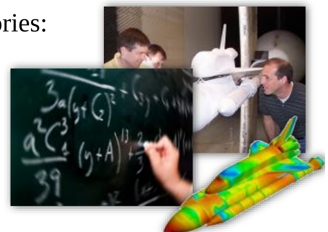
Introduction to Computational Fluid Dynamics (CFD)

School of Mechanical Engineering
Collage of Engineering
University of Tehran

Branches of Fluid Dynamics

Three boards categories:

- Experimental
- Theoretical
- Computational



Crucial to know all three:

- Each has their advantages and disadvantages.
- Require validation and verification.

Experimental Fluid Dynamics

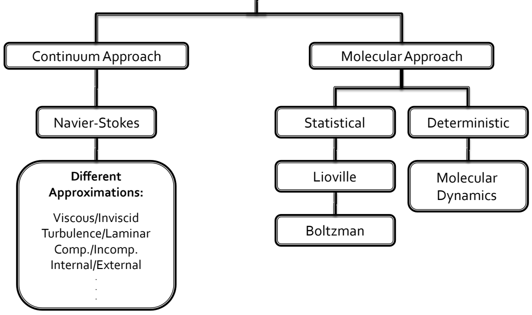
1. Only way to obtain some data.
2. Long set-up time.
3. Rapid change of some parameters.
   e.g. angle of attack
4. Often difficult to match all parameters.
   i.e. Mach NO, Reynolds NO, Froude NO, etc
5. Usually limited spatial and temporal data.
6. Need to estimate experimental errors and ensure repeatability.

Everyone believes the experimental results, except the person who took the data!!

Theoretical Fluid Dynamics

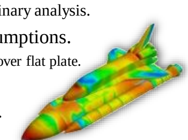
1. Time consuming and tedious.
2. Fewer and fewer people trained in theory.
3. Usually significant simplifications required.
   e.g. linear theory, simple geometry, etc
4. Must ensure that simplification are valid and that no mistakes are made in derivation.

Fluid Mathematical Description



Computational Fluid Dynamics

1. Fastest growing branch of fluid dynamics.
   Computers are getting faster with more memory.
   Algorithms and grids improve to handle more complicated flows
2. Can provide tremendous amount of data.
   Can solve for unsteady flow over complete aircraft.
   Can extract reduced order models for multi-disciplinary analysis.
3. Need to understand/reduce numeric's/assumptions.
   Can't always accurately predict laminar boundary layer over flat plate.
4. CFD is not Color Fluid Dynamics.
   Use of color should only be to better help illustrate.



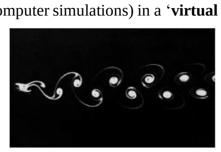
Nobody believes CFD results, except the person who ran the code!!

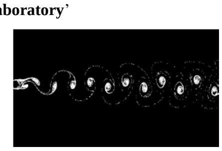
What is CFD?

Computational Fluid Dynamics (CFD) provides a qualitative (and sometimes even quantitative) prediction of fluid flows by means of:

- mathematical modeling (partial differential equations)
- numerical methods (discretization and solution techniques)
- software tools (solvers, pre and post processing utilities)

CFD enables scientists and engineers to perform '**numerical experiments**' (i.e. computer simulations) in a '**virtual flow laboratory**'





What is CFD?

Fluid (gas and liquid) flows are governed by Partial Differential Equations (PDE) represent conservation laws for the **mass**, **momentum**, and **energy**.

Computational Fluid Dynamics (CFD) is the **art of replacing such PDE systems by a set of algebraic equations for a given computer hardware.**

Experiment vs. Simulations

As a rule, CFD does not replace the measurements completely but the amount of experimentation and the overall cost can be significantly reduced.

Experiments	Simulations
expensive	Cheap(er)
Slow	Fast(er)
Sequential	Parallel
Single-purpose	Multiple-purpose

The results of a CFD simulation are never 100% reliable because

- The input data may involve too much guessing or imprecision
- The mathematical model of the problem at hand may be inadequate
- The accuracy of the results is limited by the available computing power

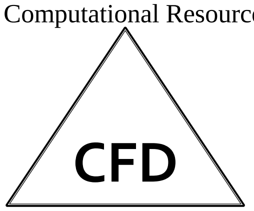
Experiment vs. Simulations

CFD gives an insight into flow patterns that are difficult, expensive or impossible to study using traditional (experimental) techniques

Experiments	Simulations
for one quantity at a time	for all desired quantities
at a limited number of points and time instants	with high resolution in space and time
for a laboratory-scale model	for the actual flow domain
for a limited range of problems and operating conditions	for virtually any problem and realistic operating conditions

Simulation Procedure in CFD

Computational Resource

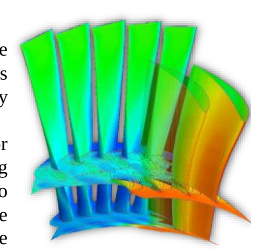


PDE Knowledge Numerical Algorithm

CFD Goals

Modern CFD has two goals:

1. To simulate flows where experimental measurement is physically challenging or very expensive.
2. To create insight into the behavior of complex flows by leveraging the detailed numerical data to generate more informative visualizations than testing alone could provide.

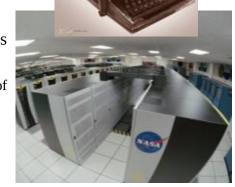


Numerical Algorithms

- Finite Difference Methods (FDM)
- Finite Volume Methods (FVM)
- Finite Element Methods (FEM)
- Boundary Element Method
- Spectral Method

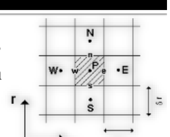
Computational Resource in CFD

- Early computational methods developed before digital computers.
  Calculations done by hand and often in teams.
- Computers were invented in 1930's.
- Digital computers are available in 1960's.
- Computer "cost" has dropped 5 to 6 orders of magnitude.
  PC's today have the power of supercomputer's of 1980's.
- CFD homework assignment today could have been PhD thesis in the 1950's to 1960's.



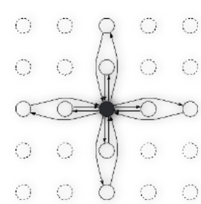
Finite Volume

- In the finite volume method, volume integrals in a partial differential equation that contain a divergence term are converted to surface integrals, using the divergence theorem.
- These terms are then evaluated as fluxes at the surfaces of each finite volume. Because the flux entering a given volume is identical to that leaving the adjacent volume, these methods are conservative.
- It is easily formulated to allow for unstructured meshes.
- Often times involve interpolation methods to derive flux interface values.

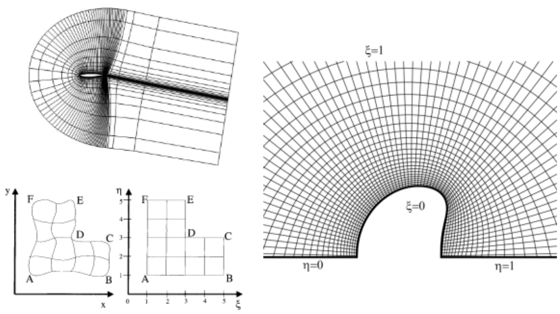


Finite Difference

- The approximation of derivatives by finite differences plays a central role in finite difference methods for the numerical solution of differential equations.
- First apply to ordinary differential equations.
- The simplest to understand.



Structured Grid



Grid Generation

STRUCTURED GRID

Generation Methods:
- Algebraic Methods
- PDE's

Using in:
- FDM
- FVM

UNSTRUCTURED GRID

Generation Methods:
- Delaunay
- Advancing Front

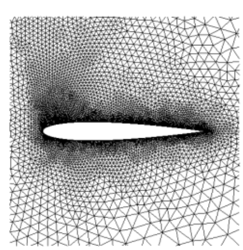
Using in:
- FVM
- FEM

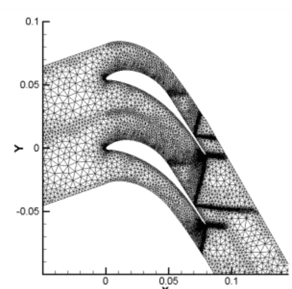
Why Use CFD?

Numerical simulations of fluid flow (will) enable

- **Architects** to design comfortable and safe living environments
- **Designers of vehicles** to improve the aerodynamic characteristics
- **Chemical engineers** to maximize the yield from their equipment
- **Petroleum engineers** to devise optimal oil recovery strategies
- **Surgeons** to cure arterial diseases (computational hemodynamics)
- **Meteorologists** to forecast the weather and warn of natural disasters
- **Safety exper**ts to reduce health risks from radiation and other hazards
- **Military organizations** to develop weapons and estimate the damage
- CFD practitioners to make big bucks by selling colorful pictures :-)

Unstructured Grid





CFD Analysis Process

1. **Problem statement**	information about the flow
2. **Mathematical model**	IBVP = PDE + IC + BC
3. **Mesh generation**	nodes/cells, time instants
4. **Space discretization**	coupled ODE/DAE systems
5. **Time discretization**	algebraic system Ax = b
6. **Iterative solver**	discrete function values
7. **CFD software**	implementation, debugging
8. **Simulation run**	parameters, stopping criteria
9. **Postprocessing**	Visualization, analysis of data
10. **Verification**	model validation/adjustment

How Does CFD Make Predictions?

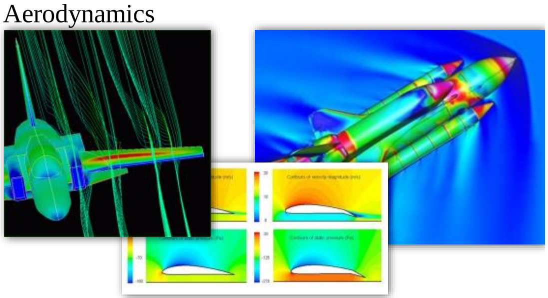
CFD uses a computer to solve the mathematical equations for the problem at hand. The main components of a **CFD design cycle** are as follows:

- **the human being** (**analyst**) who states the problem to be solved
- **scientific knowledge** (**models, methods**) expressed mathematically
- **the computer code** (**software**) which embodies this knowledge and provides detailed instructions (algorithms) for
- **the computer hardware** which performs the actual calculations
- **the human being** who inspects and interprets the simulation results

CFD is a highly interdisciplinary research area which lies at the interface of **physics**, **applied mathematics**, and **computer science.**

CFD Applications

Aerodynamics



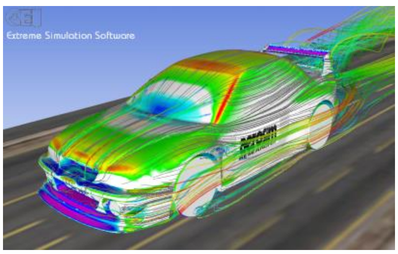
CFD Simulations

The computing times for a flow simulation depend on:

- the choice of numerical algorithms and data structures
- Linear algebra tools, stopping criteria for iterative solvers
- discretization parameters (mesh quality, mesh size, time step)
- cost per time step and convergence rates for outer iterations
- programming language (most CFD codes are written in Fortran)
- many other things (hardware, parallelization etc.)

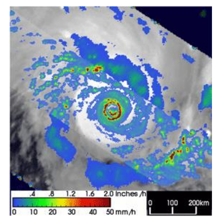
CFD applications

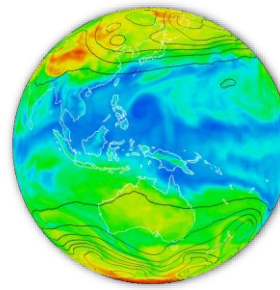
Automobiles



CFD Applications

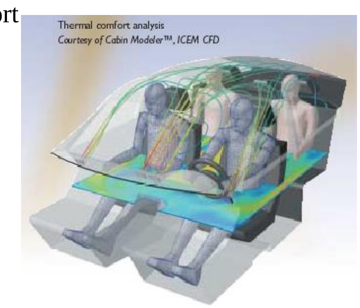
Weather Prediction





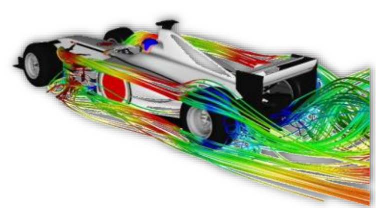
CFD Applications

Interior Comfort



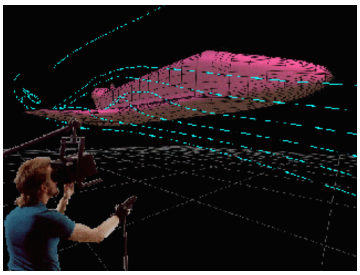
CFD Applications

Automobiles



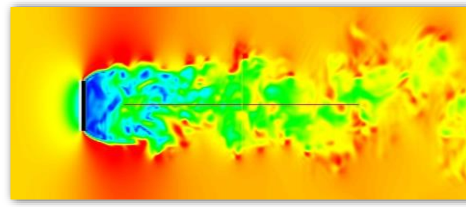
CFD Applications

NASA Ames Virtual Wind Tunnel



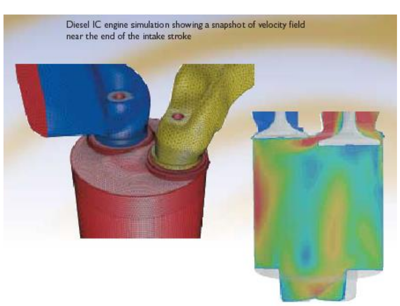
CFD Applications

Turbulence Modeling



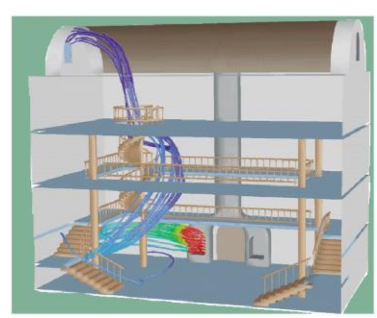
CFD Applications

Engine Combustion



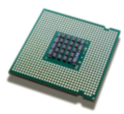
CFD Applications

HVAC



Hardware

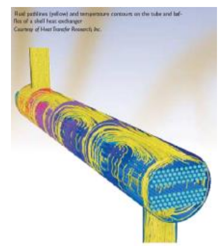
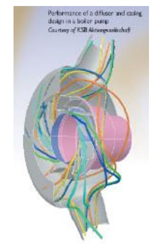
- Processing Unit
- Memory
- Connection





CFD Applications

Process Engineering

Processing Trend

Year	Supercomputer	Calculations
1960	UNIVAC LARC	250 KFLOPS
1976	IBM 7030	1.2 MFLOPS
1983	Cray-1	250 MFLOPS
1985	Cray-2	3.9 GFLOPS
1990	NEC SX-3/44R	23.2 GFLOPS
1993	Intel Paragon XP/S 140	143.40 GFLOPS
2002	NEC Earth Simulator	35.86 TFLOPS
2005	IBM Blue Gene/L	280.6 TFLOPS
2008	IBM Roadrunner	1.105 PFLOPS
2009	Cray Jaguar	1.759 PFLOPS

FLOPS

FLOPS

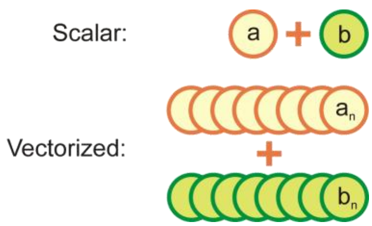
Floating Point Operation per Second

Moore's Law

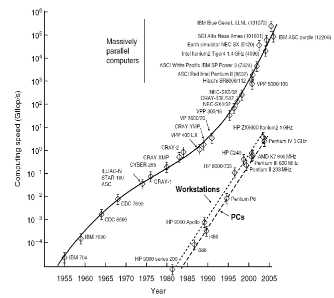
Processors Doubled *approximately* every 18 months

Vector Processing

- Vectorized Computing



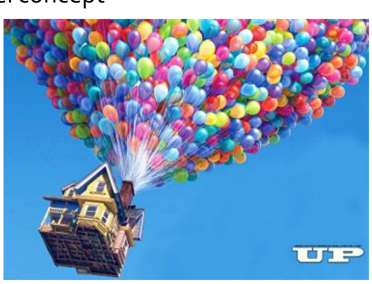
Supper Computer Development



Introduction to Computational Fluid Dynamics (CFD) 37

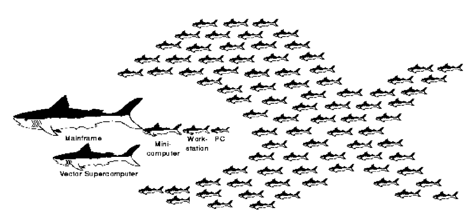
Parallel Concept

- Parallel concept



Parallel Concept

- Parallel concept



Structure of the Course — Computational Fluid Dynamics

1. **Introduction**
2. **Partial Differential Equations**
3. **Finite Difference Calculus**

   Mid Term #1
4. **FDM for Parabolic PDE's**
5. **Stability of FDM**
6. **FDM for Elliptic PDE's**

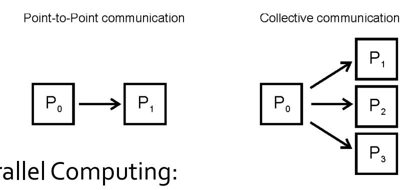
   Midterm #2
7. **Finite Volume Method**
8. **Hyperbolic**
9. **Hyperbolic**

   Final Exam

Quizes	(5/20)
Projects	(1/20, 1/20)
HW's	(2/20)
Midterm #1	(3/20)
Midterm #2	(3/20)
Final Exam	(5/20)

Total: 20/20

Parallel Processing



- Parallel Computing:
  - Domain Decomposition
  - MPI Library
  - Point-to-Point Communication
  - Collective Communication

Thanks!