Finite Difference Method to Solve Poisson Equation

Vishnu Raghuraman

April 2021

1 One-Dimensional Case

In 1D, the Poisson equation can be written as follows

$$\frac{d^2\phi}{dx^2} = -\rho(x) \tag{1}$$

Now, consider the domain to be $\{x_0, x_1, x_2, \cdots, x_n, x_{N+1}\}$ - these points are equally spaced, with the spacing defined as

$$d = \frac{x_{N+1} - x_0}{N}, \tag{2}$$

where x_0 and x_N are boundary points. The second derivative at $x = x_i$ can be written as

$$\left(\frac{d^2\phi}{dx^2}\right)_{x=x_i} = \frac{\phi_{i+1} - 2\phi_i + \phi_{i-1}}{d^2} \tag{3}$$

Inserting this into the Poisson equation

$$\frac{\phi_{i+1} - 2\phi_i + \phi_{i-1}}{d^2} = -\rho_i \tag{4}$$

1.1 Dirichlet Boundary Conditions

This can be recast into matrix form as follows

$$\frac{1}{d^2}\underbrace{\begin{bmatrix} 1 & -2 & 1 & \cdots & \cdots & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & 1 & -2 & 1 \end{bmatrix}}_{N-1\,\times\,N+1} \underbrace{\begin{bmatrix} \phi_0 \\ \phi_1 \\ \vdots \\ \phi_N \end{bmatrix}}_{N+1\,\times\,1} = -\underbrace{\begin{bmatrix} \rho_1 \\ \vdots \\ \rho_{N-1} \end{bmatrix}}_{N\,\times\,1} \tag{5}$$

To make the coefficient matrix square, we can write the boundary terms separately.

$$\frac{1}{d^2}\begin{bmatrix} -2 & 1 & 0 & \cdots & \cdots & 0 \\ 1 & -2 & 1 & \ddots & \cdots & \vdots \\ 0 & 1 & -2 & 1 & \ddots & \vdots \\ \vdots & 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \vdots & \ddots & 1 & -2 & 1 \\ 0 & \cdots & \cdots & 0 & 1 & -2 \end{bmatrix} \begin{bmatrix} \phi_1 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \phi_{N-1} \end{bmatrix} + \frac{1}{d^2}\begin{bmatrix} \phi_0 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \phi_N \end{bmatrix} = -\begin{bmatrix} \rho_1 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \rho_{N-1} \end{bmatrix} \tag{6}$$

This is now a straightforward matrix-vector equation that can be easily solved using LAPACK/BLAS routines in Fortran.

2 Two-Dimensional Case

In 2D, the Poisson equation can be written as follows

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = -\rho(x, y) \tag{7}$$

We now need to consider a 2D grid. Let's say our grid goes from x_0 to x_N along the x-axis and y_0 to y_N along the y-axis. The spacing between any two points along the x-axis is d_x and along the y-axis is d_y. The differential equation can then be discretized as follows

$$\frac{\phi_{i+1,j} - 2\phi_{i,j} + \phi_{i-1,j}}{d_x^2} + \frac{\phi_{i,j+1} - 2\phi_{i,j} + \phi_{i,j-1}}{d_y^2} = -\rho_{ij} \tag{8}$$

$$\frac{1}{d_x^2}\phi_{i+1,j} + \frac{1}{d_x^2}\phi_{i-1,j} + \frac{1}{d_y^2}\phi_{i,j+1} + \frac{1}{d_y^2}\phi_{i,j-1} - \left(\frac{2}{d_x^2} + \frac{2}{d_y^2}\right)\phi_{i,j} = \rho_{ij} \tag{9}$$

We can express this in a matrix-vector form. This is a bit more complicated than the 1D case. To simplify, instead of considering a general N × N grid, let's take a 5 × 5. The below is what the co-ordinate mapping (x, y) or (i, j) is going to look like for the grid

$$\begin{bmatrix} (0,4) & (1,4) & (2,4) & (3,4) & (4,4) \\ (0,3) & (1,3) & (2,3) & (3,3) & (4,3) \\ (0,2) & (1,2) & (2,2) & (3,2) & (4,2) \\ (0,1) & (1,1) & (2,1) & (3,1) & (4,1) \\ (0,0) & (1,0) & (2,0) & (3,0) & (4,0) \end{bmatrix} \tag{10}$$

For this grid, the discretized Poisson equation can be expressed in matrix vector form as follows

$$\begin{bmatrix} \begin{pmatrix} -\frac{2}{d_x^2} - \frac{2}{d_y^2} & \frac{1}{d_x^2} & 0 \\ \frac{1}{d_x^2} & -\frac{2}{d_x^2} - \frac{2}{d_y^2} & \frac{1}{d_x^2} \\ 0 & \frac{1}{d_x^2} & -\frac{2}{d_x^2} - \frac{2}{d_y^2} \end{pmatrix} & \frac{1}{d_y^2}\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \\ \frac{1}{d_y^2}\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} & \begin{pmatrix} -\frac{2}{d_x^2} - \frac{2}{d_y^2} & \frac{1}{d_x^2} & 0 \\ \frac{1}{d_x^2} & -\frac{2}{d_x^2} - \frac{2}{d_y^2} & \frac{1}{d_x^2} \\ 0 & \frac{1}{d_x^2} & -\frac{2}{d_x^2} - \frac{2}{d_y^2} \end{pmatrix} & \frac{1}{d_y^2}\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \\ \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & \frac{1}{d_y^2}\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} & \begin{pmatrix} -\frac{2}{d_x^2} - \frac{2}{d_y^2} & \frac{1}{d_x^2} & 0 \\ \frac{1}{d_x^2} & -\frac{2}{d_x^2} - \frac{2}{d_y^2} & \frac{1}{d_x^2} \\ 0 & \frac{1}{d_x^2} & -\frac{2}{d_x^2} - \frac{2}{d_y^2} \end{pmatrix} \end{bmatrix} \begin{bmatrix} \phi_{11} \\ \phi_{21} \\ \phi_{31} \\ \phi_{12} \\ \phi_{22} \\ \phi_{32} \\ \phi_{13} \\ \phi_{23} \\ \phi_{33} \end{bmatrix} \tag{11}$$

$$\underbrace{+ \frac{1}{d_y^2}\begin{bmatrix} \phi_{10} \\ \phi_{20} \\ \phi_{30} \\ 0 \\ 0 \\ 0 \\ \phi_{14} \\ \phi_{24} \\ \phi_{34} \end{bmatrix} + \frac{1}{d_x^2}\begin{bmatrix} \phi_{01} \\ 0 \\ \phi_{41} \\ \phi_{02} \\ 0 \\ \phi_{42} \\ \phi_{03} \\ 0 \\ \phi_{43} \end{bmatrix}}_{\text{Boundary Conditions}} = \begin{bmatrix} \rho_{11} \\ \rho_{21} \\ \rho_{31} \\ \rho_{12} \\ \rho_{22} \\ \rho_{32} \\ \rho_{13} \\ \rho_{23} \\ \rho_{33} \end{bmatrix} \tag{12}$$

This structure will hold for any $N \times N$. This matrix-vector equation can be solved to get the potential at all the internal grid points. Note that we now need 4 functions which will serve as boundary conditions.