Adaptive Scattered Data Interpolation with Multilevel Nonuniform B-Splines

J. Lin[†] and Z. Huang[‡]

National University of Singapore[§]

Abstract

We present an adaptive method for scattered data interpolation. The method is based on multilevel nonuniform B-splines. It makes use of a coarse-to-fine hierarchy of control lattices to generate a sequence of bicubic nonuniform B-spline functions whose sum approaches the desired interpolation function. Experimental results demonstrate that the method performs better than the method using uniform B-splines.

1. Introduction

Scattered data interpolation is the practice of fitting a smooth surface through a scattered distribution of data samples. This is often applied in science and engineering where data are measured or generated at sparse and irregular positions. Interpolation is to estimate an underlying function that can be evaluated at any position. The use of bicubic B-spline surfaces to represent functions is very popular due to the advantages, such as C^2 continuity, of such surfaces.

There has been much work in this area. However, scattered data interpolation is still a difficult and computationally expensive problem. Much of the work suffers from limitations in smoothness, time complexity or allowable data distributions [5]. Lee et al. [7] have significantly improved performance by applying multilevel B-spline interpolation.

Based on the idea of multilevel interpolation, we present a different way to adaptively interpolate the scattered data samples. It is an extension of the method by Lee et al., where uniform B-splines are used. They also propose an adaptive representation of the control lattice hierarchy but a linear array is used. We will show the use of nonuniform B-splines is a better representation in our method.

[†] Email: scip6151@mango.math.nus.edu.sg. Department of Mathematics, Faculty of Science, NUS, Singapore 119260.

[‡] Email: huangzy@comp.nus.edu.sg. Department of Computer Science, School of Computing, NUS, Singapore 119260.

[§] This work was supported by the NUS under Research Scholarship and grant RP3982704.

Published by the Eurographics Association, ISSN 1017-4656.

2. Other Relevant Work

There is much work devoted to scattered data interpolation. An excellent review can be found in [1]. The well-known methods include approaches based on Shepard's method [5], radial basis functions [3], thin plate splines [2], finite element method [8], a hierarchical B-Spline approach [4], and optimization based techniques [6].

3. The Algorithm

3.1. Overview

The algorithm is outlined as follows:

1. Initial approximation: use a coarse lattice Φ_0 to calculate a uniform B-spline surface f_0 based on point set P. We use index i to represent the current layer of subdivision. So $i = 0$ at this step. More details can be found in Lee et al. [7].
2. Derivation of the deviation: increase index i by 1, i.e., $i = i + 1$. Then, derive the deviation function Δ^i of the last approximation in layer $i - 1$.
3. Refinement of the approximation: check the error and gradient of the surface f_{i-1} on each subdividing region of Φ_{i-1}:
   a. If the test results for all current regions are below the specified threshold, sum $f_0, f_1, \cdots, f_{i-1}$ to get the final approximation function f. Stop.
   b. Otherwise, go to next step.
4. Adaptive subdivision: for each region in which the test fails, further subdivide it. A new lattice Φ_i is derived.

5. Derive an approximation to the deviation: derive a nonuniform B-spline function f_i to approximate Δ^i, go to step 2 for deriving the deviation.

3.2. Adaptive Subdivision

Consider a hierarchy of control lattices, $\Phi_0, \Phi_1, \cdots, \Phi_h$, overlaid on domain Ω. We derive Φ_0 from the initial approximation. As the lattice Φ_0 is very coarse, the initial approximation f_0 leaves large errors at the data points in P. Assume f_0 leaves a deviation $\Delta^1 z_c = z_c - f_0(x_c, y_c)$ for each point (x_c, y_c, z_c) in P. The next finer control lattice Φ_1 is then used to obtain function f_1 to approximate the difference $P_1 = \{(x_c, y_c, \Delta^1 z_c)\}$. From this step, all successively finer lattices serve to approximate and remove the residual error. In general, for level k in the hierarchy, the function f_k is derived by using control lattice Φ_k to approximate $P_k = \{(x_c, y_c, \Delta^k z_c)\}$, where $\Delta^k z_c = z_c - \sum_{i=0}^{k-1} f_i(x_c, y_c) = \Delta^{k-1} z_c - f_{k-1}(x_c, y_c)$, and $\Delta^0 z_c = z_c$. This process continues incrementally until some conditions are met, e.g., $\Delta^{h+1} z_c < \varepsilon$. Then, we get f_h to approximate $\Delta^h z_c$ over lattice Φ_h. The final approximation function f is derived by summing the f_k:

$$f = \sum_{k=0}^{h} f_k. \tag{1}$$

In this way, the scattered data $P = \{(x_c, y_c, z_c)\}$ is approximated by f, the sum of multilevel B-spline functions.

Lee et al. have proposed a sufficient condition for the function f to become an interpolant from an approximant: if no two data points share a control point in their 4×4 neighborhoods, i.e., each control point in Φ_k contains at most one data point in its *proximity data set*[7].

The sufficient condition means that a multilevel interpolation requires the control point spacing in the finest lattice Φ_h becomes sufficiently small. However, it is not necessary to use a fine uniform lattice to overlay the whole domain Ω in which not all regions have data points. Lee et al. noticed the problem and proposed an *adaptive control lattice hierarchy* to tackle the problem. Their purpose is only to save memory. Thus a linear array is used to represent a hierarchy of two dimensional lattices.

Noticing the same problem, we believe the use of nonuniform B-spline functions is a better choice:

1. Less control points are needed than the multilevel approximation using uniform B-splines.
2. More accurate and efficient:
   a. the highly varying regions of the data points are approximated by finer control lattices.
   b. the low varying regions by coarser lattices.
3. Easier to maintain the data structure than the use of a linear array for a hierarchy of two dimensional lattices.

We generalize the subdivision condition of Lee et al. who proposed that a region needs further division if it has data points. It is true that a region without data points does not need to be divided. However, a region with data points does not need subdivision either if the approximation is already accurate enough.

We measure the accuracy by checking the error at data points and gradient of the approximant for each region. Thus, a more general condition for adaptive subdivision is given: a subdivision is necessary for a region if its error or the magnitude of gradient is larger than threshold. The error of a region R is derived by:

$$err_k(R) = \max_{(x_c, y_c) \in R} |\Delta^k z_c|, \tag{2}$$

where k is the layer of subdivision.

Using error as the subdivision condition is intuitive. If a surface interpolates all data points in a region, it approximates the underlying function well in this region. Another important factor is gradient. The gradient of a function g is defined as:

$$\nabla g = \frac{\partial g}{\partial x}\vec{i} + \frac{\partial g}{\partial y}\vec{j}. \tag{3}$$

In a region R, a high magnitude of gradient means the function changes rapidly. Thus R needs a finer subdivision. In our estimation, we use the intermediate B-spline function f_k to approximate $\Delta^k z_c$. We calculate the maximal magnitude of partial derivatives $max_{(x,y)\in R}\left|\frac{\partial f_k}{\partial x}\right|$ and $max_{(x,y)\in R}\left|\frac{\partial f_k}{\partial y}\right|$. The results are compared with specified threshold to determine whether subdivisions for R are necessary in x and y directions respectively.

Before estimating the maximal magnitude of partial derivatives, we check the length ℓ and width w of R. If ℓ (w) is smaller than a specified threshold, no further subdivision is necessary in x (y) direction in R. Therefore, there is no need to estimate partial derivative with respect to x (y).

We derive the knot vectors s and t of nonuniform B-spline functions directly from the subdivision of lattice. The parametric st-space is same as the object xy-space in our implementation. The method to derive the knot vectors is consistent with the approximation principle that a highly variable region needs a finer lattice. The knot vectors are derived by

$$\begin{aligned} s &= \{x_{-3}, x_{-2}, x_{-1}, x_0, \cdots, \\ &\quad x_m, x_{m+1}, x_{m+2}, x_{m+3}\}, \\ t &= \{y_{-3}, y_{-2}, y_{-1}, y_0, \cdots, \\ &\quad y_n, y_{n+1}, y_{n+2}, y_{n+3}\}. \end{aligned} \tag{4}$$

$\{x_i | i = 0, \cdots, m\}$ and $\{y_i | i = 0, \cdots, n\}$ are the subdivisions of the current layer in x and y directions respectively. The additional knots x_{-3}, x_{-2}, and x_{-1} are chosen so that $x_0 - x_{-1} = x_{-1} - x_{-2} = x_{-2} - x_{-3} = (x_m - x_0)/m$. Similarly, x_{m+1}, x_{m+2}, and x_{m+3} are added after x_m. $y_{-3}, y_{-2}, y_{-1}, y_{n+1}, y_{n+2}$, and y_{n+3} are derived in a similar way.

Table 1: *Comparison results of Test 1.*

	nonuniform		uniform	
layer	memory	NRMS (%)	memory	NRMS (%)
1	7×7	20.564	7×7	20.564
2	11×11	10.250	11×11	10.250
3	15×15	5.034	19×19	5.334
4	19×19	4.058	35×35	4.928

Note: After four subdivisions, the memory cost of nonuniform *AMBA* method is less than half of that of uniform method. NRMS is Normalized Root Mean Square error.

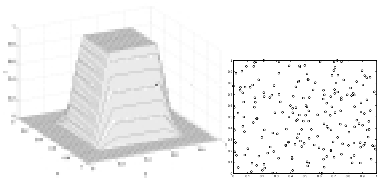

Figure 1: *Original hat function (left). Random points used in Test 1 (right).*

4. Results

In the first test we used a hat function shown in Figure 1 (left). We took 200 randomly distributed data points in its domain (Figure 1 (right)) and reconstructed the function. We compared the results between the *AMBA* (Adaptive Multilevel B-spline Approximation) method and uniform B-spline based method. The reconstructed results are shown in Figure 2 with the final subdivisions shown in Figure 3. We observe the *AMBA* method performs a little better than uniform method in accuracy but with a much less memory cost (Table 1). More subdivisions are dedicated to the highly variable regions. Notice that some regions with data points are not further subdivided such as the top of the hat.

In the second test we used another function shown in Figure 4 (left). We took 120 data points, 40 in each of the two nonzero regions and 40 randomly distributed in the entire domain. The reconstructed result is shown in Figure 4 (right).

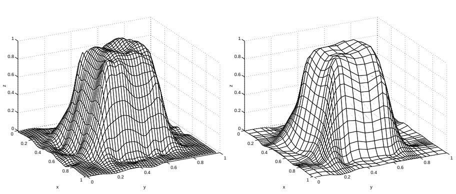

Figure 2: *Approximation to the hat function using uniform (left) and nonuniform (right) B-splines.*

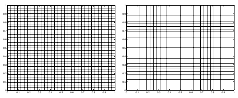

Figure 3: *The control lattices on Ω of layer 4 in examples shown in Figure 2.*

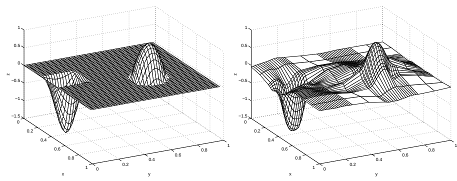

Figure 4: *Original function (left). Result using nonuniform B-splines (right).*

References

1. U. Dietz and J. Hoschek, Smooth B-Spline Surface Approximation to Scattered data. *Reverse Engineering*, J. Hoschek and W. Dankwort (eds.), pp. 143-151, 1996.
2. J. Duchon, Splines Minimizing Rotation-Invariant Semi-Norms in Sobolev Spaces. C. Chui, L. Schumaker, and J. Ward, (eds.), *Approximation Theory*, pp. 85-100, Basel, Switzerland: Birkhauser, 1975.
3. N. Dyn, Interpolation and Approximation by Radial and Related Function. C. Chui, L. Schumaker, and J. Ward, (eds.), *Approximation Theory VI*, pp. 211-234, San Diego, Calif.: Academic Press, 1989.
4. D. R. Forsey and R. H. Bartels, Surface Fitting with Hierarchical Splines. *ACM Trans. Graphics*, vol. 14, no. 2, pp. 134-161, 1995.
5. R. Franke and G. M. Nielson, Scattered Data Interpolation and Applications: A Tutorial and Survey. *Geometric Modeling: Methods and Their Application*, H. Hagen and D. Roller, (eds.), pp. 131-160, Berlin: Springer-Verlag, 1991.
6. G. Greiner and K. Hormann, Interpolating and Approximating Scattered 3D Data with Hierarchical Tensor Product B-Splines. *Surface Fitting and Multiresolution Methods*, A. Mehaute, C. Rabut, and L.L. Schumaker (eds.), pp. 163-172, 1997.
7. S. Lee, G. Wolberg, and S. Y. Shin, Scattered Data Interpolation with Multilevel B-Splines. *IEEE Trans. Visualization and Computer Graphics*, vol. 3, no. 3, pp. 228-244, 1997.
8. L. Schumaker, Fitting Surfaces to Scattered Data. C. Chui, L. Schumaker, and G. Lorentz, (eds.), *Approximation Theory*, pp. 203-268, New York: Wiley, 1976.