Numerical approximations of derivatives and integrals

Gergely Kristóf
6-th September 2017

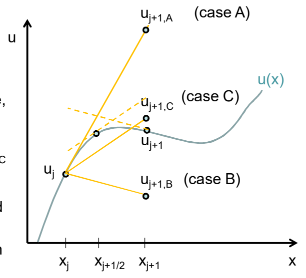
		Lecture	Laboratory
		Wednesday 10-12	Thursday 10-12
		K. g.f. 87	CFD lab
1	szept.6	Differencing schemes	ICEM tutorial (1) - 2D elbow
2	szept.13	Finite volume method	ICEM tutorial (2) - 3D elbow
3	szept.20	Uni. Sport Day	ICEM tutorial (3) - pipe-blade
4	szept.27	Pressure-velocity coupling	Pump model
5	okt.4	Solution of linear systems	Jet fan model
6	okt.11	Compressible flows	Individual assignment (1)
7	okt.18	Multiphase flows	Individual assignment (2)
8	okt.25	Multiphase flows	Individual assignment (3)
9	nov.1	National event	Group assignment (1)
10	nov.8	1st midterm	Group assignment (2)
11	nov.15	Turbulence modelling	Scientific student competition
12	nov.22	Turbulence modelling	Group assignment (3)
13	nov.29	Turbulence modelling	Multiphase flow tutorial
14	dec.6	2nd midterm	Presentation
	dec.14	Retake	

Finite difference method error and convergence

We shall calculate the change of exact solution u(x) by integrating the derivative on section $x_{j+1}-x_j=\Delta x$:
A) from the initial derivative,
B) from the terminal derivative,
C) from midpoint derivative.
The values of the approximate solution are: $u_{j+1,A}$, $u_{j+1,B}$, $u_{j+1,C}$

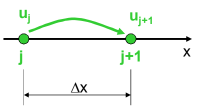
The approximation error $u_{j+1}-u_{j+1,A}$ reduces with reduced interval size.

Some schemes are better than the other...



XLS demo

Forward Differencing Scheme (FDS)



From the Taylor polynomial we can express a differencing scheme of first order accuracy:

$$u'_j = \frac{u_{j+1} - u_j}{\Delta x} + o(1)$$

Note that, the error term is one degree of magnitude higher.

Taylor polynomial of the exact solution from point j to point j+1:

$$u_{j+1} = u_j + u'_j \Delta x + u''_j \frac{\Delta x^2}{2} + \ldots$$

$$u_{j+1} = u_j + u'_j \Delta x + o(\Delta x)$$

This is an integration scheme of first order accuracy.

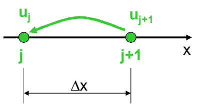
When the differential equation is given in the explicit form:

$$u'_j = f(u_j, x_j)$$

we can integral step by step, by assuming:

$$u_{j+1} \cong u_j + f(u_j, x_j)\Delta x$$

Backward Differencing Scheme (BDS), implicit discretisation method



When F is evaluated in j+1, we may end up with a more complicated expression for u_{j+1}. This kind of discretization is called implicit:

Another first order scheme:

$$u_j = u_{j+1} + u'_{j+1}(-\Delta x) + o(\Delta x)$$

from the backward Euler scheme we get:

$$u'_{j+1} = \frac{u_{j+1} - u_j}{\Delta x} + o(1)$$

Now, we assume the differential equation is given in the following form:

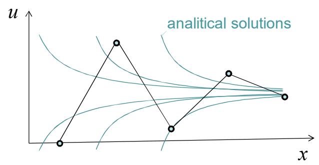
$$F(u', u, x) = 0$$

unknown

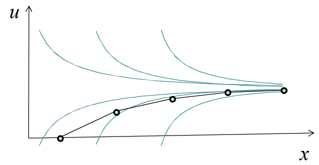
$$F(u'_{j+1}, u_{j+1}, x_{j+1}) = 0 \longrightarrow F\left(\frac{u_{j+1} - u_j}{\Delta x}, u_{j+1}, x_{j+1}\right) \cong 0$$

Different behavior…

Physical processes lead to a temporal equilibrium in many cases.

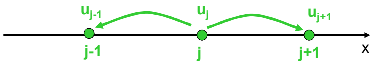


Explicit Euler method:



Implicit Euler method:

Central Differencing Scheme (CDS)



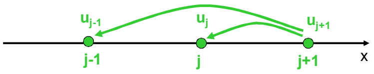
$$u_{j+1} = u_j + u'_j\,\Delta x + u''_j\,\frac{\Delta x^2}{2} + o(\Delta x^2)$$

$$u_{j-1} = u_j + u'_j(-\Delta x) + u''_j\,\frac{\Delta x^2}{2} + o(\Delta x^2)$$

$$u'_j = \frac{u_{j+1} - u_{j-1}}{2\,\Delta x} + o(\Delta x)$$

Extensively used **in CFD for spatial discretization**.

An implicit differencing scheme with second order accuracy



$$u_j = u_{j+1} + u'_{j+1}(-\Delta x) + u''_{j+1}\,\frac{\Delta x^2}{2} + o(\Delta x^2)$$

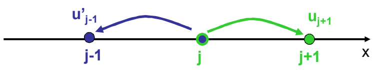
$$u_{j-1} = u_{j+1} + u'_{j+1}(-2\Delta x) + u''_{j+1}\,2\Delta x^2 + o(\Delta x^2)$$

$$u_j - \frac{u_{j-1}}{4} = \frac{3}{4}u_{j+1} + u'_{j+1}\left(-\frac{\Delta x}{2}\right) + o(\Delta x^2)$$

$$u'_{j+1} = \frac{\frac{3}{2}u_{j+1} - 2u_j + \frac{1}{2}u_{j-1}}{\Delta x} + o(\Delta x)$$

Can be used for discretizing the **boundary layer equation**.

Adams-Basforth scheme



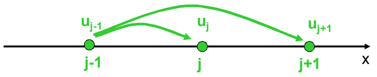
$$u_{j+1} = u_j + u'_j\,\Delta x + u''_j\,\frac{\Delta x^2}{2} + o(\Delta x^2)$$

$$u'_{j-1} = u'_j + u''_j(-\Delta x) + o(\Delta x) \quad / \quad + \ldots \times \frac{\Delta x}{2}$$

$$u_{j+1} = u_j + \frac{3}{2}u'_j\,\Delta x - \frac{1}{2}u'_{j-1}\,\Delta x + o(\Delta x^2)$$

An explicit integrating scheme with second order accuracy
It is often used **for integrating the Navier-Stoket equations**.

A 2 step 2nd order explicit Runge-Kutta type scheme



1st step: Using the Euler method we can calculate approximate values: $\tilde{u}_j$ and $\tilde{u}'_j$

$$u_j = u_{j-1} + u'_{j-1}\,\Delta x + o(\Delta x) = \tilde{u}_j + o(\Delta x)$$

$$u'_j = f(u_j, x_j) = f(\tilde{u}_j + o(\Delta x), x_j) = f(\tilde{u}_j, x_j) + \left.\frac{\partial f}{\partial u}\right|_{u_j, x_j} \cdot o(\Delta x) = \tilde{u}'_j + o(\Delta x)$$

2nd step: Use CDS scheme around point j:

$$u_{j+1} = u_{j-1} + u'_j\, 2\,\Delta x + o(\Delta x^2) = u_{j-1} + \tilde{u}'_j\, 2\,\Delta x + o(\Delta x^2)$$

Can be used for calculating compressible flows (**eg. Lax-Wendroff method**).

Further important properties of numerical methods

1. **Consistency** The discretization of a PDE should become exact as the mesh size tends to zero (truncation error should vanish)
2. **Stability** Numerical errors which are generated during the solution of discretized equations should not be magnified
3. **Convergence** The numerical solution should approach the exact solution of the PDE and converge to it as the mesh size tends to zero
4. **Conservation** Underlying conservation laws should be respected at the discrete level (artificial sources/sinks are to be avoided)
5. **Boundedness** Physical quantities like densities, temperatures, concentrations etc. should remain nonnegative and free of spurious wiggles

These properties must be verified for each (component of the) numerical scheme.