In Class Work Sheet XII: Elliptic PDEs

1. Laplace $\nabla^2\phi = 0$
2. Poisson $\nabla^2\phi = f$
3. Helmholtz $\nabla^2\phi + \alpha^2\phi = 0$

These are boundary value problems in all directions (2D and 3D) and can be subject to the following types of boundary conditions:

1. Neumann conditions $\frac{\partial\phi}{\partial n} = g$
2. Dirichlet conditions $c_1\phi = g$
3. Robin (or mixed conditions) $c_1\phi + \frac{\partial\phi}{\partial n} = g$

Central Differencing to Laplace Consider uniform grid in x and y with $\Delta x = \Delta y = \Delta$. Let there be $M+1$ points in x and y directions labeled from $0, 1, 2....M$ and each point the domain is represented by a pair (x_i, y_j).

How many equations? What about Dirichlet conditions?

Can we arrange them in the form $\mathbf{A}\bar{x} = \bar{b}$?
Discretized equation can be written in the form

$$a_P \phi_P = \sum a_{\text{nbr}} \phi_{\text{nbr}} + b_P \tag{1}$$

1 Point-Jacobi Iterative Method

Starting with $k = 0$ and some initial guess for $\phi_{i,j}^{(k)}$,

$$a_P \phi_P^{(k+1)} = \sum_{\text{nbr}} a_{\text{nbr}} \phi_{\text{nbr}}^{(k)} + b_P \tag{2}$$

or

$$\phi_P^{(k+1)} = \frac{\sum a_{\text{nbr}} \phi_{\text{nbr}}^{(k)} + b_P}{a_P} \tag{3}$$

Example of Point-Jacobi

$$T_1 = 0.4T_2 + 0.2 \tag{4}$$

$$T_2 = T_1 + 1 \tag{5}$$

Point Jacobi uses old values for all unknowns and gets the new values using the following equation

$$T_1^{(k+1)} = 0.4T_2^{(k)} + 0.2 \tag{6}$$

$$T_2^{(k+1)} = T_1^{(k)} + 1 \tag{7}$$

2 Gauss-Seidel

However, notice that, if we solve the equation for T_1 first, we will already have $T_1^{(k+1)}$ that can be used in the equation for T_2. Point Jacobi does not do this, but this is exactly what's done for Gauss-Seidel. **Example Gauss-Seidel** Consider

$$T_1^{(k+1)} = 0.4T_2^{(k)} + 0.2 \tag{8}$$

$$T_2^{(k+1)} = T_1^{(k+1)} + 1 \tag{9}$$

$$4\phi_P^{(k+1)} = \phi_{i+1,j}^{(k)} + \phi_{i-1,j}^{(k+1)} + \phi_{i,j+1}^{(k)} + \phi_{i,j-1}^{(k+1)} - \Delta^2 f_{i,j} \tag{10}$$

For large M, this is faster than Point-Jacobi, but still very slow method. Note that it uses the **latest** value available for each neighboring point value. So if we are going from $i = 0$ to higher i and low $j = 0$ to higher j in our *loop*, then the left side and bottom side values are already at $k + 1$ level and are readily used here.

3 Successive Over Relaxation (SOR)

Example Successive Over Relaxation (SOR) Consider our test problem, with the Gauss-Seidel iteration. For SOR we write

$$
\begin{aligned}
T_1^{(k+1*)} &= 0.4T_2^{(k)} + 0.2 && (11)\\
T_1^{(k+1)} &= T_1^{(k)} + \alpha(T_1^{(k+1*)} - T_1^{(k)}) && (12)\\
T_2^{(k+1*)} &= T_1^{(k+1)} + 1 && (13)\\
T_2^{(k+1)} &= T_2^{(k)} + \alpha(T_2^{(k+1*)} - T_2^{(k)}) && (14)
\end{aligned}
$$

Try $\alpha = 1.2$ (over-relaxation)

Successive Over Relaxation (SOR) Consider the discretized equation for Gauss-Seidel iteration. Starting with $k = 0$ and some initial guess for $\phi_{i,j}^{(k)}$,

$$\phi_P^{(k+1*)} = \frac{1}{4}(\phi_{i+1,j}^{(k)} + \phi_{i-1,j}^{(k+1*)} + \phi_{i,j+1}^{(k)} + \phi_{i,j-1}^{(k+1*)} - \Delta^2 f_{i,j}) \tag{15}$$

The change in ϕ_P during this iteration is $d = \phi_P^{(k+1*)} - \phi_P^{(k)}$. In stead of accepting $\phi_P^{(k+1*)}$ as the new solution, we write,

$$\phi_P^{(k+1)} = \phi_P^{(k)} + \alpha(\phi_P^{(k+1*)} - \phi_P^{(k)}) \tag{16}$$

If $\alpha = 1$, we get Gauss-Seidel.

If $0 \leq \alpha \leq 1$ we get under-relaxation.

If $\alpha > 1$, we get over-relaxation which can accelerate convergence!

4 Does iterative method always converge?

Consider

$$T_1 = 0.4T_2 + 0.2 \tag{17}$$
$$T_2 = T_1 + 1 \tag{18}$$

Re-arrange and get the following set of equations:

$$T_1 = T_2 - 1 \tag{19}$$
$$T_2 = 2.5T_1 - 0.5 \tag{20}$$

Apply Gauss-Seidel (relaxation factor = 1) to get

Does iterative method always converge? No! A *sufficient* condition for convergence is the Scarborough criterion:

For the finite-difference approximation of the type

$$a_P\phi_P = \sum a_{\mathrm{nbr}}\phi_{\mathrm{nbr}} + b_P \tag{21}$$

A sufficient condition for convergence is :

$$\frac{\sum |a_{\mathrm{nbr}}|}{|a_P|}\begin{cases} \leq \ 1 & \text{for all equations} \\ < \ 1 & \text{for at least one equation} \end{cases} \tag{22}$$

Point-Jacobi

Iteration #	0	1	2	3	4	5	6	7
T_1	0	0.2	0.6	0.68	0.84	0.872	0.936	
T_2	0	1	1.2	1.6	1.68	1.84	1.872	

However, notice that, if we solve the equation for T_1 first, we will already have $T_1^{(k+1)}$ that can be used in the equation for T_2. Point Jacobi does not do this, but this is exactly what's done for Gauss-Seidel.

Gauss-Seidel
Consider

$$T_1^{(k+1)} = 0.4T_2^{(k)} + 0.2 \tag{23}$$
$$T_2^{(k+1)} = T_1^{(k+1)} + 1 \tag{24}$$

Iteration #	0	1	2	3	4	5	6	7
T_1	0	0.2	0.68	0.872	0.9488	0.979	0.9918	
T_2	0	1.2	1.68	1.872	1.9488	1.979	1.9918	

Notice faster convergence than Point-Jacobi.

SOR

Consider our test problem, with the Gauss-Seidel iteration. For SOR we write

$$T_1^{(k+1*)} = 0.4T_2^{(k)} + 0.2 \tag{25}$$
$$T_1^{(k+1)} = T_1^{(k)} + \alpha(T_1^{(k+1*)} - T_1^{(k)}) \tag{26}$$
$$T_2^{(k+1*)} = T_1^{(k+1)} + 1 \tag{27}$$
$$T_2^{(k+1)} = T_2^{(k)} + \alpha(T_2^{(k+1*)} - T_2^{(k)}) \tag{28}$$

Try $\alpha = 1.2$ (over-relaxation)

Iteration #	0	1	2	3	4	5	6	7
T_1	0	0.24	0.90624	1.01389	1.0061981	..		
T_2	0	1.488	1.989888	2.0187003	2.0036977	..		

Scarborough
Consider

$$T_1 = 0.4T_2 + 0.2 \tag{29}$$
$$T_2 = T_1 + 1 \tag{30}$$

Re-arrange and get the following set of equations:

$$T_1 = T_2 - 1 \tag{31}$$
$$T_2 = 2.5T_1 - 0.5 \tag{32}$$

Apply Gauss-Seidel (relaxation factor = 1) to get

Iteration #	0	1	2	3	4	5	6	7
T_1	0	-1	-4	-11.5	-30.25	..		
T_2	0	-3	-10.5	-29.25	-76.13	..		