MECH 479/587

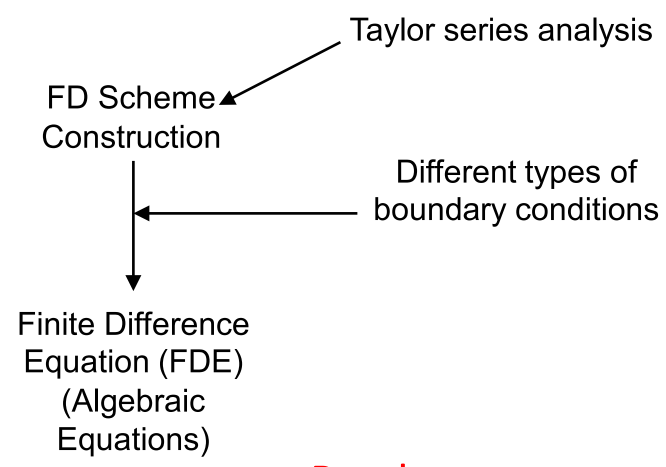
Computational Fluid Dynamics

Module 3: Finite Difference Approximation

Rajeev K. Jaiman
Winter Term 1, 2022

Outline

- ❑ Introduction of Taylor series
  - ► Point difference operators
  - ► Construction of finite difference equations (FDEs)
- ❑ Generalized finite difference approximation
  - ► Matrix difference operators
  - ► Boundary conditions
- ❑ Numerical Properties of FDEs
  - ► Order of Accuracy
  - ► Consistency
  - ► Stability
  - ► Lax Equivalence Theorem



Roadmap

Measurable Learning Outcomes

- Understand and apply Taylor series for derivative approximation
- Implement a finite difference discretization to solve a representative PDE (or set of PDEs) from an engineering application
- Understand basic numerical properties (accuracy, stability and convergence) of finite difference equation (FDE)

The Finite Difference (FD) Method

- To replace continuous PDE problem to finite difference approximation
  - Using the definition of derivative

$$u_x \equiv \left(\frac{\partial u}{\partial x}\right) = \lim_{\Delta x \to 0} \frac{u(x+\Delta x) - u(x)}{\Delta x}$$

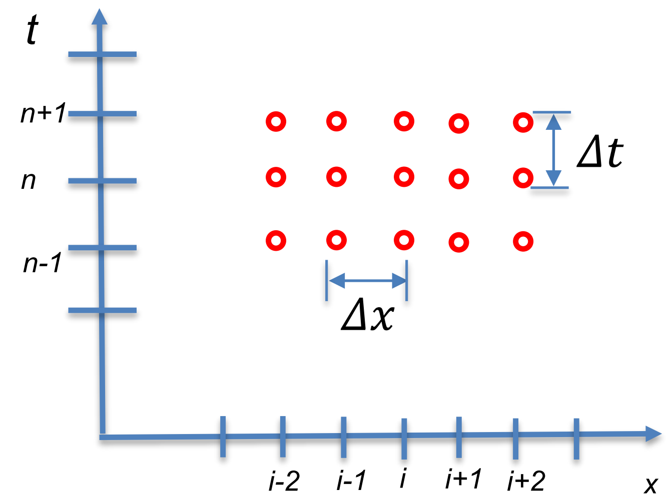
If Δx is small but finite the expression on the right side is an approximation to the exact value of

  - Using the properties of Taylor expansions
    - Function can be differentiated many times

- Relies on uniform meshes with high degree of regularity
  - Mesh must setup in a structured way

The Finite Difference (FD) Method

- Relies on uniform meshes with high degree of regularity
  - Mesh is generally setup in a structured way
  - Finite difference equations can be constructed for dependent variable *u* as functions of the independent variables *t*, and *x*,*y*,*z*



For example, consider dependent variable u in one-dimension as function of independent variables (x,t) with $u = u(x,t)$, where

$$x = x_i = i\Delta x$$

$$t = t_n = n\Delta t$$

Taylor Series Expansion

Taylor series expansion is a powerful method that will allow us to construct an FD scheme to any order of accuracy that we desire. It can also be used to check the consistency of a given FD scheme.

By assuming that the function $u(x)$ can be differentiated many times, then the general single variable Taylor series expansion of function $u(x)$ (expanded around x_0) is given as

$$u(x_0+\Delta x) = u(x_0) + \overset{\text{First Approx.}}{(\Delta x)\left.\frac{\partial u}{\partial x}\right|_{x=x_0}} + \overset{\text{Consider Slope}}{\frac{1}{2!}(\Delta x)^2\left.\frac{\partial^2 u}{\partial x^2}\right|_{x=x_0}} + \overset{\text{Consider Curvature}}{\frac{1}{3!}(\Delta x)^3\left.\frac{\partial^3 u}{\partial x^3}\right|_{x=x_0}} + \cdots$$

Using the formulation above, we can generate FD schemes to any desired order of accuracy for derivatives of any order.

Finite Difference Notation

u_i^n

Time location

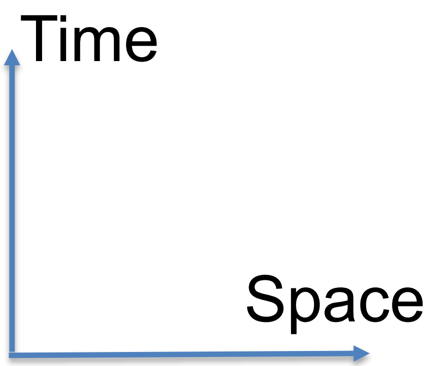
Spatial mesh point/node

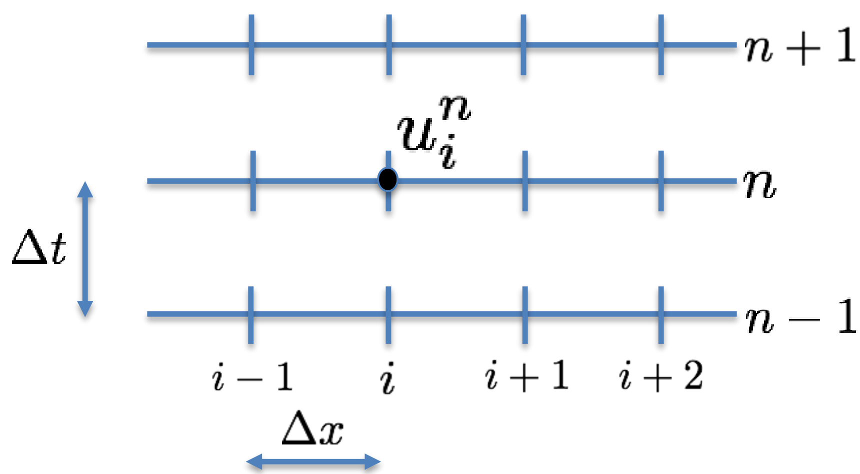
$$u_i^n = u\left(t, x_i\right)$$

$$u_i^{n+1} = u\left(t + \Delta t, x_i\right)$$

$$u_{i+1}^n = u\left(t, x_i + h\right)$$

$$u_{i-1}^n = u\left(t, x_i - h\right)$$



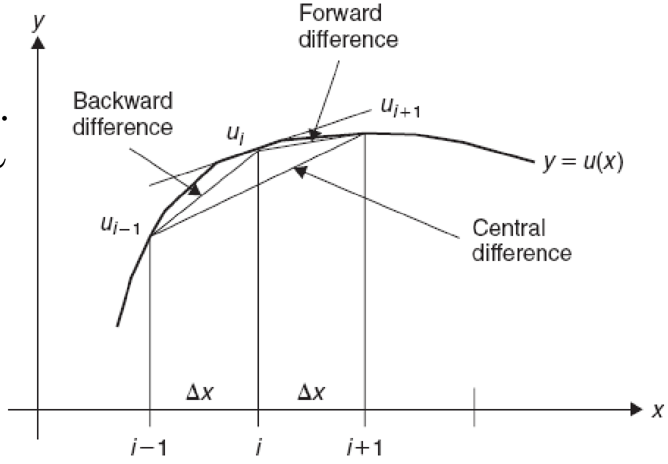


Taylor Series

- Forward differencing

$$\left(u_x\right)_i = \left(\frac{\partial u}{\partial x}\right)_i = \frac{u_{i+1}-u_i}{\Delta x} \underbrace{- \frac{\Delta x}{2}\left(u_{xx}\right)_i - \frac{\Delta x^2}{6}\left(u_{xxx}\right)_i + \ldots}_{\text{Truncation error}}$$

$$= \frac{u_{i+1}-u_i}{\Delta x} + O(\Delta x)$$



- Backward differencing

$$\left(u_x\right)_i = \left(\frac{\partial u}{\partial x}\right)_i = \frac{u_i-u_{i-1}}{\Delta x} \underbrace{+ \frac{\Delta x}{2}\left(u_{xx}\right)_i - \frac{\Delta x^2}{6}\left(u_{xxx}\right)_i + \ldots}_{\text{Truncation error}}$$

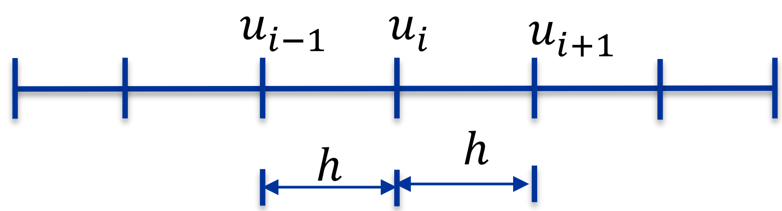
$$= \frac{u_i-u_{i-1}}{\Delta x} + O(\Delta x)$$

- Centered

$$\left(u_x\right)_i = \left(\frac{\partial u}{\partial x}\right)_i = \frac{u_{i+1}-u_{i-1}}{2\Delta x} \underbrace{- \frac{\Delta x^2}{6}\left(u_{xxx}\right)_i + \ldots}_{\text{Truncation error}}$$

$$= \frac{u_{i+1}-u_{i-1}}{2\Delta x} + O(\Delta x^2)$$

Spatial First Derivative



Let $u = u(x)$, then for small h

$$u(x+h) = u(x) + \left.\frac{\partial u}{\partial x}\right|_x h + \left.\frac{\partial^2 u}{\partial x^2}\right|_x \frac{h^2}{2!} + \left.\frac{\partial^3 u}{\partial x^3}\right|_x \frac{h^3}{3!} + \cdots$$

$$= u(x) + u'(x).h + u''(x)\frac{h^2}{2!} + u'''(x)\frac{h^3}{3!} + \cdots$$

$$u(x-h) = u(x) - \left.\frac{\partial u}{\partial x}\right|_x h + \left.\frac{\partial^2 u}{\partial x^2}\right|_x \frac{h^2}{2!} - \left.\frac{\partial^3 u}{\partial x^3}\right|_x \frac{h^3}{3!} + \cdots$$

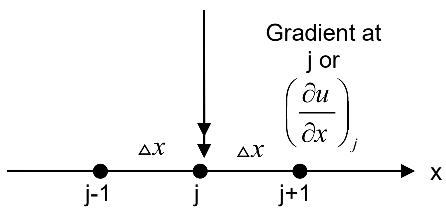
$$= u(x) - u'(x).h + u''(x)\frac{h^2}{2!} - u'''(x)\frac{h^3}{3!} + \cdots$$

In Central differencing (to n=2 which is second order of accuracy),

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

$$= \left(\frac{\partial u}{\partial x}\right)_j + \frac{1}{6}\Delta x^2 \left(\frac{\partial^3 u}{\partial x^3}\right)_j + \cdots$$

with Taylor series expansion:



$$= \left(\frac{\partial u}{\partial x}\right)_j + O\left(\Delta x^2\right)$$

in forward differencing (to n=1 first order of accuracy)

The symbol δ is used to represent a finite difference approximation to a derivative such that

$$\delta_x \approx \partial_x, \quad \delta_{xx} \approx \partial_{xx}$$

$$
\begin{aligned}
\left(\delta_x u\right)_j &= \frac{u_{j+1} - u_j}{\Delta x} \\
&= \left(\frac{\partial u}{\partial x}\right)_j + \frac{1}{2}\Delta x \left(\frac{\partial^2 u}{\partial x^2}\right)_j + \cdots \\
&= \left(\frac{\partial u}{\partial x}\right)_j + O\left(\Delta x\right)
\end{aligned}
$$

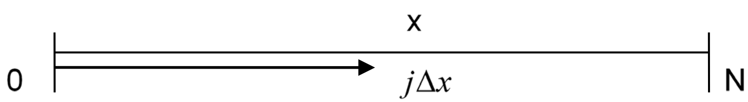
in Backward differencing (to n=1, first order of accuracy),

$$
\begin{aligned}
\left(\delta_x u\right)_j &= \frac{u_j - u_{j-1}}{\Delta x} \\
&= \left(\frac{\partial u}{\partial x}\right)_j + O\left(\Delta x\right)
\end{aligned}
$$

- Hence difference operators are non-unique with different (n) orders of approximation depending on the nodal points employed.

- In general, in order to get higher order of approximation (i.e. more accurately), one would require more nodal points for representation of the differential operator.

(if $x = j\Delta x$, $u(j\Delta x + h) = u_j + u_j' h + u_j'' \frac{h^2}{2!} + u_j''' \frac{h^3}{3!} + \cdots$)



Similarly,

$$u(x-h) = u(x) - u'(x).h + u''(x)\frac{h^2}{2!} - u'''(x)\frac{h^3}{3!} + \cdots \qquad \boxed{\begin{array}{l} x = j\Delta x \\ h = \Delta x \end{array}}$$

$$= u_j - u_j'.h + u_j''\frac{h^2}{2!} - u_j'''\frac{h^3}{3!} + \cdots$$

By adding, we have

$$u(x+h) + u(x-h) = 2u(x) + h^2 u''(x) + o\left(h^4\right) + \cdots$$

$$\Rightarrow u''(x) = \frac{1}{h^2}\left[u(x+h) - 2u(x) + u(x-h)\right] + O\left(h^2\right)$$

$$\equiv \frac{u_{j+1} - 2u_j + u_{j-1}}{\Delta x^2} + O\left(h^2\right) \qquad \text{for} \quad h = \Delta x$$

i.e. up to $o(\Delta x)^2$ which is 2nd order.

By simple subtracting, we obtain:

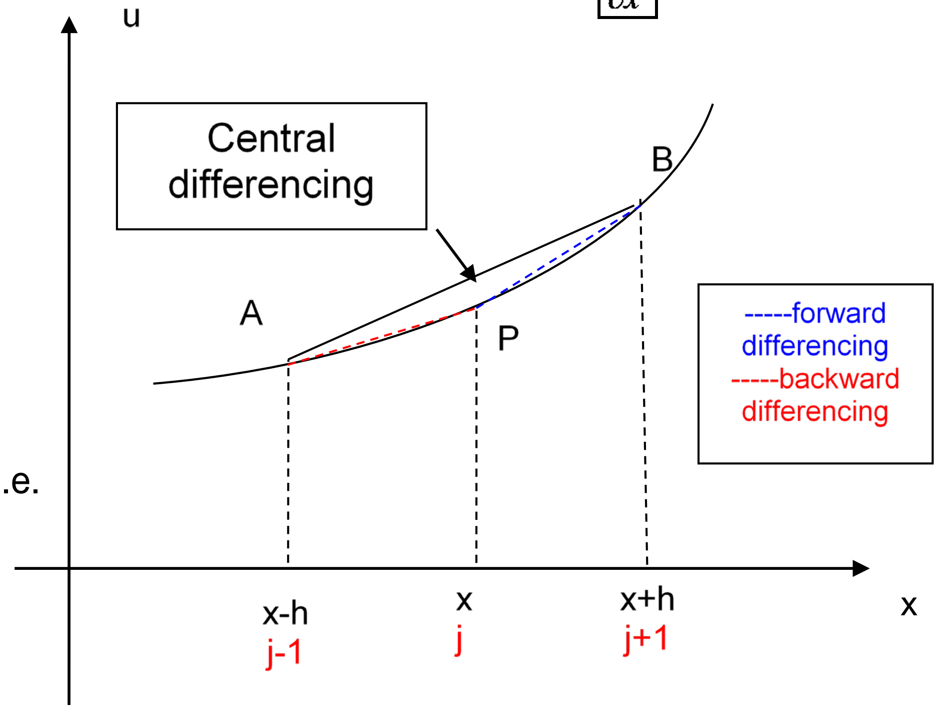
$$\Rightarrow u'(x) = \frac{1}{2h}\left\{u(x+h) - u(x-h)\right\} + O\left(h^2\right)$$

$$\equiv \frac{u_{j+1} - u_{j-1}}{2\Delta x} + O\left(h^2\right) \quad \text{for } h = \Delta x$$

2nd order accuracy.

Central differencing formulas for $\frac{\partial^2 u}{\partial x^2}$ & $\frac{\partial u}{\partial x}$

Graphically, the central difference approximation for $\boxed{\frac{\partial u}{\partial x}}$ can be viewed as thus:



The exact gradient $\left(\frac{\partial u}{\partial x}\right)_x$ at P is given by the slope AB, i.e.

$$\frac{u(x+h)-u(x-h)}{2h} \qquad (h \equiv \Delta x)$$

$$\equiv \frac{u_{j+1}-u_{j-1}}{2\Delta x}$$

For forward difference approximation, the gradient $\left(\frac{\partial u}{\partial x}\right)_p$ at P is given by the *slope BP*, i.e.

$$\frac{u(x+h)-u(x)}{h} \equiv \frac{u_{j+1}-u_j}{\Delta x}.$$

For backward difference approximation, the gradient $\left(\frac{\partial u}{\partial x}\right)_p$ at P is given by the *slope AP*, i.e.

$$\frac{u(x)-u(x-h)}{h} \equiv \frac{u_j-u_{j-1}}{\Delta x}.$$

Second-Order One-Sided Difference

- Example

$$(u_x)_i = \frac{au_i + bu_{i-1} + cu_{i-2}}{\Delta x} + O(\Delta x^2)$$

$$u_{i-2} = u_i - 2\Delta x (u_x)_i + \frac{(2\Delta x)^2}{2}(u_{xx})_i - \frac{(2\Delta x)^3}{6}(u_{xxx})_i + \ldots$$

$$u_{i-1} = u_i - \Delta x (u_x)_i + \frac{(\Delta x)^2}{2}(u_{xx})_i - \frac{(\Delta x)^3}{6}(u_{xxx})_i + \ldots$$

Consider

$$au_i + bu_{i-1} + cu_{i-2} = (a+b+c)u_i - \Delta x(b+2c)(u_x)_i + \frac{(\Delta x)^2}{2}(b+4c)(u_{xx})_i + O(\Delta x^3)$$

- Solution:

$$a+b+c=0; \quad (b+2c)=-1; \quad b+4c=0$$

$$c = \frac{1}{2}, \ b=-2, \ a=\frac{3}{2}$$

$$(u_x)_i = \frac{3u_i - 4u_{i-1} + u_{i-2}}{2\Delta x} + O(\Delta x^2)$$

Generalized Finite Difference Equation

- Systematic way of generating finite difference equations (FDEs) to different orders
  - ► In contrast to just simple adding and subtracting of the operators

Suppose we are interested to obtain the difference operator for $\frac{\partial u}{\partial x}$ up to order O$(\Delta x)^p$

The idea is to determine values of $\alpha_{-1,}\alpha_0,\alpha_{+1},\cdots$ such that

$$\boxed{u_j^{'}+\alpha_{-1}u_{j-1}+\alpha_0 u_j+\alpha_1 u_{j+1}+\cdots=o(\Delta x)^p}$$

$u_j^{'}$ - Differential operator of interest to have minimum truncation error $O(\Delta x)^p$

- Construction of <u>Taylor Tables</u>
  - ► Matrix representation of the difference relations

- To express u'_j in terms of nodal points at $u_{j-1}, u_j, u_{j+1}, \cdots$ about u_j and present them in a table (see below).
- In general, more nodal points which also means more α's are required
  - if one wants a higher order representation of the differential operator in this case u'_j and is given by *p*.
- Here *p* is the order of approximation. (α's and *p* are to be solved.)

		u_j	u'_j	u''_j	u'''_j
u'_j	$=$	0	1	0	0
$\alpha_{-1}u_{j-1}$	$=$	α_{-1}	$-\Delta x\alpha_{-1}$	$\frac{\Delta x^2}{2}\alpha_{-1}$	$-\frac{\Delta x^3}{6}\alpha_{-1}$
$\alpha_0 u_j$	$=$	α_0	0	0	0
$\alpha_1 u_{j+1}$	$=$	α_1	$\Delta x\alpha_1$	$\frac{\Delta x^2}{2}\alpha_1$	$\frac{\Delta x^3}{6}\alpha_1$
LHS	$=$	RHS1 +	RHS2 +	RHS3 +	RHS4 + …

- Adding up in each column, we have

$$\text{LHS} = \text{RHS1} + \text{RHS2} + \text{RHS3} + \text{RHS4} + \cdots$$

where

$$
\begin{aligned}
LHS &= u_j' + \sum_{k=-1}^{k=1} \alpha_k u_{j+k} \\
RHS1 &= \left[\alpha_{-1} + \alpha_0 + \alpha_1\right] u_j \\
RHS2 &= \left[1 - \Delta x \alpha_{-1} + \Delta x \alpha_1\right] u_j' \\
RHS3 &= \left[\frac{1}{2} \Delta x^2 \left(\alpha_{-1} + \alpha_1\right)\right] u_j'' \\
RHS4 &= \frac{\Delta x^3}{6} \left(\alpha_1 - \alpha_{-1}\right) u_j'''
\end{aligned}
$$

- Now we seek to make as many as possible RHS$_i$ (starting from the left or 1st column and progressing towards the right) to vanish by appropriate choices of α_k.
- Let's say that we have only 3 unknowns $\alpha_{-1}, \alpha_0, \alpha_1$, hence we can set three equations of *RHS* equal to zero (starting from 1st column), i.e.

$$
\begin{aligned}
RHS\,1 &= 0 \\
RHS\,2 &= 0 \\
RHS\,3 &= 0
\end{aligned}
$$

  and hence 3 independent equations with 3 unknowns (α_k).
- Expanding or re-arranging, we therefore have

$$
\begin{pmatrix} 1 & 1 & 1 \\ -1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} \alpha_{-1} \\ \alpha_0 \\ \alpha_1 \end{pmatrix} = \begin{pmatrix} 0 \\ -\dfrac{1}{\Delta x} \\ 0 \end{pmatrix}
$$

Solving, we obtain

$$\boxed{\begin{aligned}\alpha_{-1} &= \frac{1}{2\Delta x}\\ \alpha_0 &= 0\\ \alpha_1 &= -\frac{1}{2\Delta x}\end{aligned}}$$

and the first non-zero term on the right hand side, $RHS4 = -\frac{\Delta x^2}{6} u_j''' $.

Substituting back, we get

$$\overbrace{u_j' - \frac{1}{2\Delta x}\left(u_{j+1} - u_{j-1}\right)}^{LHS} = \overbrace{-\frac{1}{6}\Delta x^2 u_j''' + \ldots}^{\sum_j RHS_j}$$

- The lowest order term on the RHS which is $\overbrace{\frac{1}{6}\Delta x^2 u_j''' + \ldots}^{\sum RHS_j}$ then represents the truncation error for the difference approximation of u_j' (the order *p* of approximation). Hence

$$u_j' = \frac{u_{j+1} - u_{j-1}}{2\Delta x} + O\left(\Delta x\right)^2$$

i.e. *p* = 2 as worked out

- The term $O(\Delta x)^2$ with *p* =2 is therefore 2nd order accurate central finite difference representation for $\left(\frac{\partial u}{\partial x}\right)_j$ .Of course, using the same method, we can construct a 2nd-order <u>one-sided</u> (i.e. setting *k* either to the right or left of *j*) finite difference approximation for $\left(\frac{\partial u}{\partial x}\right)_j$ and others.

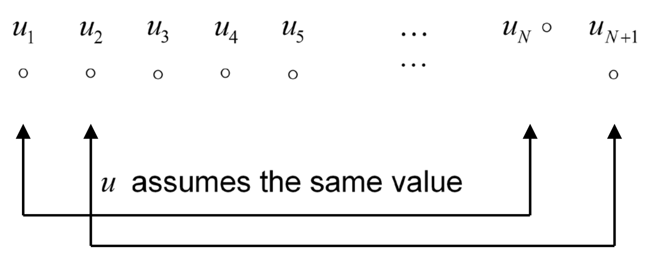
Boundary Conditions for Spatial Discretization

There are three common types of Boundary Conditions (BCs) which are

i) Dirichlet condition: function u is given at boundary points which may or may not depend on time.

ii) Neumann condition: the spatial derivative of function, $\frac{\partial u}{\partial x}$ is given at boundary points.

iii) Periodic: function u is to repeat itself again and again after certain fixed spatial dimension.



A less common type is the mixed boundary condition (also known as Robin boundary condition). For example

$$u(x=2\pi)=u_0+\left.\frac{\partial u}{\partial x}\right|_{x=2\pi}=0$$

Differential vs. Difference Operators

Differential operators are uniquely designated as $\frac{\partial u}{\partial x}$, $\frac{\partial^2 u}{\partial x^2}$, $\frac{\partial u}{\partial t}\cdots$ while difference operators depend on how we choose to discretize the differential operators.

For example, for the differential operator, $\frac{\partial u}{\partial x}$, we can have a few corresponding difference operators or FD scheme such as

- Central difference - $\bar{\delta}_x = \left(\frac{\partial u}{\partial x}\right)_j + O(\Delta x^2)$
- Forward difference - $\vec{\delta}_x = \left(\frac{\partial u}{\partial x}\right)_j + O(\Delta x)$
- Backward difference - $\overleftarrow{\delta}_x = \left(\frac{\partial u}{\partial x}\right)_j + O(\Delta x)$

As mentioned, these difference operators are not unique and have different orders of accuracy.

Example: Heat Diffusion Problem

Consider the generation of FDE of the 1-D diffusion equation

$$\frac{\partial u}{\partial t} = \upsilon \frac{\partial^2 u}{\partial x^2}, \quad x \in [0, \pi]$$

First, we divide the interval $x \in [0, \pi]$ into equal segments of Δx with $N = M + 2$ total number of grid points. Therefore, $\Delta x = \frac{\pi}{M+1} = \frac{\pi}{N-1}$ and the coordinates of the grid points are $x_j = (j-1)\Delta x$ where $j = 1, 2, \ldots, M+2$.

Next, we generate the FDE using truncated Taylor series

$$u_j = u(x_j)$$

$$u_{j+1} = u(x_j + \Delta x) = u_j + \Delta x\, u_j^{(1)} + \frac{\Delta x^2}{2!} u_j^{(2)} + \frac{\Delta x^3}{3!} u_j^{(3)} + \cdots$$

$$u_{j-1} = u(x_j - \Delta x) = u_j - \Delta x\, u_j^{(1)} + \frac{\Delta x^2}{2!} u_j^{(2)} - \frac{\Delta x^3}{3!} u_j^{(3)} + \cdots$$

We can then approximate $u_j^{(2)} \equiv \frac{\partial^2 u}{\partial x^2}$ by appropriate linear combination of u_j, u_{j+1} and u_{j-1} with an error term, ε.

$$u_j^{(2)} + \varepsilon = a u_{j-1} + b u_j + c u_{j+1}$$

Or course, if we wish to have $\varepsilon \to 0$ then we will need an infinite order of terms.
We choose the coefficients a, b and c to eliminate u_j
and $u_j^{(1)}$. Finally, we obtain

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_j + \varepsilon = \frac{u_{j-1} - 2u_j + u_{j+1}}{\Delta x^2} = u_j^{(2)} + \underbrace{\frac{\Delta x^2}{12} u_j^{(4)} + O(\Delta x^4)}_{=\varepsilon}$$

We have:

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_j + \varepsilon = \frac{u_{j-1} - 2u_j + u_{j+1}}{\Delta x^2} = u_j^{(2)} + \underbrace{\frac{\Delta x^2}{12} u_j^{(4)} + O(\Delta x^4)}_{=\varepsilon}$$

So, $\varepsilon \in O(\Delta x^2)$ and $\varepsilon \to 0$ as $\Delta x \to 0$. (Convince yourself this is the case; hint $a = c = 1/\Delta x^2,\ b = -2/\Delta x^2$)

The above FD scheme for the 2nd derivative $u_j^{(2)} = \frac{\partial^2 u}{\partial x^2}$ is sometimes called the 3-point central difference scheme, yielding

$$\left(\frac{\partial u}{\partial t}\right)_j = \frac{\upsilon}{\Delta x^2}\left(u_{j-1} - 2u_j + u_{j+1}\right) + O(\Delta x^2)$$ for $j = 1, 2, \ldots, M$ (interior points)

The boundary condition is either u or $\frac{\partial^2 u}{\partial x^2}$ at $j = 0,\ M+1$

It should be noted that the use of FDE necessarily implied that we are actually solving

$$\frac{\partial u}{\partial t} = \upsilon \frac{\partial^2 u}{\partial x^2} - \frac{\Delta x^2}{12} u_j^{(4)} + \cdots$$

which suggests a discretization error of $O(\Delta x^2)$.

Matrix Form of FDE

We can write out the above FDE at each interior point where $j = 1, 2, \ldots, M$.

$$\begin{aligned}
\frac{du_1}{dt} &= \frac{\upsilon}{\Delta x^2}\left(u_0 - 2u_1 + u_2\right) \\
\frac{du_2}{dt} &= \frac{\upsilon}{\Delta x^2}\left(u_1 - 2u_2 + u_3\right) \\
&\vdots \\
\frac{du_M}{dt} &= \frac{\upsilon}{\Delta x^2}\left(u_{M-1} - 2u_M + u_{M+1}\right)
\end{aligned}$$

In matrix form, we have

$$\frac{d}{dt}\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{M-1} \\ u_M \end{bmatrix} = \frac{\upsilon}{\Delta x^2}\underbrace{\begin{bmatrix} -2 & 1 & 0 & & 0 \\ 1 & -2 & 1 & & \\ 0 & 1 & -2 & \ddots & \\ & & \ddots & \ddots & 1 \\ 0 & & & 1 & -2 \end{bmatrix}}_{=\mathbf{A}}\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{M-1} \\ u_M \end{bmatrix} + \begin{bmatrix} \frac{\upsilon}{\Delta x^2}u_0 \\ 0 \\ \vdots \\ 0 \\ \frac{\upsilon}{\Delta x^2}u_{M+1} \end{bmatrix}$$

- Let $\vec{u} = [u_1 \quad u_2 \quad \cdots \quad u_{M-1} \quad u_M]^T$ and $\overrightarrow{BC} = [\frac{\upsilon}{\Delta x^2}u_0 \quad 0 \quad \cdots \quad 0 \quad \frac{\upsilon}{\Delta x^2}u_{M+1}]^T$. The size of matrix $\mathbf{A}$ is M by M. The difference form of the equation becomes

$$\boxed{\frac{d\vec{u}}{dt} = \mathbf{A}\vec{u} + \overrightarrow{BC}}$$

- We have managed to convert a continuous PDE, $\frac{\partial u}{\partial t} = \upsilon\frac{\partial^2 u}{\partial x^2}$ into a **semi-discrete equation**, $\frac{d\vec{u}}{dt} = \mathbf{A}\vec{u} + \overrightarrow{BC}$.
- Expressing finite difference schemes in vector-matrix forms is useful for numerical analysis
  - finite difference, finite element, finite volume or spectral method schemes

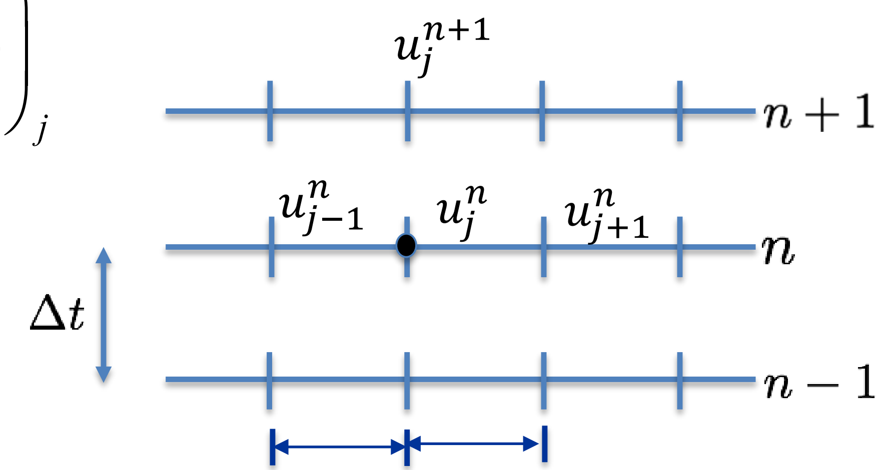
Example

- ❑ A numerical approximation

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial x^2}$$

can be constructed by the

$$\left(\frac{\partial u}{\partial t}\right)_j^n + c\left(\frac{\partial u}{\partial x}\right)_j^n = \nu\left(\frac{\partial^2 u}{\partial x^2}\right)_j^n$$



Example

- Using shorthand notation

$$u_j^n = u\left(t, x_j\right);\ u_j^{n+1} = u\left(t+\Delta t, x_j\right);$$

$$u_{j+1}^n = u\left(t, x_j + h\right); u_{j-1}^n = u\left(t, x_i - h\right)$$

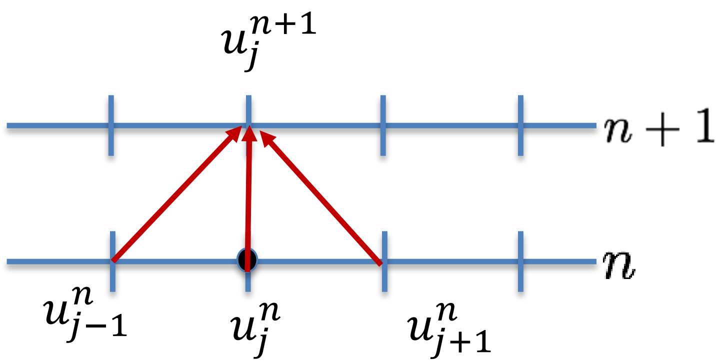
- FDM formulas are:

$$\left(\frac{\partial u}{\partial t}\right)_j^n = \frac{u_j^{n+1} - u_j^n}{\Delta t} + O(\Delta t)$$

$$\left(\frac{\partial u}{\partial x}\right)_j^n = \frac{u_{j+1}^n - u_{j-1}^n}{2h} + O(h^2)$$

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_j^n = \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} + O(h^2)$$

Example



- Substituting into the equation

$$\left(\frac{\partial u}{\partial t}\right)_j^n + c\left(\frac{\partial u}{\partial x}\right)_j^n = \nu\left(\frac{\partial^2 u}{\partial x^2}\right)_j^n$$

gives:

$$\frac{u_j^{n+1}-u_j^n}{\Delta t} + c\,\frac{u_{j+1}^n-u_{j-1}^n}{2h} = \nu\,\frac{u_{j+1}^n-2u_j^n+u_{j-1}^n}{h^2} + O(\Delta t, h^2)$$

- Update formula

$$u_j^{n+1} = u_j^n - \frac{c\Delta t}{2h}\left(u_{j+1}^n - u_{j-1}^n\right) + \frac{\nu\Delta t}{h^2}\left(u_{j+1}^n - 2u_j^n + u_{j-1}^n\right)$$

Coding Example:

We consider the evolution of sine wave, which 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.

```
% MECH 479 - CFD
% EX2: A simple code for three integration methods
% 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;
u=zeros(n,1); y=zeros(n,1); ex=zeros(n,1); time=0.0;
for i=1:n,
    u(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(u,'linewidt',2); axis([1 n -2.0, 2.0]); % plot solution
hold on; plot(ex,'r','linewidt',2);pause; % plot exact solution
y=u; % store the solution
for i=2:n-1,
u(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;
u(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
u(1)=u(n); % periodic boundaries
time=time+dt;
end;
```

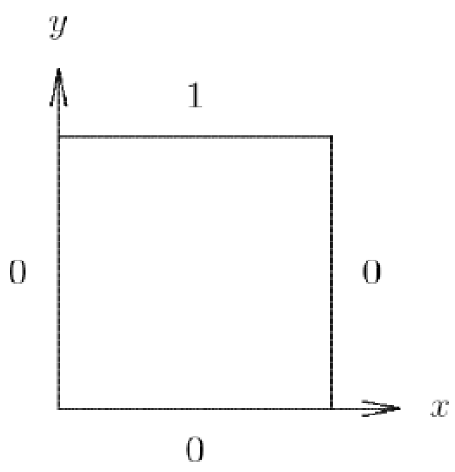
Example: 2D Laplace Equation (Whiteboard)

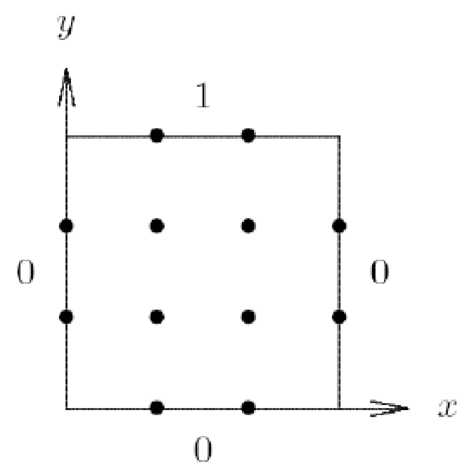
- FDM procedure
  - Define mesh points
  - Replace derivatives by finite difference approximations
  - Seek numerical solution at mesh points

- Consider Laplace equation

$$u_{xx} + u_{yy} = 0$$

on unit square with boundary conditions shown below left



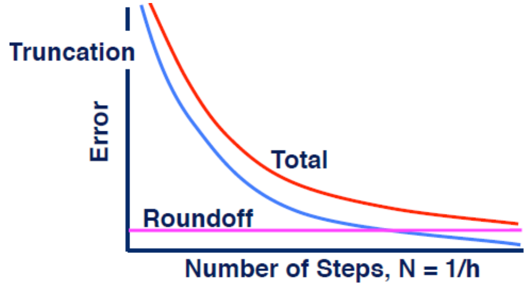


NUMERICAL ANALYSIS OF FINITE DIFFERENCE APPROXIMATIONS

Basic Numerical Properties

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

Numerical Errors



❑ Round-off error

- ► Errors in computer representation of numbers
  - ◇ Due to finite precision, a finite number of digits in the arithmetic operations
- ► Instead of true infinite continuum of real numbers, there must be a maximum or minimum number

$$relative\ round\text{-}off\ \ error = \frac{|true\ number - computer\ \ number|}{|true\ number|} \leq \epsilon$$

❑ Discretization error

- ► Replacing continuous problem by a discrete one
- ► Errors in computer representation of functions and function operators, e.g.

$$error = u(x) - u_{num}(x)$$

- ► The difference between the exact solution of PDE (round-off free) and the exact solution of the FDEs (round-off free)
  - ◇ Errors due to the truncation error
  - ◇ Any errors introduced by the treatment of boundary conditions

Consistency:

Consistency defines the extent to which the FDEs approximate the PDEs. A finite difference representation of a PDE is said to be **consistent** if

$$\lim_{\text{mesh}\to 0} \text{TE} = \lim_{\text{mesh}\to 0} \text{PDE} - \text{FDE} = 0$$

Order of accuracy (OOA):

This term refers to the order of the truncation error. For example, if we consider the FD corresponding to the first derivative of the function. Using **backward** difference, where $\frac{\partial u}{\partial x} \approx \frac{u_i - u_{i-1}}{\Delta x}$

$$\text{TE} = \frac{1}{2}\frac{\partial^2 u}{\partial x^2}(\Delta x)^1$$

Hence, the OOA of the backward difference method is order 1, i.e. $O(\Delta x)$

Order of accuracy (OOA):

If we use **central** difference instead, where $\frac{\partial u}{\partial x} \approx \frac{u_{i+1} - u_{i-1}}{2\Delta x}$ then

$$\mathrm{TE} = -\frac{1}{3}\frac{\partial^3 u}{\partial x^3}(\Delta x)^2$$

The OOA of the central difference method is order 2, i.e. $O((\Delta x)^2)$

Furthermore, the OOA of a PDE takes the lowest OOA of each derivative approximation. For example if we have:

$$\frac{\partial u}{\partial t} = \upsilon \frac{\partial^2 u}{\partial x^2}$$

Say we use backward difference method for the $\frac{\partial u}{\partial t}$ term where OOA is order 1 and we use the central difference for the $\frac{\partial^2 u}{\partial x^2}$ term (with OOA = 2), then the OOA of the whole PDE is equal to order 1.

Accuracy: Example

- Scalar conservation equation

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

- First-order

$$\frac{\partial u}{\partial t}\left(x,t^{n}\right)=\frac{u\left(x,t^{n+1}\right)-u\left(x,t^{n}\right)}{\Delta t}+O(\Delta t)$$

$$\frac{\partial f}{\partial x}\left(x_i,t\right)=\frac{f\left(u\left(x_{i+1},t\right)\right)-f\left(u\left(x_{i},t\right)\right)}{\Delta x}+O(\Delta x)$$

- Second-order space and time

$$\frac{\partial u}{\partial t}\left(x,t^{n}\right)=\frac{u\left(x,t^{n+1}\right)-u\left(x,t^{n-1}\right)}{2\Delta t}+O(\Delta t^2)$$

$$\frac{\partial f}{\partial x}\left(x_i,t\right)=\frac{f\left(u\left(x_{i+1},t\right)\right)-f\left(u\left(x_{i-1},t\right)\right)}{2\Delta x}+O(\Delta x^2)$$

Consistency

- Using the finite difference equation, we are effectively solving an equation that is slightly different than the original partial differential equations.
  - Does the FDE approach to the PDE in the limit of vanishing time step and grid size?
  - Truncation errors should vanish as the mesh size and time step tend to zero

Discretized equation $\longleftrightarrow$ Differential equation

Example: Consistency

1D Convection PDE: $\dfrac{\partial u}{\partial t} + c\dfrac{\partial u}{\partial x} = 0$

Central difference in space and forward Euler in time

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + c\frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} = O\left[(\Delta t)^q, (\Delta x)^p\right]$$

Taylor series expansions: $u_i^{n+1} = u_i^n + \Delta t\left(\dfrac{\partial u}{\partial t}\right)_i^n + \dfrac{\Delta t^2}{2}\left(\dfrac{\partial^2 u}{\partial t^2}\right)_i^n +$

$$u_{i\pm1}^n = u_i^n \pm \Delta x\left(\frac{\partial u}{\partial x}\right)_i^n + \frac{\Delta x^2}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)_i^n \pm \frac{\Delta x^2}{6}\left(\frac{\partial^3 u}{\partial x^3}\right)_i^n + \ldots$$

Hence

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + c\frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} - \left(\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x}\right)_i^n + \epsilon_\tau = 0$$

$$\epsilon_\tau = -\frac{\Delta t}{2}\left(\frac{\partial^2 u}{\partial t^2}\right)_i^n - c\frac{\Delta x^2}{6}\left(\frac{\partial^3 u}{\partial x^3}\right)_i^n + O\left[(\Delta t)^2, (\Delta x)^4\right]$$

ϵ_τ is termed as the truncation error.

The Dufort-Frankel Differencing

- 1D Diffusion PDE: $$\frac{\partial u}{\partial t} = \upsilon \frac{\partial^2 u}{\partial x^2}.$$

- The FD scheme we are going to use is called Dufort-Frankel scheme and is given as

$$u_j^{n+1} = u_j^{n-1} + \frac{2\upsilon \Delta t}{\Delta x^2}\left[u_{j-1}^n - (u_j^{n+1} + u_j^{n-1}) + u_{j+1}^n\right]$$

- Using Taylor series to expand the terms u_j^{n+1}, u_j^{n-1}, u_{j-1}^n and u_{j+1}^n about point $(x = x_j,\ t = t_n)$ and substituting them into the FD scheme gives an equivalent PDE as follows

$$\frac{\partial u}{\partial t} = \upsilon \frac{\partial^2 u}{\partial x^2} - \upsilon \left(\frac{\Delta t}{\Delta x}\right)^2 \frac{\partial^2 u}{\partial t^2} + O(\Delta t^2) + O(\Delta x^2)$$

- We find that in general, this is **not consistent** with our original diffusion PDE since

$$\lim_{\Delta t, \Delta x \to 0} \frac{\Delta t}{\Delta x} \neq 0$$

Numerical Stability

A stable FD scheme means that any error that arises from sources such as round-off, truncation or 'approximation' are <u>not permitted to grow</u> in the sequence of numerical computation. Basically, stability involves the actual solution and exact solution of an FDE.

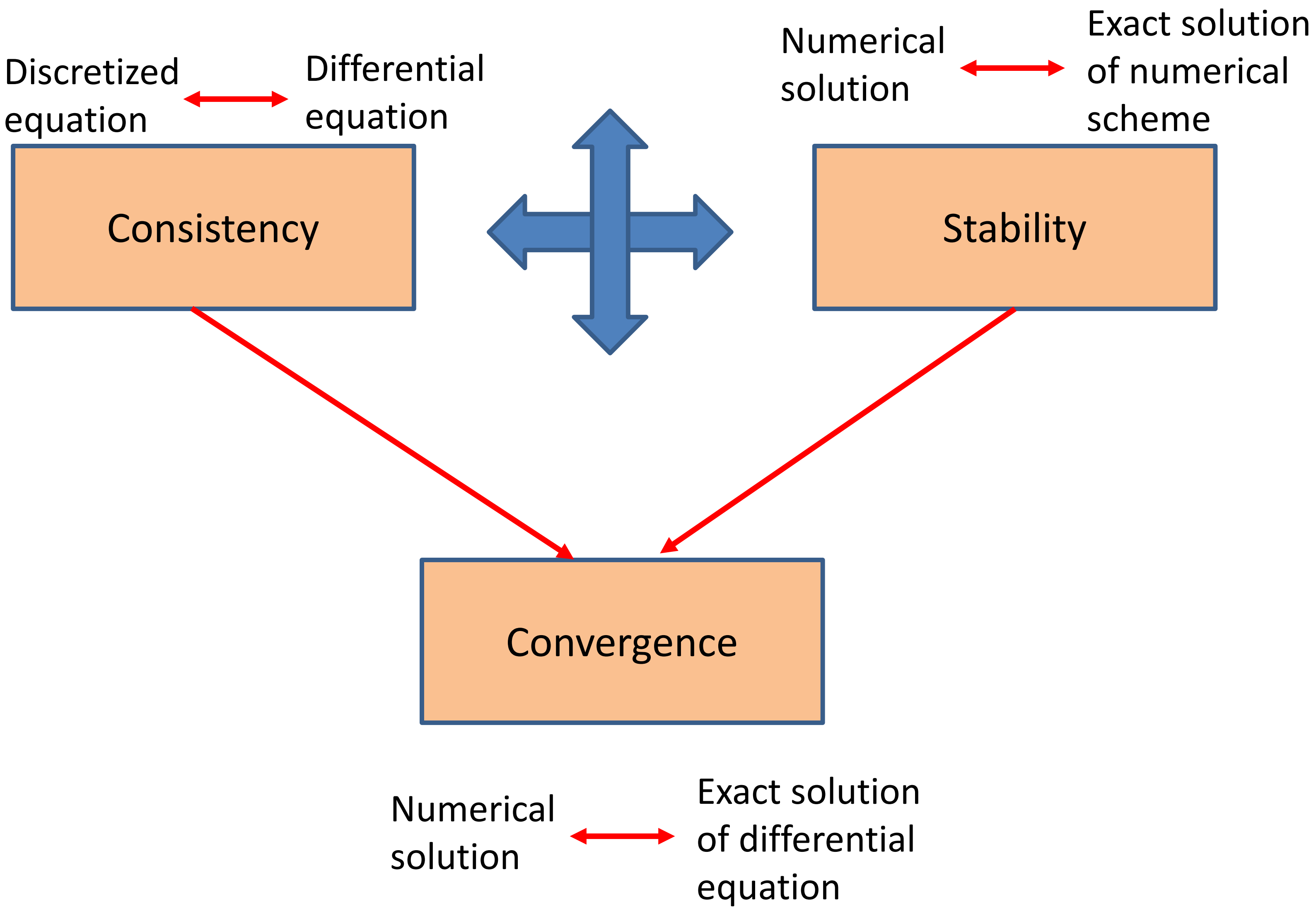
Convergence

The solution to the FDE should approach <u>the true solution</u> of the PDE (provided they have the same initial and boundary conditions) as the mesh or time step is refined.

The determination of convergence involves exact solutions of PDE and FDE. Usually, it is very difficult to show that a FD scheme would or would not converge at all since the exact solution to the PDE is not known. This difficulty can be removed by using Lax's equivalence theorem.

Consistency, Stability and Convergence

- Lax Equivalence Theorem



Lax’s Equivalence Theorem

- For a linear, well-posed initial value problem, a consistent FD approximation is convergent if and only if it is stable.

- With this theorem, we only need to prove that a FD method is stable to guarantee convergence.

- Although the Lax’s equivalence theorem is only proven for linear initial value problems, it has been widely applied to nonlinear and boundary value/initial-boundary value problems in engineering.

Summary (1)

- In fully discrete finite difference method:
  - ► replace continuous domain of equation by discrete mesh of points
  - ► replace all derivatives in PDE by finite difference approximations (i.e., FDEs)
  - ► determine numerical solution as table of approximate values at selected points in space and time
- Accuracy of approximate solution depends on step sizes in both space and time

Summary (2)

- Replacement of all partial derivatives by finite differences results in system of algebraic equations for unknown solution at discrete set of sample points
  - Discrete system may be linear or nonlinear, depending on underlying PDE

- Fully discrete methods for time-dependent PDEs discretize in both space and time
  - Lax Equivalence Theorem: Consistency and stability are together necessary and sufficient for convergence as step sizes in space and time jointly go to zero

Rajeev K. Jaiman
Email: rjaiman@mech.ubc.ca