positivity constraint.* PhD thesis, School of Computing, University of Leeds, 2000.

[Bar77] BARNHILL R.: Representation and approximation of surfaces. In *Mathematical Software III*, J.R.Rice, (Ed.). Academic Press, New York, 1977, pp. 69–120.

[BBG73] BARNHILL R., BIRKHOFF G., GORDON W.: Smooth interpolation in triangles. *Journal of Approximation Theory 8* (1973), 114–128.

[BM99] BERRY M. W., MINSER K. S.: Algorithm 798: High-dimensional interpolation using the modified Shepard method. *ACM Transactions on Mathematical Software 25*, 3 (1999), 353–366.

[CO01] CHAN E., ONG B.: Range restricted scattered data interpolation using convex combination of cubic Bezier triangles. *Journal of Computational and Applied Mathematics 136* (2001), 135–147.

[Fle87] FLETCHER R.: *Practical Methods of Optimization.* Wiley, 1987.

[FN80] FRANKE R., NIELSON G.: Smooth interpolation of large sets of scattered data. *International Journal of Numerical Methods in Engineering 15* (1980), 1691–1704.

[GW78] GORDON W. J., WIXOM J. A.: Shepard's method of metric interpolation to bivariate and multivariate interpolation. *Mathematics of Computation 32*, 141 (1978), 253–264.

[LF00] LODHA S. K., FRANKE R.: Scattered data techniques for surfaces. In *Scientific Visualization, Dagstuhl 97 Proceedings*. IEEE Computer Society Press, 2000, pp. 189–230.

[LS86] LANCASTER P., SALKAUSKAS K.: *Curve and Surface Fitting : An Introduction*. Academic Press, London, 1986.

[MS94] MULANSKY B., SCHMIDT J.: Powell-Sabin splines in range restricted interpolation of scattered data. *Computing 53* (1994), 137–154.

[NAG03] NAG: NAG Library routine E01SGF, 2003. See http://www.nag.co.uk.

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

[Nie93] NIELSON G. M.: Scattered data modelling. *IEEE Computer Graphics and its Applications* (1993), 60–70.

[NIW03] NIWA: New Zealand National Institute of Water and Atmospheric Research: Rainfall data, 2003. See http://www.niwa.co.nz.

[OW96] ONG B., WONG H.: A C^1 positivity preserving scattered data interpolation scheme. In *Advanced Topics in Multivariate Approximation*. World Scientific, Singapore, 1996, pp. 259–274.

[RB99] RENKA R. J., BROWN R.: Algorithm 792: Accuracy tests of ACM algorithms for interpolation of scattered data in the plane. *ACM Transactions on Mathematical Software 25*, 1 (1999), 78–94.

[Ren88a] RENKA R. J.: Algorithm 660: QSHEP2D: Quadratic Shepard method for bivariate interpolation of scattered data. *ACM Transactions on Mathematical Software 14*, 2 (1988), 149–150.

[Ren88b] RENKA R. J.: Multivariate interpolation of large sets of scattered data. *ACM Transactions on Mathematical Software 14*, 2 (1988), 139–148.

[Ren99a] RENKA R. J.: Algorithm 790: CSHEP2D: Cubic Shepard method for bivariate interpolation of scattered data. *ACM Transactions on Mathematical Software 25*, 1 (1999), 70–73.

[Ren99b] RENKA R. J.: Algorithm 791: TSHEP2D: Cosine series Shepard method for bivariate interpolation of scattered data. *ACM Transactions on Mathematical Software 25*, 1 (1999), 74–77.

[She68] SHEPARD D.: A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the 23rd National Conference* (New York, 1968), ACM, pp. 517–523.

[Utr85] UTRERAS F. I.: Positive thin plate splines. *J. Approximation Theory and its Applications 1*, 3 (1985), 77–108.

[XW99] XIAO Y., WOODBURY C.: Constraining global interpolation methods for sparse data volume visualization. *International Journal of Computers and Applications 21*, 2 (1999), 59–64.