Math 511. Winter 2009

Numerical Solution of PDE
Prof. Vianey Villamizar

Grid's Project

This is an individual project is due Wednesday March 25, 2009 in class

1. Write a code to generate grids on planar regions using the **"length algorithm" or Amsden-Hirt's algorithm** combining fimite difference approximations with SOR iteration. Your code should have at least two subroutines and a main program.

   Details:

   a) One of your subroutine"gridgenAH" should contain the commands to generate the grid and should include at least the following parameters:
   gridgenAH(x,y,n1,n2,ω,tol,n), where
   x: Matrix for the x coordinates of the grid points in the physical domain.
   y: Matrix for the y coordinates of the grid points in the physical domain.
   n_1: Number of grid points in the ξ-direction.
   n_2: Number of grid points in the η-direction.
   ω: Relaxation parameter of SOR iteration.
   tol: Tolerance used to establish stop criteria of SOR iteration.
   n: Maximum number of iterations of SOR procedure.

   b) Write another subroutine: **gridquality(x,y,n1,n2)** to evaluate the quality of the grids obtained. It should include computation of
   $J_{i,j}$: the jacobian at every grid point $(x_{i,j}, y_{i,j})$,
   $\theta_{i,j}$: angle between grid curves at every grid point $(x_{i,j}, y_{i,j})$,
   ADO: Average Deviation from Orthogonality, and
   MDO: Maximum Deviation from Orthogonality.

   c) Write a main program which calls all the subroutines. In this program define the data parameters needed by the subroutines. You may also define the boundary transformation ∂T that transform the four sides of the rectangular computational domain into the boundaries of the physical domain and the branch cut (for multiply connected domains). Also, an initial grid may be defined in this main program. This grid will be used as an initial guess to start the iterative algorithm defined in gridgenAH.

   d) Apply your code for the Amsden-Hirt's algorithm to the following simply connected physical domains:

      i) Dome (Convex Domain)
      Region enclosed by coordinate axes: x=0, and y=0, vertical line: x=1, and curve: $y = -4(x - 1/2)^2 + 2$

      ii) Swan Domain
      Region enclosed by coordinate axes: x=0, and y=0, and curves: $y = 1 - 2x + 2x^2, \quad x = 1 + 2y - 2y^2$

iii) Prototypical Antenna Domain
Region enclosed by segments:
y=1, $-3 \le x \le 0$, y=0, $-3 \le x \le 0$,
x=0, $1 \le y \le 3 + 1/2$, x=0, $-3 + 1/2 \le y \le 0$, and
semicircle: $x \ge 0$, $x^2 + (y - 1/2)^2 = 3^2$

e) Perform two experiments for each domain:
i) n1=41 n2=61; ii) n1=61,n2=61.
Use $tol = 10^{-5}$, $n = 5000$ (if your problem needs more iterations to converge increase n as needed).

f) Report your results in a table that includes:
($n1 \times n2$): grid size,
MDO: Maximum deviation from orthogonality,
ADO: Average deviation from orthogonality,
$|J|_{min}$: Minimum absolute value of the jacobian,
(x_{min}, y_{min}): Point where Jacobian is minimum or maybe zero. Your code should find a way to determine if the Jacobian changes its sign (a warning message should be printed if that happens). That will be an indicator of the presence of zero jacobian.
ItsTotal: Number of iterations.
Make comments on your results. Try to establish a correlation between jacobian=0 and points of non-uniqueness.

g) For each domain, draw only the initial and final 41×61 grid. Ask MATLAB graphical subroutine to mark the points on the graph of the final grid where the jacobian is minimum and where the deviation from orthogonality is the highest. For a better analysis, draw a surface plot of the jacobian $J_{i,j}$ over each 41×61 final grid and comment on it.

h) Use the formula given in the notes to obtain the optimum ω for the system of Laplace's equation. Run your code for two other values of ω, record and report the number of iterations and verify that the theoretical optimum ω indeed leads to a faster convergence of SOR. You can be more precise by making a graph of the number of iterations versus values of ω: $NIts(\omega)$.

2. Consider the transformation T from the rectangular computational domain $\mathcal{D}'$ into the two-dimensional physical domain $\mathcal{D}$ defined by $x = x(\xi, \eta)$ and $y = y(\xi, \eta)$.

(a) Prove the following theorem.

Theorem 1 *If the jacobian of the transformation* T $J(\xi, \eta) \neq 0$ *for all* $(\xi, \eta) \in \mathcal{D}'$ *then, the inverse of the Laplace system of equations*

$$\xi_{xx} + \xi_{yy} = 0 \quad (1)$$
$$\eta_{xx} + \eta_{yy} = 0, \quad (x, y) \in \mathcal{D} \quad (2)$$

can be written as the Winslow's quasi-linear elliptic system:

$$\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} = 0, \tag{3}$$

$$\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} = 0, \quad (\xi, \eta) \in \mathcal{D}' \tag{4}$$

$$\textit{where} \quad \alpha = x_\eta^2 + y_\eta^2, \quad \beta = x_\eta x_\xi + y_\xi y_\eta, \quad \gamma = x_\xi^2 + y_\xi^2. \tag{5}$$

(b) Use centered finite difference combined with SOR iteration to obtain the finite difference equations to be used by **"Smoothness algorithm" or Winslow's algorithm**. Write a code to generate grids on simply connected planar regions using Winslow's algorithm. Your code should consist of at least two subroutines: gridgenWinslowSC(x,y,n1,n2,ω,tol,n), gridquality(x,y,n1,n2), and a main program, as proposed in problem (1).

(c) Follow the instruction and perform all the work indicated in items (1c)-(1g) for the Amsden-Hirt's algorithm, but this time for the Winslow's algorithm.

(d) Compare the tables obtained from both algorithms and make comments.

(e) Compare the graphs from both algorithms and make comments.

(f) Determine an empirical optimum relaxation parameter by running your Winslow's algorithm for different values of ω. Make a graph of ω versus number of iterations.

3. Modify your Winslow's code to generate grids on multiply connected planar regions containing holes.

a) As in the previous two cases, define the boundary conditions that form part of the BVP which determine the transformation T in your main program. However, in this case only two sides of the rectangle correspond to physical boundaries. On the other two sides of the rectangular computational domain impose continuity conditions (periodic conditions).

b) Modify the subroutine gridgenWinslowSC and create gridgenWinslowMC. Perform similar work as described in (2b)-(2c) and (2f).

c) Apply your modified code for Winslow's algorithm to the following multiply connected physical domains:

i) Three-Leafed Rose Domain
Region bounded by
Three-leafed rose: $x(t) = 0.3(2 + \cos 3t)\cos t$, $y(t) = 0.3(2 + \cos 3t)\sin t$, $0 \le t \le 2\pi$ and outer circle: $x^2 + y^2 = 2^2$

ii) Astroid Shaped Hole:
parametric equations:
$x(\theta) = 0.5(3\cos(\theta) + \cos(3\theta))$ and $y(\theta) = 0.5(3\sin(\theta) - \sin(3\theta))$, with $0 \le \theta \le 2\pi$ and outer circle: $x^2 + y^2 = 6^2$

d) Perform two experiments for each domain: i) n1=71, n2=21; ii) n1=71 n2=41. Use $tol = 10^{-5}$, $n = 5000$ (if your problem needs more iterations to converge increase n as needed).

e) Report your results in a table similar to the one described in (1f). **Make comments on your results.**

f) Draw only one grid: 71 $\times$ 41 for each domain.

Recommendations:
i) Read carefully all the items and try to give answer to all of them. In this work, we are trying to analyze the properties of different grid generators, **make sure you explain differences and analogies in your comments.**
ii) I may have overlooked something in the problems proposed, if you think so, feel free to add anything necessary.
iii) I have carefully worked out all the parts in the project myself. I think this is reasonable and doable in a week. However, you will need TO START EARLY. I will be more than willing to clarify questions as you go along.

"You are capable of something better. Give it your best. You will never again have such an opportunity. Pray about it. Work at it. Make it happen."
Pres. Hinckley at the Inauguration of Pres. Samuelson