Numerical Solution of Poisson Equation

Zhengyang Liu

(Dated: June 17, 2021)

This note documents the numerical solution of Poisson Equation. Consider the following PDE

$$\nabla^2\psi(x, y) = -\omega \tag{1}$$

in domain $x, y \in [0, 1]$.

We start by discretizing the Laplacian $\nabla^2\psi$ in Eq. 1 using the centered difference approximation, into an $N \times N$ grid. The distance between adjacent grid nodes is $h = 1/N$. Other higher order discretization schemes will be discussed later.

$$\left(\frac{\partial\psi}{\partial x}\right)_{i,j} \approx (\psi_{i+1,j} - \psi_{i,j})/h$$

$$\left(\frac{\partial\psi}{\partial x}\right)_{i-1,j} \approx (\psi_{i,j} - \psi_{i-1,j})/h$$

$$\begin{aligned}\left(\frac{\partial^2\psi}{\partial x^2}\right)_{i,j} &\approx \left(\left(\frac{\partial\psi}{\partial x}\right)_{i,j} - \left(\frac{\partial\psi}{\partial x}\right)_{i-1,j}\right)/h \\ &= \frac{(\psi_{i+1,j} - \psi_{i,j})/h - (\psi_{i,j} - \psi_{i-1,j})/h}{h} \\ &= \frac{\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}}{h^2}\end{aligned} \tag{2}$$

similarly,

$$\left(\frac{\partial^2\psi}{\partial y^2}\right)_{i,j} \approx \frac{\psi_{i,j+1} - 2\psi_{i,j} + \psi_{i,j-1}}{h^2} \tag{3}$$

combine Eqs. 2 and 3, we get

$$(\nabla^2\psi)_{i,j} = \frac{\psi_{i+1,j} + \psi_{i,j+1} - 4\psi_{i,j} + \psi_{i-1,j} + \psi_{i,j-1}}{h^2} = -\omega_{i,j} \tag{4}$$

For each node (i, j), there is a linear equation of $\mathbf{\Psi} = (\psi_{0,0}, ..., \psi_{N-1,N-1})$. These equations form a linear system with $N \times N$ equations, where $\mathbf{\Psi}$ can be solved for. The linear system can be

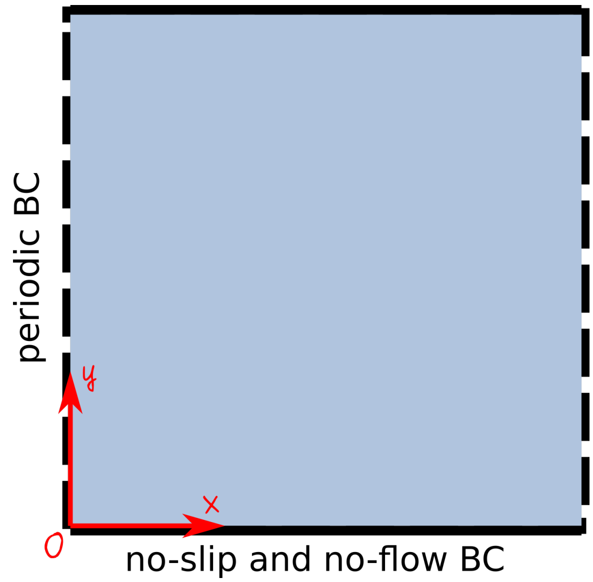

FIG. 1. Schematic of the system.

expressed in the form of $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, where

$$\boldsymbol{A} = \begin{bmatrix} \mathit{solution} \ddots & & & & & & & & & & & \\ & \ddots & & & & & & & & & & \\ & & \ddots & & & & & & & & & \\ & & & \ddots & & & & & & & & \\ \cdots & & 1 & \cdots & 1 & -4 & 1 & \cdots & 1 & \cdots \\ & & & & & & \ddots & & & & \\ & & & & & & & \ddots & & & \\ & & & & & & & & \ddots & & \\ & & & & & & & & & \ddots \end{bmatrix}_{N^2 \times N^2}, \quad \boldsymbol{x} = \begin{bmatrix} \vdots \\ \psi_{i-1,j} \\ \vdots \\ \psi_{i,j-1} \\ \psi_{i,j} \\ \psi_{i,j+1} \\ \vdots \\ \psi_{i+1,j} \\ \vdots \end{bmatrix}_{N^2} \tag{5}$$

The row shown in $\boldsymbol{A}$ is the $(iN + j)$ th row, corresponding to the (i, j) element in the discretized streamfunction ψ and vorticity ω. Multiplying this row to $\boldsymbol{x}$, we backup the general form of Eq. 4. Note that near the boundaries, this general form needs to be modified to satisfy the boundary conditions. This is what we are going to do next.

Boundary conditions

As in Fig. 1, we impose periodic boundary condition in x direction and no-slip-no-flow boundary condition in y direction. The no-slip-no-flow boundary condition is formally stated as

$$\frac{\partial \psi}{\partial y}\Big|_{y=0} = \frac{\partial \psi}{\partial y}\Big|_{y=1} = \frac{\partial \psi}{\partial x}\Big|_{y=0} = \frac{\partial \psi}{\partial x}\Big|_{y=1} = 0$$

The periodic boundary condition, to my knowledge, can only be expressed in a discretized expression.

Now let's discretize the boundary conditions. Here, we consider the conditions at $y = 0$ a backward discretization and the conditions at $y = 1$ a forward discretization. The idea is to use the BC's to generate extra "imaginary grids" so that the Laplacian at the boundaries can be evaluated more accurately. Fig. 2 shows how an imaginary node $(-1, j)$ helps evaluating the Laplacian at $(0, j)$. Formally, the boundary conditions at $y = 0$ can be written as

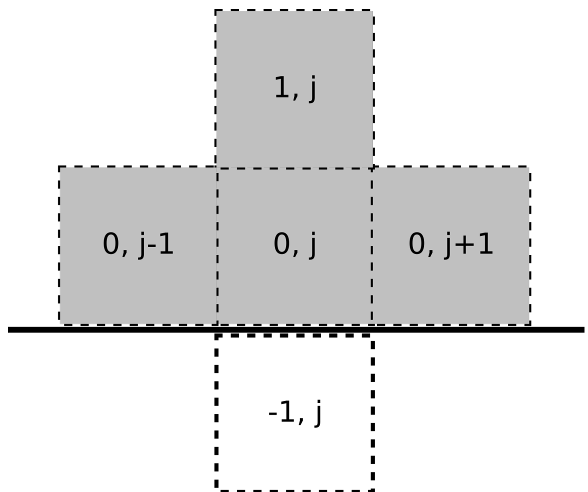

FIG. 2. Use boundary condition to generate imaginary grid nodes.

$$\frac{\psi_{0,j} - \psi_{-1,j}}{h} = 0, \; \frac{\psi_{0,j+1} - \psi_{0,j}}{h} = 0$$

Similarly, at $y = 1$

$$\frac{\psi_{N,j} - \psi_{N-1,j}}{h} = 0, \; \frac{\psi_{N-1,j+1} - \psi_{N-1,j}}{h} = 0$$

Apply the conditions above to the 5-point discretization scheme in Eq. 4, we have

$$\begin{aligned}(\nabla^2\psi)_{0,j} &= \frac{\psi_{1,j} + \psi_{0,j+1} - 4\psi_{0,j} + \psi_{-1,j} + \psi_{0,j-1}}{h^2} \\ &= \frac{\psi_{1,j} - \psi_{0,j}}{h^2}\end{aligned} \tag{6}$$

$$\begin{aligned}(\nabla^2\psi)_{N-1,j} &= \frac{\psi_{N,j} + \psi_{N-1,j+1} - 4\psi_{N-1,j} + \psi_{N-2,j} + \psi_{N-1,j-1}}{h^2} \\ &= \frac{\psi_{N-1,j} - \psi_{N-2,j}}{h^2}\end{aligned} \tag{7}$$

Eqs. 6 and 7 holds for $j = 0, 1, \ldots, N-1$ since the evaluation of the Laplacian does not require a derivative in x direction.

To impose the periodic boundary conditions at $x = 0$ and $x = 1$ (i.e. $j = 0$ and $j = N-1$), we start by noticing that

$$\psi_{i,-1} = \psi_{i,N-1}, \ \psi_{i,N} = \psi_{i,0} \tag{8}$$

Eq. 8 only considers the very edge of the boundary, but is already sufficient for performing the 5-point discretization:

$$\begin{aligned}(\nabla^2\psi)_{i,0} &= \frac{\psi_{i+1,0} + \psi_{i,1} - 4\psi_{i,0} + \psi_{i-1,0} + \psi_{i,N-1}}{h^2} \\ (\nabla^2\psi)_{i,N-1} &= \frac{\psi_{i+1,N-1} + \psi_{i,0} - 4\psi_{i,N-1} + \psi_{i-1,N-1} + \psi_{i,N-2}}{h^2}\end{aligned} \tag{9}$$

Eq. 9 hold for $i = 1, 2, \ldots, N-2$. Up to here, we have the discretized formulation for all the nodes in the field, and we are ready to go about solving the equation!

Using Eqs. 6, 7 and 8, we can construct the matrix $\boldsymbol{A}$ for the linear system. Usually, $\boldsymbol{A}$ is very large ($N^2 \times N^2$) and sparse (with only a few diagnals with nonzero numbers). Therefore, we construct sparse matrix to save computational power. Fig. 3 shows an illustration of the matrix $\boldsymbol{A}$, where 7 nonzero diagnals need to be constructed. Table. I shows the compositions of the 7 nonzero diagnals.

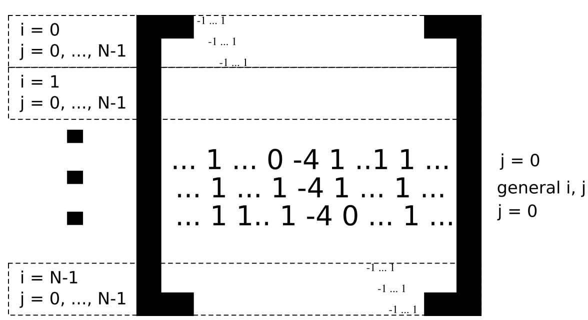

FIG. 3. An illustration of the linear system.

Now we have everything we need to solve for the function ψ in Eq. 1, or for the $\boldsymbol{x}$ in the formulation in Eq. 5. We test the method by constructing a vorticity field $\boldsymbol{\omega}$ and solve for ψ. Figure 4 shows the flow fields corresponding to a random and a sinusoidal vorticity fields.

Supplemental Content

[1] Implementation in Jupyter notebook

Name	Position	Composition
`e_3`	$-N$	$[1] \times (N-1)N + [-1] \times N$
`e_2`	$-N+1$	$[0] + ([0] \times (N-1) + [1]) \times (N-2) + [0] \times N$
`e_1`	-1	$[0] \times (N-1) + ([0] + [1] \times (N-1)) \times (N-2) + [0] \times N$
`e`	0	$[-1] \times N + [-4] \times (N-2)N + [1] \times N$
`e1`	1	$[0] \times N + ([1] \times (N-1) + [0]) \times (N-2) + [0] \times (N-1)$
`e2`	$N-1$	$[0] \times N + ([1] + [0] \times (N-1)) \times (N-2) + [0]$
`e3`	N	$[1] \times (N-1)N$

TABLE I. Nonzero diagnals in $\boldsymbol{A}$.

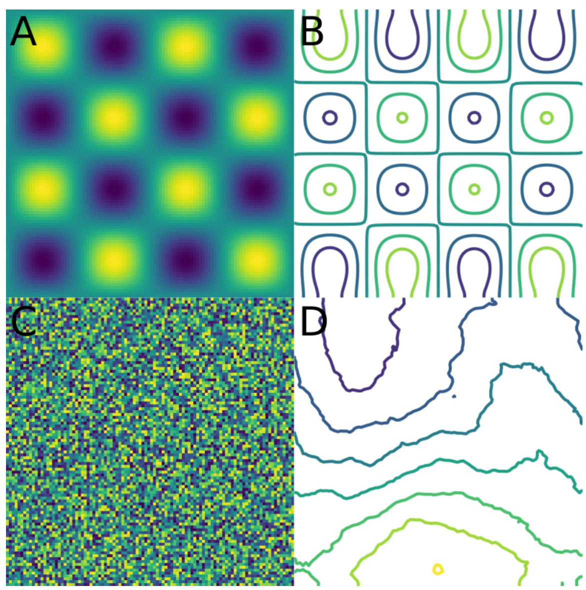

FIG. 4. Examle solutions of the Poisson equation.