On Adaptive Mesh Generation in Two-dimensions

Ed D'Azevedo
Computer Science and Mathematics Division
Oak Ridge National Laboratory
email: e6d@ornl.gov
http://www.epm.ornl.gov/~efdazedo

Quadratic model

- Assume locally $f(x,y)$ is well approximated by quadratic Taylor expansion. Consider interpolation error over piecewise linear triangle

$$E_T(x,y) = p_\ell(x,y) - f(x,y)$$

- Error curves form a family of conics. Ellipses if $\det(H) > 0$ for convex surface and hyperpolas if $\det(H) < 0$ for saddle shaped case.
- Error formula for triangles is

$$E_T(x_c + dx, y_c + dy) = \mathcal{E}_T - \frac{1}{2}[dx, dy]H[dx, dy]^t$$

where H is Hessian matrix, (x_c, y_c) center of conic, $\mathcal{E}_T = E_T(x_c, y_c)$.

Isotropic space

- Isotropic transformation $[\tilde{x}, \tilde{y}]^t = S[x,y]^t$ based on eigen-decomposition of Hessian matrix.
- Hessian matrix holds curvature information and can be interpreted as a local metric tensor for approximation errors.
- Error expression over isotropic space $(\tilde{x}, \tilde{y})$ simplifies to

$$\tilde{E}_T(\tilde{x}_c + d\tilde{x}, \tilde{y}_c + d\tilde{y}) = \mathcal{E}_T - \frac{1}{2}(d\tilde{x}^2 + \epsilon d\tilde{y}^2)$$

- Maximum error is attained either at (x_c, y_c) or at the midpoint of longest edge over isotropic space.

Coordinate transformation

- Transformation S consists of a rotation to align axes along eigenvectors , followed by a rescaling based on eigenvalues.
-

$$\begin{aligned} H &= Q^t \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} Q \\ &= S^t \begin{bmatrix} 1 & 0 \\ 0 & \epsilon \end{bmatrix} S, \quad \text{where } \epsilon = \text{sign}(\det(H)), \\ S &= \begin{bmatrix} \sqrt{|\lambda_1|} & 0 \\ 0 & \sqrt{|\lambda_2|} \end{bmatrix} Q, \end{aligned}$$

Continuous transformation

- *Continuous* transformation $[\tilde{x}(x,y), \tilde{y}(x,y)]$ for continuous Hessian matrix $H(x,y)$ such that over the isotropic space

$$[dx, dy]H[dx, dy]^t = (d\tilde{x}^2 + \epsilon d\tilde{y}^2)$$

- Classical result in differential geometry lead to a sufficient condition of

$$K_1 h_{11} + K_2 h_{12} + K_3 h_{22} = 0$$

for some constants K_1, K_2, K_3.

PLTMG

- Package to solve 2nd order PDE available for free from `www.netlib.org`.
- User guide published by SIAM, *PLTMG: A Software Package for Solving Elliptic Partial Differential Equations. Users' Guide 8.0* by R. E. Bank.
- Uses piecewise linear continuous elements.
- Capabilities for error estimation, adaptive mesh refinement, unrefinement, smoothing.
- Exercise only meshing capability, with exact solution supplied explicitly through user function `QXY`.

Mesh refinement

- Posterior error estimate $e_h \approx u - u_h$ found by quadratic interpolation from a user supplied function `QXY`.
- `PLTMG` uses Hessian information to estimate error at mid point (x_m, y_m) of edge (x_i, y_i), (x_j, y_j)

$$u(x_m, y_m) - \tfrac{1}{2}(u(x_i, y_i) + u(x_j, y_j)) \approx$$
$$-\frac{1}{8}\begin{bmatrix} x_i - x_j \\ y_i - y_j \end{bmatrix}^t \begin{bmatrix} u_{xx} & u_{xy} \\ u_{xy} & u_{yy} \end{bmatrix} \begin{bmatrix} x_i - x_j \\ y_i - y_j \end{bmatrix},$$

- Longest edge bisection of Rivara. Element with highest error selected from heap. Refinement recursively applied to neighbors to form a valid triangulation.

Mesh unrefinement

- Error of vertex is maximum error associated with neighbor elements.
- Vertex with highest error selected from heap and removed from mesh.
- Resulting vacant region is retriangulated.

Mesh smoothing

- Mesh smoothing to reduce the approximation error by adjusting coordinates of vertices.
- Gauss-Seidel like iteration where each vertex is locally adjusted while keeping all neighboring vertices fixed.
- Boundary and interface nodes are constrained to move along boundaries.
- Typically only 4 to 5 sweeps are performed.

Numerical experiments

- Starting from initial regular mesh over $[0,1] \times [0,1]$. Perform 5 cycles of
  1. unrefinement to about 500 vertices
  2. mesh smoothing
  3. refinement to desired mesh size
  4. mesh smoothing
- Mesh I is a regular mesh of squares over isotropic space and has error equidistributing property. Mesh II essentially is Mesh I with $\pi/4$ rotation and has super-convergence property.

Example 1

Example 1. A logarithmic singularity at

$$\begin{aligned} (x_0, y_0) &= (0.5, -0.2), \\ f(x,y) &= \ln((x-x_0)^2 + (y-y_0)^2)/2\,, \\ \det(H) &= -((x-x_0)^2 + (y-y_0)^2)^{-2}\,. \end{aligned}$$

Coordinate transformation is

$$\begin{aligned} \tilde{x}(x,y) &= \arctan(y-y_0, x-x_0)\,, \\ \tilde{y}(x,y) &= -\ln((x-x_0)^2 + (y-y_0)^2)/2\,. \end{aligned}$$

Regular mesh over isotropic space

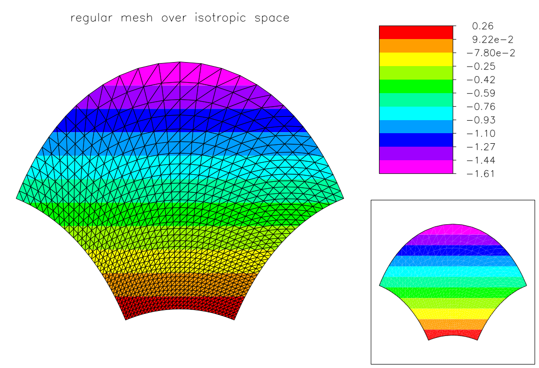

Figure 1: Regular mesh transformed to isotropic space for Example 1.

Final mesh over isotropic space

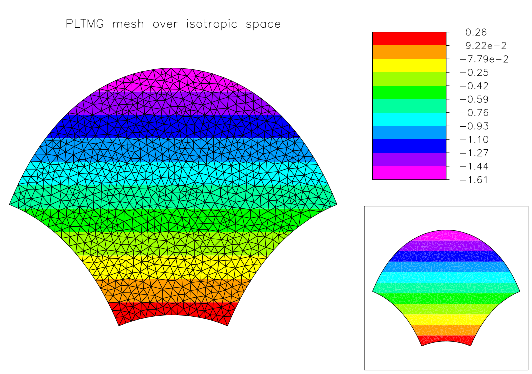

Figure 2: Final mesh by PLTMG over isotropic space for Example 1.

Final mesh produced by PLTMG

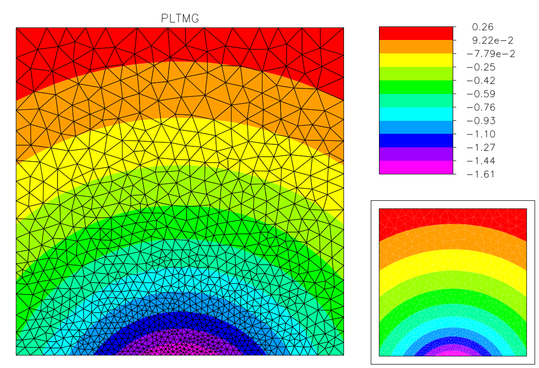

Figure 3: Final mesh produced by PLTMG for Example 1.

	Minimum error	Median error	90 percentile	Maximum error	Number of elements
Mesh I	3.56e-04	3.56e-04	3.56e-04	3.56e-04	918
Mesh II	3.44e-06	3.44e-06	3.44e-06	3.44e-06	923
PLTMG	1.53e-04	4.40e-04	6.35e-04	1.15e-03	1897

Table 1: Summary of results for Example 1.

Example 2

Example 2. A near singularity at

$$\begin{aligned}(x_0, y_0) &= (0.5, -0.2),\\ f(x,y) &= \frac{(x-x_0)^2-(y-y_0)^2}{((x-x_0)^2+(y-y_0)^2)^2},\\ \det(H) &= -36((x-x_0)^2+(y-y_0)^2)^{-4}.\end{aligned}$$

Coordinate transformation is

$$\begin{aligned}\tilde{x}(x,y) &= \sqrt{6}\left(1-\frac{x-x_0}{(x-x_0)^2+(y-y_0)^2}\right),\\ \tilde{y}(x,y) &= \sqrt{6}\frac{y-y_0}{(x-x_0)^2+(y-y_0)^2}.\end{aligned}$$

Final mesh produced by PLTMG

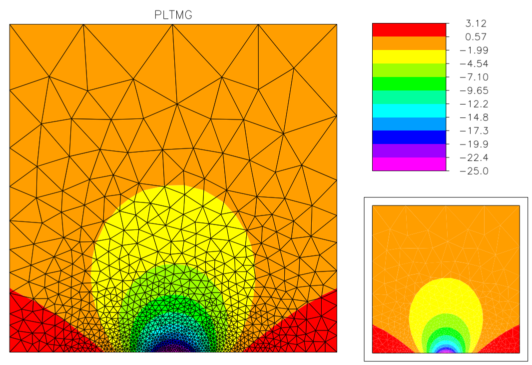

Figure 6: Final mesh produced by PLTMG for Example 2.

	Minimum error	Median error	90 percentile	Maximum error	Number of elements
Mesh I	1.30e-02	1.30e-02	1.30e-02	1.30e-02	920
Mesh II	1.27e-04	1.79e-04	3.18e-04	6.93e-04	921
PLTMG	5.32e-03	1.62e-02	2.41e-02	4.71e-02	1894

Table 2: Summary of results for Example 2.

Regular mesh over isotropic space

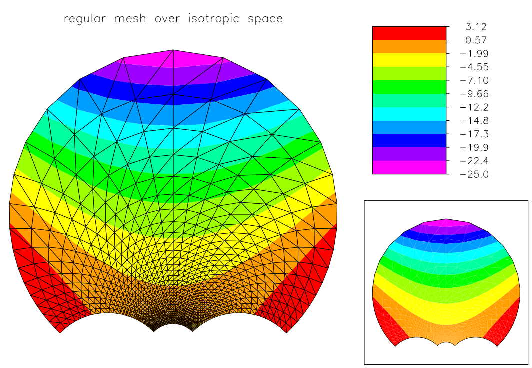

Figure 4: Regular mesh transformed to isotropic space for Example 2.

Final mesh over isotropic space

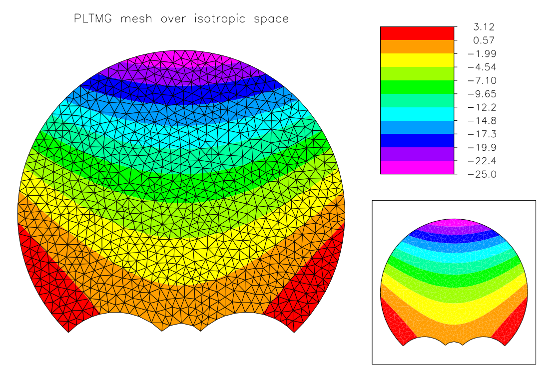

Figure 5: Final mesh by PLTMG over isotropic space for Example 2.

Example 3

Example 3. A more severe near singularity at

$$
\begin{aligned}
(x_0, y_0) &= (0.5, -0.2), \\
f(x,y) &= \frac{R^2 - 8(x-x_0)^2(y-y_0)^2}{R^4}, \\
R &= (x-x_0)^2 + (y-y_0)^2 \\
\det(H) &= -400((x-x_0)^2 + (y-y_0)^2)^{-6}.
\end{aligned}
$$

Coordinate transformation is

$$
\begin{aligned}
\tilde{x}(x,y) &= \sqrt{5}\left(1 + \frac{(y-y_0)^2 - (x-x_0)^2}{((x-x_0)^2 + (y-y_0)^2)^2}\right), \\
\tilde{y}(x,y) &= 2\sqrt{5}\frac{(x-x_0)(y-y_0)}{((x-x_0)^2 + (y-y_0)^2)^2}.
\end{aligned}
$$

Regular mesh over isotropic space

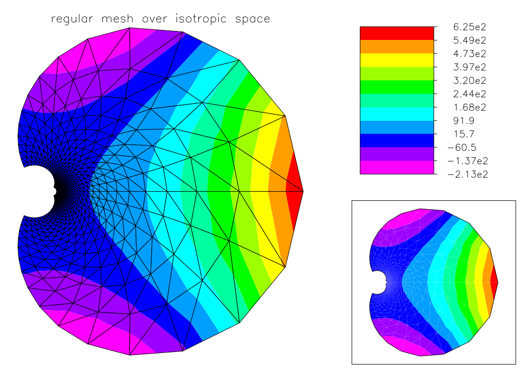

Figure 7: Regular mesh transformed to isotropic space for Example 3.

Final mesh over isotropic space

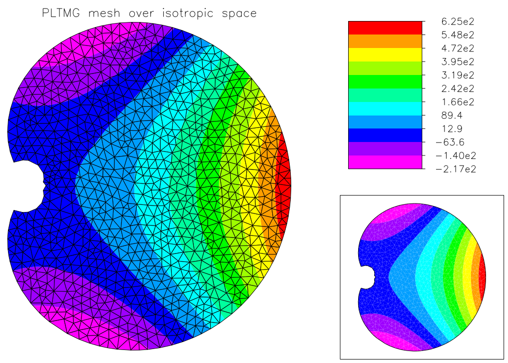

Figure 8: Final mesh by PLTMG over isotropic space for Example 3.

Final mesh produced by PLTMG

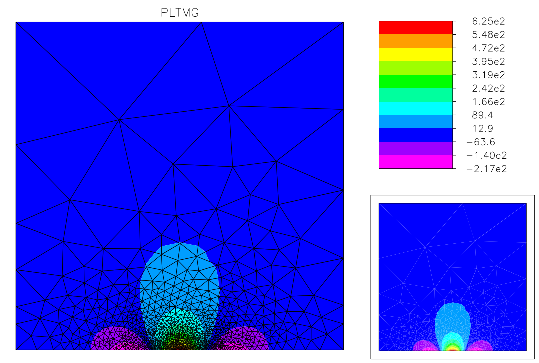

Figure 9: Final mesh produced by PLTMG for Example 3.

	Minimum error	Median error	90 percentile	Maximum error	Number of elements
Mesh I	4.54e-01	4.54e-01	4.54e-01	4.60e-01	916
Mesh II	3.69e-03	6.69e-03	1.63e-02	9.64e-02	918
PLTMG	1.58e-01	5.56e-01	8.40e-01	1.51e+00	1892

Table 3: Summary of results for Example 3.

Summary

- Graphical illustration of an effective mesh over the isotropic space.
- If we consider the median and 90 percentile errors as the basis for comparison, then PLTMG and Mesh I deviate by at most a factor of 2.
- The results suggest the solution adaptive methodology employed in PLTMG is very effective and is capable of generating meshes that approach the theoretically efficiency of optimal meshes.