MTH 602 Final Project Statement

Physics Informed Neural Networks (PINNs) Approach to Volume Tracking in Multiphase Flows

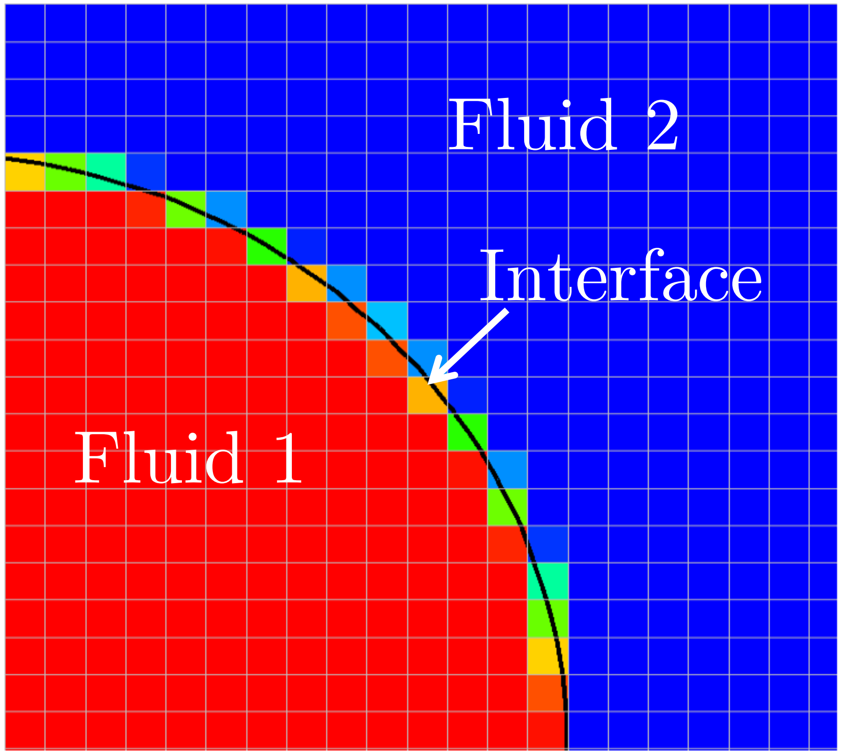
S. M. Mahfuzul Hasan

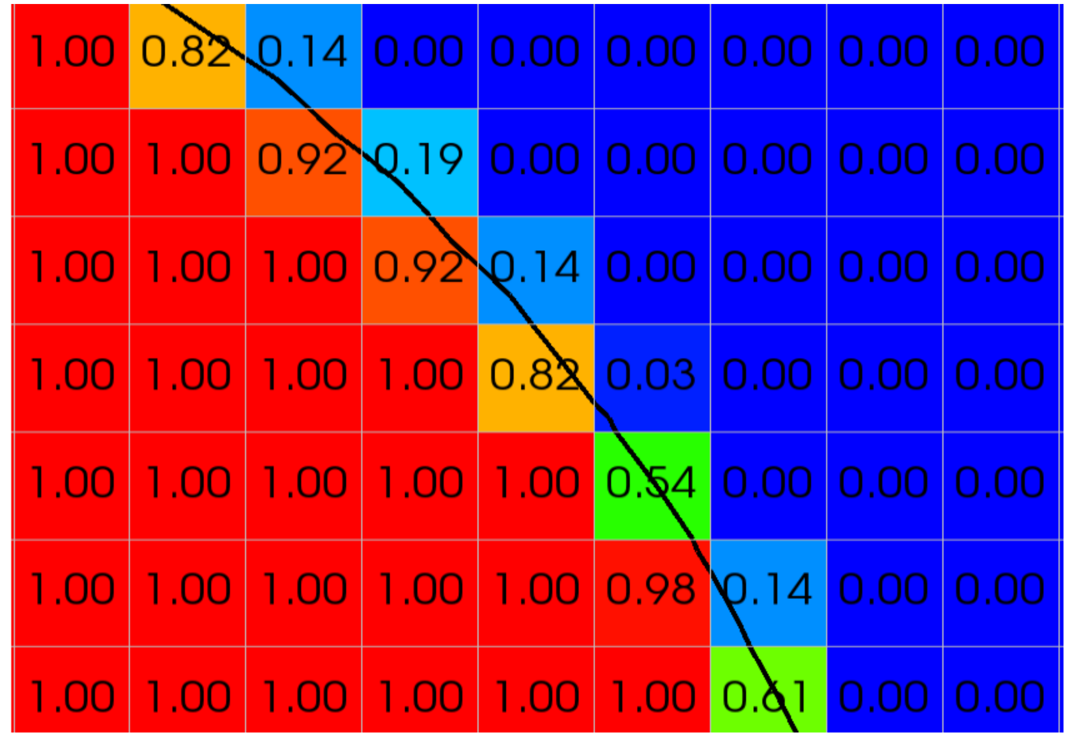
October 6, 2025

UMass | Dartmouth

A very short background

What is multiphase flow? What are the governing equations?





$$f(\vec{x}) = \begin{cases} 1, & \vec{x} \in \text{Fluid 1}, \\ 0, & \vec{x} \in \text{Fluid 2}. \end{cases}$$

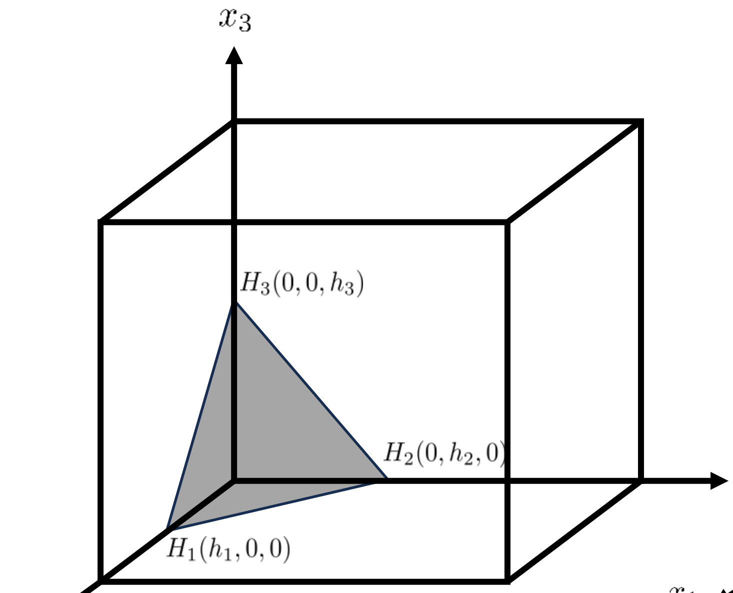
$$F = \frac{1}{V} \int_V f \, dV$$

Advection equation for scalar transport

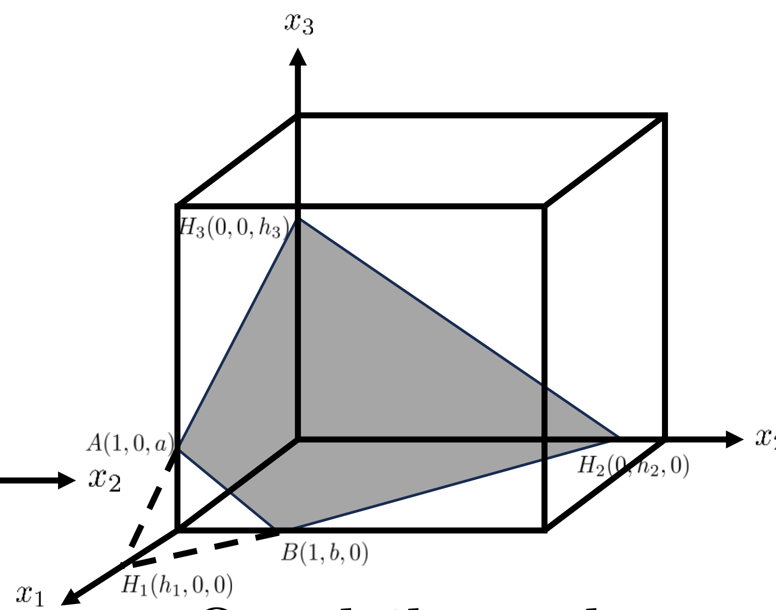
$$\frac{\partial f}{\partial t} + \vec{u} \cdot \nabla f = 0$$

Piecewise linear interface reconstruction

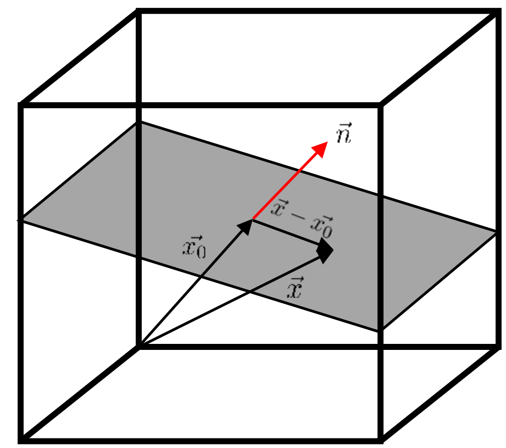
$$\vec{n} \cdot \vec{x} = \alpha \qquad \vec{n} = \nabla F$$



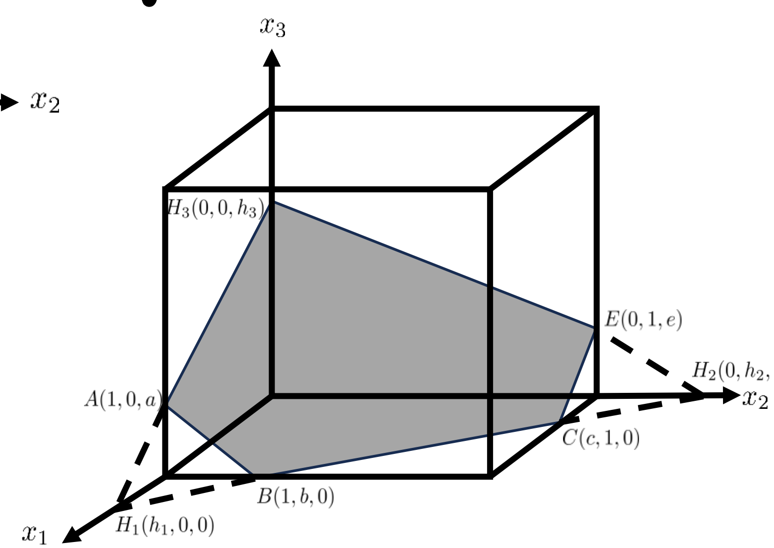
x_1 Triangular



Quadrilateral 1

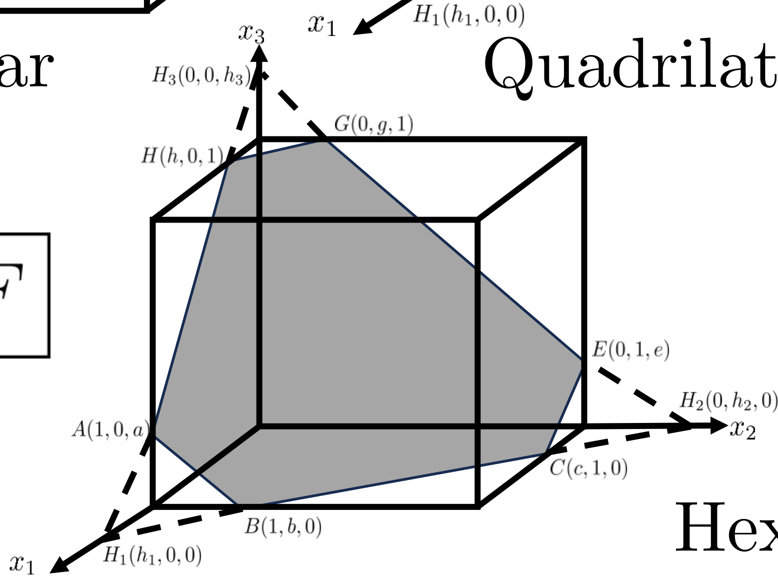


Quadrilateral 2



Pentagonal

$$\boxed{V(\alpha) = F}$$

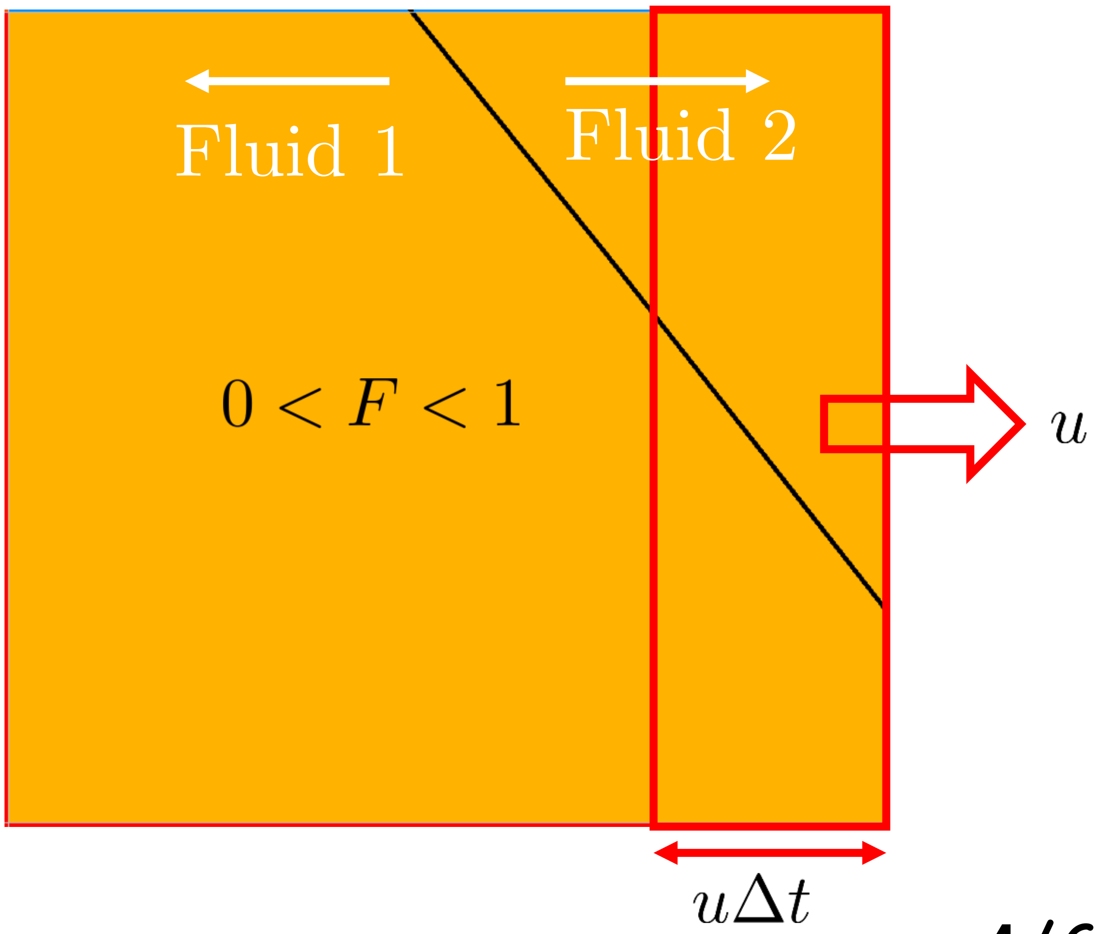


Hexagonal

Flux calculation

- After interface reconstruction, flux calculation is done by splitting the fluxes along each dimensional direction.
- After doing flux calculation in In all the directions, total flux for a cell is computed.

$$F = \frac{Total\ flux}{V}$$

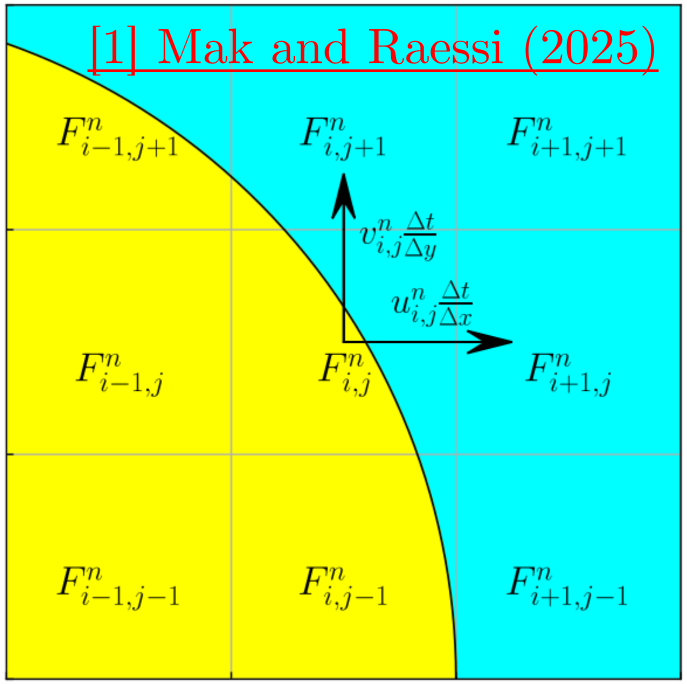


Alternate strategy

Machine Learning

- A data-driven method was applied to bypass those two steps to directly predict F^{n+1} from F^n using a **grid search** method to find an optimal NN configuration using MATLAB's Deep Learning Toolbox around 3×3 stencils of the targeted cell.

$$F_{i,j}^{n+1} = g\Big(F_{i-1,j+1}^n,\ F_{i,j+1}^n,\ F_{i+1,j+1}^n,\ F_{i-1,j}^n,\ F_{i,j}^n,\ F_{i+1,j}^n,\ F_{i-1,j-1}^n,\ F_{i,j-1}^n,\ F_{i+1,j-1}^n,\ u_{i,j}^n\frac{\Delta t}{\Delta x},\ v_{i,j}^n\frac{\Delta t}{\Delta y}\Big)$$



- Though this approach provided 4x speedup, it could not give the accuracy of the traditional Volume-of-Fluid (VoF) method.

Alternate strategy

Implementation of PINNs

Aim:

- explore the possibility of attaining the accuracy and robustness of the tradition VoF for test cases like translation and rotation.
- keep the speedup superior to traditional VoF.

- Step-1:
  - Data pre-training by just using ML model to learn geometric behavior (Minimizing $\mathrm{MSE}_{\mathrm{data}}$).
- Step-2:
  - Physics incorporation by adding loss functions for PDE, volume conservation and boundedness of the solution.
- Step-3:
  - Dropping the data loss function, and only train on physics losses.