MULTIRESOLUTION REPRESENTATION OF URBAN TERRAIN BY L_1 SPLINES, L_2 SPLINES AND PIECEWISE PLANAR SURFACES

John E. Lavery*
Computing and Information Sciences Division
Army Research Office, Army Research Laboratory
P.O. Box 12211, Research Triangle Park, NC 27709-2211

David E. Gilsinn
National Institute of Standards and Technology
100 Bureau Drive, Stop 8910
Gaithersburg, MD 20899-8910

Abstract

Cubic L_1 and L_2 interpolating splines based on C^1 smooth, piecewise cubic Sibson elements on a tensor-product grid are investigated. Computational tests were carried out for an 800 m by 800 m area of Baltimore, Maryland represented by an 801×801 set of 1-meter-spacing (posting) data set. Interpolating splines at coarser resolutions were computed along with ℓ_1, ℓ_2, and ℓ_∞ errors relative to the 800 m by 800 m data set. Piecewise planar interpolations at the coarser resolutions were also computed along with the above errors for comparative purposes.

1. Introduction

Currently, irregular geometric surfaces and, in particular, terrain are often represented by piecewise planar surfaces on triangulated networks (often called "TINs" or "triangulated irregular networks" when the triangles are irregular in shape). Such triangulated networks are convenient, because they fit within current software and hardware constraints. However, generating an accurate, error-free surface within a triangulated network framework requires extremely fine triangulations in regions of rapid change and therefore storage and manipulation of huge amounts of data. When the triangulated networks are irregular, intricate data keeping is necessary to avoid errors such as missing triangles and triangles with mismatched edges. This results in large computing time and reduced zoom-in/out capability.

The conceptual superiority of using smooth surfaces for representation of terrain and of irregular geometric surfaces in general has long been recognized. However, previously available smooth-surface techniques such as polynomial and rational splines, radial basis functions and wavelets require too much data, too much computing time, too much human interaction and/or do not preserve shape well. Within the conventional spline framework, one can prevent extraneous, "nonphysical" oscillation only if a human operator intervenes and corrects the spline in many places or if the mesh is inordinately fine (and therefore the storage and manipulation requirements are large). Neither of these options is feasible for huge terrain data sets.

The abrupt changes in elevation that are characteristic of urban terrain are particularly challenging for both piecewise planar modeling and spline modeling. These abrupt changes in elevation are doubly challenging if one restricts the grid on which one carries out the modeling to be a grid with regularly spaced nodes. Recently, a new class of cubic "L_1" splines that perform well in preserving the shape of data sets has been developed (Lavery, 2000a, 2000b, 2001). It is the accuracy of these cubic L_1 splines with regularly spaced grids for urban terrain that we wish to investigate in this paper. Any class of surfaces, such as cubic L_1 splines, that preserve shape well is ipso facto a candidate for multiresolution representation of data and this aspect of the representation of urban data by L_1 splines will be investigated in this paper. We will compare L_1 splines with a class of conventional "L_2" splines and with piecewise planar surfaces.

2. Cubic L_1 Splines, Cubic L_2 Splines and Piecewise Planar Surfaces

The cubic splines $z(x, y)$ used in this paper consist of C^1 smooth, piecewise cubic Sibson elements (Han and Schumaker, 1997; Lavery, 2001) on regularly spaced rectangular grids with nodes $(x_i, y_j) = (c_x i, c_y j), i = 0, 1, ..., I, j = 0, 1, ..., J$, where c_x and c_y are known constants. These cubic splines, which exist on the domain $D = (x_0, x_I) \times (y_0, y_J)$, are characterized by their values $z_{ij} = z(x_i, y_j)$ and the values of their derivatives $z^x_{ij} = \frac{\partial z}{\partial x}(x_i, y_j)$ and $z^y_{ij} = \frac{\partial z}{\partial y}(x_i, y_j)$ at the nodes (x_i, y_j). At each node (x_i, y_j), the elevation z_{ij} is given. To calculate a cubic spline, one must compute the values of the derivatives z^x_{ij} and z^y_{ij}.

The z_{ij}^x and z_{ij}^y of a cubic L_1 interpolating spline are calculated by minimizing the following weighted sum of the absolute values of the second derivatives of the spline and a regularization term

$$\iint_D \left[\left|\frac{\partial^2 z}{\partial x^2}\right| + 2\left|\frac{\partial^2 z}{\partial x \partial y}\right| + \left|\frac{\partial^2 z}{\partial y^2}\right|\right] dx\, dy + \varepsilon \sum_{i=0}^{I}\sum_{j=0}^{J}\left[|z_{ij}^x| + |z_{ij}^y|\right] \tag{1}$$

over all Sibson-element surfaces z that interpolate the data z_{ij}. Here, ϵ is an a priori given "regularization parameter," that is, a small positive constant that assists the algorithm for minimizing (1) in selecting a unique solution. For further information about ϵ, see Sec. 3 of Lavery, 2001. The cubic L_1 spline defined here is the same as the cubic L_1 spline of type A_2 defined in Lavery, 2001. Minimization of (1) is carried out by discretizing the integral in (1) and using the primal affine method of Lavery 2001, Vanderbei 1989, Vanderbei, Meketon and Freedman, 1986. The integral in (1) was discretized in the following manner. Express the integral as the sum of the integrals over the rectangles $[x_i, x_{i+1}] \times [y_j, y_{j+1}]$. Divide each rectangle into N^2 equal subrectangles, where $N \geq 2$. The integral over the rectangle is approximated by 1/[2N(N - 1)] times the sum of the 2N(N - 1) values of the integrand at the midpoints of the sides of the subrectangles that are in the interior of the rectangle.

The z_{ij}^x and z_{ij}^y of a conventional cubic L_2 interpolating spline are calculated by minimizing the following weighted sum of the squares of the second derivatives of the spline and a regularization term

$$\iint_D \left[\left(\frac{\partial^2 z}{\partial x^2}\right)^2 + 4\left(\frac{\partial^2 z}{\partial x \partial y}\right)^2 + \left(\frac{\partial^2 z}{\partial y^2}\right)^2\right] dx\, dy + \varepsilon^2 \sum_{i=0}^{I}\sum_{j=0}^{J}\left[(z_{ij}^x)^2 + (z_{ij}^y)^2\right] \tag{2}$$

over all Sibson-element surfaces z that interpolate the data z_{ij}. The regularization parameter ϵ in (2) is the same as the ϵ in (1). The integral in (2) was discretized in the same manner as the integral in (1).

A piecewise planar surface z is calculated by dividing each rectangle $[x_i, x_{i+1}] \times [y_j, y_{j+1}]$ into two triangles by drawing the diagonal from the corner (x_i, y_j) to the corner (x_{i+1}, y_{j+1}) and letting z inside each triangle be the linear interpolant of the data at the three corners of the triangle.

3. Multiresolution Representation of Urban Data

Computational tests were carried out on a set of 801×801 data that consists of an 800 m by 800 m portion of a 1000×1000 set of 1-meter-posting digital elevation data for downtown Baltimore, Maryland surrounding Oriole Park at Camden Yards, home of the Baltimore Orioles baseball team. (East and west are reversed in the figures below.) The data set was obtained from the Joint Precision Strike Demonstration Project Office (JPSD PO) Rapid Terrain Visualization (RTV) ACTD. For all of these computational results, N = 3 and $\epsilon = 10^{-4}/(2N(N-1))$. In Fig. 1, we present the surface for the 800 m by 800 m, 1-meter-posting subset of the Baltimore data set mentioned above. This surface was plotted by a commercial package using bilinear elements. Figs. 2-13 below were also plotted by the commercial package using bilinear elements on 1 m by 1 m cells, the z values at the corners of which are the values of the splines and piecewise planar surfaces at these corners.

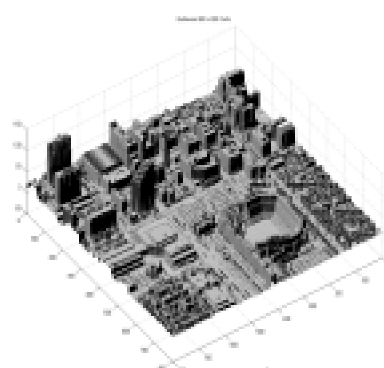

Fig. 1. Surface based on 1-meter-posting for 800 m by 800m area of Baltimore, Maryland.

In Figs. 2-5, we present for the 800 m by 800 m area of Baltimore represented in Fig. 1 the cubic L_1 interpolating splines calculated on coarse spline grids at postings (spacings) of 5 m, 10 m, 20 m and 40 m. We denote these splines by $z_{[L_1,5]}$, $z_{[L_1,10]}$, $z_{[L_1,20]}$ and $z_{[L_1,40]}$, respectively. The coarser meshes smooth out the surface. However, even at 10 m spacing major features remain recognizable. At the 20 m and 40 m spacings the features tend to smooth out in a way that makes them not readily identifiable.

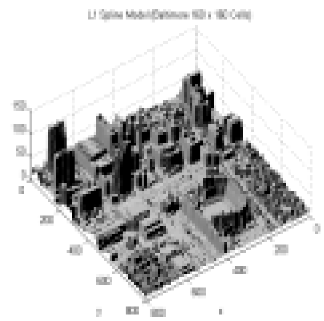

Fig. 2. L_1 spline $z_{[L_1,5]}$ based on 5-meter-spacing data for 800 m by 800 m area of Baltimore, Maryland.

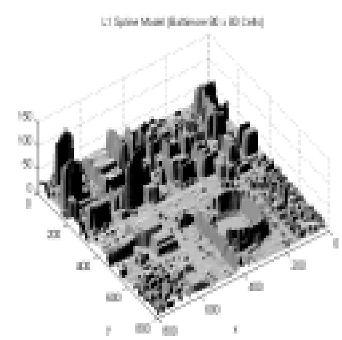

Fig. 3. L_1 spline $z_{[L_1,10]}$ based on 10-meter-spacing data for 800 m by 800 m area of Baltimore, Maryland.

In Figs. 6-9, we present for the 800 m by 800 m area of Baltimore represented in Figs. 1-5 the cubic L_2 interpolating splines calculated on coarse spline grids at postings (spacings) of 5 m, 10 m, 20 m, and 40 m. We denote these splines by $z_{[L_2,5]}$, $z_{[L_2,10]}$, $z_{[L_2,20]}$ and $z_{[L_2,40]}$, respectively. Comments concerning the smoothing obtained over the coarser meshes, similar to those above for L_1 splines, apply to the discussion about L_2 splines. As with Figures 4 and 5, Figures 8

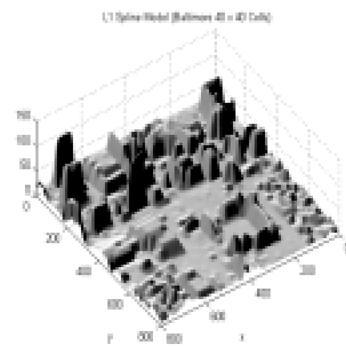

Fig. 4. L_1 spline $z_{[L_1,20]}$ based on 20-meter-spacing data for 800 m by 800 m area of Baltimore, Maryland.

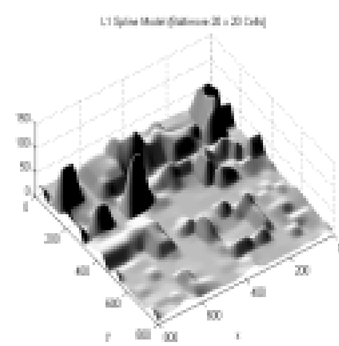

Fig. 5. L_1 spline $z_{[L_1,40]}$ based on 40-meter-spacing data for 800 m by 800 m area of Baltimore, Maryland.

and 9 show enough interpolative smoothing that many of the features begin to blend with one another.

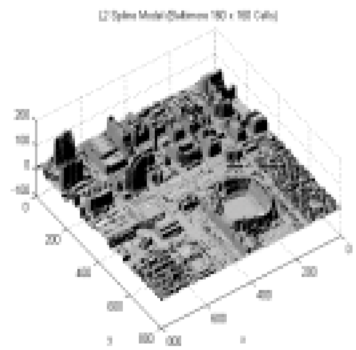

Fig. 6. L_2 spline $z_{[L_2,5]}$ based on 5-meter-spacing data for 800 m by 800 m area of Baltimore, Maryland.

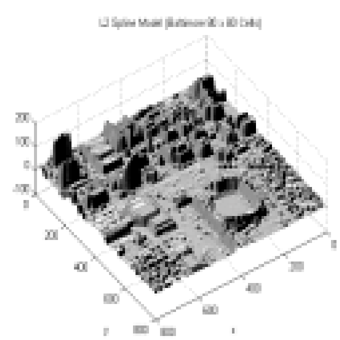

Fig. 7. L_2 spline $z_{[L_2,10]}$ based on 10-meter-spacing data for 800 m by 800 m area of Baltimore, Maryland.

In Figs. 10-13, we present for the 800 m by 800 m area of Baltimore represented in Figs. 1-5 the piecewise planar surfaces calculated on coarse spline grids at postings (spacings) of 5 m, 10 m, 20 m and 40 m. We denote these surfaces by $z_{[pp,5]}$, $z_{[pp,10]}$, $z_{[pp,20]}$ and $z_{[pp,40]}$, respectively.

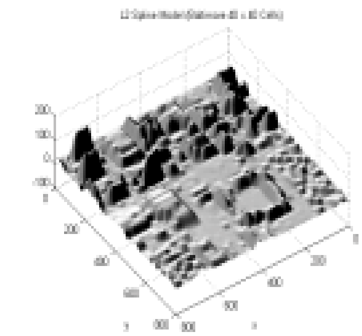

Fig. 8. L_2 spline $z_{[L_2,20]}$ based on 20-meter-spacing data for 800 m by 800 m area of Baltimore, Maryland.

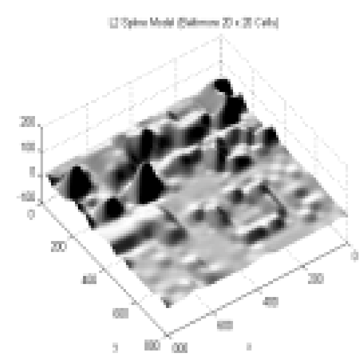

Fig. 9. L_2 spline $z_{[L_2,40]}$ based on 40-meter-spacing data for 800 m by 800 m area of Baltimore, Maryland.

In the case of the piecewise planar interpolation one begins to see the planar facets becoming more distinct as the grid spacing becomes coarser. In particular, Figures 12 and 13 show the planar facets more distinctly.

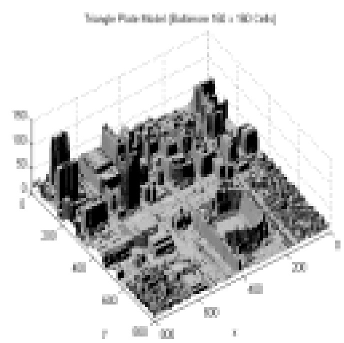

Fig. 10. Piecewise planar surface $z_{[pp,5]}$ based on 5-meter-spacing data for 800 m by 800 m area of Baltimore, Maryland.

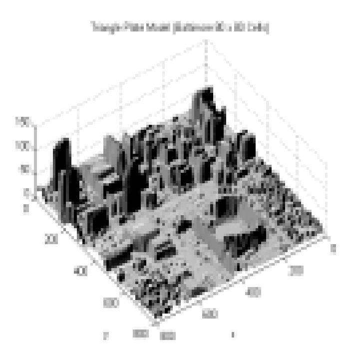

Fig. 11. Piecewise planar surface $z_{[pp,10]}$ based on 10-meter-spacing data for 800 m by 800 m area of Baltimore, Maryland.

To compare the L_1 splines, L_2 splines and piecewise planar surfaces of Figs. 2-13, we will use 1) the (normalized) ℓ_1 norm $\| \|_1$ (sum of the absolute values of the 801^2 points divided by 801^2), 2) the (normalized) ℓ_2 norm $\| \|_2$, also known as the RMS or root-mean-square norm (square root of the quotient that consists of the sum of the squares of the 801^2 points divided by 801^2) and 3) the ℓ_∞ norm $\| \|_\infty$ (maximum absolute value of the 801^2 points). In Tables 1, 2 and 3, we present the ℓ_1, ℓ_2 and ℓ_∞ norms of the error between the splines and piecewise planar surfaces of Figs. 2-13 and the original set of 801^2 data points.

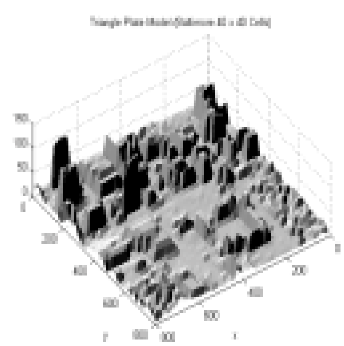

Fig. 12. Piecewise planar surface $z_{[pp,20]}$ based on 20-meter-spacing data for 800 m by 800 m area of Baltimore, Maryland.

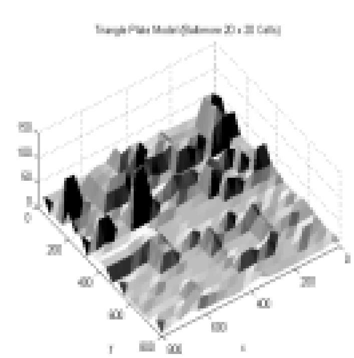

Fig. 13. Piecewise planar surface $z_{[pp,40]}$ based on 40-meter-spacing data for 800 m by 800 m area of Baltimore, Maryland.

The first row of the tables below contain the spacings, designated by "s" in meters. In the left hand column of the tables the following notation is used: $\Delta\ell_1 = \|z_{[L_1,s]} - \text{data}\|_{\ell_1}$, $\Delta\ell_2 = \|z_{[L_1,s]} - \text{data}\|_{\ell_2}$, $\Delta\ell_\infty = \|z_{[L_1,s]} - \text{data}\|_{\ell_\infty}$, where data is the original 801×801 data used to plot Fig. 1.

Table 1. Norms of differences between cubic L_1 splines on coarse grids and original data.

s (m)	5	10	20	40
$\Delta\ell_1$	1.314	2.207	3.709	5.987
$\Delta\ell_2$	3.640	5.058	7.582	11.34
$\Delta\ell_\infty$	94.50	108.4	103.1	104.2

Table 2. Norms of differences between cubic L_2 splines on coarse grids and original data.

s (m)	5	10	20	40
$\Delta\ell_1$	1.488	2.450	4.014	6.305
$\Delta\ell_2$	3.726	5.144	7.702	11.69
$\Delta\ell_\infty$	101.2	113.0	98.50	104.1

Table 3. Norms of differences between piecewise planar surfaces on coarse grids and original data.

s (m)	5	10	20	40
$\Delta\ell_1$	1.389	2.346	3.902	6.099
$\Delta\ell_2$	3.690	5.130	7.545	11.15
$\Delta\ell_\infty$	89.82	94.85	95.57	101.1

By careful visual inspection of the figures, one can see differences in the L_1, L_2 splines and the piecewise planar surfaces for the same spacing. These differences consist mainly of additional oscillation in the L_2 splines. However, one is not able to determine by visual inspection which type of interpolation, L_1 spline, L_2 spline, or piecewise planar surface is more accurate. Some information about the accuracy can be gathered from the norms of the errors in Tables 1, 2 and 3. In these tables, the ℓ_1 errors of the L_1 spline for a given spacing are always smaller than the ℓ_1 errors of the L_2 spline and the piecewise planar surface for the same spacing. In two cases, the ℓ_∞ error of the L_2 spline is smaller than the ℓ_∞ error of the corresponding L_1 spline. In the other cases, it is larger. In two cases, the ℓ_2 error and, in all cases, the ℓ_∞ error for the piecewise planar surfaces are smaller than the corresponding errors for the L_1 spline. Furthermore all of the piecewise planar surface errors are smaller than the corresponding L_2 spline errors.

Overall, this evidence indicates that L_1 splines preserve shape better for this terrain data set than do L_2 splines. With respect to the piecewise planar surface interpolation the criteria for preservation of shape depends strongly on the measure of difference between the interpolation and the original data. Piecewise planar performs better than the L_2 spline for this data set given the three measures of performance used. The comparison with the L_1 spline depends on the error measure.

4. Conclusion

L_1 splines provide shape-preserving interpolation of a surprisingly simple, piecewise polynomial nature that will result in enhanced accuracy in a multitude of applications, especially in representation of urban terrain. L_1 splines are ideal for parallel computing and computationally efficient updating of huge terrain skins because they represent local perturbations in the data by local perturbations in the spline surface.

The results in this paper indicate that L_1 splines are excellent candidates for representation of urban terrain. In this article, we have investigated the approximation properties of L_1 interpolating splines on increasingly coarse grids. On these coarse grids, we have completely ignored the presence of intermediate data. In the future, computational experiments with L_1 smoothing splines, which use all of the data, including the data between the coarse-grid nodes, will be carried out. It is expected that the performance of L_1 splines will be further enhanced by doing this.

References

Han, L. and L. L. Schumaker, "Fitting monotone surfaces to scattered data using C_1 piecewise cubics," *SIAM J. Numer. Anal.*, Vol. 34, pp. 569-585, 1997.

Lavery, J. E., "Univariate cubic L_p splines and shape-preserving, multiscale interpolation by univariate cubic L_1 splines," *Comput. Aided Geom. Design*, Vol. 17, pp. 319-336, 2000a.

Lavery, J. E., "Shape-preserving, multiscale fitting of univariate data by cubic L_1 smoothing splines," *Comput. Aided Geom. Design*, Vol. 17, pp. 715-727, 2000b.

Lavery, J. E., "Shape-preserving, multiscale interpolation by bi- and multivariate cubic L_1 splines," *Comput. Aided Geom. Design*, submitted, 2001.

Vanderbei, R. J., "Affine-scaling for linear programs with free variables," *Mathematical Programming*, Vol. 43, pp. 31-44, 1989.

Vanderbei, R. J., M. J. Meketon and B. A. Freedman, "A modification of Karmarkar's linear programming algorithm," *Algorithmica*, Vol. 1, pp. 395-407, 1986.