Fluid Flow in Nature and Engineering

- Fluids are everywhere: air, water, blood, oil...
- Applications in aerospace, civil, biomedical, and mechanical engineering
- Fluid motion governed by the Navier–Stokes equations:

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u}\right) = -\nabla p + \mu \nabla^2 \mathbf{u}, \quad \nabla \cdot \mathbf{u} = 0$$

From Continuum to Discrete Models

- Solving Navier–Stokes equations directly is complex
- Traditional CFD: finite elements, finite volumes, spectral methods
- Alternative: **mesoscopic** models like the Lattice Boltzmann Method (LBM)
- Based on kinetic theory: simulate evolution of particle distribution functions instead of macroscopic variables

What is the Idea Behind the Boltzmann Equation?

- Instead of tracking fluid velocity and pressure directly, we study how groups of particles (molecules) move and collide
- These are **fictitious particles** — statistical representatives of many real molecules
- We use a function $f(\mathbf{x}, \boldsymbol{\xi}, t)$ that describes the density of particles at position $\mathbf{x}$, with microscopic velocity $\boldsymbol{\xi}$, at time t
- This approach comes from kinetic theory and bridges microscopic and macroscopic scales

The Continuous Boltzmann Equation

- Describes the evolution of the particle distribution function:

$$\frac{\partial f}{\partial t} + \boldsymbol{\xi} \cdot \nabla f = \Omega(f)$$

- Where:
  - f: particle distribution function
  - $\boldsymbol{\xi}$: microscopic velocity
  - $\Omega(f)$: collision term — describes how particles interact and return to equilibrium
- In LBM, we simplify Ω using the BGK model:

$$\Omega(f) = -\frac{1}{\tau}(f - f^{\text{eq}})$$

The Lattice Boltzmann Equation (LBE)

- After discretizing velocity space, we get a finite set of distributions f_i along directions $\mathbf{c}_i$
- The full Lattice Boltzmann Equation reads:

$$f_i(\mathbf{x} + \mathbf{c}_i \Delta t, t + \Delta t) = f_i(\mathbf{x}, t) - \frac{\Delta t}{\tau} \left(f_i(\mathbf{x}, t) - f_i^{\mathrm{eq}}(\mathbf{x}, t) \right)$$

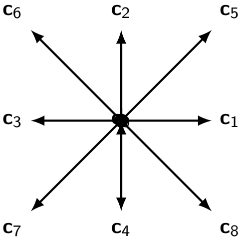
- This equation combines **collision** and **streaming** into one step
- At each point $\mathbf{x}$, and for each direction i, the distributions evolve in time

Discretization in LBM: Steps and Variables

- Discretize:
  - **Time**: steps of size Δt
  - **Space**: nodes on a regular lattice
  - **Velocity**: finite set of discrete directions $\{\mathbf{c}_i\}$
- Each node stores distribution functions $f_i(\mathbf{x}, t)$, one per direction $\mathbf{c}_i$
- At each time step:
  1. **Collision** (local relaxation):
  $$f_i^*(\mathbf{x}, t) = f_i(\mathbf{x}, t) - \frac{1}{\tau}(f_i - f_i^{\mathrm{eq}})$$
  2. **Streaming** (shift to neighbor node):
  $$f_i(\mathbf{x} + \mathbf{c}_i, t + 1) = f_i^*(\mathbf{x}, t)$$
- Macroscopic quantities:
$$\rho(\mathbf{x}, t) = \sum_i f_i(\mathbf{x}, t)$$
$$\mathbf{u}(\mathbf{x}, t) = \frac{1}{\rho} \sum_i f_i(\mathbf{x}, t)\mathbf{c}_i$$

Popular Lattice Models: D2Q9

- Two-dimensional, 9-velocity model (D2Q9)
- 1 rest direction, 4 axial, 4 diagonal



Useful Video Resources on LBM

- Introduction to Lattice Boltzmann winter 23/24 - YouTube
- Introduction to Lattice Boltzmann Method
- Introduction to Lattice Boltzmann Method @ NASA Glenn (2013)
- LBM Fluid Simulation in Python with JAX — van Karman Vortex Street
- Simple Lattice-Boltzmann Simulator in Python — CFD for Beginners
- Create your own LBM Simulation Using Python — Physics Simulations
- Lattice Boltzmann Method - Lid Driven
- Of Foxes, Attackers... and the Lattice Boltzmann Method

Tutorial

Krüger, Timm, et al. "The lattice Boltzmann method." Springer International Publishing 10.978-3 (2017) "The lattice Boltzmann method The book has its own GitHub page Github

Tasks for the nearest Weeks

- **Task 1: Literature review** (figure out the topic)
- **Task 2: Python simulations**
    - Channel (Poiseuille) flow
    - Flow around a cylinders (vortex shedding)