



Computational Fluid Dynamics
<http://www.nd.edu/~gtryggva/CFD-Course/>

Computational Fluid Dynamics

Lecture 1
January 18, 2017

Grétar Tryggvason



Computational Fluid Dynamics
Introduction

What is CFD
Beginning of CFD
Course goals
Course content
Schedule
Homework and projects

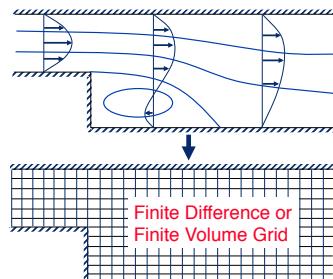


Computational Fluid Dynamics
Introduction

What is Computational Fluid Dynamics (CFD)?

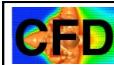
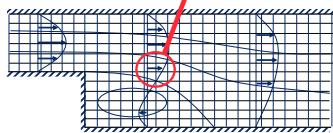
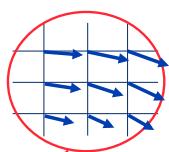


Computational Fluid Dynamics
Introduction



Computational Fluid Dynamics
Introduction

Grid must be sufficiently fine to resolve the flow



Computational Fluid Dynamics
Introduction

Using CFD to solve a problem:

Preparing the data (preprocessing):
Setting up the problem, determining flow parameters and material data and generating a grid.

Solving the problem

Analyzing the results (postprocessing):
Visualizing the data and estimating accuracy. Computing forces and other quantities of interest.



Computational Fluid Dynamics Introduction

CFD is used by many different people for many different things

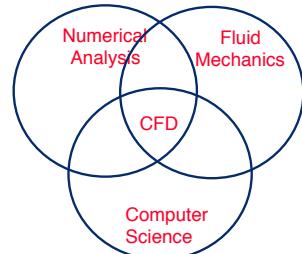
Industrial problems: The goal is generally to obtain data (quantitative and qualitative) that can be used in design of devices or processes. Often it is necessary to use "subgrid" or closure models for unresolved processes

Academic problems: The goal is to understand the physical aspects of the process, often the goal is to construct "subgrid" or closure models for industrial computations



Computational Fluid Dynamics Introduction

CFD is an interdisciplinary topic



Computational Fluid Dynamics Introduction

Many website contain information about fluid dynamics and computational fluid dynamics specifically. Those include

NASA site with CFD images
<http://www.nas.nasa.gov/SC08/images.html>

CFD Online: An extensive collection of information but not always very informative
<http://www.cfd-online.com/>

eFluids.com is a monitored site with a large number of fluid mechanics material
<http://www.e-fluids.com/>



Computational Fluid Dynamics Introduction

Computational Fluid Dynamics:

The definition of CFD depends on the whom you ask. Here I suggest:

Computing multidimensional flow governed by the Navier Stokes or Euler equations with the goal of obtaining both a physical insight into the dynamics and quantitative predictions.

Thus, development of methods for general PDEs, 1D flow and method development for its own sake is excluded

If we define CFD this way, identifying the person and the place originating CFD becomes straight-forward. The convergence of powerful computers, important problems and creative people took place at Los Alamos around the middle of the 20th Century



Computational Fluid Dynamics

Beginning of CFD



Computational Fluid Dynamics Beginning of CFD

Theory is good, but mankind has always needed numbers

Methods to produce numbers thus go back a long way: Computing Pi; Newton–Raphson method; Gauss and Jacobi iterations; Rayleigh–Ritz method; etc.

Books and papers published in the early part of the 20th Century include:

L. F. Richardson, The approximate arithmetical solution by finite differences of physical problems involving differential equations with an application to stresses in a masonry dam, Phil. Trans. Royal Soc. London, A, 210 (1910), pp. 307–357.

E. T. Whittaker and G. Robinson, The Calculus of Observations. A Treatise on Numerical Mathematics, London, 1924.

R. Courant, K. O. Friedrichs, and H. Lewy, U'ber die partiellen Differenzengleichungen der mathematischen Physik, Mathematische Annalen 100 (1928), pp. 32–74.

J. B. Scarborough, Numerical Mathematical Analysis, Baltimore, 1930.

http://history.siam.org/pdf/nahist_Benzi.pdf



Computational Fluid Dynamics Beginning of CFD

Iterative methods for the solution of finite difference approximation to elliptic equation (Richardson, 1910; Lieberman, 1918; Southwell, 1940; Frankel, 1950)

Stability analysis of hyperbolic PDE's (Courant, Friedrichs, and Lewy, 1928)

Point vortex simulations of shear layer roll-up Rosenhead, 1931)

von Neumann method of stability analysis of finite difference approximations (O'Brien, Hyman, and Kaplan, 1950)

Conservation law form of the governing equations of fluid dynamics (Lax, 1954)

Alternating Direction Implicit (ADI) methods for elliptic PDE's (Peaceman and Rachford, 1955; Douglas and Rachford, 1956)

Lax-Wendroff method (Lax and Wendroff, 1960)



Computational Fluid Dynamics Beginning of CFD

CFD at Los Alamos

The MANIAC at Los Alamos had already stimulated considerable interest in numerical solutions at the Laboratory. However, CFD in the modern sense started with the formation of the T3 group. Early development in the Group focused on:

Compressible flows: Particles in Cells (1955): Eulerian –Lagrangian method where particles move through a fixed grid

Low-speed (incompressible) flows

Vorticity streamfunction for homogeneous

(mostly 2D) flows (1963)

Marker and Cell for free surface and

Multiphase flows in primitive variables (1965)



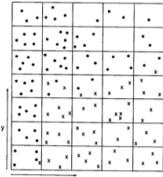
All-speed flows—ICE (1968, 1971)

Arbitrary–Lagrangian–Eulerian (ALE) methods

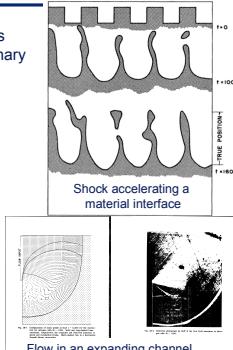


Computational Fluid Dynamics Beginning of CFD

In the Particle-In-Cell (PIC) method for compressible multi-material flows, particles with constant mass move through a stationary Eulerian grid



From Harlow, F. H., Dickman, D. O., Harris, D. E., and Martin, R. E., "Two Dimensional Hydrodynamic Calculations," Los Alamos Scientific Laboratory report LA-2301 (September 1959).



Computational Fluid Dynamics Beginning of CFD

Incompressible flows—Vorticity-Streamfunction Method



Computations of the development of a von Karman vortex street behind a blunt body by the method developed by J. Fromm. Time goes from left to right showing the wake becoming unstable.

From: J. E. Fromm and F. H. Harlow, Numerical Solution of the Problem of Vortex Street Development: *Phys. Fluids* 6 (1963), 975.



Computational Fluid Dynamics Beginning of CFD

Incompressible flows—the MAC Method

Primitive variables (velocity and pressure) on a staggered grid

The velocity is updated using splitting where we first ignore pressure and then solve a pressure equation with the divergence of the predicted velocity as a source term

Marker particles used to track the different fluids

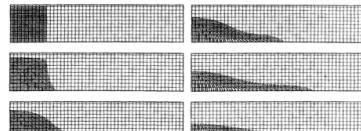
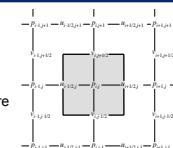


Fig. 5. Configuration of marker particles for the liquid–gas interface at times $t = 0, 0.5, 1.0, 1.5, 2.0, 2.5$. There were four computational cells for each square of the grid shown here.



The dam breaking problem simulated by the MAC method, assuming a free surface. From F. H. Harlow and J. E. Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface. *Phys. Fluids*, 8: 2182–2189, 1965.



Computational Fluid Dynamics Beginning of CFD

Multiphase flows with the MAC method

F. H. Harlow and J. E. Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface. *Phys. Fluids*, 8: 2182–2189, 1965. Introduces the MAC method and shows two sample computations of the so-called dam breaking problem.

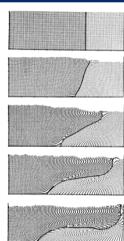
F. H. Harlow and J. E. Welch. Numerical study of large-amplitude free-surface motions. *Phys. Fluids*, 9:842–851, 1966. Examines the Rayleigh-Taylor problem.

F. H. Harlow and J. P. Shannon. The splash of a liquid drop. *J. Appl. Phys.*, 38:3855–3868, 1967. Studies the splash when a droplet hits a liquid surface.

B. J. Daly. Numerical study of the effect of surface tension on interface instability. *Phys. Fluids*, 12:1340–1354, 1969. Two-fluid Rayleigh-Taylor instability.

B. J. Daly and J. E. Prahl. Numerical study of density-current surges. *Phys. Fluids*, 11:15–30, 1968. Two fluid density currents.

B. J. Daly. A technique for including surface tension effects in hydrodynamic calculations. *J. Comput. Phys.*, 4:97–117, 1969. Rayleigh-Taylor instability with surface tension.



A two fluid Rayleigh-Taylor instability (left) and viscous density current (above) computed by the two-fluid MAC method

Computational Fluid Dynamics Beginning of CFD

Early Publicity:

SCIENTIFIC AMERICAN

F.H. Harlow, J.P. Shannon, Distortion of a splashing liquid drop, *Science* 157 (August) (1967) 547–550.

F.H. Harlow, J.P. Shannon, J.E. Welch, Liquid waves by computer, *Science* 149 (September) (1965) 1092–1093.

Fig. 1. Sequence of frames, produced by a computer, depicting the escape of water from a spherical cavity. The sequence shows the initial cavity, followed by a series of frames showing the cavity expanding and then collapsing, creating a complex pattern of liquid motion.

Computational Fluid Dynamics Beginning of CFD

From: Computing & Computers: Weapons Simulation Leads to the Computer Era, by Francis H. Harlow and N. Metropolis

"While most of the scientific and technological world maintained a disdainful distaste (or at best an amused curiosity) for computing, the power of the stored-program computers came rapidly into its own at Los Alamos during the decade after the War."

Another type of opposition occurred in our interactions with editors of professional journals, and with scientists and engineers at various universities and industrial laboratories. One of the things we discovered in the 1950s and early 1960s was that there was a lot of suspicion about numerical techniques. Computers and the solutions you could calculate were said to be the playthings of rich laboratories. You couldn't learn very much unless you did studies analytically."

From: *Journal of Computational Physics* 195 (2004) 414–433
Review: *Fluid dynamics in Group T-3 Los Alamos National Laboratory (LA-UR-03-3852)*, by Francis H. Harlow

Computational Fluid Dynamics Beginning of CFD

CFD goes mainstream:

Early studies using the MAC method:

- R. K.-C. Chan and R. L. Street, A computer study of finite-amplitude water waves, *J. Comput. Phys.*, 6:68–94, 1970.
- G. B. Foote, A numerical method for studying liquid drop behavior: simple oscillations, *J. Comput. Phys.*, 11:507–530, 1973.
- G. B. Foote, The water drop rebound problem: dynamics of collision, *J. Atmos. Sci.*, 32:390–402, 1975.
- R. B. Chapman and M. S. Plesset, Nonlinear effects in the collapse of a nearly spherical cavity in a liquid, *Trans. ASME, J. Basic Eng.*, 94:142, 1972.
- T. M. Mitchell and F. H. Hammitt, Asymmetric cavitation bubble collapse, *Trans ASME, J. Fluids Eng.*, 95:29, 1973.

Beginning of commercial CFD—Imperial College (Spalding)
Computations in the Aerospace Industry (Jameson and others)
And others!

Computational Fluid Dynamics Beginning of CFD

The Legacy—as seen from the Laboratory

While the origin of computing at Los Alamos is widely acknowledged, the profound impact the T3 Group had on CFD is more unevenly recognized. Within the Laboratory there is a strong appreciation of the remarkable progress made in the T3 group and how creative the work was.

Computational Fluid Dynamics Beginning of CFD

Even though the focus in the T3 group at Los Alamos was originally on compressible flows, it is probably in computations of incompressible flows, particularly multifluid flows where the techniques devised have had the most direct impact.

While the combination of particles and fixed meshes is no longer used in mainstream CFD, a large number of other innovations are, such as:

- Explicit solution of the Navier-Stokes equations by a projection method on structured meshes
- Staggered Mesh for fluid mechanics (and also as the Yee mesh in FDTD simulations of electromagnetic waves)
- When particles are used on fixed meshes they are now usually connected to represent interfaces and imbedded boundaries

Computational Fluid Dynamics Beginning of CFD

Available online at www.sciencedirect.com
SCIENCE @ DIRECT[®]
JOURNAL OF COMPUTATIONAL PHYSICS
[www.elsevier.com/locate/jcp](http://www elsevier.com/locate/jcp)

Review
Fluid dynamics in Group T-3
Los Alamos National Laboratory^{*}
(LA-UR-03-3852)
Francis H. Harlow^{*}

Los Alamos National Laboratory, MS-B276, Los Alamos, NM 87544, USA
Received 12 June 2003; revised in revised form 25 August 2003; accepted 3 September 2003

The development of computer fluid dynamics has been closely associated with the evolution of large high-speed computers. At first the principal incentive was to produce numerical techniques for solving problems related to national defense. Soon, however, it was recognized that numerous additional scientific and engineering applications could be accomplished by means of modified techniques that extended considerably the capabilities of the early procedures. This paper describes some of this work at The Los Alamos National Laboratory, where many types of problems were solved for the first time with the newly emerging sequence of numerical capabilities. The discussions focus principally on those with which the author has been directly involved.



Computational Fluid Dynamics Beginning of CFD

Although the methods developed at Los Alamos were put to use in solving practical problems and picked up by researchers outside the Laboratory, considerable development took place that does appear to be only indirectly motivated by it. Those development include:

- Panel methods for aerodynamic computations (Hess and Smith, 1966)
- Specialized techniques for free surface potential and Stokes flows (1976) and vortex methods
- Spectral methods for DNS of turbulent flows (late 70's, 80's)
- Monotonic advection schemes for compressible flows (late 70's, 80's)
- Steady state solutions (SIMPLE, aeronautical applications, etc)
- Commercial CFD

However, with outside interest in Multifluid simulations (improved VOF, level sets, front tracking) around 1990, the legacy became obvious



Computational Fluid Dynamics

Commercial Codes



Computational Fluid Dynamics Commercial Codes

CHAM (Concentration Heat And Momentum) founded in 1974 by Prof. Brian Spalding was the first provider of general-purpose CFD software. The original PHOENICS appeared in 1981.

The first version of the FLUENT code was launched in October 1983 by Creare Inc. Fluent Inc. was established in 1988.

STAR-CD's roots go back to the foundation of Computational Dynamics in 1987 by Prof. David Gosman,

The original codes were relatively primitive, hard to use, and not very accurate.



Computational Fluid Dynamics Commercial Codes

What to expect and when to use commercial package:

The current generation of CFD packages generally is capable of producing accurate solutions of simple flows. The codes are, however, designed to be able to handle very complex geometries and complex industrial problems. When used with care by a knowledgeable user CFD codes are an enormously valuable design tool.

Commercial CFD codes are rarely useful for state-of-the-art research due to accuracy limitations, the limited access that the user has to the solution methodology, and the limited opportunities to change the code if needed



Computational Fluid Dynamics Commercial Codes

Major current players include

Ansys (Fluent and other codes)
<http://www.ansys.com/>

Siemens: (starCD and other codes)
<http://mdx.plm.automation.siemens.com>

Others

CHAM: <http://www.cham.co.uk/>
CFD2000: <http://www.adaptive-research.com/>
COMSOL: <https://www.comsol.com/cfd-module>
OpenFOAM: <http://www.openfoam.com>



Computational Fluid Dynamics

Computational Resources

Computational Fluid Dynamics has traditionally been one of the most demanding computational application. It has therefore been the driver for the development of the most powerful computers

The number of grid points (or control volumes) available determines the complexity of the problem that can be solved and the accuracy of the solution

CFD Computational Fluid Dynamics

Bits and Bytes:
 $64 \text{ bits} = 8 \text{ bytes} = \text{one number with } \sim 16 \text{ digits precision}$
 Memory requirement for 3-D calculations
 $100^3 = 10^6 \times 10 \text{ bytes} \times 10 \text{ numbers/node} = 0.1 \text{ GB}$
 $200^3 = 8 \times 10^6 \times 10 \text{ bytes} \times 10 \text{ numbers/node} \sim 1 \text{ GB}$
 $1000^3 = 10^9 \times 10 \text{ bytes} \times 10 \text{ numbers/node} \sim 100 \text{ GB}$

FLOPS (Floating-point operations per second)
 CRAY-1 (1976) - 133 Megaflops
 ASCI White (2000) - 12.28 Teraflops
 Cray XT5 Jaguar (ORNL 2009) 1.759 Petaflops
 Tianhe-1A (China 2010) 2.507 Petaflops

→ Increase by a million in a quarter century !!

CFD Computational Fluid Dynamics

Currently Largest!

11 PFLOP/s Simulations of Cloud Cavitation Collapse

Diego Rossinelli¹, Babak Hejazialhosseini¹, Panagiotis Hadjidoukas¹, Costas Bekas², Alessandro Curioni², Adam Bertsch², Scott Futral², Steffen J. Schmidt², Nikolaus A. Adams¹ and Petros Koumoutsakos¹

We present unprecedented, high throughput simulations of cloud cavitation collapse on 1.6 million cores of Sequoia reaching 55% of its nominal peak performance, corresponding to **11 PFLOP/s**. The destructive power of cavitation reduces the lifetime of energy critical systems such as internal combustion engines and hydraulic turbines, yet it has been harnessed for water purification and kidney lithotripsy. The present two-phase flow simulations enable the quantitative prediction of cavitation using **13 trillion grid points** to resolve the collapse of 15,000 bubbles. We advance by one order of magnitude the current state-of-the-art in terms of time to solution, and by two orders the geometrical complexity of the flow. The software successfully addresses the challenges that hinder the effective solution of complex flows on contemporary supercomputers, such as limited memory bandwidth, I/O bandwidth and storage capacity. The present work redefines the frontier of high performance computing for fluid dynamics simulations.

CFD Computational Fluid Dynamics

Single processor computers
 Vector computers (80' s)
 Parallel computers (90' s)
 Shared Memory
 Distributed Memory
 Multi-core parallel nodes (00' s)
 More heterogeneous cores, including with GPUs etc. are likely to be the future

CFD Computational Fluid Dynamics
 World's fastest computers

For up-to-date information about the World's fastest computers, see:
<http://www.top500.org/>

As of this fall, the World's fastest computer, Jaguar at ORNL, has been upgraded to Titan. See:
[http://en.wikipedia.org/wiki/Titan_\(supercomputer\)](http://en.wikipedia.org/wiki/Titan_(supercomputer))



<http://en.wikipedia.org/wiki/File:Titan1.jpg>

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	National Supercomputing Center in Wuxi, China	Sunway TaihuLight - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway NRPC	10,649,600	93,014.6	125,435.9	15,371
2	National Super Computer Center in Guangzhou, China	Tianhe-2 [MilkyWay-2] - TH-IVB-FEP Cluster, Intel Xeon E5-2692 12C 2.200GHz, TH Express-2, Intel Xeon Phi 31S1P NUDT	3,120,000	33,862.7	54,902.4	17,808
3	DOE/SC/Oak Ridge National Laboratory United States	Titan - Cray XK7 , Opteron 6274 16C 2.200GHz, Cray Gemini interconnect, NVIDIA K20x Cray Inc.	560,640	17,590.0	27,112.5	8,209
4	DOE/NNSA/LLNL United States	Sequoia - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom IBM	1,572,864	17,173.2	20,132.7	7,890

<https://www.top500.org/lists/2016/11/>

CFD Computational Fluid Dynamics
<http://www.nd.edu/~gtryggva/CFD-Course/>

This Course

Grétar Tryggvason
 Spring 2017

CFD Computational Fluid Dynamics Instructor

Prof. Gretar Tryggvason
Ph.D. Brown University 1985 (Postdoc at Courant Inst. 84-85)

Professor of Mechanical Engineering
University of Michigan 1985-2000

Professor and Head, Mechanical Engineering
Worcester Polytechnic Institute 2000-2010

Viola D. Hank Professor of Aero & Mech Engrg
University of Notre Dame and Dept. Chair since 2010

Short term visiting/research positions: Courant, Caltech,
NASA Glen, University of Marseilles, University of Paris VI

Fellow of the APS, ASME and AAAS & ASME Fluids Award (2012)

Editor-in-chief of the Journal of Computational Physics
(2002-2015)

CFD Computational Fluid Dynamics This Course

Coarse Goals:
Learn how to solve the Navier-Stokes and Euler equations for engineering problems using both customized codes and a commercial code

Hear about various concepts to allow continuing studies of the literature.

Ways:
Detailed coverage of selected topics, such as: simple finite difference methods, accuracy, stability, etc.
Rapid coverage of other topics, such as: multigrid, monotone advection, unstructured grids.

CFD Computational Fluid Dynamics This Course

Background needed:

Undergraduate Numerical Analysis and Fluid Mechanics

Graduate Level Fluid Mechanics (can be taken concurrently).

Basic computer skills. We will use MATLAB for some of the homework.

CFD Computational Fluid Dynamics This Course

I. Introduction
A very brief introduction to how to solve the Navier-Stokes equations

II. Numerical solutions of PDEs
A relatively standard exposure to numerical solution of Partial Differential Equations with special emphasize on the Navier-Stokes equation

III. Introduction to advanced topics
Multiscale and multiphysics, achieving predictive simulations, large-scale coupled systems

IV. Introduction to Multiphase DNS
Resolving moving interfaces

CFD Computational Fluid Dynamics This Course

Grading
Four projects, homework, quizzes.

Collaborations
You are free to discuss the homework problems and the projects with your fellow students, but your solution should be your own work.

Reading Material
Lecture material along with references to recommended reading material will be available on the course webpage

Course homepage: <http://www.nd.edu/~gtryggva/CFD-Course/>

CFD Computational Fluid Dynamics Early Textbooks (Sample)

Ferziger, J. H. and Peric, M., *Computational Methods for Fluid Dynamics*, Springer, 1999.
Hirsch, C., *Numerical Computation of Internal and External Flows, I and II*, Wiley, 1988.
Tannehill, J. C., Anderson, D. A., and Pletcher, R. H., *Computational Fluid Mechanics and Heat Transfer*, 2nd ed., Taylor and Francis, 1997.
Wesseling, P., *Principles of Computational Fluid Dynamics*, Springer 2000
Ferziger, J. H., *Numerical Methods for Engineering Application*, Wiley, 1981
Peyret, R. and Taylor, T. D., *Computational Methods for Fluid Flow*, Springer, 1983.
Patankar, S. V., *Numerical Heat Transfer and Fluid Flow*, McGraw-Hill, 1980



Computational Fluid Dynamics
<http://www.nd.edu/~gtryggva/CFD-Course/>

Warm-up: One-Dimensional Conservation Equation

Grétar Tryggvason
Spring 2017



Computational Fluid Dynamics
Objectives:

Introduce the basic concepts needed to solve a partial differential equation using finite difference methods.

Discuss basic time integration methods, ordinary and partial differential equations, finite difference approximations, accuracy.

Show the implementation of numerical algorithms into actual computer codes.



Computational Fluid Dynamics
Outline

- Solving partial differential equations
 - Finite difference approximations
 - The linear advection-diffusion equation
 - Matlab code
 - Accuracy and error quantification
 - Stability
 - Consistency
 - Multidimensional problems
 - Steady state



Computational Fluid Dynamics

The Advection-Diffusion Equation



Computational Fluid Dynamics
Model Equations

We will use the model equation:

$$\frac{\partial f}{\partial t} + U \frac{\partial f}{\partial x} = D \frac{\partial^2 f}{\partial x^2}$$

to demonstrate how to solve a partial equation numerically.

Although this equation is much simpler than the full Navier Stokes equations, it has both an advection term and a diffusion term.

Before attempting to solve the equation, it is useful to understand how the analytical solution behaves.



Computational Fluid Dynamics
Model Equations

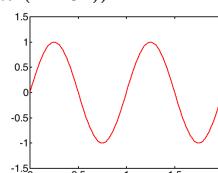
For initial conditions of the form:

$$f(x, t=0) = A \sin(2\pi kx)$$

It can be verified by direct substitution that the solution is given by:

$$f(x, t) = e^{-Dk^2 t} \sin(2\pi k(x - Ut))$$

which is a decaying traveling wave





Computational Fluid Dynamics Conservation equations

Most physical laws are based on CONSERVATION principles: In the absence of explicit sources or sinks, f is neither created nor destroyed.

Consider a simple one dimensional pipe of uniform diameter with a given velocity

$$f(t) \quad U(x) \longrightarrow \quad x \longrightarrow$$

Given f at the inlet as a function of time (as well as everywhere at time zero), how do we predict the $f(t,x)$?



Computational Fluid Dynamics Conservation equations

To predict the evolution of f everywhere in the pipe, we assume that f is conserved. Thus, for any section of the pipe:

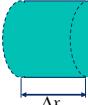
Amount of f in a control volume after a given time interval Δt	=	Amount of f in the control volume at the beginning of the time interval Δt	+	Amount of f that flows into the control volume during Δt	-	Amount of f that flows out of the control volume during Δt
--	---	--	---	--	---	--

$$f(t) \quad [\quad] \quad U(x) \longrightarrow \quad x \longrightarrow$$



Computational Fluid Dynamics Conservation equations

How much does the total amount of f in the Control Volume change during a short time Δt ? Restating the conservation law from the last slide, we have



Total $f(t+\Delta t)$ - total $f(t)$ = Total inflow of f - Total outflow of f

$$\text{Approximate: } f_{Total} = \int_x f dx \approx f_{av} \Delta x$$

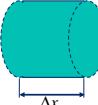
Denote the flow rate of f into the control volume by F then:

$$[f_{av}(t+\Delta t) - f_{av}(t)] \Delta x = \Delta t [F_{in} - F_{out}]$$



Computational Fluid Dynamics Conservation equations

We can also state the conservation principle in differential form. From last slide



Divide by $\Delta t \Delta x$

$$\frac{[f_{av}(t+\Delta t) - f_{av}(t)]}{\Delta t} = - \frac{[F_{out} - F_{in}]}{\Delta x}$$

or:

$$\frac{\Delta f_{av}}{\Delta t} = - \frac{\Delta F}{\Delta x}$$

Taking the limit, we get:

$$\frac{\partial f}{\partial t} + \frac{\partial F}{\partial x} = 0$$

F : Flux of f



Computational Fluid Dynamics Conservation equations

The general form of the one-dimensional conservation equation is:

$$\frac{\partial f}{\partial t} + \frac{\partial F}{\partial x} = 0$$

Taking the flux to be the sum of advective and diffusive fluxes:

$$F = Uf - D \frac{\partial f}{\partial x}$$

Gives the advection diffusion equation

$$\frac{\partial f}{\partial t} + U \frac{\partial f}{\partial x} = D \frac{\partial^2 f}{\partial x^2}$$



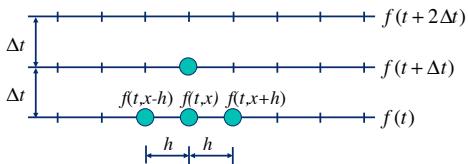
Computational Fluid Dynamics

Finite Difference Approximations of the Derivatives



Computational Fluid Dynamics Finite Difference Approximations

Derive a numerical approximation to the governing equation, replacing a relation between the derivatives by a relation between the discrete nodal values.



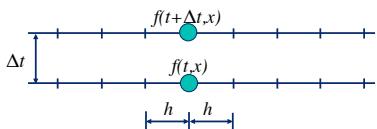
Computational Fluid Dynamics Finite Difference Approximations

The Time Derivative



Computational Fluid Dynamics Finite Difference Approximations

The Time Derivative is found using a FORWARD EULER method. The approximation can be found by using a Taylor series



Computational Fluid Dynamics Finite Difference Approximations

Time derivative

$$f(t + \Delta t) = f(t) + \frac{\partial f(t)}{\partial t} \Delta t + \frac{\partial^2 f(t)}{\partial t^2} \frac{\Delta t^2}{2} + \dots$$

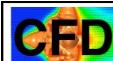
Solving this equation for the time derivative gives:

$$\frac{\partial f(t)}{\partial t} = \frac{f(t + \Delta t) - f(t)}{\Delta t} - \frac{\partial^2 f(t)}{\partial t^2} \frac{\Delta t}{2} + \dots$$



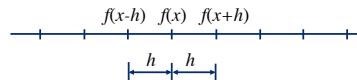
Computational Fluid Dynamics Finite Difference Approximations

The Spatial First Derivative



Computational Fluid Dynamics Finite Difference Approximations

When using FINITE DIFFERENCE approximations, the values of f are stored at discrete points,



The derivatives of the function are approximated using a Taylor series



Computational Fluid Dynamics Finite Difference Approximations

Start by expressing the value of $f(x+h)$ and $f(x-h)$ in terms of $f(x)$

$$f(x+h) = f(x) + \frac{\partial f(x)}{\partial x} h + \frac{\partial^2 f(x)}{\partial x^2} \frac{h^2}{2} + \frac{\partial^3 f(x)}{\partial x^3} \frac{h^3}{6} + \frac{\partial^4 f(x)}{\partial x^4} \frac{h^4}{24} + \dots$$

$$f(x-h) = f(x) - \frac{\partial f(x)}{\partial x} h + \frac{\partial^2 f(x)}{\partial x^2} \frac{h^2}{2} - \frac{\partial^3 f(x)}{\partial x^3} \frac{h^3}{6} + \frac{\partial^4 f(x)}{\partial x^4} \frac{h^4}{24} - \dots$$

Subtracting the second equation from the first:

$$f(x+h) - f(x-h) = 2 \frac{\partial f(x)}{\partial x} h + 2 \frac{\partial^3 f(x)}{\partial x^3} \frac{h^3}{6} + \dots$$



Computational Fluid Dynamics Finite Difference Approximations

The result is:

$$f(x+h) - f(x-h) = 2 \frac{\partial f(x)}{\partial x} h + 2 \frac{\partial^3 f(x)}{\partial x^3} \frac{h^3}{6} + \dots$$

Rearranging this equation to isolate the first derivative:

$$\frac{\partial f(x)}{\partial x} = \frac{f(x+h) - f(x-h)}{2h} - \frac{\partial^3 f(x)}{\partial x^3} \frac{h^2}{6} + \dots$$



Computational Fluid Dynamics Finite Difference Approximations

The Spatial Second Derivative



Computational Fluid Dynamics Finite Difference Approximations

Start by expressing the value of $f(x+h)$ and $f(x-h)$ in terms of $f(x)$

$$f(x+h) = f(x) + \frac{\partial f(x)}{\partial x} h + \frac{\partial^2 f(x)}{\partial x^2} \frac{h^2}{2} + \frac{\partial^3 f(x)}{\partial x^3} \frac{h^3}{6} + \frac{\partial^4 f(x)}{\partial x^4} \frac{h^4}{24} + \dots$$

$$f(x-h) = f(x) - \frac{\partial f(x)}{\partial x} h + \frac{\partial^2 f(x)}{\partial x^2} \frac{h^2}{2} - \frac{\partial^3 f(x)}{\partial x^3} \frac{h^3}{6} + \frac{\partial^4 f(x)}{\partial x^4} \frac{h^4}{24} - \dots$$

Adding the second equation to the first:

$$f(x+h) + f(x-h) = 2f(x) + 2 \frac{\partial^2 f(x)}{\partial x^2} \frac{h^2}{2} + 2 \frac{\partial^4 f(x)}{\partial x^4} \frac{h^4}{24} + \dots$$



Computational Fluid Dynamics Finite Difference Approximations

The result is:

$$f(x+h) + f(x-h) = 2f(x) + 2 \frac{\partial^2 f(x)}{\partial x^2} \frac{h^2}{2} + 2 \frac{\partial^4 f(x)}{\partial x^4} \frac{h^4}{24} + \dots$$

Rearranging this equation to isolate the second derivative:

$$\frac{\partial^2 f(x)}{\partial x^2} = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} - \frac{\partial^4 f(x)}{\partial x^4} \frac{h^2}{12} + \dots$$



Computational Fluid Dynamics Finite Difference Approximations

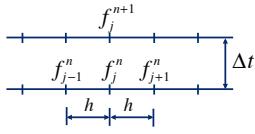
Solving the partial differential equation



Computational Fluid Dynamics Finite Difference Approximations

Use the notation $f^n = f(t)$
For space and time we will use:

$$\begin{aligned}f_j^n &= f(t, x_j) \\f_j^{n+1} &= f(t + \Delta t, x_j) \\f_{j+1}^n &= f(t, x_j + h) \\f_{j-1}^n &= f(t, x_j - h)\end{aligned}$$



Computational Fluid Dynamics Finite Difference Approximations

A numerical approximation to

$$\frac{\partial f}{\partial t} + U \frac{\partial f}{\partial x} = D \frac{\partial^2 f}{\partial x^2}$$

Is found by replacing the derivatives by the following approximations

$$\left(\frac{\partial f}{\partial t} \right)_j + U \left(\frac{\partial f}{\partial x} \right)_j = D \left(\frac{\partial^2 f}{\partial x^2} \right)_j$$



Computational Fluid Dynamics Finite Difference Approximations

Using the shorthand notation

$$\begin{aligned}f_j^n &= f(t, x_j) && \text{gives} \\f_j^{n+1} &= f(t + \Delta t, x_j) && \left(\frac{\partial f}{\partial t} \right)_j = \frac{f_j^{n+1} - f_j^n}{\Delta t} + O(\Delta t) \\f_{j+1}^n &= f(t, x_j + h) && \left(\frac{\partial f}{\partial x} \right)_j = \frac{f_{j+1}^n - f_j^n}{2h} + O(h^2) \\f_{j-1}^n &= f(t, x_j - h) && \left(\frac{\partial^2 f}{\partial x^2} \right)_j = \frac{f_{j+1}^n - 2f_j^n + f_{j-1}^n}{h^2} + O(h^2)\end{aligned}$$



Computational Fluid Dynamics Finite Difference Approximations

Substituting these approximations into:

$$\frac{\partial f}{\partial t} + U \frac{\partial f}{\partial x} = D \frac{\partial^2 f}{\partial x^2}$$

gives

$$\frac{f_j^{n+1} - f_j^n}{\Delta t} + U \frac{f_{j+1}^n - f_{j-1}^n}{2h} = D \frac{f_{j+1}^n - 2f_j^n + f_{j-1}^n}{h^2} + O(\Delta t, h^2)$$

Solving for the new value and dropping the error terms yields

$$f_j^{n+1} = f_j^n - \frac{U\Delta t}{2h}(f_{j+1}^n - f_{j-1}^n) + \frac{D\Delta t}{h^2}(f_{j+1}^n - 2f_j^n + f_{j-1}^n)$$

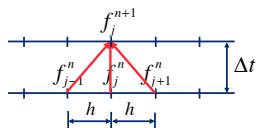


Computational Fluid Dynamics Finite Difference Approximations

Thus, given f at one time (or time level), f at the next time level is given by:

$$f_j^{n+1} = f_j^n - \frac{U\Delta t}{2h}(f_{j+1}^n - f_{j-1}^n) + \frac{D\Delta t}{h^2}(f_{j+1}^n - 2f_j^n + f_{j-1}^n)$$

The value of every point at level $n+1$ is given explicitly in terms of the values at the level n



Computational Fluid Dynamics

Example



Computational Fluid Dynamics

A short MATLAB program

The evolution of a sine wave is followed as it is advected and diffused. Two waves of the infinite wave train are simulated in a domain of length 2. To model the infinite train, periodic boundary conditions are used. Compare the numerical results with the exact solution.



Computational Fluid Dynamics EX2

```
% one-dimensional advection-diffusion by the FTCS scheme
n=21; nstep=100; length=2.0; h=length/(n-1); dt=0.05; D=0.05;
f=zeros(n,1); y=zeros(n,1); ex=zeros(n,1); time=0.0;
for i=1:n, f(i)=0.5*sin(2*pi*h*(i-1)); end; % initial conditions
for m=1:nstep, m, time
    for i=1:n, ex(i)=exp(-4*pi*pi*D*time)*...
        0.5*sin(2*pi*(h*(i-1)-time)); end; % exact solution
    hold off; plot(f,'linewidt',2); axis([1 n -2.0, 2.0]); % plot solution
    hold on; plot(ex,'r','linewidt',2); pause; % plot exact solution
    y=f;
    for i=2:n-1,
        f(i)=y(i)-0.5*(dt/h)*(y(i+1)-y(i-1))+...
            D*(dt/h^2)*(y(i+1)-2*y(i)+y(i-1)); % advect by centered differences
    end;
    f(n)=y(n)-0.5*(dt/h)*(y(2)-y(n-1))+...
        D*(dt/h^2)*(y(2)-2*y(n)+y(n-1)); % do endpoints for
    f(1)=f(n); % periodic boundaries
    time=time+dt;
end;
```



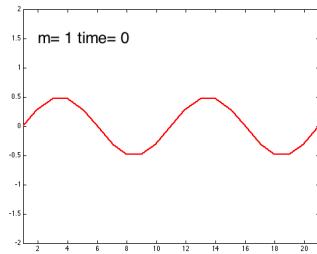
Computational Fluid Dynamics

Evolution for

$U=1$;
 $D=0.05$;
 $k=1$

$N=21$
 $\Delta t=0.05$

— Exact
— Numeric



Computational Fluid Dynamics

It is clear that although the numerical solution is qualitatively similar to the analytical solution, there are significant quantitative differences.

The derivation of the numerical approximations for the derivatives showed that the error depends on the size of h and Δt .