REVIEW NO. 2

NUMERICAL DIFFERENTIATION

- Find a ***discrete*** approximation to differentiation
- Use numerical differentiation to solve o.d.e.'s and p.d.e.'s on a computer
    - Recall that a computer doesn't do differential/integral mathematics and only deals with discrete functional values

Generic Method to Derive a Difference Formula:

$$f_i^{(p)} - E = \frac{a_\alpha f_\alpha + a_\beta f_\beta + \dots a_\lambda f_\lambda}{h^p}$$

where

$f_\alpha, f_\beta, \dots, f_\lambda$ = functional values at nodes

$a_\alpha, a_\beta, \dots, a_\lambda$ = coefficients of the formula being derived

$E = O(h)^N$ when $p + N$ nodes are used (or better for some central approximations)

- Procedure:
  - Substitute Taylor Series expansions for f_α, f_β etc. about node i
  - Rearrange equations such that coefficients multiply equal order derivatives at node i and generate algebraic equations by setting coefficients of $f_i^{(p)}$ equal to 1 and the $p + N - 1$ other coefficients equal to zero
  - Solve for a_α, a_β etc.

Numerical Differentiation Formulae Using Interpolating Polynomials:

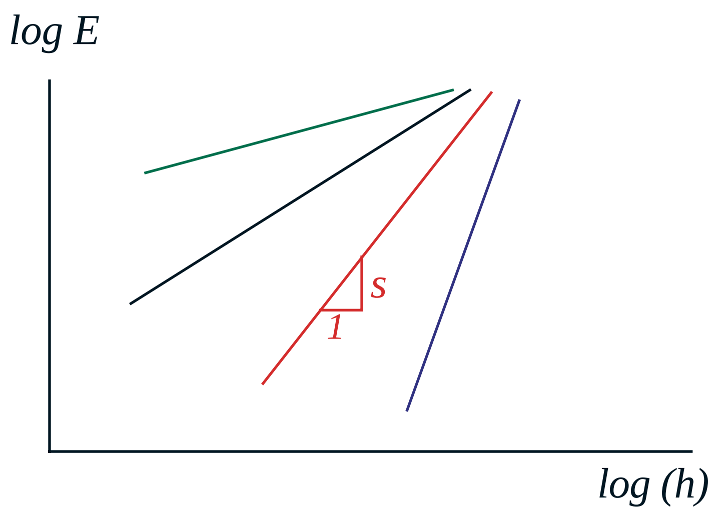
- Use at least $p + 1$ nodes with the interpolation formula to establish an approximation to the p^{th} derivative
- Any interpolating technique/formula can be used
- The numerical differencing formula is simply the differentiated interpolating polynomial evaluated at one of the nodes used for interpolation
- The error can be computed based on the error of the interpolating formula

Errors for Numerical Differentiation

$$f_i^{(p)} = \frac{\sum_{i=-N}^{M} a_i f_i}{h^p} + E$$

$$E = C f^{(r)}(\xi) h^s$$

s = the order of the method



- Error can be derived by
  - Applying Taylor series expansions to the terms in the differentiation formula
  - Differentiating the error in the interpolation function (assuming that this error is expressed as a series and does not depend on ξ)

- Higher order accuracy leads to better answers for larger h (presumably less work for computer for a given level of accuracy)

- However, higher order accuracy is not ***always*** better
  - Recall that for interpolation, piecewise linear was sometimes better than high order interpolation
  - The same holds true for differentiation. It depends on the function
  - Also there are trade-offs in order of accuracy versus actual implementation cost on a computer

INTRODUCTION TO O.D.E.'s AND P.D.E.'S

Partial Differentiation

- Simply apply numerical differentiation formulae relative to the independent variable/ direction in which you are differentiating while holding indices associated with all other independent variables/directions constant.

Solving Single Equation O.D.E. I.V.P.'s

- Solve $\frac{dy}{dt} = f(y, t)$ with a specified i.c. $y(t_o) = y_o$
- Apply the Euler method by approximating the d.e. at $t = t_j$ and applying a forward difference approximation for the first derivative

$$\frac{y_{j+1} - y_j}{\Delta t} = f(y_j, t_j) \quad \Rightarrow$$

$$\boldsymbol{y_{j+1} = y_j + \Delta t \; f(y_j, t_j)}$$

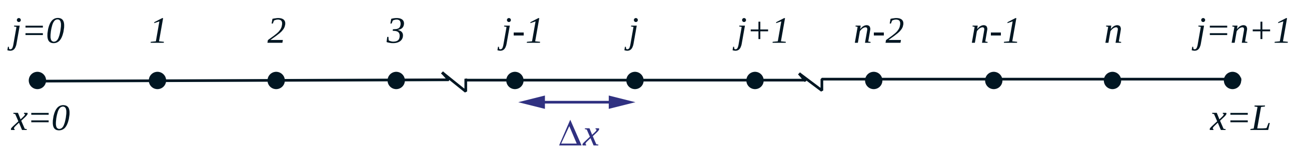
- Simply advance from one time level to the next by substituting known values (y_j, t_j) into the right hand side and solving for y_{j+1} $\rightarrow$ "time marching"

Solving O.D.E. Boundary Value Problems

- A BVP is defined by a second or higher order o.d.e. with boundary conditions specified at two values of the independent variable
    - e.g.

$$\frac{d^2y}{dx^2} + D\frac{dy}{dx} + E = h(x), \quad y(0) = y_o, \quad y(L) = y_l$$

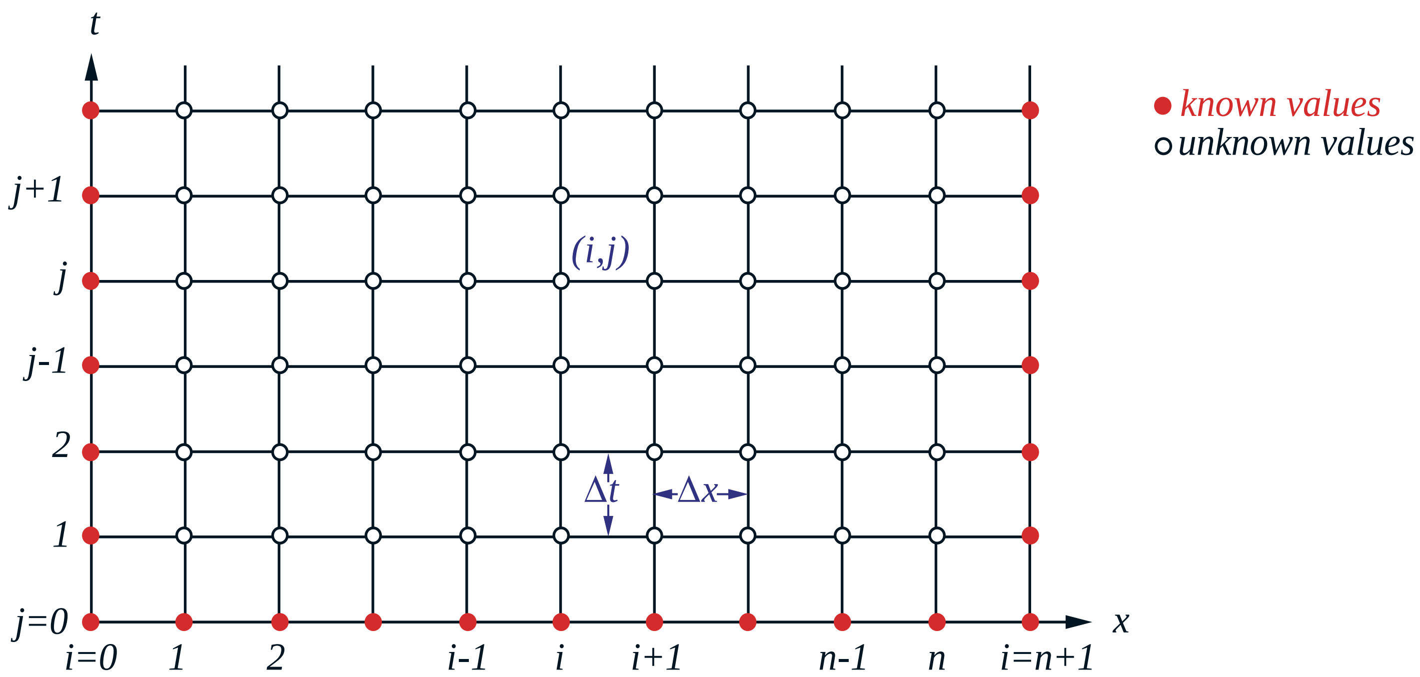
- Define a “grid” of discrete nodes and write discrete forms of the o.d.e. at these nodes by approximating differential terms by finite difference approximations.



- Incorporate discrete forms of the boundary conditions
- Solve the resulting system of simultaneous algebraic equations.

Solutions to P.D.E.'s

- Space-time dependent p.d.e.'s are solved by defining a space-time grid of discrete nodes



- Discrete solutions are advanced in time by writing a sufficient number of algebraic equations to allow the solution of all unknowns at a given time level → "time marching"
  - Solution is found at a specific time level prior to advancing to the next time level
  - Finite difference approximations are used to discretize the p.d.e.'s at either the known or unknown time levels.

- Time discretizations for first time derivatives
  - Explicit: approximate the time derivative at the known time level j using a first order forward approximation in time and evaluate all spatial derivatives at the known time level j
  - Implicit: approximate the time derivative at the unknown time level $j+1$ using a backward approximation in time and evaluate all spatial derivatives at the unknown time level $j+1$
  - Crank-Nicolson: approximate the time derivative at an intermediate time level $j+1/2$ using a second order central approximation in time and evaluate the spatial derivatives at the intermediate time level $j+1/2$ (applying linear interpolation to express the intermediate node values at $j+1/2$ in terms of full node values at j and $j+1$).
- Spatial differentiation is typically implemented using central finite difference approximations
- Explicit methods do not lead to systems of simultaneous equations since there is no coupling in the discrete equations between nodes at the unknown time level.
- Implicit and Crank-Nicolson methods do lead to systems of simultaneous equations since there is coupling in the discrete equations between nodes at the unknown time level.

NUMERICAL INTEGRATION

- Solve

$$I = \int_a^b f(x)dx$$

- Usage
  - Compute cumulative values

$$\mathbf{V} = \int_{t_o}^{t} Q(t)dt$$

  - Integration is the cornerstone of finite element based methods

Basic Development of Integration Formulae

- ***All methods are based on integrating an interpolating function over a sub-interval***
- ***Extended integration rules are obtained by summing integrands over sub-intervals***

- Steps to derive integration formulae
  - Define a sub-interval
  - Define interpolation/integration points over the sub-interval as well as the interpolation function type (and order) (e.g. Lagrange or Hermite)

$$g(x) = \sum_{i=0}^{N} f_i V_i(x) \quad \text{or} \quad g(x) = \sum_{i=0}^{N} f_i \alpha_i(x) + \sum_{i=0}^{N} f_i^{(1)} \beta_i(x)$$

  - Integrate the approximating interpolating function over the sub-interval
  - Now sum up integrals over all sub-intervals to develop the "extended formulae"

Newton Cotes Closed Formulae

- Derived by using equispaced interpolation points and Lagrange interpolation
- Integration points are always at the end points of the sub-interval
  - Trapezoidal Rule and Simpson's $1/3$ Rule

Newton Cotes Open Formulae

- Same as the closed formulae except that the sub-interval now extends beyond interpolation (or integration) points

Gauss-Legendre Integration Formulae

- Derived by using non-equispaced interpolation points and Hermite interpolation
- Specifically the integration points are selected such that the integrals of the $\beta_i(x)$ functions (associated with the derivative terms) are equal to zero.

$$I = \int_{-1}^{+1} g(x)dx$$

$$g(x) = \sum_{i=0}^{N} \alpha_i(x) f_i + \sum_{i=0}^{N} \beta_i(x) f_i^{(1)}$$

- Thus

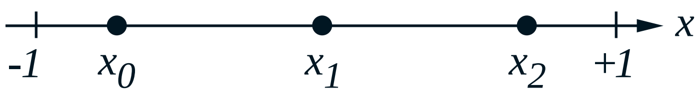
$$I = \sum_{i=0}^{N} f_i \left(\int_{-1}^{+1} \alpha_i(x)dx \right) + \sum_{i=0}^{N} f_i^{(1)} \left(\int_{-1}^{+1} \beta_i(x)dx \right)$$

- Now we select the integration points x_i such that $\int_{-1}^{+1} \beta_i(x)dx = 0$

- ***We can show that***

$$\int_{-1}^{+1} \beta_i(x)dx = 0$$

when the interpolation/integration points are selected to be the roots of an appropriate order Legendre polynomial



- *This results in integration formulae which only involve* f_i ***(not*** $f_i^{(1)}$***)*** *and are twice the order of accuracy as Newton Cotes formulae with same number of interpolation/integration points.*

Methods for Evaluating the Accuracy of Integration Methods For a Sub-interval

Method 1: Develop Taylor Series expansions for f(x) and the functional values at the nodes

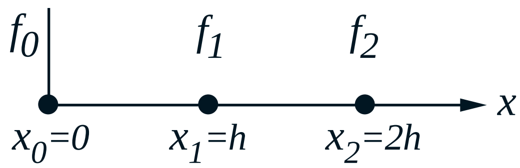
- Equation for Simpson's 1/3 rule

$$E = \int_0^{2h} f(x)dx - \frac{h}{3}(f_o + f_1 + f_2)$$

where

$$\int_0^{2h} f(x)dx = \text{the exact integral}$$

$$\frac{h}{3}(f_o + 4f_1 + f_2) = \text{the integration formula}$$



- Develop Taylor Series expansions for $f(x)$, f_o, f_1 and f_2 about the midpoint of the interval $x_1 = h$
- Substitute in and integrate

Method 2: Use error terms from the interpolating function

- Apply the error in the interpolation function in series form (using the first few terms)
- ***DO NOT*** use $e(x)$ based on evaluations of the last derivative at ξ where ξ is a point in the interval since $\xi = \xi(x)$!

$$e(x) = f(x) - g(x) \quad \Rightarrow$$

$$E = \int f(x)dx - \int g(x)dx \quad \Rightarrow$$

$$\boldsymbol{E = \int e(x)dx}$$

Accuracy of Extended Integration Methods

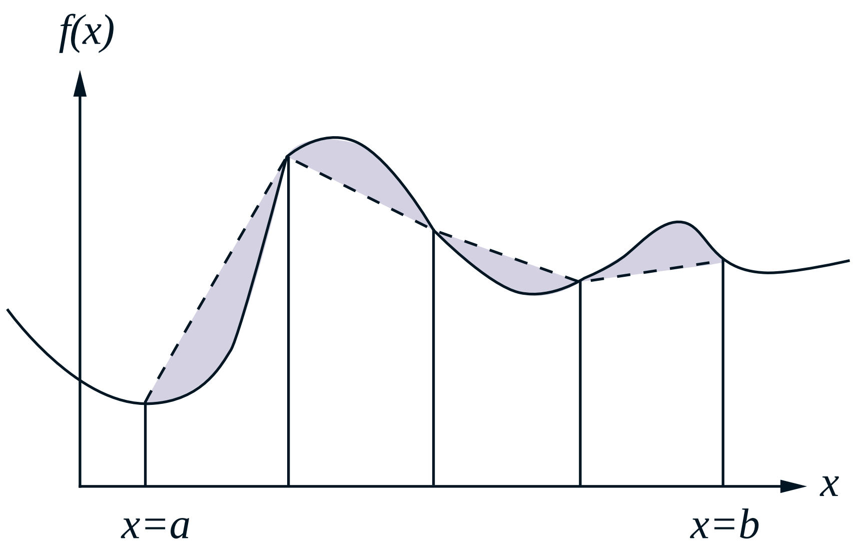
- Sub-interval error is defined as:

$$E_{SI-i} = C\ h^n\ f^{(m)}(\bar{x}_i)$$

where $f^{(m)}(\bar{x}_i)$ = m^{th} derivative at the mid point of i^{th} sub-interval

- Simply sum errors over the sub-intervals to obtain the error for the extended rule

$$E = Ch^n \sum_{i=1}^{N} f^{(m)}(\bar{x}_i)$$



- However $N = \frac{b-a}{h} \Rightarrow h = \frac{b-a}{N}$

$$E = C(b-a)h^{n-1}\left[\frac{1}{N}\sum_{i=1}^{N} f^{(m)}(\bar{x}_i)\right]$$

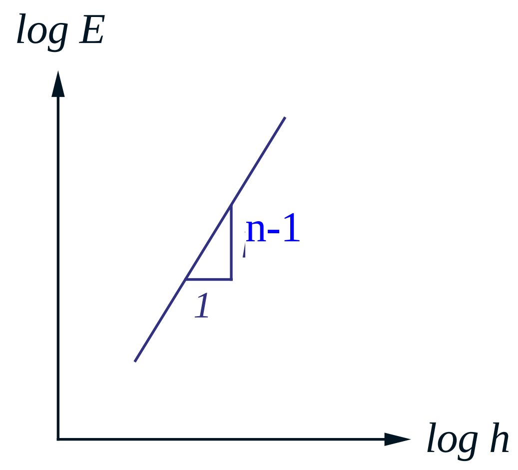
- We note that

$$\frac{1}{N}\sum_{i=1}^{N} f^{(m)}(\bar{x}_i) = \overline{f^{(m)}(\bar{x}_i)}$$

- Substituting

$$\boldsymbol{E = C(b-a)h^{n-1}\,\overline{f^{(m)}(\bar{x}_i)}}$$

  - $b-a$ remains constant no matter what sub-interval spacing h you choose
  - $f^{(m)}(\bar{x}_i)$ remains approximately constant over the interval $[a, b]$ no matter what sub-interval h you choose.


log E
n-1
1
log h

Romberg Integration

- Based on knowing the general form of the error series and two or more approximate numerical integrals, Romberg integration solves for the coefficients of the error series and thus extrapolates a higher order accurate solution.
- Apply to extended integration rules
- Total error is known from Taylor series analysis (summed over the intervals)
    - e.g. for extended trapezoidal rule

$$I = \tilde{I}_h + Ch^2 + Dh^4 + Eh^6 + Fh^8$$

where

I = the exact integral

$\tilde{I}_h$ = the approximate integral using the integration formula

C, D, E = coefficients of the error series which include derivatives

- The form of the series depends on the method

- Evaluate the integral using two or more spacings. This allows us to evaluate I, C, D etc.

- Evaluate $\tilde{I}_h$ with h; $\tilde{I}_{2h}$ with $2h$
  - Solve for I, C
  - Improve accuracy by two orders

- Evaluate $\tilde{I}_h, \tilde{I}_{2h}, \tilde{I}_{4h}$
  - solve for I, C, D etc.
  - Improve accuracy by four orders!