Computational fluid dynamics for physicists and engineers

Ruchi Mishra
Nicolaus Copernicus Astronomical Center

Inverted CERN school of Computing (iCSC), CERN, 16-18 March 2020
Content

What is Computational Fluid Dynamics?
Applications in different fields
How does CFD work?
Conservation equations
Discretization method
Riemann Problem
Example: Shock tube Problem
Example: KORAL Simulation
Fluid flows are governed by a system of partial differential equations (PDEs) which describe the conservation of mass, momentum and energy.

Computational fluid dynamics (CFD) solves these PDEs by replacing them with algebraic equations.
CFD APPLICATION IN DIFFERENT FIELDS

Aerospace

Automobile

Biomedical

Astrophysical Simulation

Building architect

CFD Application

Thermal analysis of instruments

Numerical weather prediction
HOW DOES CFD WORK?

FLUID PROBLEM

Physics of Fluid

Navier–Stokes equations

Discretization

GRID

Analysis

Simulation Results

Computer Programme
EQUATIONS GOVERNING FLUID FLOW

Mass conservation equation / continuity equation

\[ - \frac{\partial \rho}{\partial t} = \frac{\partial (\rho u)}{\partial x} \]

Momentum conservation equation

Navier-Stokes equation

\[ \rho \left( \frac{Dv}{Dt} \right) = -\nabla P + \mu \nabla^2 v + F_x \]

mass X acceleration  Pressure gradient  Internal Forces  External Forces

Energy conservation equation

\[ \frac{\partial e}{\partial t} + v \cdot \nabla e = -\frac{P}{\rho} \nabla \cdot v \]

Equation of state

\[ p = (\gamma - 1) \rho e \]
FROM INPUT TO OUTPUT

Initial conditions
Boundary conditions

Unknown Physical Quantities

\[ \begin{align*}
\rho & \quad \text{Density} \\
\nu & \quad \text{Velocity} \\
p & \quad \text{Pressure} \\
\varepsilon & \quad \text{Energy}
\end{align*} \]
DIFFERENT APPROACHES TO MODEL FLUID IN CFD

Grid based hydrodynamics
- Solves the fluid dynamics equations by calculating the flux of conserved quantities through adjacent cell boundaries

Eulerian Approach

Lagrangian Approach

Smooth particle hydrodynamics (SPH)
- Calculates the properties on each particle by averaging over its nearest neighbour
- Satisfies mass conservation without extra computation as the particles themselves represent mass

IN THIS LECTURE WE ARE GOING TO COVER ONLY THE GRID BASED HYDRODYNAMICS
We replace our equations by simpler ones.

Original Navier-Stokes equation reduces to Euler equations.

\[ \frac{\partial \rho}{\partial t} + \frac{\partial (\rho v)}{\partial x} = 0 \]
\[ \frac{\partial (\rho v)}{\partial t} + \frac{\partial (\rho v^2 + P)}{\partial x} = 0 \]
\[ \frac{\partial e}{\partial t} + \frac{\partial ((e + P)v)}{\partial x} = 0 \]

\[ \frac{\partial}{\partial t} \left[ \text{Conserved quantity} \right] + \frac{\partial}{\partial x} \left[ \text{Flux} \right] = 0 \]

How to solve it numerically?
HOW TO SOLVE THEM NUMERICALLY?

Unknown Physical Quantities

\[ \rho \quad v \quad p \quad e \]

Density \quad Velocity \quad Pressure \quad Energy

\[ \frac{\partial}{\partial t} \left[ \text{Conserved quantity} \right] + \frac{\partial}{\partial x} \left[ \text{Flux} \right] = 0 \]

Physical domain: space \((x,y,z)\) and time \(t\)
Physical quantities: \(\rho v p e\)

Since we will solve equations numerically, we have to discretize
1) Physical domain
2) Physical quantities (aka equation discretization)
DOMAIN DISCRETIZATION

MESH GRID - Division of a continuous geometric space into discrete geometric cells

Model of flow around cylinder using **cartesian grid**.

Example of triangle mesh representing a dolphin

Structured curvilinear grid

Unstructured curvilinear grid

Hybrid grid
THE DISCRETIZED PHYSICAL DOMAIN

Grid generation

- A simple method of placing points in the domain
- Each point is labeled using $i$ for spatial discretization and $n$ for time discretization
- The spacing can be of variable size
DISCRETIZATION OF PHYSICAL QUANTITIES

Equation discretization

Backward difference

\[ \frac{\partial a_i^n}{\partial x} \approx \frac{a_i^n - a_{i-1}^n}{\Delta x} \]

Forward difference

\[ \frac{\partial a_i^n}{\partial x} \approx \frac{a_{i+1}^n - a_i^n}{\Delta x} \]

Central difference

\[ \frac{\partial a_i^n}{\partial x} \approx \frac{a_{i+1}^n - a_{i-1}^n}{2\Delta x} \]
SPECIFYING INPUT THROUGH INITIAL AND BOUNDARY CONDITIONS

Some of the boundary conditions used in CFD

1- inlet condition
2- symmetric condition
3- periodic boundary condition
4- reflective boundary condition
5- outlet condition
**Boundary Conditions**

**Periodic Boundary Condition**

**Symmetric Boundary Condition**

*Ghost cells* are used here to extend the grid beyond physical boundary to accommodate boundary condition.
STORING DATA IN GRIDS

Finite-difference grid: Data is stored at grid edges

Cell-centered finite-difference grid: Data is stored at cell centers

Finite-volume grid—the average value of the function is stored within each zone.
EQUATION DISCRETIZATION USING
FINITE DIFFERENCE METHOD

\[
\frac{\Delta x}{\Delta t} \quad \frac{\partial U}{\partial t} = - \frac{\partial F}{\partial x}
\]

Finite difference grid with ghost cell at each end

\[
\frac{U_{i}^{n+1} - U_{i}^{n}}{\Delta t} = \frac{-1}{\Delta x} \left[ F_{i}^{n} - F_{i-1}^{n} \right]
\]

Time evolution

\[
U_{i}^{n+1} = \frac{-\Delta t}{\Delta x} \left[ F_{i}^{n} - F_{i-1}^{n} \right] + U_{i}^{n}
\]
FINITE VOLUME METHOD

_fluxes are calculated at cell edges \((i \pm 1/2)\)
EQUATION DISCRETORIZATION USING FINITE DIFFERENCE METHOD

Finite volume grid with two ghost cells at both ends

\[ \frac{\partial U}{\partial t} = - \frac{\partial F}{\partial x} \]

\[ \frac{U_{i}^{n+1} - U_{i}^{n}}{\Delta t} = \frac{-1}{\Delta x} \left[ F_{i+\frac{1}{2}}^{n+\frac{1}{2}} - F_{i-\frac{1}{2}}^{n-\frac{1}{2}} \right] \]

Time evolution

\[ U_{i}^{n+1} = \frac{-\Delta t}{\Delta x} \left[ F_{i+\frac{1}{2}}^{n+\frac{1}{2}} - F_{i-\frac{1}{2}}^{n-\frac{1}{2}} \right] + U_{i}^{n} \]
At the interface there will be a jump. How do we calculate flux at the interface?

For flux evaluation at half time, we need information of state U at half time

\[
[F_{i+\frac{1}{2}}^{n+\frac{1}{2}}] = f(U_{i+\frac{1}{2}}^{n+\frac{1}{2}})
\]
Two states separated by a discontinuity.

This is called a Riemann problem.

Solution to Riemann problem results in single state at interface.

\[ U_{i+\frac{1}{2}}^{n+\frac{1}{2}} = R(U_{i+\frac{1}{2}}^{n+\frac{1}{2},L}, U_{i+\frac{1}{2}}^{n+\frac{1}{2},R}) \]
The exact solution of Riemann problem at every interface is very expensive!

Use approximate Riemann solver instead!
Solution is computed from two wave speeds $S_L$ and $S_R$

If we have algorithm to track these wave speeds an approximate intercell flux can be calculated from it.

$$F_{i+\frac{1}{2}}^{\text{hll}} = \begin{cases} F_L & \text{if } 0 \leq S_L \\ S_R F_L - S_L F_R + S_L S_R (U_R - U_L) \over S_R - S_L & \text{if } S_L \leq 0 \leq S_R \\ F_R & \text{if } 0 \geq S_L \end{cases}$$
We have a new problem while discretizing the time.

When we discretize the time, the step must be less than the time it takes for the information to propagate across a single zone.

This is called **CFL** (Courant–Friedrichs–Lewy) condition

\[ \Delta t \leq \frac{\Delta x}{v} \]

\[ C = \frac{v \Delta t}{\Delta x} \]

\[ C \leq 1 \]

Necessary condition for stability
SHOCK TUBE OR SOD PROBLEM IN 1D

Gary A. Sod (1978)

Commonly used problem to test accuracy of CFD codes using Riemann Solver.

Initial Condition

\[ \begin{align*}
    v_L &= 0 \\
    \rho_L &= 1 \\
    P_L &= 1 \\
    v_R &= 0 \\
    \rho_R &= 0.1 \\
    P_R &= 0.125
\end{align*} \]

1. The fluid (gas) is initially at rest separated by a wall.

2. The sudden breakdown of the wall generates a high-speed flow resulting in a shock wave, which propagates to the right.
THE ALGORITHM

EQUATION IN CONSERVED FORM

\[ \frac{\partial \rho}{\partial t} + \frac{\partial (\rho v)}{\partial x} = 0 \]

\[ \frac{\partial (\rho v)}{\partial t} + \frac{\partial (\rho v^2 + P)}{\partial x} = 0 \]

\[ \frac{\partial e}{\partial t} + \frac{\partial ((e + P)v)}{\partial x} = 0 \]

Conserved quantities:

\[
\begin{pmatrix}
\rho \\
\rho v \\
\rho e + \frac{1}{2} \rho v^2
\end{pmatrix}
\]

Fluxes:

\[
\begin{pmatrix}
\rho v \\
\rho v^2 + P \\
(\rho e + \frac{1}{2} \rho v^2 + P)v
\end{pmatrix}
\]

Physical quantities:

\[
\begin{pmatrix}
\rho \\
v \\
e
\end{pmatrix}
\]

Discretize physical domain by setting grid cells

Set initial conditions for physical quantities

Reconstruct
Calculation left step \( p_L \) and right state of physical quantity \( p_R \)

Calculate flux (from \( p_R \) and \( p_L \) , \( F_L \) and \( F_R \) are calculated)

Convert physical quantity to conserved quantity (P to U)

Solve the Reimann Problem

Check CFL condition

UPDATE IN TIME

Convert conserved quantity to physical quantity (U to P)
INITIAL CONDITION

\[
t = 0
\]
TIME EVOLUTION

t = 0.1
TIME EVOLUTION

\[ t = 0.2 \]
TIME EVOLUTION

\[ t = 0.3 \]
Why do we do simulation in Astrophysics?

SIMULATION IN ASTROPHYSICS

Simulation enables us to build a model of a system.

It allows us to do virtual experiments to understand how this system reacts to a range of conditions and assumptions.
General Relativistic Radiative Magnetohydrodynamics

Using KORAL
TAKE HOME MESSAGE:

● CFD enables us to predict fluid flow
● The fundamentals of CFD lie in solving the set of partial differential equations that describe the fluid flow (e.g. \textit{Navier-Stokes equation})
● In Eulerian grid based approach, the physical domain is discretized into large number of cells
● In each of these cells, Navier-Stokes equations can be rewritten as algebraic equations
● These equations are then solved numerically
● At the end we get the complete description of flow throughout the domain
THANK YOU