ORNL/TM-13086

Computer Science and Mathematics Division

Mathematical Sciences Section

COEFFICIENT ADAPTIVE TRIANGULATION FOR STRONGLY ANISOTROPIC PROBLEMS

E.F. D'Azevedo
C.H. Romine
J.M. Donato

Mathematical Sciences Section
Oak Ridge National Laboratory
P.O. Box 2008, Bldg. 6012
Oak Ridge, TN 37831-6367

Date Published: November 1997

Research supported by the Applied Mathematical Sciences subprogram of the Office of Energy Research, U.S. Department of Energy

Prepared by the
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831
managed by
Lockheed Martin Energy Research Corp.
for the
U.S. DEPARTMENT OF ENERGY
under Contract No. DE-AC05-96OR22464

Contents

List of Figures

COEFFICIENT ADAPTIVE TRIANGULATION FOR STRONGLY ANISOTROPIC PROBLEMS

E.F. D'Azevedo
C.H. Romine
J.M. Donato

Abstract

Second order elliptic partial differential equations arise in many important applications, including flow through porous media, heat conduction, and the distribution of electrical or magnetic potential. The prototype is the Laplace problem, which in discrete form produces a coefficient matrix that is relatively easy to solve in a regular domain. However, the presence of anisotropy produces a matrix whose condition number is increased, making the resulting linear system more difficult to solve.

In this work, we take the anisotropy into account in the discretization by mapping each anisotropic region into a "stretched" coordinate space in which the anisotropy is removed. The region is then uniformly triangulated, and the resulting triangulation mapped back to the original space. The effect is to generate long slender triangles that are oriented in the direction of "preferred flow." Slender triangles are generally regarded as numerically undesirable since they tend to cause poor conditioning; however, our triangulation has the effect of producing effective isotropy, thus improving the condition number of the resulting coefficient matrix.

1. Introduction

Second order elliptic partial differential equations arise in many important applications, including flow through porous media, heat conduction, and the distribution of electrical potential. A simple prototype is the piecewise constant coefficient equation,

$$\frac{\partial}{\partial x}\left(K_x \frac{\partial P}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y \frac{\partial P}{\partial y}\right) = -q \tag{1}$$

which reduces to the Laplace problem for $K_x = K_y = 1$. The discretization of the Laplace problem over a regular grid produces a coefficient matrix that is relatively easy to solve. However, the presence of strong anisotropy ($K_x \gg K_y$) produces a poorly conditioned matrix, making the resulting linear system more difficult to solve.

In this report, we take the anisotropy into account in the discretization by mapping each anisotropic region into a "stretched" coordinate space in which the anisotropy is removed. The region is then uniformly triangulated, and the resulting triangulation mapped back to the original space. The effect is to generate long slender triangles that are oriented in the direction of "preferred flow." Slender triangles are generally regarded as numerically undesirable; however, our triangulation has the effect of producing effective isotropy, thus producing a coefficient matrix with a smaller condition number. Furthermore, our initial experiments suggest such "coefficient-adaptive" triangulation suffers no degradation in approximation accuracy.

The idea of using special approximation basis functions that depend on the rough coefficients has also been proposed by Falk and Osborn [3] in the analysis of mixed finite element methods for problems with rough coefficients. A technique of using Delaunay triangulation under an anisotropic transformation has been examined by Letniowski [9] and Forsyth [4]. The idea in their work is to ensure that the coefficient matrix resulting from the standard Galerkin finite element approximation of the second-order diffusion operator is an M-matrix. Our work differs in that our focus is on improving the conditioning of the linear system.

In Section 2, we describe the motivation for coefficient-adaptive mesh generation, using a simple example for illustration. In Section 3, we describe in more detail the approach that we have taken to discretizing anisotropic problems. Section 4 presents our sequence of test problems, based on Stone's third problem [10]. Section 5 presents the results of our empirical studies, comparing our coefficient-adaptive discretization to the standard 5-point Laplacian discretization on the test problems. Finally, in Section 6, we summarize our conclusions and discuss how the results can be expanded.

2. Fourier Analysis of Anisotropic Problem

In this section, we use a Fourier analysis technique to analyze the condition of the coefficient matrix arising from the 5-point finite difference discretization of a model anisotropic problem. The analysis we present here follows the technique described by Chan and Elman [1] and Donato and Chan [2].

The problem we analyze is

$$\frac{\partial}{\partial x}\left(K_x \frac{\partial P}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y \frac{\partial P}{\partial y}\right) = -q$$

on the unit square with Neumann boundary conditions, where K_x and K_y are constant. Clearly, the differential equation with pure Neumann boundary conditions is not well-posed, and the resulting coefficient matrix will be rank-deficient. Rather than imposing an additional constraint, we define the *modified condition number* of the coefficient matrix to be

$$\tilde{\kappa} = \frac{\lambda_{max}}{\tilde{\lambda}_{min}}, \tag{2}$$

where $\tilde{\lambda}_{min}$ is the smallest *nonzero* eigenvalue of the matrix.

Let h_x, h_y be the grid spacing in x and y, so that $n_x = h_x^{-1}$ and $n_y = h_y^{-1}$ are the number of grid points in x and y, respectively. We wish to compute the modified condition number as a function of K_x, K_y, h_x and h_y. More accurately, we will compute the modified condition number as a function of the *degree of anisotropy* $\alpha = K_y/K_x$ and the *grid aspect ratio* $\rho = h_y/h_x$. The corresponding finite difference equations are:

$$K_x \frac{-P_{i-1,j} + 2P_{ij} - P_{i+1,j}}{{h_x}^2} + K_y \frac{-P_{i,j-1} + 2P_{ij} - P_{i,j+1}}{{h_y}^2} = q_{ij}. \tag{3}$$

On the boundary of the square, the Neumann boundary conditions can be imposed using centered differences, yielding

$$P_{i,j+1} = P_{i,j-1} \quad \text{for } i = 0 \text{ and } i = n_x;$$

$$P_{i+1,j} = P_{i-1,j} \quad \text{for } j = 0 \text{ and } j = n_y.$$

If we scale (3) symmetrically so that the coefficient of P_{ij} is 1, we have

$$P_{ij} + b(P_{i-1,j} + P_{i+1,j}) + c(P_{i,j-1} + P_{i,j+1}) = \tilde{q}_{ij} \tag{4}$$

where $b = -\frac{1}{2}(1 + \alpha/\rho^2)^{-1}$ and $c = -\frac{1}{2}(1 + \rho^2/\alpha)^{-1} = -\frac{1}{2} - b$. A straightforward analysis of

the spectrum of the difference operator leads to the following expression for the eigenvalues:

$$\lambda_{ij} = 1 + 2b\cos\theta_i + 2c\cos\phi_j, \tag{5}$$

where $\theta_i = i\pi h_x$ for $i = 0, \ldots, n_x - 1$ and $\phi_j = j\pi h_y$ for $j = 0, \ldots, n_y - 1$.

The pure Neumann boundary conditions mean that $(1, 1, \ldots, 1)^T$ is an eigenvector, with corresponding eigenvalue $\lambda_{00} = 0$. Furthermore, since b and c are both negative, λ_{max} occurs at $i = n_x - 1$ and $j = n_y - 1$, yielding $\lambda_{max} = 2$. By inspection, $\tilde{\lambda}_{min} = \min\{\lambda_{10}, \lambda_{01}\}$. To determine which of these eigenvalues is smaller, we use the Taylor approximation $\cos\theta = 1 - \theta^2/2 + O(\theta^4)$. This yields

$$\lambda_{10} = \tfrac{1}{2}(1 + \alpha/\rho^2)^{-1}(\pi^2 h_x^2) + O(h_x^4)$$

$$\lambda_{01} = \tfrac{1}{2}(1 + \rho^2/\alpha)^{-1}(\pi^2 h_y^2) + O(h_y^4).$$

If we ignore the fourth-order terms, simple algebraic manipulation verifies that $\lambda_{01} = \alpha\lambda_{10}$. Hence, if $K_x > K_y$ so that $\alpha < 1$, then $\tilde{\lambda}_{min} = \lambda_{01}$. Conversely, if $K_x < K_y$, $\tilde{\lambda}_{min} = \lambda_{10}$.

We now have an expression for the modified condition number, $\tilde{\kappa} = \lambda_{max}/\tilde{\lambda}_{min}$ as a function of α and ρ. If $K_y > K_x$, then

$$\tilde{\kappa}(\alpha, \rho) \approx \frac{4(1 + \alpha/\rho^2)}{\pi^2 h_x^2}.$$

If $K_y < K_x$, then

$$\tilde{\kappa}(\alpha, \rho) \approx \frac{4(1 + \rho^2/\alpha)}{\pi^2 h_y^2}.$$

Now we have the machinery in place to answer our primary question concerning the construction of a discretization grid: For a given degree of anisotropy (fixed α) and a given number of unknowns N, what is the grid aspect ratio (ρ) that minimizes the (modified) condition number of the resulting coefficient matrix? If we assume that $\alpha > 1$, then we seek to minimize

$$\tilde{\kappa} \approx \frac{4(1 + \alpha/\rho^2)}{\pi^2 h_x^2}.$$

Using the fact that $N = n_x n_y = h_x^{-1} h_y^{-1}$ we can rewrite this as

$$\tilde{\kappa} = 4\pi^2 N(\rho + \alpha/\rho)$$

for which the minimum occurs at $\rho = \sqrt{\alpha}$. The same result is obtained for $\alpha < 1$. We use this result in defining the mapping described in Section 3. The smallest condition number is obtained when $\alpha = 1$, indicating that the isotropic case produces the best conditioned matrix.

To summarize, for $\rho = \sqrt{\alpha}$, we have

$$\tilde{\kappa} \approx \frac{8N}{\pi^2}(\max\{K_y/K_x, K_x/K_y\})^{1/2}.$$

We used a series of MATLAB tests to verify the conclusions of the analysis described above. Table 1 is in three sections. The first four rows display the close agreement between the condition number $\tilde{\kappa}_{FDM}$ of the coefficient matrix and the theoretical condition number $\tilde{\kappa}$, both computed by MATLAB. Two digits of agreement are obtained even for very small problems.

The second section of the table verifies that as the degree of anisotropy (α) is multiplied (or divided) by 100, the optimal condition number $\tilde{\kappa}$ increases by $\sqrt{100} = 10$, as predicted. The second and third sections of the table together illustrate that choosing the correct aspect ratio ρ can significantly improve the condition number of the resulting coefficient matrix.

α	n_x	n_y	$n_x * n_y$	# unknowns	$\tilde{\kappa}$	$\tilde{\kappa}_{FDM}$
1	12	12	144	169	1.1672e+02	1.1739e+02
2	14	10	140	165	1.6049e+02	1.6117e+02
4	16	8	128	153	2.0751e+02	2.0817e+02
10	21	6	126	154	3.2463e+02	3.2524e+02
5	5,981	2,674	15,993,194	16,001,850	2.8987e+07	–
500	18,914	845	15,982,330	16,002,090	2.8968e+08	–
1/500	845	18,914	15,982,330	16,002,090	2.8968e+08	–
5	4,000	4,000	16,000,000	16,008,001	3.8907e+07	–
500	4,000	4,000	16,000,000	16,008,001	3.2488e+09	–
1/500	4,000	4,000	16,000,000	16,008,001	3.2488e+09	–

Table 1: MATLAB results verifying the analysis.

3. Coefficient Adaptive Triangulation

We modified an existing triangular mesh generation package, GEOMPACK [7], to generate coefficient adaptive triangulations. GEOMPACK is a mathematical software package written in Fortran 77 for the generation of convex polygon decompositions and triangular meshes in two-dimensional polygonal regions.

GEOMPACK generates a triangular mesh by first decomposing the polygonal domain into simpler convex polygons [6]. The decomposition can be further controlled by equidistribution of a user supplied density function. We chose to use $(K_x K_y)^{1/2}$ as the density function, to allocate proportionally *more* triangles in regions where K_x or K_y is small. GEOMPACK then generates a uniform triangulation within the interior of each convex subdomain [5]. A final step handles the mesh connection between neighboring subdomains [8] to generate a Delau-

nay triangulation.

GEOMPACK was modified to perform a rescaling by $K_x^{-1/2}$ and $K_y^{-1/2}$ in the x and y directions before the generation of a uniform mesh within each convex subdomain. The node coordinates of this triangulation are then mapped back into the original space.

4. Stone's Problem

We tested the technique for coefficient adaptive triangulation on a variant of Stone's third problem [10] in solving

$$\frac{\partial}{\partial x}\left(K_x \frac{\partial P}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y \frac{\partial P}{\partial y}\right) = -q$$

on the $[0, 30] \times [0, 30]$ with Neumann boundary conditions. Locations and strengths of point sources and sinks are

$$\begin{aligned} q_1(3, 3) = 1.0, \quad q_2(3, 37) = 0.5, \quad q_3(23, 4) = 0.6, \\ q_4(14, 15) = -1.83, \quad q5(27, 27) = -0.27\ . \end{aligned} \tag{6}$$

The distribution of material properties, K_x, K_y were (see Figure 1)

$$(K_x, K_y) = \begin{cases} (1, \gamma) & \text{if } (x_i, y_j) \in B, \quad 14 \le i \le 30,\ 0 \le j \le 16, \\ (\gamma, 1) & \text{if } (x_i, y_j) \in C, \quad 5 \le i \le 12,\ 5 \le j \le 12, \\ (0, 0) & \text{if } (x_i, y_j) \in D, \quad 12 \le i \le 19,\ 21 \le j \le 28, \\ (1, 1) & \text{if } (x_i, y_j) \in A. \end{cases} \tag{7}$$

A 31×31 regular grid with $\gamma = 100$ was used in the original problem. Note that region D with $K_x = K_y = 0$ was modeled as a hole in the plate. We chose a consistent discretization scheme based on linear triangular elements in a Galerkin Finite Element formulation. Even on a regular rectangular grid, we generate a triangulation by consistently splitting each rectangle into two right-angle triangles. For the regular rectangular grid, this yields the identical standard 5-point finite difference stencil, with the exception of the nodes at interfaces between different materials.

5. Results

We chose MATLAB to perform the numerical experiments and visualization. We used linear triangular elements in a Galerkin Finite Element formulation to perform the matrix assembly. Initially, we imposed a Dirichlet condition at the origin $(0, 0)$ to avoid exact rank

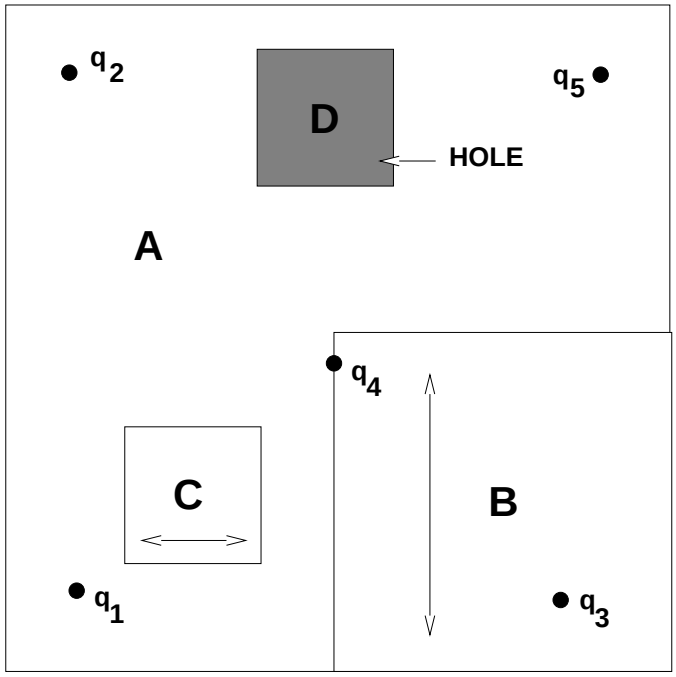

Figure 1: Stone's third problem.

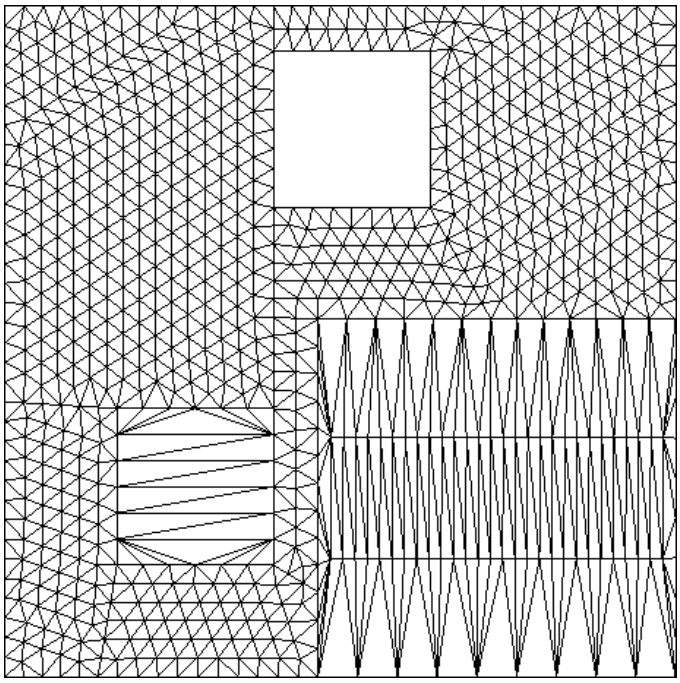

Figure 2: Coefficient adaptive triangulation of Stone's Third Problem, $\gamma = 100$.

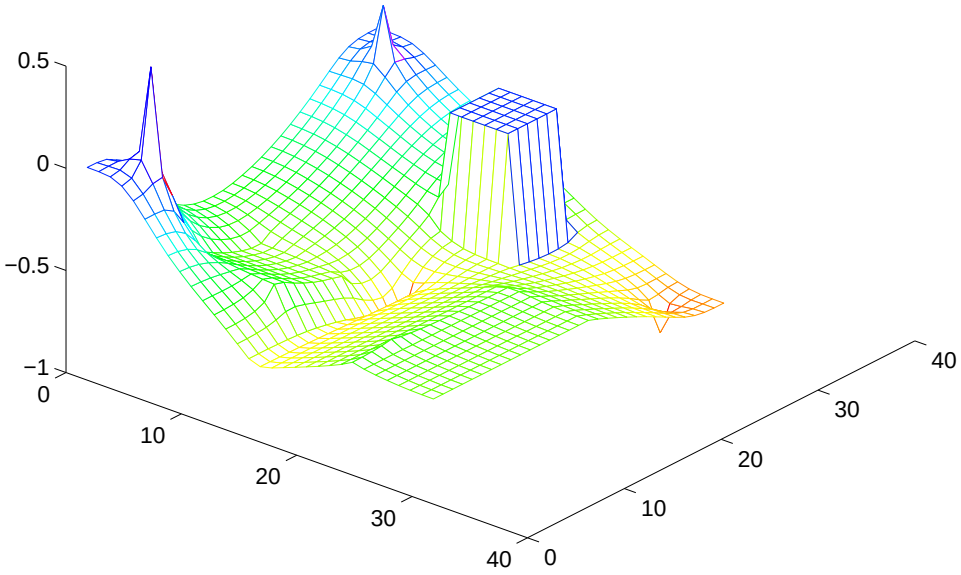

Figure 3: Accurate solution on 121×121 grid with $\gamma = 100$.

deficiency. However, subsequent testing revealed that the CG iteration applied to the original semi-definite problem converged more rapidly than CG iteration on the problem with a Dirichlet condition imposed. We have reported the times for both problems in the interest of completeness. To simplify visualization, all solutions were interpolated and compared on a 31×31 regular rectangular mesh. Since region D is modeled as a hole, nodes within region D were set to zero for simplicity. We used the solution obtained from a 121×121 grid as an accurate solution. Note that the sources and sinks introduce point singularities; thus in comparing solution accuracies we ignore the errors within one mesh block of the point singularities.

Figure 2 displays the coefficient adaptive unstructured triangular mesh for $\gamma = 100$. Note the orientation of slender triangles within regions B and C. Figure 3 displays the accurate solution obtained with the 121×121 mesh for $\gamma = 100$. Notice the flat solution profiles in regions C and D (see also Figure 1).

The linear systems resulting from the discretization were first scaled to unit diagonal, before they were solved using the conjugate gradient (CG) method with no further preconditioning. We used a relative reduction in the initial residual as our termination criterion:

$$\|r_k\|_2 \leq 10^{-10} \|r_0\|_2 \,,$$

where r_k is the residual vector on the k-th iteration. Tables 2 and 3 show the number of CG iterations, maximum discretization error and condition number.[1]

	Regular Grid			Adaptive Triangulation				
γ	iters	err[a]	$\tilde{\kappa}$	iters	err[a]	$\tilde{\kappa}$	nelts[b]	N[c]
1	180	3.7e-2	8.9e+2	170	8.6e-3	8.5e+2	1653	900
10	210	6.7e-2	1.8e+3	168	4.9e-2	9.3e+2	1635	894
100	367	8.4e-2	1.3e+4	202	3.2e-2	1.7e+3	1589	870
1000	629	7.7e-2	1.2e+5	278	1.7e-2	8.9e+3	1579	865

Table 2: Diagonally Scaled PCG on Stone's Third Problem, No Dirichlet Boundary Condition Imposed

	Regular Grid		Adaptive Triangulation			
γ	iters	κ	iters	κ	nelts[b]	N[c]
1	207	1.9e+4	195	1.8e+4	1653	900
10	243	5.0e+4	190	2.1e+4	1635	894
100	418	3.6e+5	227	4.5e+4	1589	870
1000	665	3.4e+6	305	2.2e+5	1579	865

Table 3: Diagonally Scaled PCG on Stone's Third Problem, Dirichlet Boundary Condition Imposed

[a] Maximum discretization error, estimated by comparing a highly accurate solution of the resulting linear system to the 'exact' solution.

[b] Number of triangular elements in the discretization.

[c] Total number of unknowns in the resulting linear system.

We also tested the problems using CG with SSOR preconditioning. If matrix $A = I - (L + L^T)$ represents the global assembled matrix with unit diagonal, and L is strictly lower triangular, then the SSOR factorization used is

$$Q \quad = \quad (I - \omega L)(I - \omega L^T) \, . \tag{8}$$

A few preliminary runs on various problem show the optimal ω to be between 1.4 and 1.6. A value of $\omega = 1.5$ was consistently used for all runs for simplicity. All matrices were also consistently reordered using the bandwidth reducing RCM (Reverse Cuthill-McKee) ordering to minimize the effect of matrix ordering on convergence. The results are summarized in Table 4.

The results from Tables 2 and 4 show coefficient adaptive triangulations generate better conditioned matrices with no loss of approximation accuracy even using slender triangles.

[1] For the semidefinite case, we report the 'modified' condition number (2).

	Regular Grid	Adaptive Triangulation
γ	iters	iters
1	47	51
10	56	53
100	104	70
1000	207	111

Table 4: Results of SSOR PCG on Stone's Third Problem

6. Summary

We have explored the use of coefficient adaptive mesh generation techniques on strongly anisotropic problems. The initial results on Stone's problem suggest there is no loss in approximation accuracy even with slender triangles and the resulting discretization produces better conditioned matrices.

More extensive testing with more realistic problems is required. It is straightforward to extend this approach to generating tetrahedral meshes in three-dimensions.

Acknowledgements

The authors thank Professor Barry Joe for providing GEOMPACK. We also thank Pat Worley and Steve Lee for their insightful comments, which improved the paper.

7. References

[1] T. CHAN AND H. ELMAN, *Fourier analysis of iterative methods for elliptic problems*, SIAM Review, 31 (1989), pp. 20–49.

[2] J. DONATO AND T. CHAN, *Fourier analysis of incomplete factorization preconditioners for three-dimensional anisotropic problems*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 319–338.

[3] R. FALK AND J. E. OSBORN, *Remarks on mixed finite element methods for problems with rough coefficients*, Math. Comp., 62 (1994), pp. 1–19.

[4] P. A. FORSYTH, *A control volume finite element approach to NAPL groundwater contamination*, SIAM J. Sci. Statist. Comput., 12 (1991), pp. 1029–1057.

[5] B. JOE, *Delaunay triangular meshes in convex polygons*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 514–539.

[6] ——, *On the correctness of a linear-time visibility polygon algorithm*, Intern. J. Computer Math, 32 (1990), pp. 155–172.

[7] ——, *GEOMPACK – A software package for the generation of meshes using geometric algorithms*, Adv. Eng. Software, 13 (1991), pp. 325–331. GEOMPACK is available by contacting Professor Barry Joe (barry@cs.ualberta.ca) at the University of Alberta, Canada.

[8] B. JOE AND R. B. SIMPSON, *Triangular meshes for regions of complicated shape*, Intern. J. Num. Meth. Eng., 23 (1986), pp. 751–778.

[9] F. LETNIOWSKI, *Three-dimensional Delaunay triangulations for finite element approximations to a second-order diffusion operator*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 765–770.

[10] H. L. STONE, *Iterative solution of implicit approximations of multidimensional partial differential equations*, SIAM J. Numer. Anal., 5 (1968), pp. 530–558.