Progressive Triangulations for Scattered Data

Martin Bertram* Bernd Hamann* Kenneth I. Joy*

September 27, 1999

1 Introduction

Many problems in scientific visualization deal with the representation of surfaces such as shock waves, material boundaries, isosurfaces (surfaces of equal value in 3D scalar fields), and terrain models. In many cases, these surfaces are represented by a set of sampled points that can be projected locally on a plane. For visualization applications, however, a triangulated surface representation is required. The number of triangles has a major impact on the efficiency of the visualization process.

We present an efficient algorithm to obtain triangulated graph surfaces for sets of scattered points $(x_i\ y_i)^T$, $i = 1...n$, with associated function values f_i at multiple levels of detail. The approximation error measured in z-direction satisfies a prescribed error bound for each level of detail.

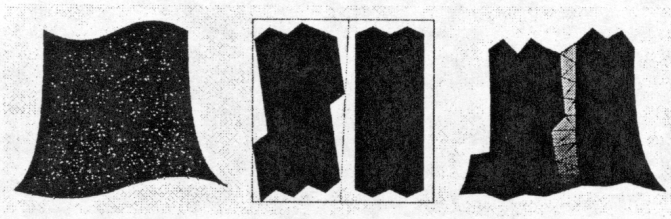

Figure 1: Left: scattered points, sampled from a smooth graph surface; middle: clusters with locally optimal triangulations; right: final triangulation.

Our algorithm consists of three major steps: (i) subdividing the given discrete data set into clusters such that each cluster can be approximated by a quadratic polynomial within a prescribed tolerance; (ii) optimally triangulating each quadratic polynomial; and (iii) "stitching" the triangulations of all graph surfaces together. Figure 1 illustrates the overall construction approach. For step (ii) we use the optimal triangulation approach in [5]. Our algorithm was published in [1]. For related methods concerning more general triangulated surfaces with non-planar topology we refer to [3, 4].

*Center for Image Processing and Integrated Computing (CIPIC), Department of Computer Science, University of California, Davis, CA 95616-8562, USA. E-mail: {bertram, hamann, joy}@cs.ucdavis.edu

2 Adaptive Clustering

To approximate the scattered data locally by quadratic polynomials of the form

$$f(x,y) = c_{20}x^2 + c_{11}xy + c_{02}y^2 + c_{10}x + c_{01}y + c_{00},$$

we use *least squares fitting*, see [2]. A binary space partition tree (BSP tree) is constructed in the xy-plane by recursive subdivision of cluster regions. Subdivision terminates when for all cluster regions a quadratic polynomial is found that approximates the data within a prescribed error bound.

3 Triangulating Quadratic Polynomials

For each cluste region the corresponding quadratic polynomial is approximated by a triangulated surface following the approach in [5]. The number of triangles is minimized subject to a prescribed approximation tolerance with respect to the polynomial.

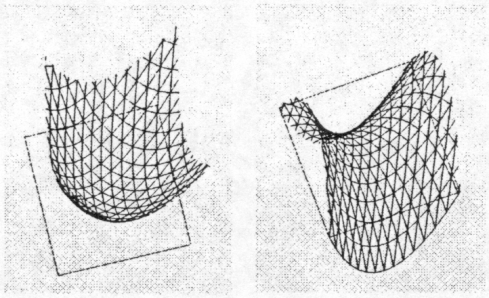

Figure 2: Optimal triangulations for elliptic and hyperbolic cases.

From the *principal axes transformation*, see [2], we obtain the *principal curvatures*, i.e., the minimal and maximal second derivatives of the polynomial and the corresponding directions in the xy-plane. After changing the basis to these directions and neglecting all linear terms, the polynomial is transformed to

$$f(x,y) = x^2 + y^2 \quad \text{(elliptic case) or}$$

$$f(x,y) = x^2 - y^2 \quad \text{(hyperbolic case).}$$

In both cases a regular, optimal triangulation exists, shown in Figure 2, that is transformed back into the canonic basis. All triangles that lie entirely in the cluster region are drawn. In cases of nearly flat or parabolic surfaces a special treatment is required.

4 Merging Triangulations

Finally, the individual triangulations are stitched together by triangulating the gaps along the cluster boundaries. This is done in the reverse order of cluster subdivision that is stored in the BSP tree. Only two triangulations are stitched together at a time, see Figure 3.

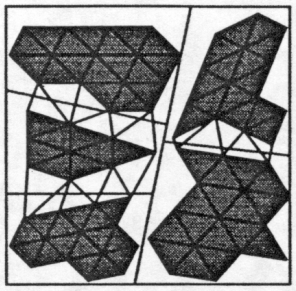

Figure 3: Triangulations are stitched together along a straight line.

5 Numerical Examples

We have applied our algorithm to approximate the terrain model "Crater Lake" that consists of 159,272 sampled points. Numerical results for different error bounds are shown in Table 1 and Figure 4.

Error [%]	No. Clusters	No. Triangles	Comp. Time [sec] [1] Clustering	Triangles
10.0	114	948	6.6	0.168
5.0	400	3541	18.0	0.371
3.0	790	7256	32.5	0.680

Table 1: Numerical results for "Crater Lake" terrain model.

6 Conclusions and Future Work

We have introduced a progressive method for the construction of data-dependent triangulations for scattered bivariate data. The computational cost for generating the BSP tree dominates the cost for obtaining a triangulation. However, the BSP tree needs to be established only once for all levels of detail.

We plan to generalize our approach to approximate scalar fields defined by scattered points in 3D space with associated function values. The goal is to obtain progressive tetrahedral meshes that approximate a scalar field within user-defined error bounds.

[1] The computation times are based on our implementation on a SGI O^2 workstation with a 180 MHz R5000SC processor.

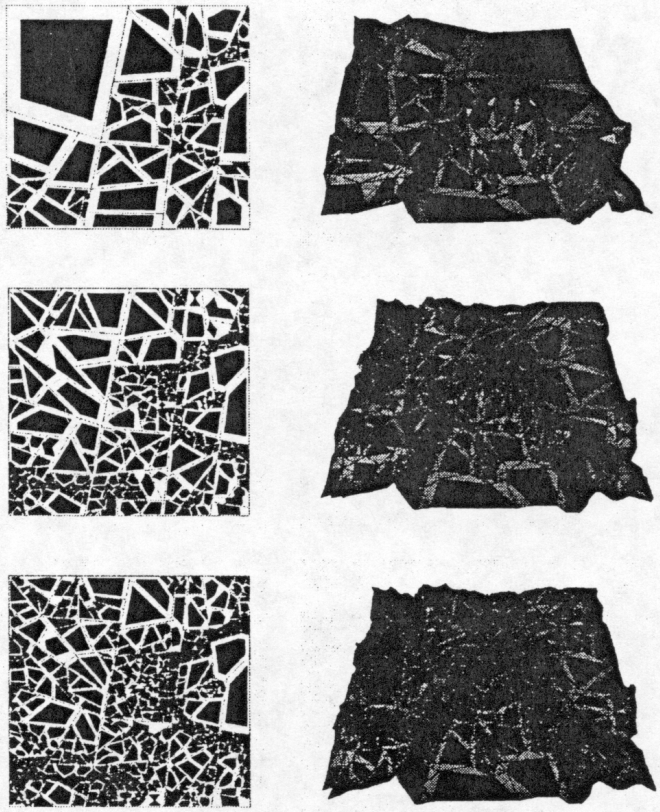

Figure 4: Triangulations for dataset "Crater Lake". Left: optimal cluster triangulations in xy-plane at three different resolutions, right: corresponding final triangulations in xyz-space.

References

[1] M. Bertram, J. C. Barnes, B. Hamann, K. I. Joy, H. Pottmann, and D. Wushour, *Piecewise optimal triangulations of Scattered Data in the Plane*, submitted to Computer Aided Geometric Design, 1999.

[2] W. Boehm and H. Prautzsch, *Geometric Concepts for Geometric Design*, A K Peters, Ltd., Wellesley, Massachusetts, 1994.

[3] B. Heckel, A. Uva, and B. Hamann, *Clustering-based generation of hierarchical surface models*, Late Breaking Hot Topics Proceedings, Visualization '98 IEEE Computer Society Press, 1998, pp. 41–44.

[4] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, *Surface Reconstruction from Unorganized Points*, SIGGRAPH '92, ACM, 1992, pp. 71–78.

[5] H. Pottmann, R. Krasauskas, B. Hamann, K. I. Joy and W. Seibold, *On piecewise linear approximation of quadratic functions*, submitted to Computer Aided Geometric Design, 1999.