Lecture 5

Discretization & Numerical Differentiation

역학을 다루려면 미분이란 개념이 반드시 필요 !
미분에 대한 정의가 필요하다

Jung-Il Choi

School of Mathematics and Computing (Computational Science and Engineering)

YONSEI UNIVERSITY

1. Discretization methods

- **Basic concept**

➔ Approximation of derivatives

$$\frac{d}{dx}u(\boldsymbol{x}) = \lim_{h\to 0}\frac{u(\boldsymbol{x}+h)-u(\boldsymbol{x})}{h} \sim \frac{u(\boldsymbol{x}+h)-u(\boldsymbol{x})}{h} \quad + O(h)$$

➔ Approximation of functions **Weighted sum 하면 되겠다!!**

$$u(x) = \sum_{i=0}^{\infty} \underline{u_i}\phi_i(x) \sim \sum_{i=0}^{n} u_i\phi_i(x)$$

이산화!

- **Truncation error**

➔ Let L be a linear operator and L_h be an approximation then,

$$\|(L_h - L)u\|_\infty \leq \sum_{i=0}^{\infty} c_i h^i$$

➔ Truncation error is said to be order k if k is the smallest integer for which $c_k \neq 0$.

1. Discretization methods

- Finite difference methods (straightforward to apply, usually for regular grid) and finite volumes and finite element methods (usually for irregular meshes)

- Each type of methods above yields the same solution if the grid is fine enough. However, some methods are more suitable to some cases than others

- Finite difference methods for spatial derivatives with different order of accuracies can be derived using Taylor expansions, such as 2nd order upwind scheme, central differences schemes, etc.

2차 정확도 까진 보장되어야 한다.
이유는 2차 가공 시 error를 줄이기 위해서

- Higher order numerical methods usually predict higher order of accuracy for CFD, but more likely unstable due to less numerical dissipation

- Temporal derivatives can be integrated either by the explicit method (Euler, Runge-Kutta, etc.) or implicit method (e.g., Beam-Warming method)

1. Discretization methods

- Explicit methods can be easily applied but yield conditionally stable Finite Different Equations (FDEs), which are restricted by the time step; Implicit methods are unconditionally stable, but need efforts on efficiency. 공간에선 사용하지 않음, 시간에서 고려함

- Usually, higher-order temporal discretization is used when the spatial discretization is also of higher order.

- Stability: A discretization method is said to be stable if it does not magnify the errors that appear in the course of numerical solution process.

- Pre-conditioning method is used when the matrix of the linear algebraic system is ill-posed, such as multi-phase flows, flows with a broad range of Mach numbers, etc.

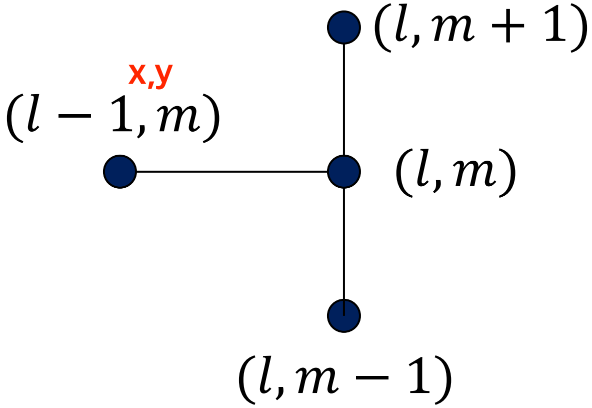
- Selection of discretization methods should consider efficiency, accuracy and special requirements, such as shock wave tracking.

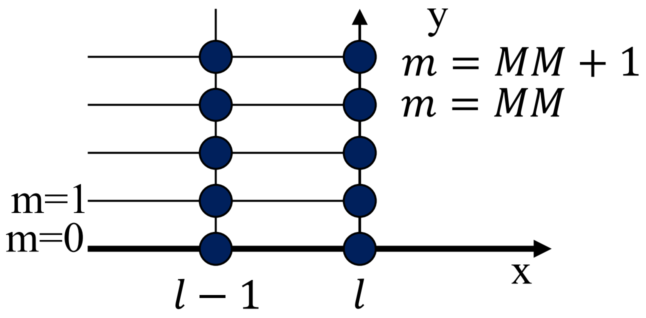
1. Discretization methods

- **2D incompressible laminar flow boundary layer**

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

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{\partial}{\partial x}\left(\frac{p}{e}\right) + \mu\frac{\partial^2 u}{\partial y^2}$$





$$\boxed{u\frac{\partial u}{\partial x} = \frac{u_m^l}{\Delta x}\left[u_m^l - u_m^{l-1}\right]}$$

$$\boxed{\mu\frac{\partial^2 u}{\partial y^2} = \frac{\mu}{\Delta y^2}\left[u_{m+1}^l - 2u_m^l + u_{m-1}^l\right]}$$

$$v\frac{\partial u}{\partial y} = \frac{v_m^l}{\Delta y}\left[u_{m+1}^l - u_m^l\right]$$

FD Sign(v_m^l)<0

$$= \frac{v_m^l}{\Delta y}\left[u_m^l - u_{m-1}^l\right]$$

BD Sign(v_m^l)>0

2nd order central difference
i.e., theoretical order of accuracy
P_{kest}= 2.

1st order upwind scheme, i.e., theoretical order of accuracy P_{kest}= 1

바람에 따라 정보량이 달라짐
예) 왼쪽 -> 오른쪽 or 오른쪽 -> 왼쪽

1. Discretization methods

$$\overset{\color{red}{B_2}}{\left[\frac{u_m^l}{\Delta x} + v_m^l \frac{-\frac{1}{\Delta y}FD}{\frac{1}{\Delta y}BD} - \frac{2\mu}{\Delta y^2}\right]} u_m^l + \overset{\color{red}{B_3}}{\left[\frac{\mu}{\Delta y^2} + \frac{v_m^l}{\Delta y}FD\right]} u_{m+1}^l + \overset{\color{red}{B_1}}{\left[\frac{\mu}{\Delta y^2} - \frac{v_m^l}{\Delta y}BD\right]} u_{m-1}^l = \overset{\color{red}{B_4}}{\frac{u_m^l}{\Delta x}} u_m^{l-1} - \frac{\partial}{\partial x}(p/e)_m^l$$

$$B_1 u_{m-1}^l + B_2 u_m^l + B_3 u_{m+1}^l = B_4 u_m^{l-1} - \frac{\partial}{\partial x}(p/e)_m^l$$

$$\begin{bmatrix} B_2 & B_3 & 0 & 0 & 0 & 0 & 0 & 0 \\ B_1 & B_2 & B_3 & 0 & 0 & 0 & 0 & 0 \\ & & \bullet & \bullet & \bullet & \bullet & & \\ 0 & 0 & 0 & 0 & 0 & B_1 & B_2 & B_3 \\ 0 & 0 & 0 & 0 & 0 & 0 & B_1 & B_2 \end{bmatrix} \times \begin{bmatrix} u_1^l \\ \bullet \\ \bullet \\ \bullet \\ u_{mm}^l \end{bmatrix} = \begin{bmatrix} B_4 u_1^{l-1} - \frac{\partial}{\partial x}\left(\frac{p}{e}\right)_1^l \\ \bullet \\ \bullet \\ \bullet \\ B_4 u_{mm}^{l-1} - \frac{\partial}{\partial x}\left(\frac{p}{e}\right)_{mm}^l \end{bmatrix}$$

Solve it using Thomas algorithm

To be stable, Matrix has to be Diagonally dominant.

2. Numerical Differentiation – Finite Difference

- **Basic Ideas**
  - Construction of Difference Formulas Using Taylor Series
    - ➔ The derivative of $f(x)$ at the point x_j

$$f(x_{j+1}) = f(x_j) + (x_{j+1} - x_j)f'(x_j) + \frac{(x_{j+1} - x_j)^2}{2} f''(x_j) + \cdots$$

    - ➔ We want to calculate $f'(x)$ so that,

$$f'(x_j) = \frac{f(x_{j+1}) - f(x_j)}{x_{j+1} - x_j} - \frac{(x_{j+1} - x_j)}{2} f''(x_j) + \cdots$$
$$= \frac{f(x_{j+1}) - f(x_j)}{\Delta x_j} - \underline{\frac{\Delta x_j}{2} f''(x_j)} + \cdots = \frac{f(x_{j+1}) - f(x_j)}{\Delta x_j} + O(\Delta x_j)$$

leading error

f’’이 finite 하기만 하다면
뒤에는 무시 가능
h를 무한히 줄일 수 있으니까.
그래서 leading error라고 함

    - ➔ Where $\Delta x_j = x_{j+1} - x_j$ is the mesh size

2. Numerical Differentiation – Finite Difference

• Difference formulas and order of error

- first-order difference

➔ Let h be the uniform mesh size(spacing) and $f(x_j) = f_j$, then,

✓ The first-order *forward difference* with the *first-order truncation error* formular

$$f_j' = \frac{f_{j+1} - f_j}{h} + O(h)$$

1차 정확도에 대한 error

✓ The first-order *backward difference* with the *first-order truncation error* formular

$$f_j' = \frac{f_j - f_{j-1}}{h} + O(h)$$

✓ The first-order *central difference* with the *second-order truncation error* formular

$$f_{j+1} = f_j + hf_j' + \frac{h^2}{2!} f_j'' + \frac{h^3}{3!} f_j^{(3)} + \cdots$$

2차 정확도에 대한 error

$$f_{j-1} = f_j - hf_j' + \frac{h^2}{2!} f_j'' - \frac{h^3}{3!} f_j^{(3)} + \cdots$$

$$f_j' = \frac{f_{j+1} - f_{j-1}}{2h} - \frac{h^2}{3!} f_j^{(3)} + \cdots = \frac{f_{j+1} - f_{j-1}}{2h} + O(h^2)$$

Ocsillation이 있어
(disversion error가 많음)

2. Numerical Differentiation – Finite Difference

- **Difference formulas and order of error**

➔ Then, how can we construct the *fourth-order truncation error* for the first-order difference formular? (DIY)

$$f_j' = \frac{f_{j+2} - 8f_{j+1} + 8f_{j-1} - f_{j-2}}{12h} + O(h^4)$$

- Second-order central difference

$$f_j'' = \frac{f_{j+1}' - f_{j-1}'}{h} + O(h^2) = \left[\frac{f_{j+1} - f_j}{h} - \frac{f_j - f_{j-1}}{h}\right]/h + O(h^2) = \frac{f_{j+1} - 2f_j + f_{j-1}}{h^2} + O(h^2)$$

$$f_j'' \sim \frac{f_{j+1} - 2f_j + f_{j-1}}{h^2}$$

2. Numerical Differentiation

- **General Technique for Construction of Finite Difference Schemes**

(feat. Taylor Series Expansion)

- Richardson's extrapolation

➔ To construct the most accurate difference scheme for the first-order that involves the functional values at points j, $j+1$, and $j+2$.

✓ The desired finite difference formula can be written as

$$f_j' + \sum_{k=0}^{2} a_k f_{j+k} = O(?)$$

TAYLOR TABLE

	f_j	f_j'	f_j''	f_j'''
f_j'	0	1	0	0
$a_0 f_j$	a_0	0	0	0
$a_1 f_{j+1}$	a_1	$a_1 h$	$a_1 \frac{h^2}{2}$	$a_1 \frac{h^3}{6}$
$a_2 f_{j+2}$	a_2	$2ha_2$	$a_2 \frac{(2h)^2}{2}$	$a_2 \frac{(2h)^3}{6}$

$$f_{j+1} = f_j + hf_j' + \frac{h^2}{2}f_j'' + \frac{h^3}{6}f_j''' + \cdots$$

$$f_{j+2} = f_j + 2hf_j' + \frac{(2h)^2}{2}f_j'' + \frac{(2h)^3}{6}f_j''' + \cdots$$

• General Technique for Construction of Finite Difference Schemes

(feat. Taylor Series Expansion)

- Richardson's extrapolation

$$a_0 f_j + a_1 f_{j+1} + a_2 f_{j+2} = a_0 f_j + \left(a_1 f_j + a_1 h f_j' + a_1 \frac{h^2}{2!} f_j'' + \cdots\right) + \left(a_2 f_j + a_2 (2h) f_j' + a_2 \frac{(2h)^2}{2!} f_j'' + \cdots\right)$$

$$f_j' + \sum_{k=0}^{2} a_k f_{j+k} = (a_0 + a_1 + a_2) f_j + (1 + a_1 h + 2h a_2) f_j' + \left(a_1 \frac{h^2}{2!} + a_2 \frac{(2h)^2}{2!}\right) f_j'' + \underline{\left(a_1 \frac{h^3}{3!} + a_2 \frac{(2h)^3}{3!}\right) f_j^{(3)} + \cdots}$$

➔ Three unknowns with three equations,

$$a_0 + a_1 + a_2 = 0$$
$$a_1 h + 2h a_2 = -1$$
$$a_1 \frac{h^2}{2} + 2 a_2 h^2 = 0$$

$$a_0 = \frac{3}{2h}, \qquad a_1 = -\frac{2}{h}, \qquad a_2 = \frac{1}{2h}$$

➔ Forward second-order scheme for the first derivative

$$f_j' = \frac{-3 f_j + 4 f_{j+1} - f_{j+2}}{2h} + O(h^2)$$

애초에 함숫값이 정확도가 떨어지면,
결과도 정확도가 떨어짐

$$f_{j-1} = f_j - hf_j' + \frac{h}{2} f_j'' - \frac{h}{6} f_j''' + \cdots$$

2. Numerical Differentiation

• General Technique for Construction of Finite Difference Schemes

(feat. Taylor Series Expansion)

- Padé Approximation

차수가 높아 질 수록, 정보가 많이 필요하니까 한 점을 기준으로
systematic 하게 구해보자! 라는 관점

$$f_j' + a_0 f_j + a_1 f_{j+1} + a_2 f_{j-1} + a_3 f_{j+1}' + a_4 f_{j-1}' = O(?)$$

TAYLOR TABLE FOR A PADÉ SCHEME

	f_j	f_j'	f_j''	f_j'''	$f_j^{(iv)}$	$f_j^{(v)}$
f_j'	0	1	0	0	0	0
$a_0 f_j$	a_0	0	0	0	0	0
$a_1 f_{j+1}$	a_1	$a_1 h$	$a_1 \frac{h^2}{2}$	$a_1 \frac{h^3}{6}$	$a_1 \frac{h^4}{24}$	$a_1 \frac{h^5}{120}$
$a_2 f_{j-1}$	a_2	$-a_2 h$	$a_2 \frac{h^2}{2}$	$-a_2 \frac{h^3}{6}$	$a_2 \frac{h^4}{24}$	$-a_2 \frac{h^5}{120}$
$a_3 f_{j+1}'$	0	a_3	$a_3 h$	$a_3 \frac{h^2}{2}$	$a_3 \frac{h^3}{6}$	$a_3 \frac{h^4}{24}$
$a_4 f_{j-1}'$	0	a_4	$-a_4 h$	$a_4 \frac{h^2}{2}$	$-a_4 \frac{h^3}{6}$	$a_4 \frac{h^4}{24}$

$$\begin{cases} a_0 + a_1 + a_2 = 0 \\ a_1 h - a_2 h + a_3 + a_4 = -1 \\ a_1 \dfrac{h^2}{2} + a_2 \dfrac{h^2}{2} + a_3 h + a_4 h = 0 \\ a_1 \dfrac{h}{3} - a_2 \dfrac{h}{3} + a_3 + a_4 = 0 \\ a_1 \dfrac{h}{4} + a_2 \dfrac{h}{4} + a_3 - a_4 = 0. \end{cases}$$

$$a_0 = 0, \quad a_1 = -\frac{3}{4h}, \quad a_2 = \frac{3}{4h}, \quad a_3 = a_4 = \frac{1}{4}$$

2. Numerical Differentiation

• General Technique for Construction of Finite Difference Schemes

(feat. Taylor Series Expansion)

- Padé Approximation

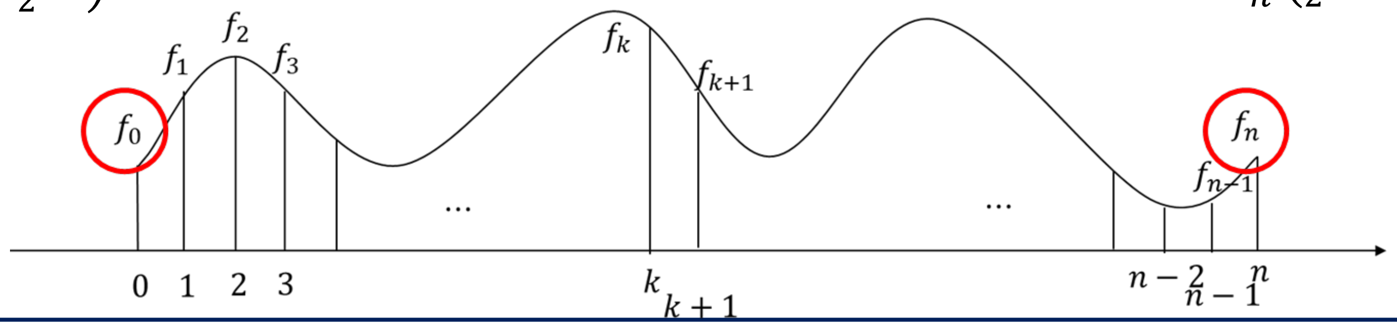
$$f'_{j+1} + f'_{j-1} + 4f'_j = \frac{3}{h}\left(f_{j+1} - f_{j-1}\right) + \frac{h^4}{30} f_j^{(5)} + \cdots = \frac{3}{h}\left(f_{j+1} - f_{j-1}\right) + O(h^4)$$

System of equations

➔ Special treatment is required near the boundaries, lower order one-sided difference formulas are used to approximate f'_0 and f'_n. (DIY, Derive it yourself)

$$f'_0 + 2f'_1 = \frac{1}{h}\left(-\frac{5}{2}f_0 + 2f_1 + \frac{1}{2}f_2\right) + O(h^3)$$

$$f'_n + 2f'_{n-1} = \frac{1}{h}\left(\frac{5}{2}f_n - f_{n-1} - \frac{1}{2}f_{n-2}\right) + O(h^3)$$



• General Technique for Construction of Finite Difference Schemes

(feat. Taylor Series Expansion)

- Padé Approximation TDMA

$$\begin{bmatrix} 1 & 2 & 0 & 0 & 0 & \cdots & 0 \\ 1 & 4 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 4 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 1 & 4 & 1 \\ 0 & 0 & 0 & 0 & \cdots & 2 & 1 \end{bmatrix} \begin{bmatrix} f_0' \\ f_1' \\ f_2' \\ \vdots \\ \vdots \\ f_{n-1}' \\ f_n' \end{bmatrix} = \frac{1}{h} \begin{bmatrix} -\frac{5}{2}f_0 + 2f_1 + \frac{1}{2}f_2 \\ 3(f_2 - f_0) \\ 3(f_3 - f_1) \\ \vdots \\ \vdots \\ 3(f_n - f_{n-2}) \\ \frac{5}{2}f_n - 2f_{n-1} - \frac{1}{2}f_{n-2} \end{bmatrix}.$$

➔ For the three-point central stencil the following fourth-order truncation error formula

$$\frac{1}{12}f''_{i-1} + \frac{10}{12}f_i + \frac{1}{12}f''_{i+1} = \frac{(f_{i+1} - 2f_i + f_{i-1})}{h^2}$$

2. Numerical Differentiation

• General Technique for Construction of Finite Difference Schemes

(feat. Lagrange Interpolation)

- Recall, the basic concepts of discretization methods are

➔ Approximation of derivatives

$$\frac{d}{dx}u(\boldsymbol{x}) = \lim_{h\to\infty}\frac{u(\boldsymbol{x}+h)-u(\boldsymbol{x})}{h} \sim \frac{u(\boldsymbol{x}+h)-u(\boldsymbol{x})}{h}$$

➔ Approximation of functions

$$u(x) = \sum_{i=0}^{\infty} u_i\phi_i(x) \sim \sum_{i=0}^{n} u_i\phi_i(x)$$

- We could take an approach using the interpolation methods with the interpolation error

(The concepts of interpolation will be dealt with the next lecture.)

$$f(x) = p(x) + \frac{1}{(n+1)!}f^{n+1}(\xi_x)\omega(x), \qquad \text{where } \omega(x) = \prod_{i=0}^{n}(x-x_i)$$

2. Numerical Differentiation

• General Technique for Construction of Finite Difference Schemes

(feat. Lagrange Interpolation)

➔ The derivative of ω is given by,

$$\omega'(x) = \sum_{i=0}^{n} \prod_{\substack{j=0 \\ j\neq i}}^{n} (x - x_j) \quad \Rightarrow \quad \omega'(x_k) = \prod_{\substack{j=0 \\ j\neq k}}^{n} (x_k - x_j)$$

➔ Since $\omega(x_k) = 0$,

$$\begin{aligned} f'(x_k) &= p'(x_k) + 0 + \frac{1}{(n+1)!} f^{(n+1)}(\xi_x)\omega'(x_k) \\ &= p'(x_k) + \frac{1}{(n+1)!} f^{(n+1)}(\xi_x) \prod_{\substack{j=0 \\ j\neq k}}^{n} (x_k - x_j) \end{aligned}$$

2. Numerical Differentiation

• General Technique for Construction of Finite Difference Schemes

(feat. Lagrange Interpolation)

- Let consider the Lagrange interpolation (in other words, the approximations based on Lagrange basis) for n discretization ($n+1$ points) on $[a, b]$.

$$\boxed{l_i(x) = \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j}}$$

$$f'(x_k) = \sum_{i=0}^{n} f(x_i) l_i'(x_i) + \frac{1}{(n+1)!} f^{(n+1)}(\xi_x) \prod_{\substack{j=0 \\ j\neq k}}^{n} (x_k - x_j)$$

함수를 모두 polynomial 입장으로 풀고 있다!

- If n = 2, then the values of derivatives at point x_1 is

$$f'(x_1) = f(x_0)\frac{x_1 - x_2}{(x_0 - x_1)(x_0 - x_2)} + f(x_1)\frac{2x_1 - x_0 - x_2}{(x_1 - x_0)(x_1 - x_2)} + f(x_2)\frac{x_1 - x_0}{(x_2 - x_0)(x_2 - x_1)} + E(\xi_x)$$

➔ Let consider the uniform mesh spacing with h, then,

$$f'(x_1) = -\frac{1}{3h} f(x_0) - \frac{1}{h} f(x_1) + \frac{4}{3h} f(x_2) + O(h^2)$$

$$y - y_0 = \frac{y_1 - y_0}{x_1 - x_0}(x - x_0)$$

$\frac{y-y_0}{y_1-y_0} = \frac{x-x_0}{x_1-x_0}$ 를 이용해서 cardinality function (Lagrange basis)를 만들 수 있음

• Alternative Measure for the Accuracy of Finite Differences

(feat. Fourier Series Expansion) sin, cos으로 basis 잡는 것

- The modified wavenumber approach
  - ➔ Arbitrary periodic functions can be decomposed into their Fourier components
  - ➔ Consider a pure harmonic function of period L,

$$f(x) = c_k e^{ikx}, \qquad \text{where } k \text{ is the wavenumber (or frequency)}$$

$$k = \frac{2\pi}{L} n, \qquad n = 0, 1, \dots, N/2$$

  - ➔ The exact derivative is $f' = ikc_k f$
  - ➔ Let us discretize a portion of the x axis of length L with a uniform mesh, and $h = L/N$ is the mesh size

$$x_j = \frac{L}{N} j = hj, \qquad j = 0, 1, 2, \dots, N-1$$

$$\left.\frac{\delta f}{\delta x}\right|_j = \frac{f_{j+1} - f_{j-1}}{2h} = \frac{e^{i2\pi n(j+1)/N} - e^{i2\pi n(j-1)/N}}{2h} = \frac{e^{i2\pi n/N} - e^{-i2\pi n/N}}{2h} f_j$$

$$\left.\frac{\delta f}{\delta x}\right|_j = i\frac{\sin(2\pi n/N)}{h} f_j = ik' f_j, \qquad \text{where } k' = \frac{\sin(2\pi n/N)}{h}$$

2. Numerical Differentiation

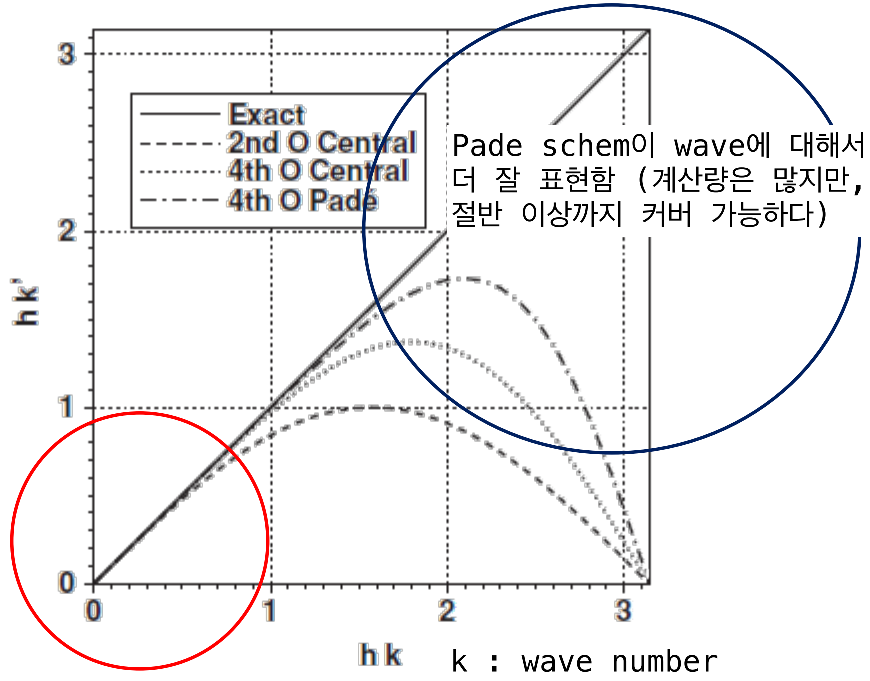
• Alternative Measure for the Accuracy of Finite Differences

(feat. Fourier Series Expansion)

● The modified wavenumber approach for three finite difference schemes

➔ This is expected because for small values of k, f is slowly varying and the finite difference scheme is sufficiently accurate for numerical differentiation.

➔ For higher value of k, however, f varies rapidly in the domain, and the finite difference scheme provides a poor approximation for its derivative.

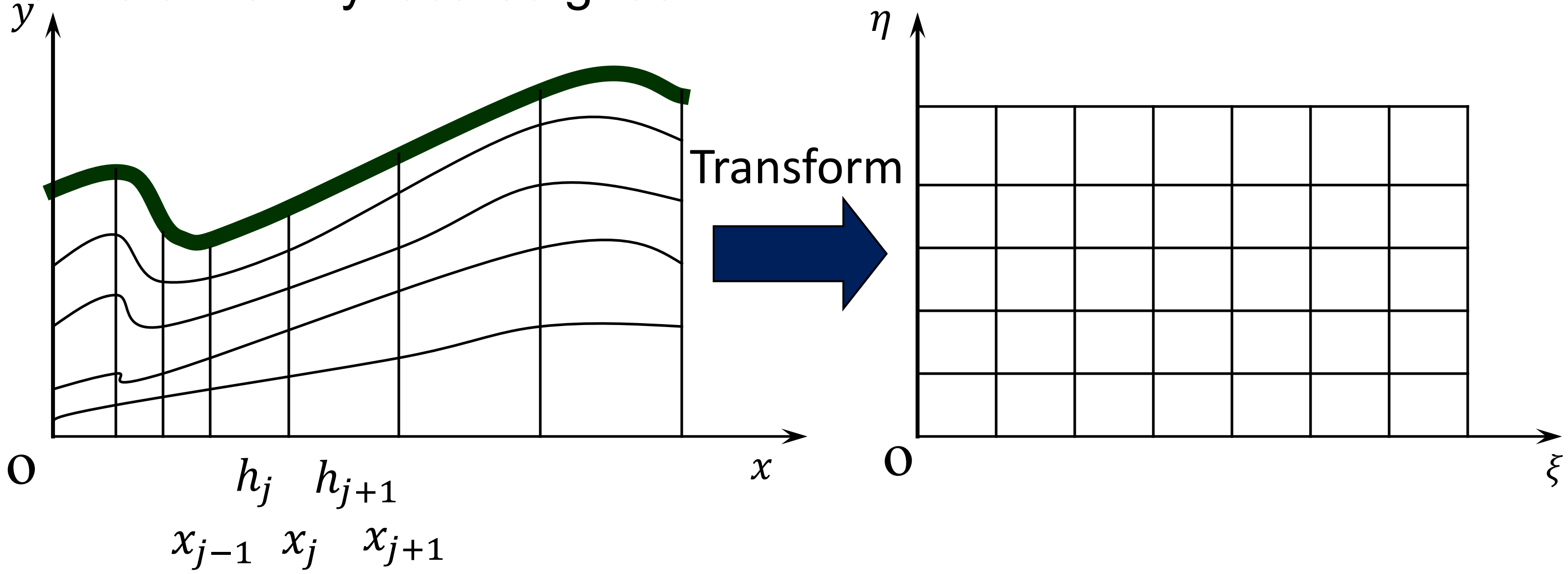


- **Non-uniform grids**
  - Strictly a first-order approximation whereas its counterpart on a uniform mesh is second-order accurate.

$$f_j' = \frac{f_{j+1} - f_{j-1}}{x_{j+1} - x_{j-1}} = \frac{f_{j+1} - f_{j-1}}{h_{j+1} + h_j} + O(h^2)$$

$$f_j'' = 2\left[\frac{f_{j-1}}{h_j(h_j + h_{j+1})} - \frac{f_j}{h_j h_{j+1}} + \frac{f_{j+1}}{h_{j+1}(h_j + h_{j+1})}\right] + O(h^2)$$

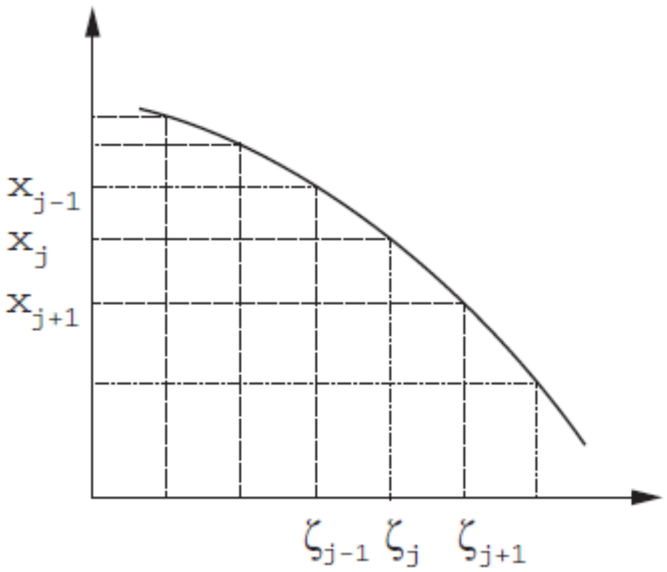
  - Transformation to the uniformly located grids.



2. Numerical Differentiation

- **Non-uniform grids**

➔ Uniform mesh spacing in ζ corresponds to non-uniform mesh spacing in x.



$$\zeta = g(x)$$

$$\frac{df}{dx} = \frac{d\zeta}{dx}\frac{df}{d\zeta} = g'\frac{df}{d\zeta}$$

$$\frac{d^2 f}{dx^2} = \frac{d}{dx}\left[g'\frac{df}{d\zeta}\right] = g''\frac{df}{d\zeta} + (g')^2\frac{d^2 f}{d\zeta^2}$$

- Finite difference approximation for uniform meshes are then used to approximate $\frac{df}{d\zeta}$ and $\frac{d^2 f}{d\zeta^2}$.

Q&A Thanks for listening