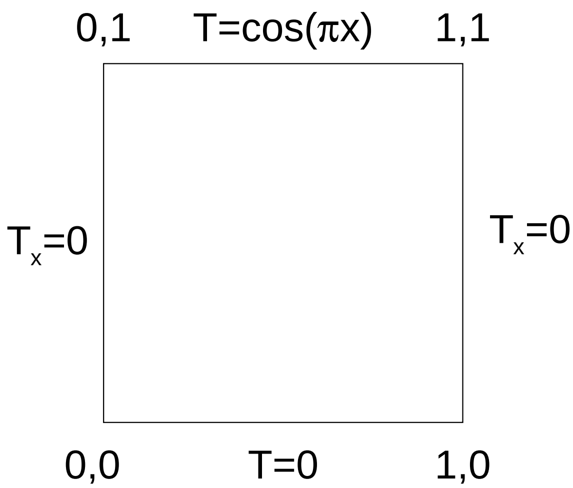
Programming Assignment 1

Mech 510

Laplace's Equation. This problem covers the solution of the Laplace equation with mixed boundary conditions, as shown below. The exact solution to this problem is

$$T(x,y) = \frac{\cos(\pi x)\sinh(\pi y)}{\sinh \pi}$$



When you turn in this assignment online, please also submit an electronic copy of your program. Not only do I give feedback about code, I can often find that one last bug that students can't, just because I've done this lots of times now.

1. *Boundary conditions.*

   If you intend to use ghost cells to apply boundary conditions, write a subroutine that sets the correct ghost cell values for the given boundary conditions. Verify that your subroutine is correct by setting the solution initially to zero everywhere for a 10×10 mesh, then using your subroutine to set values in the ghost cells. Check the ghost cell values by hand.

If you intend to implement boundary conditions by modifying your flux integration scheme (in this case, this is equivalent to modifying your Gauss-Seidel iteration), you'll need to verify that you are computing the correct fluxes. Use the exact solution as data for this.

You do not need to include any documentation of this step in your report; if your boundary conditions are wrong, I'll be able to tell that immediately.

2. *Point Gauss-Seidel iteration scheme.* Write a subroutine that performs a single Gauss-Seidel iteration with no over-relaxation. In addition to updating the solution, your subroutine should return the maximum change in solution for any control volume. Loop through the mesh first in the i-direction, then in the j-direction.
   Test your subroutine by using a 10×10 mesh with initial solution of zero; compare the result with a hand computation for this same problem. (Again, you don't need to write up the results of this test in your report.)
   Then use your Gauss-Seidel routine to solve the given Laplace problem for a 10×10 mesh with a starting solution of zero, iterating until the maximum change in solution, as reported by your subroutine, is smaller than 10^{-7}. Plot your solution error (computed - exact). Also, report the number of iterations required and the L_2 norm of the error in your converged solution.

3. *Over-relaxation.* Modify your Gauss-Seidel subroutine to perform over-relaxation. Repeat the previous test with $\omega = 1.5$, and report the number of iterations required to reach convergence. (This should be significantly less than without over-relaxation.)

4. *Convergence behavior.* For a 20×20 mesh, and values of ω of 1, 1.3, and 1.5, run your program to steady state. Plot, on a single graph, the maximum change in solution at each iteration as a function of iteration count. Be sure that your plot is scaled appropriately to show what's actually going on here.

5. *Accuracy.* Finally, run your code for a 40×40 mesh and an 80×80 mesh. Tabulate or plot the L_2 norms of solution error for each mesh (including those you've already done), and estimate the actual order of accuracy of your code. You may want to converge your solution a bit farther than a maximum solution change of 10^{-7} to ensure that the error you are measuring is discretization error, not lack of convergence.

Application: Poisson problem for pressure calculation for incompressible flows One of the ways to get around the problem of coupling pressure and velocity in incompressible flows involves solving a Poisson problem for the pressure. This equation is derived by taking the divergence of the momentum equations (in non-dimensional form):

$$
\begin{aligned}
\frac{\partial}{\partial x}\left\{\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}\right\} &= \frac{\partial}{\partial x}\left\{-\frac{\partial P}{\partial x}+\frac{1}{\mathrm{Re}}\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)\right\} \\
\frac{\partial}{\partial y}\left\{\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}\right\} &= \frac{\partial}{\partial y}\left\{-\frac{\partial P}{\partial y}+\frac{1}{\mathrm{Re}}\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right)\right\}
\end{aligned}
$$

Summing these and expanding, we get:

$$
\begin{aligned}
&\frac{\partial}{\partial x}\frac{\partial u}{\partial t}+\frac{\partial u}{\partial x}\frac{\partial u}{\partial x}+u\frac{\partial^2 u}{\partial x^2}+\frac{\partial v}{\partial x}\frac{\partial u}{\partial y}+v\frac{\partial^2 u}{\partial x\partial y} \\
&+\frac{\partial}{\partial y}\frac{\partial v}{\partial t}+\frac{\partial u}{\partial y}\frac{\partial v}{\partial x}+u\frac{\partial^2 v}{\partial x\partial y}+\frac{\partial v}{\partial y}\frac{\partial v}{\partial y}+v\frac{\partial^2 v}{\partial y^2} = -\frac{\partial^2 P}{\partial x^2}+\frac{1}{\mathrm{Re}}\frac{\partial}{\partial x}\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right) \\
&\qquad -\frac{\partial^2 P}{\partial y^2}+\frac{1}{\mathrm{Re}}\frac{\partial}{\partial y}\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right)
\end{aligned}
$$

If u and v are three times continuously differentiable, we can switch the order of differentiation of some terms and rearrange their order to get:

$$
\begin{aligned}
&\left[\frac{\partial}{\partial t}\left\{\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}\right\}\right] \\
&+\left[u\frac{\partial}{\partial x}\left\{\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}\right\}\right]+\left[v\frac{\partial}{\partial y}\left\{\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}\right\}\right] \\
&+\frac{\partial u}{\partial x}\frac{\partial u}{\partial x}+\frac{\partial v}{\partial x}\frac{\partial u}{\partial y}+\frac{\partial u}{\partial y}\frac{\partial v}{\partial x}+\frac{\partial v}{\partial y}\frac{\partial v}{\partial y} = -\left(\frac{\partial^2 P}{\partial x^2}+\frac{\partial^2 P}{\partial y^2}\right) \\
&\qquad +\frac{1}{\mathrm{Re}}\left[\frac{\partial^2}{\partial x^2}\left\{\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}\right\}\right]+\frac{1}{\mathrm{Re}}\left[\frac{\partial^2}{\partial y^2}\left\{\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}\right\}\right]
\end{aligned}
$$

Each term in this form has an exactly corresponding term in the previous version. The quantity in $\{\}$ is the divergence of velocity, which is zero for incompressible flow, so all terms in [] can be eliminated. With some simplification, we get:

$$
\frac{\partial^2 P}{\partial x^2}+\frac{\partial^2 P}{\partial y^2}=-\left(\left(\frac{\partial u}{\partial x}\right)^2+2\frac{\partial v}{\partial x}\frac{\partial u}{\partial y}+\left(\frac{\partial v}{\partial y}\right)^2\right) \tag{1}
$$

While most schemes that use a pressure Poisson problem solve a somewhat different form than this for numerical reasons, this will do for our purposes.

Modify your Laplace equation code from the first part of this assignment to use a source term, computed using Equation 1 for the following velocities:

$$\begin{aligned} u &= x^3 - 3xy^2 \\ v &= -3x^2y + y^3 \end{aligned}$$

Compute the pressure on the unit square domain above, using a sequence of meshes with the finest mesh at least as fine as 40×40 (if possible, use at least an 80×80). As boundary conditions, use $\frac{\partial P}{\partial n} = 0$ at $x = 0$ and $y = 0$. On $x = 1$, use $P = 5 - \frac{1}{2}\left(1 + y^2\right)^3$; on $y = 1$, use $P = 5 - \frac{1}{2}\left(1 + x^2\right)^3$. Use your sequence of computed solutions to estimate the solution value P at $(x, y) = \left(\frac{1}{2}, \frac{1}{2}\right)$. (Hint: this will require some sort of interpolation based on nearby control volume values.) Also, provide an error bound on this value, based on the procedure described in the ASME solution accuracy handout.