An Introduction to Finite Difference Methods

A Tour Through Partial Differential Equations

Manuel A. Diaz

InspiraSTEM Internship

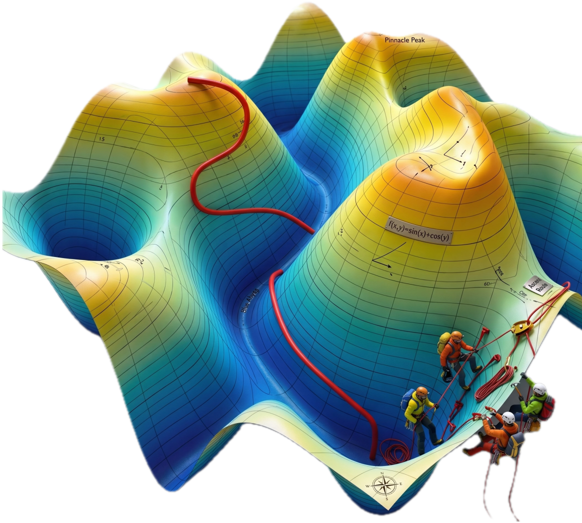
June 13, 2026

Outline

Motivation

PDEs are fundamental in modeling physical phenomena across science and engineering.

- **Transport** of pollutants in the atmosphere and oceans (advection).
- **Diffusion** in solids (diffusion).
- **Electrostatic potential** in a region (Poisson).
- **Fluid flow, wave propagation, and many more**.



NOTES:

- PDEs are fundamental in modeling physical phenomena across science and engineering.
- Analytical solutions are rare, especially for complex geometries and nonlinear problems.
- Numerical methods, like finite difference methods (FDM), provide a powerful tool to approximate solutions to PDEs.
- FDM is conceptually straightforward, making it an excellent starting point for learning numerical PDEs.
- Understanding FDM lays the groundwork for more advanced techniques like finite element and spectral methods (See for example an advanced introduction for CFD methods Pawar and San 2019).

Motivation: Incompressible Navier–Stokes

Following Lorena Barba's 12-steps to Navier-Stokes[1], we are interested in solving:

The Incompressible Navier–Stokes Equations

$$\nabla \cdot \mathbf{u} = 0. \qquad \text{(Continuity)}$$

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{1}{\rho}\nabla p + \nu \Delta \mathbf{u} + \mathbf{f}, \qquad \text{(Momentum)}$$

Where $\mathbf{u}$ is the velocity field, p the scalar pressure, ρ density, ν kinematic viscosity and $\mathbf{f}$ external forcing.

Here, the key features are:

- $(\mathbf{u} \cdot \nabla)\mathbf{u}$ is the nonlinear advection term,
- $-\frac{1}{\rho}\nabla p$ is the pressure gradient term $\Rightarrow$ enforces $\nabla \cdot \mathbf{u} = 0$, and
- $\nu \Delta \mathbf{u}$ is the viscous diffusion term.

[1] Barba and Forsyth 2018.

NOTES:

The incompressible Navier–Stokes equations are a cornerstone of fluid dynamics, modeling the motion of fluids in various contexts. They are nonlinear, coupled PDEs that require careful numerical treatment, especially when

- the pressure gradient is the term responsible forenforcing incompressibility (divergence-free velocity field), and
- the nonlinear advection term can lead to complex flow phenomena like turbulence.

We will build up to this problem by first understanding simpler PDEs that share these key features.

Outline

Unified form of Time-Marching PDEs

A compact and linear formulation that covers advection, diffusion and Poisson problems:

Semi-explicit formulation

$$\frac{\partial u}{\partial t} = \mathcal{L}u + f(\mathbf{x}, t),$$

where $\mathcal{L}$ is a linear spatial operator and f is a source term modeling external forcing or heat sources.

- Advection: $\mathcal{L}u := -\mathbf{c} \cdot \nabla u$ (transport, unsteady when $\partial_t u \neq 0$).
- Heat (diffusion): $\mathcal{L}u := \alpha \Delta u$ (parabolic, time-dependent).
- Burgers Equation: $\boldsymbol{\mathcal{L}u} := -\boldsymbol{u} \cdot \nabla \boldsymbol{u} + \alpha \Delta \boldsymbol{u}$ (unsteady nonlinear advection).

NOTES:

- Writing problems in operator form highlights linearity, separates spatial and temporal discretization, and makes it straightforward to build a unified finite-difference framework.
- At this point we are interested in understanding the numerical solution of
  - Unsteady Advection Equation: (notebooks 5 and 6)

$$\partial_t u + c_x \partial_x u + c_y \partial_y u = 0$$

  - Unsteady Heat Equation: (notebook 7)

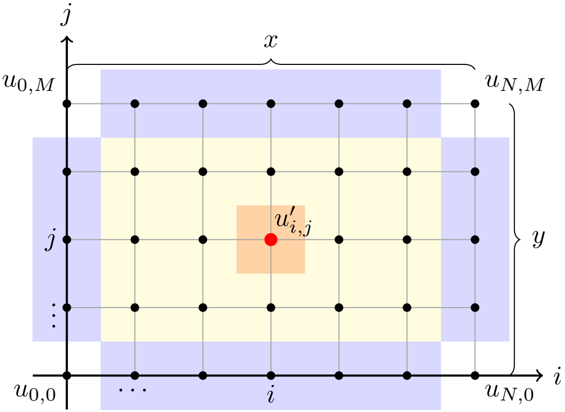
$$\partial_t u - \alpha \left(\partial_{xx} u + \partial_{yy} u \right) = 0$$

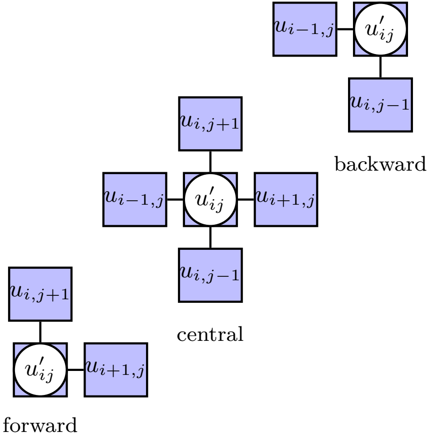
  - Burgers Equation for $\boldsymbol{u} : (u, v)$: (notebook 8)

$$\partial_t \boldsymbol{u} + u \partial_x \boldsymbol{u} + v \partial_y \boldsymbol{u} = \alpha \left(\partial_{xx} \boldsymbol{u} + \partial_{yy} \boldsymbol{u} \right)$$

- Appropriate initial and boundary conditions complete each problem.

The same principles for 1D finite difference methods apply for 2D, but we need to be careful with the boundary conditons:





NOTES:

When extending finite difference methods to 2D, we need to consider the grid structure, how to implement boundary conditions correctly, and the resulting linear systems that arise from discretization. Structured grids are easier to implement but may not be suitable for complex geometries, while unstructured grids offer more flexibility at the cost of increased complexity in implementation. Properly handling boundary conditions is essential for ensuring the accuracy and stability of the numerical solution. Finally, the larger linear systems that arise in 2D require efficient solvers to obtain solutions in a reasonable time frame.

Outline

Discretization of the Incompressible Navier–Stokes equation

Incompressible Navier–Stokes equations

$$\nabla \cdot \mathbf{u} = 0. \qquad \text{(Continuity)}$$

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{1}{\rho}\nabla p + \nu \Delta \mathbf{u} + \mathbf{f}, \qquad \text{(Momentum)}$$

In 2D, splitting into components:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0,$$

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) + f_x,$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) + f_y,$$

The key problem: **Where Does Pressure Come From?**

The key problem: Where Does Pressure Come From?

- Unlike the momentum equations, there's no time derivative for pressure.
- Pressure acts as a Lagrange multiplier that enforces the divergence-free constraint.
- Lorena Barba Barba and Forsyth 2018 uses a pressure Poisson equation (PPE) derived by taking the divergence of the momentum equation and invoking continuity.

Namely, apply the divergence operator to the momentum equation:

$$\nabla \cdot \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} \right) = \nabla \cdot \left(-\frac{1}{\rho} \nabla p + \nu \Delta \mathbf{u} + \mathbf{f} \right),$$

use the distributive property of linear operators and apply $\nabla \cdot \mathbf{u} = 0$.

However, we immediatly notice an inconsistency: discretely,the time derivative and spatial derivatives cannot be interchanged. As a result a compromise is needed: we assume that $\nabla \cdot \mathbf{u}$ is the variable that we are explicitly solving!

Deriving the Pressure Poisson Equation I

- In the contiuum limit $(\Delta t, \Delta x, \Delta y \to 0)$, take the $\nabla \cdot$ of the momentum equation, namely:

$$\nabla \cdot \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} \right) = \nabla \cdot \left(-\frac{1}{\rho}\nabla p + \nu \Delta \mathbf{u} + \mathbf{f} \right),$$

distribute the divergence operator and use $\nabla \cdot \mathbf{u} = 0$:

$$\frac{\partial}{\partial t}(\cancel{\nabla \cdot \mathbf{u}}) + \nabla \cdot [(\mathbf{u} \cdot \nabla)\mathbf{u}] = -\frac{1}{\rho}\Delta p + \nu \Delta (\cancel{\nabla \cdot \mathbf{u}}) + \nabla \cdot \mathbf{f}.$$

we get:

$$\nabla \cdot [(\mathbf{u} \cdot \nabla)\mathbf{u}] = -\frac{1}{\rho}\Delta p + \nabla \cdot \mathbf{f}.$$

So that the Poisson equation for pressure is:

$$\Delta p = \rho \left[\nabla \cdot [(\mathbf{u} \cdot \nabla)\mathbf{u}] - \nabla \cdot \mathbf{f} \right].$$

NOTES:

Expand the advection-divergence source term in 2D ($\mathbf{u} = (u, v)$):

$$
\begin{aligned}
\nabla \cdot [(\mathbf{u} \cdot \nabla)\mathbf{u}] &= \begin{bmatrix} \partial_x \\ \partial_y \end{bmatrix} \cdot \left[\begin{bmatrix} u\partial_x, & v\partial_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \right] \\
&= \partial_x(u\partial_x u + v\partial_y u) + \partial_y(u\partial_x v + v\partial_y v) \\
&= u_x^2 + uu_{xx} + v_x u_y + vu_{xy} + u_y v_x + uu_{xy} + v_y^2 + vv_{yy} \\
&= u_x^2 + 2v_x u_y + v_y^2 + u(u_{xx} + v_{xy}) + v(v_{yy} + u_{xy}) \\
&= u_x^2 + 2v_x u_y + v_y^2 + u\partial_x(u_x + v_y) + v\partial_y(v_y + u_x) \\
&= u_x^2 + 2v_x u_y + v_y^2 + (\mathbf{u} \cdot \nabla)(\nabla \cdot \mathbf{u}) \\
&\overset{\nabla \cdot \mathbf{u} = 0}{=} u_x^2 + 2u_y u_x + v_y^2.
\end{aligned}
$$

This is why in the 12-steps to Navier-Stokes, the convevtion term is expanded as

$$
\nabla \cdot [(\mathbf{u} \cdot \nabla)\mathbf{u}] = \left(\frac{\partial u}{\partial x}\right)^2 + 2\frac{\partial u}{\partial y}\frac{\partial v}{\partial x} + \left(\frac{\partial v}{\partial y}\right)^2.
$$

Deriving the Pressure Poisson Equation II

- Discretely, $(\Delta t, \Delta x, \Delta y \neq 0)$. Retake

$$\frac{\partial}{\partial t}(\nabla \cdot \mathbf{u}) + \nabla \cdot [(\mathbf{u} \cdot \nabla)\mathbf{u}] = -\frac{1}{\rho}\Delta p + \nu\Delta(\nabla \cdot \mathbf{u}) + \nabla \cdot \mathbf{f}.$$

We can interchange spatial derivatives but not with time derivatives, so that:

$$\frac{1}{\Delta t}\left((\nabla \cdot \mathbf{u})^{n+1} - (\nabla \cdot \mathbf{u})^{n}\right) + \nabla \cdot [(\mathbf{u} \cdot \nabla)\mathbf{u}] = -\frac{1}{\rho}\Delta p + \nabla \cdot \mathbf{f}.$$

However, we request that $(\nabla \cdot \mathbf{u})^{n+1} = 0$, so that:

$$-\frac{1}{\Delta t}(\nabla \cdot \mathbf{u})^{n} + \nabla \cdot [(\mathbf{u} \cdot \nabla)\mathbf{u}] = -\frac{1}{\rho}\Delta p + \nabla \cdot \mathbf{f}.$$

Rearranging for the pressure Poisson equation:

$$\Delta p = \rho\left[\frac{(\nabla \cdot \mathbf{u})^{n}}{\Delta t} - \nabla \cdot [(\mathbf{u} \cdot \nabla)\mathbf{u}] - \nabla \cdot \mathbf{f}\right].$$

The pressure Poisson equation can be solved using various methods (Pawar and San 2019):

- Direct solve
- Iterative methods (Gauss-Seidel, Jacobi, SOR, etc.)
- Multigrid methods (V-cycle, W-cycle, etc.)
- Preconditioned conjugate gradient method (PCG)

In this introduction, we will focus on two methods:

- The Direct method https://www.youtube.com/watch?v=YotrBNLFen0
- Jacobi method https://www.youtube.com/watch?v=YotrBNLFen0

The pressure Poisson equation is the last key component to write the fully discrete incompressible Navier–Stokes solver. Happy coding!

References I

Barba, Lorena A and Gilbert F Forsyth (2018). "CFD Python: the 12 steps to Navier-Stokes equations". In: *Journal of Open Source Education* 2.16, p. 21.

Pawar, Suraj and Omer San (2019). "CFD Julia: A learning module structuring an introductory course on computational fluid dynamics". In: *Fluids* 4.3, p. 159.