Numerical Simulation Methods

Part 1 - Fundamental Theory

Lecture 2: Introduction to Discretisation

- Objective is to develop and understand numerical methods to solve (Navier-Stokes) eqns.

TODAY

- Introduce the fundamental concepts of discretisation and ‘time-marching’.
- Consider forms of the equations.

- Briefly discuss implications of the forms of the equations,
i.e. conservative v non-conservative.

Discretisation and Time-Marching

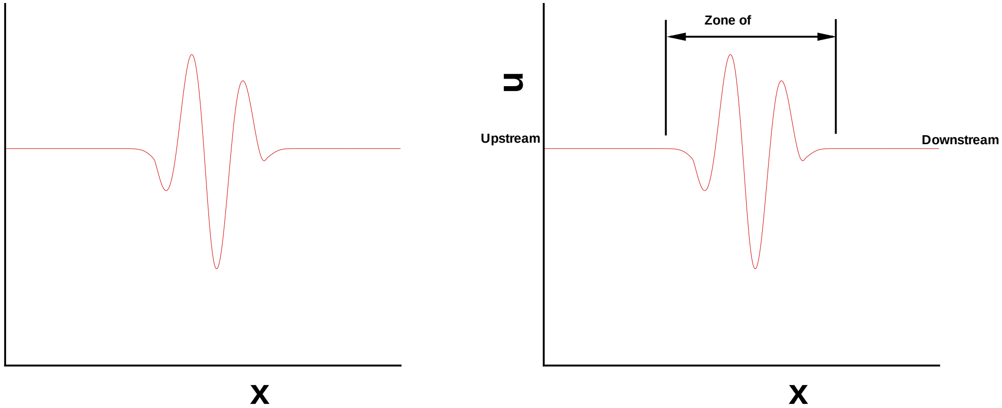
Discretisation

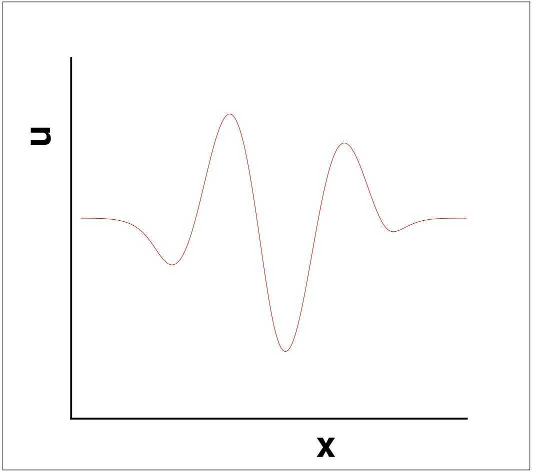
A real solution is continuous and exists at all points in space, and all (previous) points in time. However, we don't want (and don't need) to store the solution everywhere.

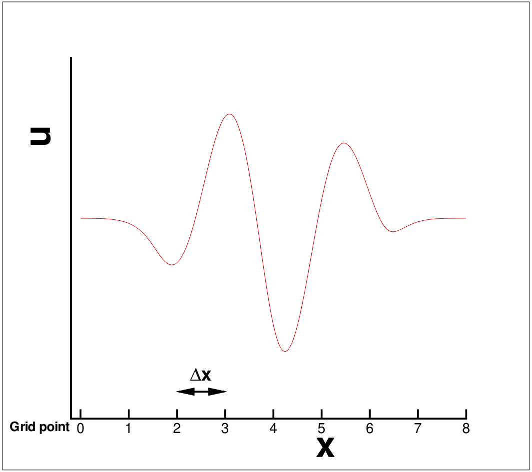
Instead of the real continuous solution, we represent this by a discrete solution; we only have solution values stored at a discrete number of points in space, and time (consider time later).

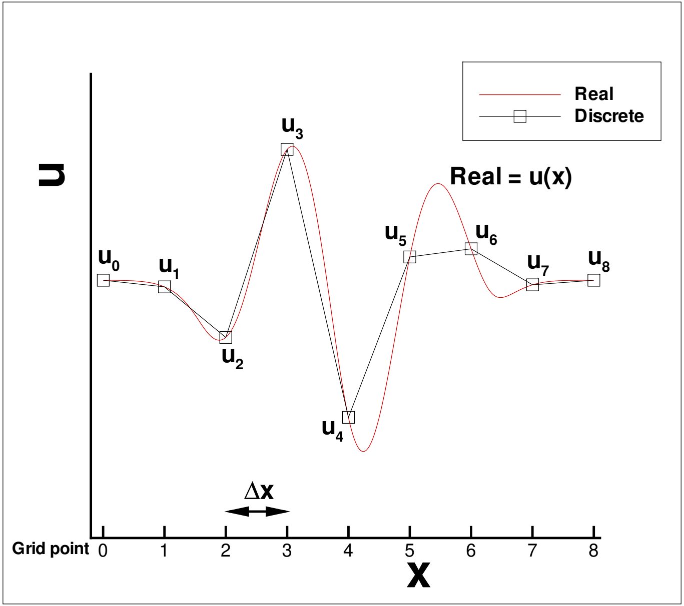
Clearly the accuracy of the discrete solution will depend on the number of points used to represent it.

Consider the variation in space of some scalar quantity, u. Assume this is some external quantity, so exists from $-\infty$ to $+\infty$. However, only a certain part of the domain is of interest. Hence, we reduce the real *physical* domain to a smaller *computational* domain.


X
Zone of
u
Upstream
Downstream
X


u
x


u
Δx
Grid point 0 1 2 3 4 5 6 7 8
x


Real
Discrete
u
u_3
Real = u(x)
u_5
u_6
u_0
u_1
u_7
u_8
u_2
u_4
Δx
Grid point 0 1 2 3 4 5 6 7 8
x

Aspects to the discretisation:

- Temporal discretisation: this is commonly a series of 'copies' of the spatial discretisation, stored as different time 'levels', often indexed as n, $n+1$, $n+2$ and similar. Nearly all codes will use this approach - apart from some of mine!
- Spatial discretisation: broadly this splits in to finite difference (FD), finite volume (FV) and finite element. FD uses points and Taylor series-based differencing, FV uses arbitrary control volume across which properties are assumed constant, and FE uses volume across which there may be linear or higher order variations of properties (this is where higher order methods such as DG or 'discontinouous Galerkin' and spectral methods fit in). In this course we will mostly refer to a 1D spatial model, with x_i^n denoting a variable at time level n and spatial node position i
- I say *nearly* all codes, because it is in fact possible to have a single discretisation that covers both space and time - a *spacetime* mesh. It sounds like a Hollywood concept, but it's just a mesh or grid in x, y, z, t
- There is an obvious point to notice - frequencies with wavelength less than two intervals are not going to be captured (Nyquist frequency). You're out of luck or you need a finer mesh (this can happen in space or time)

A static determinate beam problem perhaps makes space and time discretisation more approachable. A cantilever can be solver with only space marching, starting at the root and moving to the tip

$$\frac{d^2y}{dx^2} = \frac{M}{EI}$$

$$\frac{y_{i+1} - 2y_i + y_{i-1}}{\Delta x^2} = \frac{M(x)}{EI}$$

which you could rearrange as

$$y_{i+1} = 2y_i - y_{i-1} + \Delta x^2 \frac{M(x)}{EI}$$

and this would permit a space marching method moving from the root to the tip of the beam.

Space marching a finite difference

x	y	M/EI
0	0	1
0.02	0	0.98
0.04	0.000392	0.96
0.06	0.001168	0.94
0.08	0.00232	0.92
0.1	0.00384	0.9
...	...	...
0.92	0.28428	0.08
0.94	0.294032	0.06
0.96	0.303808	0.04
0.98	0.3136	0.02
1	0.3234	0

Table: Finite differenced beam integration (linear M/EI from root to tip)

Beam example

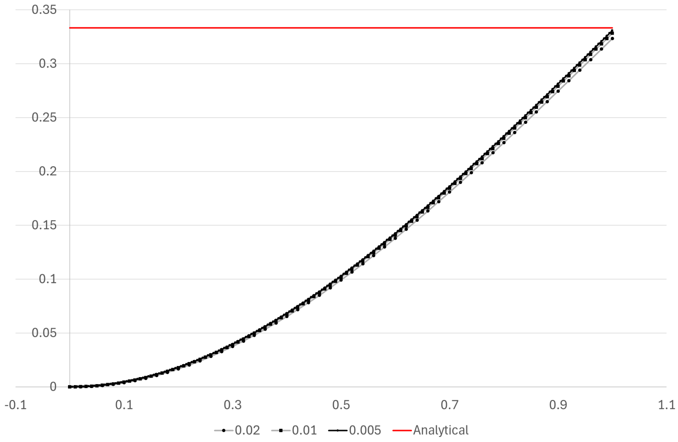


Figure: Beam deflections for varying Δx. Tip deflections are 0.3234, 0.3284, 0.33084 as Δx is reduced, approaching the $\frac{1}{3}$ exact result

A dynamic beam

$$\mu \frac{\partial^2 y}{\partial t^2} = -EI \frac{\partial^4 y}{\partial x^4}$$

$$\mu \frac{y_i^{n+1} - 2y_i^n + y_i^{n-1}}{\Delta t^2} = -EI \frac{\partial^4 y}{\partial x^4}_{n,i}$$

which you could rearrange as

$$y_i^{n+1} = 2y_i^n - y_i^{n-1} - \Delta t^2 \frac{EI}{\mu} \frac{\partial^4 y}{\partial x^4}_{n,i}$$

and this would permit a (rather basic) explicit time marching method (because the right had side is entirely at time level n). Other better integration approaches exist - this is just for info - don't go and apply it!

Now consider a simple one-dimensional PDE (this is a close representation of the first order in time fluid system) (u is any scalar):

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0$$

For a steady flow, we know that all temporal derivatives are zero, i.e.

$$\frac{\partial u}{\partial t} = 0 \quad \Rightarrow \quad \frac{\partial f(u)}{\partial x} = 0.$$

Hence, if we are solving for a steady flow, then if $\frac{\partial u}{\partial t} \neq 0$ at any point, there is an 'error' in the solution at that point.

Time-Marching

Time-marching exploits the fact that we have a temporal derivative to give an updating procedure for the solution.

- 1. Start with an initial guess. For example we might say:

$$u_0 = u_1 = u_2 = u_3 = u_4 = u_5 = 0, \quad u_6 = u_7 = u_8 = 1$$

- 2. Then approximate $\frac{\partial f(u)}{\partial x}$ $(= ERROR)$ at each point.
- 3. Then use a simple approximation to the temporal gradient at each point, for example

$$\frac{\partial u}{\partial t} \approx \frac{u(t + \Delta t) - u(t)}{\Delta t} \quad \Rightarrow \quad \frac{u(t + \Delta t) - u(t)}{\Delta t} + ERROR = 0$$

- 4. Then for each point

$$u(t + \Delta t) \approx u(t) - \Delta t \times ERROR$$

  or

$$u^{future} \approx u^{current} - \Delta t \times ERROR$$

- 5. Repeat 2-4 at each point until solution stops changing.

Actually, the updating procedure is:

$$u^{future} = u^{current} - \Delta t \times ERROR + \text{Truncation Error}$$

The truncation error will be considered more in future lectures, and gives us very useful information about the updating procedure. It is actually a function of $(\Delta x^p, \Delta t^q)$ where p, q give us a measure of the accuracy of the scheme.
This is analogous to the beam example and the basis of all explicit integration approaches.

Conservation Form of the Equations

We have derived our equations in the form

$$\frac{\partial()}{\partial t} + \frac{\partial()}{\partial x} + \frac{\partial()}{\partial y} + \frac{\partial()}{\partial z} = 0$$

so that there are no factors outside the derivatives. This is the Eulerian CONSERVATIVE form of the equations. There are many other forms of the equations - non-conservative forms.
Consider 2D Euler equations

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} = 0$$

$$\frac{\partial \rho u}{\partial t} + \frac{\partial \rho u^2}{\partial x} + \frac{\partial \rho uv}{\partial y} = -\frac{\partial P}{\partial x}$$

or

$$\frac{\partial \rho u}{\partial t} + \frac{\partial (\rho u^2 + P)}{\partial x} + \frac{\partial \rho uv}{\partial y} = 0$$

Now expand

$$\frac{\partial \rho u}{\partial t} = \rho \frac{\partial u}{\partial t} + u \frac{\partial \rho}{\partial t}$$
$$\frac{\partial \rho u^2}{\partial x} = \rho u \frac{\partial u}{\partial x} + u \frac{\partial \rho u}{\partial x}$$
$$\frac{\partial \rho u v}{\partial y} = \rho v \frac{\partial u}{\partial y} + u \frac{\partial \rho v}{\partial y}$$

Collect

$$\frac{\partial \rho u}{\partial t} + \frac{\partial (\rho u^2 + P)}{\partial x} + \frac{\partial \rho u v}{\partial y} = \rho \left[\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right]$$
$$+ \frac{\partial P}{\partial x} + u \left[\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} \right] = 0$$

Hence, we can write the X-momentum equation as

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho} \frac{\partial P}{\partial x}$$

This form is written in many textbooks. It can also be done in vector calculus notation as...

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \mathbf{u} = 0 \tag{1}$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot \rho \mathbf{u}\mathbf{u} = -\nabla p \tag{2}$$

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\frac{\partial \rho}{\partial t} + \rho \mathbf{u} \cdot \nabla \mathbf{u} + \mathbf{u}\nabla \cdot \rho \mathbf{u} = -\nabla p \tag{3}$$

terms two and four of the left side are then identified at the mass equation multiplied by velocity giving

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p \tag{4}$$

which is also possible to write in Lagrangian (moving with fluid form) as

$$\frac{D\mathbf{u}}{Dt} = -\frac{\nabla p}{\rho} \tag{5}$$

and you can identify this as $F = ma$, because gradient of pressure is (inviscid) force per unit volume, and density is mass per unit volume.

$$\frac{D\mathbf{u}}{Dt} = -\frac{\nabla p}{\rho} \tag{6}$$

This is the basis of many Lagrangian methods - for example smoothed particle hydrodynamics (SPH). Particle information provides an approximation of the pressure force on the right, and then particles are integrated/moved forwards in time. It's a remarkably simple method with a notable applicability to free surface flows.

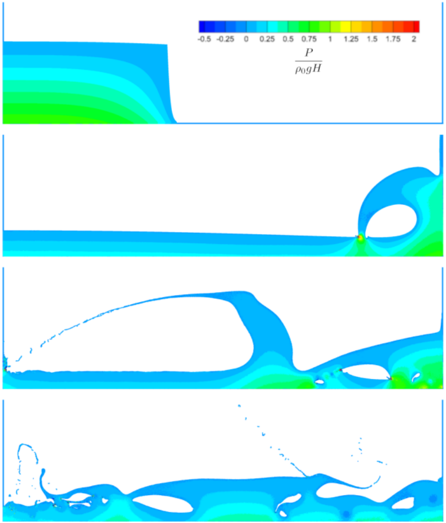
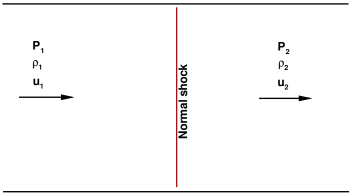
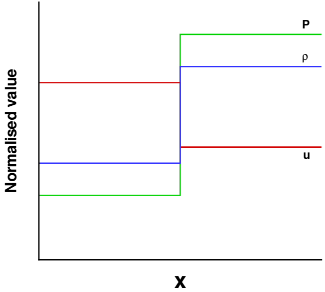


Figure: SPH dam break modelling with artificial compressibility based SPH, from Joe's paper https://doi.org/10.1016/j.cma.2023.116700. Those smooth pressure contours are what it's all about!

Why might the nonconservative form struggle?
Recall shockwaves from your second year work. These are approximately 60 mean free paths thick, which is much, much smaller than any likely mesh spacing. These shocks will therefore appear as step changes in the non-conservative flow variables.
Shockwaves on aircraft introduce significant changes in lift, drag and moment coefficients.
The original purpose of much CFD development was in fact to capture shocks, so it isn't advisable to use a scheme that can struggle with them (see later).

The non-conservative form of the equations is poor for flows with shockwaves. Consider 1-D shock-tube, i.e. discontinuity in P, ρ, u.



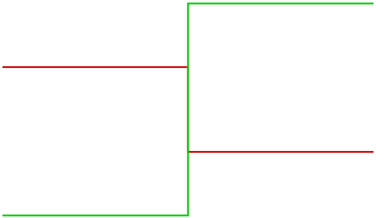


Now $\frac{\partial P}{\partial x} \to \infty$, $\frac{\partial u}{\partial x} \to \infty$. Not great for numerical computations.

But, for conservative form,

$$\frac{\partial \rho u}{\partial t} + \frac{\partial(\rho u^2 + P)}{\partial x} = 0$$

and for steady flow $\frac{\partial \rho u}{\partial t} = 0$, so $\frac{\partial(\rho u^2+P)}{\partial x} = 0$ or $\rho u^2 + P = const$ even across a shock wave.



Hence this causes no problem for numerical approximations. Therefore, the conservative form of the equations are preferred, as they involve differentiation of continuous functions, which can easily be done computationally using finite differences.

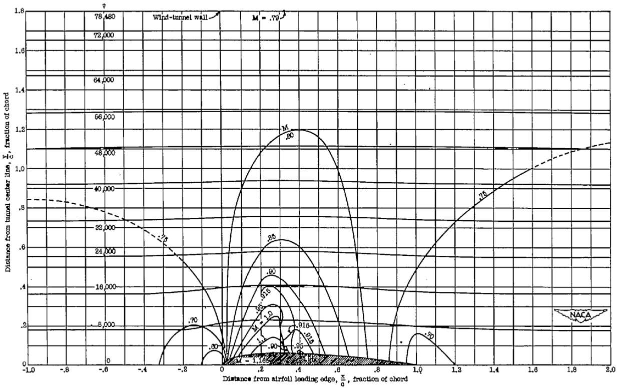
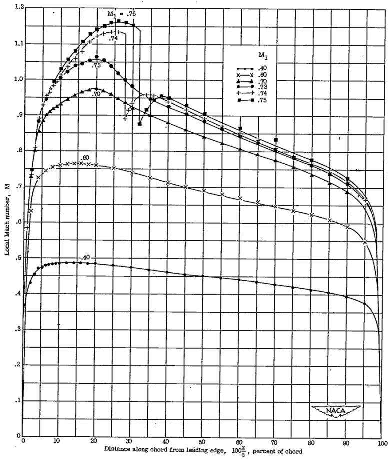


Figure 5.- Streamlines and constant Mach number lines for flow of a compressible fluid past an NACA 0012 airfoil in a wind tunnel. $\alpha = 0$; $M_1 = 0.75$.



(a) In wind tunnel.

Early CFD - no computer, no problem, just a room full of people

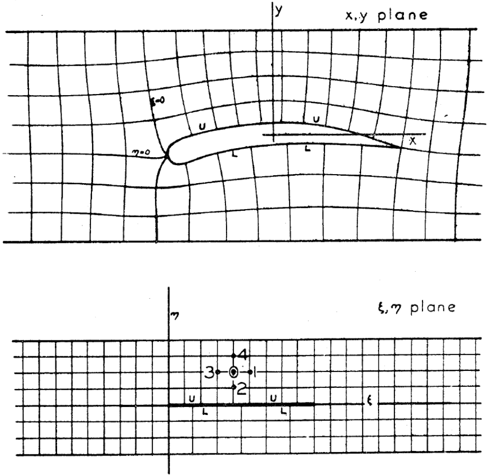


Fig. 4.

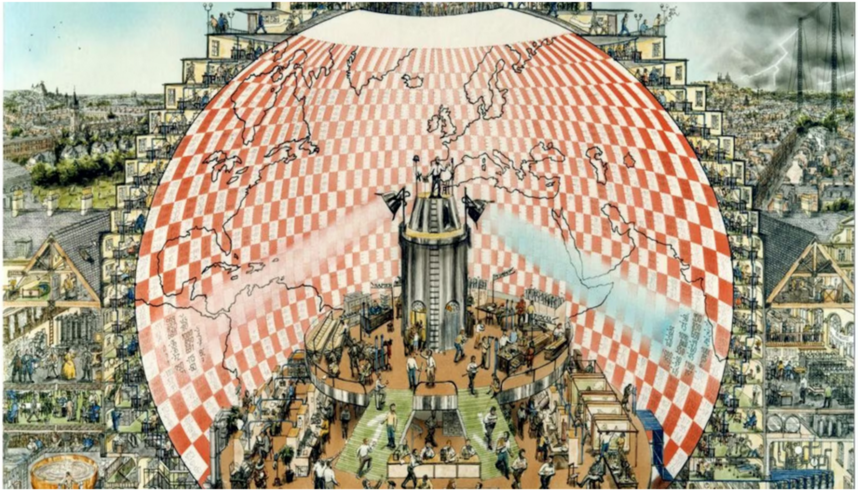


Figure: A 1986 painting illustrating what Lewis Fry Richardson's 'forecast factory' might have looked like. I think it looks much more fun than HPC!

What happens if there is a mistake?

You might think everyone gave up and went home, but not so. The iterations eliminate errors, whether they are numerical or human introduced. Human errors were of course made, but the iterations eliminated them again later providing stability was maintained - it just increased the time taken to converge. I imagine there was an error rate of at least 1/100 or 1/1000 but it didn't really matter.
Clever, huh?

Summary

• Discretisation is the concept of representing a real continuous funtion, $u(x,t)$, with a discrete solution, i.e. only stored/known at discrete points in time and space, $u(i\Delta x, n\Delta t)$.
• Equations we want to solve are non-linear PDEs with a temporal derivative, and this is exploited to produce a time-marching approach to steady flows.
Begin with a guessed (wrong) initial value, $u(x,0)$, continue to update u until it stops changing, i.e. $\frac{\partial u}{\partial t} \to 0$.
$\frac{\partial u}{\partial t}$ is the 'error' in the spatial gradients.
• Conservative form of the equations must be solved in numerical methods, due to difficulty in representing large gradients discretely.

NEXT LECTURE: Example aerodynamic simulation and introduce a model equation.