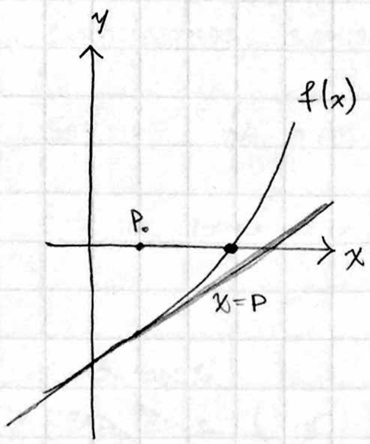
ENGS 91 9/16/2015

ROOT FINDING:



FIND P SUCH THAT $f(P)=0$

NEWTON'S METHOD:

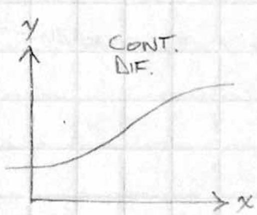
- GUESS P_0
- USE ⊤ @ P_0 TO FIND P_1
- ITERATE TO FIND P_j @ DETERMINED ACCURACY

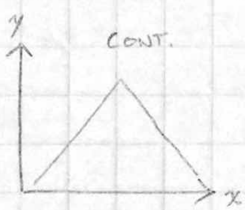
50% THEORY : 50% ANALYSIS

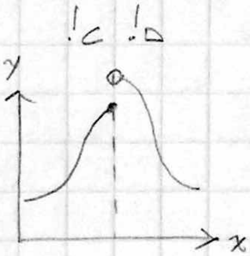
EMAIL SOURCE CODE DIRECTLY

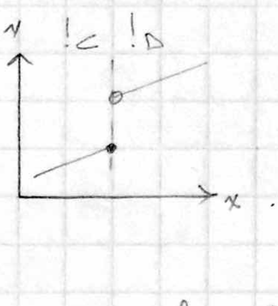
NOTES:

MATHEMATICAL PRELIMS









CONTINUITY: $\lim_{x \to x_0} f(x) = f(x_0)$ EXISTS AND IS SINGLE VALUED (REGARDLESS OF DIR.)

DIFFERENTIABILITY: $\lim_{x \to x_0} \frac{f(x) - f(x_0)}{x - x_0} = \left.\frac{\partial f(x)}{\partial x}\right|_{x_0} = f'(x_0)$ EXISTS : IS SINGLE VALUED

PRESENTATION'S ON FRIDAY'S
- SHOULD BE PROFESSIONAL IN NATURE & ON MATERIAL PRESENTED THAT WEEK

CONSTITUTIONAL SUPERCOOLING

SEA-ICE: INDICATOR & PERPETRATOR OF CLIMATE CHANGE

SEA ICE GOVERNED BY "DYNAMIC DUO"

DYNAMICS THERMODYNAMICS

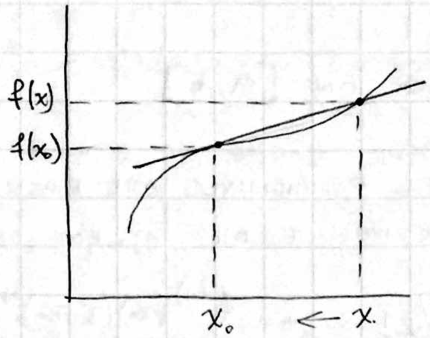
ASSIGNMENTS
* READ BOOK (THE ICE MASTER)
  - REPORT ON LEADERSHIP, THE GOOD & BAD. DUE OCTOBER 7
  - DID THEY USE TRIBAL KNOWLEDGE?

* MAKE A LIST
  - TOP 10 THINGS TO KNOW ABOUT SEA ICE. SUMMARY OF IMPORTANT THINGS
  - E.G. SUMMARY FOR VICE PRESIDENT

- LEADERSHIP STYLE, STRENGTHS, & WEAKNESSES
- HOW DID THEY RESPOND TO ADVERSITY, SUCCESS?
- INDIGEOUS KNOWLEDGE -- DID THEY USE IT?

9/15/2015 READING NOTES

CONTINUITY / DIFFERENTIABILITY



$$f'(x_0) = \lim_{x \to x_0} \frac{f(x) - f(x_0)}{x - x_0}$$

$x(1,:)$

MATLAB (ROWS, COLUMNS)

ABSOLUTE ERROR: $|P - \underbrace{P^*}_{}|$ ← APPROXIMATION

RELATIVE ERROR: $\dfrac{|P - P^*|}{|P|}$, SAME AS ABSOLUTE, BUT NORMALIZED BY $|P|$

* LAB 1 PSEUDOCODE

n	J_n
0	
1	
2	
3	
4	

COMPUTE J_2 FROM $\underbrace{J_2(x)}_{\text{U.K.}} \frac{2(2)}{x} = J_{2+1}(x) + J_{2-1}(x)$

NOTES CONT.

- STABLE: SMALL CHANGES IN INPUT → SMALL CHANGES IN OUTPUT
  - ↳ COND. STABLE = STABLE BUT ONLY FOR CERTAIN CHOICES OF INITIAL DATA

STABILITY OFTEN ATTRIBUTED TO ERROR

BESSELJ → $J_\nu(z)$

- $E_n \approx C n E_0$ (LINEAR)
- $E_n \approx C^n E_0$ (EXPONENTIAL) → OFTEN UNSTABLE

$J_0 = \ldots$ (FOR $x = 1, 8, 25$)
$J_1 = \ldots$ (FOR $x = 1, 8, 25$)
FOR $x = 1, 8, 25$
  FOR $n = 1:25$
    $J_{n+1}(x) = \frac{2n}{x} J_n(x) - J_{n-1}(x)$
    ↳ FORWARD SCHEME

$n = 1$
$J_2(x) = \frac{2n}{x} J_1 - J_0(x)$

$n = 2$
$J_3(x) = \frac{2n}{x} J_2(x) - J_1(x)$

9/18/2015

* TAYLOR'S THEORUM

IF $f(x)$ HAS $n+1$ DERIVATIVES THAT EXIST ON $[a, b]$
THEN $f(x) = P_N(x) + R_N(x)$

- N^{TH} TAYLOR <u>POLYNOMIAL</u>

USEFUL BC POLYNOMIALS ARE EASY TO WORK WITH. IE. DIFF, INT, ECT.

$$P_N(x) = f(x_0) + f'(x_0)(x-x_0) + f''(x_0)(x-x_0)^2/2! \cdots + \frac{f^{(N)}(x_0)(x-x_0)^N}{N!}$$

$$= \sum_{K=0}^{N} \frac{f^{(N)}(x_0)}{K!}(x-x_0)^K \quad \text{FOR} \quad x_0 \in [a,b]$$

IMPORTANT

$$x \in [a,b]$$

-NOTE: $P_N(x_0) = f(x_0)$, $K = 0, 1, 2 \ldots N$

- TAYLOR SERIES:

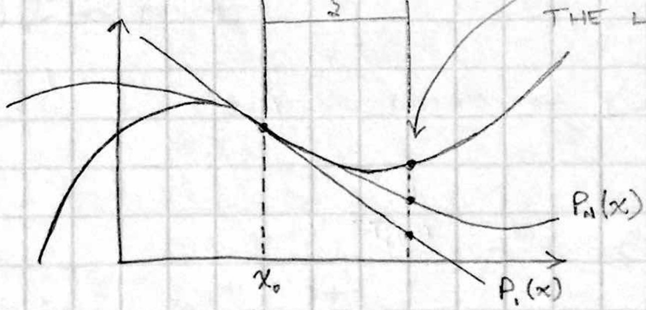
$$n \to \infty \quad \to \quad \underset{\hookrightarrow \infty}{P_N(x)} = f(x)$$

- TRUNCATION ERROR (REMAINDER)

$$R_N(x) = \frac{f^{(n+1)}(\xi)\,(x-x_0)^{n+1}}{(N+1)!} \qquad \xi \in [x, x_0]$$

WE DON'T KNOW THIS

FATHER AWAY FROM x_0 WE GET, THE LARGER R_N GETS (IN GENERAL)



9/18/2015

MATH IN COMPUTERS:

$\frac{1}{3} = 0.333333\ldots$ → MUST CHOP-OFF / ROUND

- RESULTS IN ROUND OFF ERROR

$$\left.\begin{aligned} (A+B)+C &= A+(B+C) \\ (A^*B)^*C &= A^*(B^*C) \\ A^*(B+C) &= (A^*B)+(A^*C) \end{aligned}\right\} = \text{ BECOMES } \neq \text{ W/ COMPUTERS}$$

IN GENERAL, AS # OF OPERATIONS INCREASES, SO DOES ROUND OFF ERROR

* CLASSES OF ERROR GROWTH:

ε_0 = INITIAL ERROR IN SOLUTION

ε_N = ERROR AFTER N STEPS

1.) $\varepsilon_N \propto K N \varepsilon_0$ (LINEAR GROWTH)

$\varepsilon_N \propto K^N \varepsilon_0$ (EXPONENTIAL ERROR GROWTH)

* RECURSIVE RELATIONS

A.) $P_N = \frac{1}{3} P_{N-1}$ B. $P_N = \frac{10}{3} P_{N-1} - P_{N-2}$

- ASSUME CONSTANTS ARE EXACT

$P_N = \frac{1}{3} P_{N-1}$ EXACT

$\hat{P}_N = \frac{1}{3} \hat{P}_{N-1}$ COMPUTER

→ $P_N - \hat{P}_N = \frac{1}{3}\left(P_{N-1} - \hat{P}_{N-1}\right)$

$\varepsilon_N = \frac{1}{3} \varepsilon_{N-1}$

$\varepsilon_1 = \frac{1}{3} \varepsilon_0$

$\varepsilon_2 = \frac{1}{3} \varepsilon_1 = \frac{1}{3}\left(\frac{1}{3} \varepsilon_0\right) = \left(\frac{1}{3}\right)^2 \varepsilon_0$

$\varepsilon_3 = \frac{1}{3} \varepsilon_0 = \left(\frac{1}{3}\right)^3 \varepsilon_0$ → $\varepsilon_N = \frac{1}{3} \varepsilon_{N-1} = \left(\frac{1}{3}\right)^N \varepsilon_0$ → $\lim_{n\to\infty} \varepsilon_N = \lim_{n\to\infty} \left(\frac{1}{3}\right)^N \varepsilon_0 = 0$

↳ CANNOT ASSUME CONSTANTS EXACT

9/18/2015

LET $\hat{1/3} = \frac{1}{3} - \delta$

$$P_N = \frac{1}{3} P_{N-1} \rightarrow \hat{P}_N = (1/3 - \delta)\hat{P}_{N-1} \rightarrow \varepsilon_N = 1/3\,\varepsilon_{N-1} + \delta \hat{P}_{N-1}$$

$$\varepsilon_1 = 1/3\,\varepsilon_0 + \delta \hat{P}_0$$

$$\varepsilon_2 = \left(\frac{1}{3}\right)^2 \varepsilon_0 + 1/3\,\delta\hat{P}_0 + \delta\hat{P}_1$$

$$\varepsilon_3 = (1/3)^3 \varepsilon_0 + (1/3)^2 \delta\hat{P}_0 + 1/3\,\delta\hat{P}_1 + \delta\hat{P}_2$$

$$\varepsilon_N = \left(\frac{1}{3}\right)^N \varepsilon_0 + \underbrace{\left(\frac{1}{3}\right)^{n-1}\delta\hat{P}_0 + \ldots \delta\hat{P}_{N-1}}_{\text{N TERMS, EACH SMALLER THAN } \delta\hat{P}_0}$$

$\varepsilon_N \leq (1/3)^n \varepsilon_0 + n\delta\hat{P}_0 \longrightarrow$ ERRORS ONCE INTRODUCED GROW LINEARLY

B.

$$P_N = \frac{10}{3} P_{N-1} - P_{N-2} \quad \text{(EXACT)}$$
$$\hat{P}_N = \frac{10}{3} \hat{P}_{N-1} - \hat{P}_{N-2} \quad \text{(COMPUTE)}$$

$P_N - \hat{P}_N = \frac{10}{3}(P_{N-1} - \hat{P}_{N-1}) - (P_{N-2} - \hat{P}_{N-2})$ DIFFERENCE EQ. W/ CONSTANT COEFFICIENTS

ASSUME $\varepsilon_N \propto \lambda^n$

$\lambda^n = \frac{10}{3}\lambda^{n-1} - \lambda^{n-2} \rightarrow \lambda^2 - \frac{10}{3}\lambda + 1 = 0$ CHARACTERISTIC OR AUXILARY EQ.

$$\lambda = \frac{-B \pm (B^2 - 4AC)^{1/2}}{2A}$$

$$\varepsilon_N = \underbrace{C_1 (1/3)^n}_{\to 0} + \underbrace{C_2 (3)^n}_{\to \infty} \rightarrow \lim_{n\to\infty} = \infty$$

ONE ROOT IN CHARAC. EQ > 1

ONLY THE LARGEST ROOT MATTERS

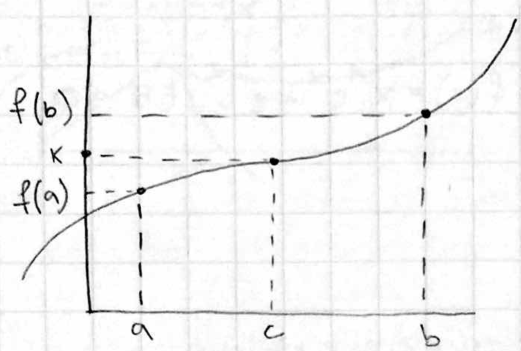
- FOR COMPLEX ROOTS, IT IS THE MAGNITUDE THAT MATTERS

9/21/2015

A FEW IMPORTANT THEORUMS

1. INTERMEDIATE VALUE THEORUM

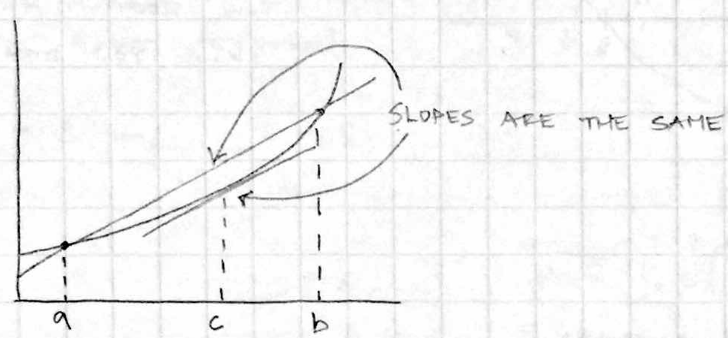
IF $f \in C[a,b]$ AND K IS BETWEEN $f(a)$ AND $f(b)$, THEN THERE EXISTS A NUMBER $c \in (a,b)$ FOR WHICH $f(c) = K$



2. MEAN VALUE THEORUM (MVT)

IF $f \in C[a,b]$ AND DIFFERENTIABLE ON (a,b), THEN A NUMBER $c \in (a,b)$ EXISTS w/

$$f'(c) = \frac{f(b) - f(a)}{b - a}$$



9/21/2015

$$\varepsilon_n \propto \lambda^n \quad \rightarrow \quad \varepsilon_N = \sum_{J=0}^{N} c_j \lambda_j^n, \quad \text{IF } |\lambda_j| > 1 \rightarrow \underline{\underline{\text{EXPONENTIAL}}}$$

* SOLUTIONS TO NON-LINEAR EQUATIONS

"ROOT" FINDING PROBLEMS. FINDING P SUCH THAT $f(p) = 0$

EG.

$$x^4 e^x - e^{-2x} = 8 \quad \rightarrow \quad f(x) = x^4 e^x - e^{-2x} - 8 = 0$$

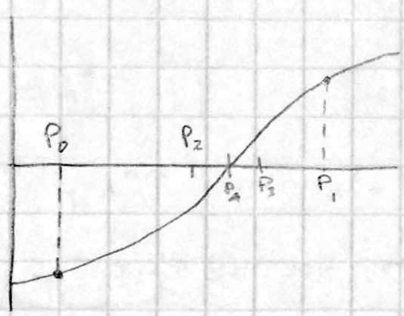
→ ITERATIVE METHODS ←

0.) "SCAN" x UNTIL $f(x) \simeq 0$

- WHAT STEP SIZE? → TERRIBLE METHOD

1.) BISECTION METHOD

IF $f(a)f(b) < 0$, THEN IVT GUARENTEE'S A ROOT $p \in [a, b]$



PICK TWO POINTS, EVALUATE, THEN CUT IN HALF, BRACKET THE ROOT & THROW AWAY HALF YOU DON'T WANT

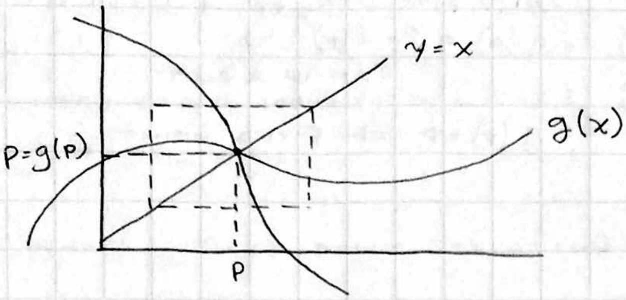
BISECTION METHOD:

+ SIMPLE
+ ALWAYS FINDS A ROOT
- NEED $[a, b]$ SUCH THAT $f(a)f(b) < 0$
- SLOW, $\varepsilon_N \leq \frac{b-a}{2^n}$

9/21/2015

* FIXED POINT SCHEME

- WILL BE USEFUL FOR ANALYZING CONVERGENCE RATES OF VARIOUS ROOT FINDING TECHNIQUES



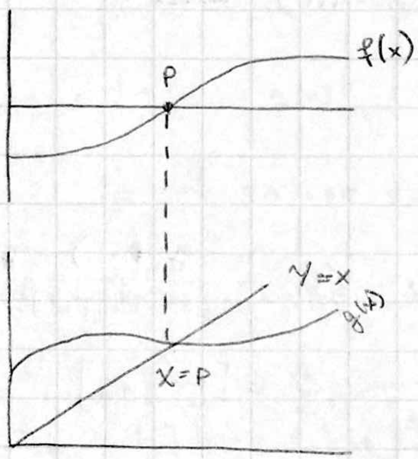
PROVABLE COND. FOR FIXED POINT

P IS A FIXED POINT WHEN $g(p) = p$

i.) $g(x)$ IS CONTINUOS ON $[a,b]$

ii.) $g(x)$ IS BOUNDED ON $[a,b]$ BY $[a,b]$

iii.) $|g'(x)| < 1 \quad x \in [a,b]$



$g(x) = x - f(x)$ ← WHEN $f(x) = 0$, $\underbrace{g(x) = x}_{\text{BY DEF A FIXED POINT}}$

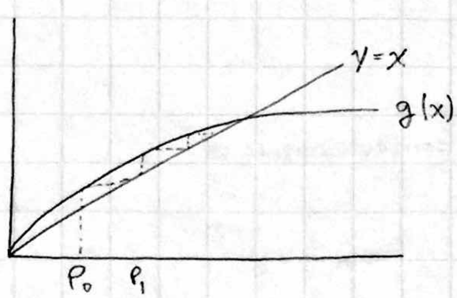
$g(p) = p - f(p)$

$g(p) = p$

HOW TO FIND FIXED POINT GIVEN $g(x)$?

→ FIXED POINT ITERATION

$$p_N = g(p_{N-1})$$

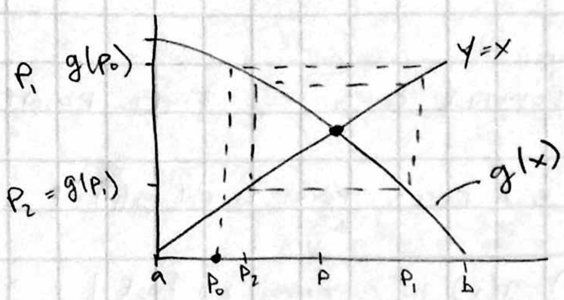


NEED $P_0 \in [a, b]$

9/22/2015

* FIXED POINT ITERATION

$$P_N = g(P_{N-1})$$



$g(x) = x - f(x)$

$g(p) = p - \underbrace{f(p)}_{=0 \text{ IF A ROOT}}$

$g(p) = p \Rightarrow$ FIXED POINT

ESTIMATE ERROR:

- ASSUME FIXED POINT CONTINVOS i.) ii.) iii.) HOLD

$P_n = g(P_{n-1})$ (COMPUTE)

$P = g(P)$ (EXACT)

$|P_n - P| = |g(P_{n-1}) - g(P)| \rightarrow$ MVT: $g'(c) = \dfrac{g(P_{n-1}) - g(P)}{P_{n-1} - P}$

$c \in [P, P_{n-1}]$

$\varepsilon_N = |g'(c)||P_{N-1} - P|$

$\varepsilon_N = \underbrace{|g'(c)|}_{<1} \varepsilon_{N-1}$ (MANDATED, RULE 3)

$\varepsilon_N = K \varepsilon_{N-1} = K^n \varepsilon_0 \quad K < 1$

$\lim_{n \to \infty} \varepsilon_N = \underbrace{K^N}_{=0} \varepsilon_0 = 0$

GUARANTEED TO CONVERGE

A FEW OBSERVATIONS:

A.) $|g'(x)|$ SMALL $\rightarrow$ QUICK CONVERGANCE

B.) $|g'(x)| > 1 \rightarrow$ DIVERGE

C.) IF $|g'(x)| < 1$ FOR $x \in [a, b]$, CONVERGE FOR ANY $P_0 \in [a, b]$
IF $|g(P)| < 1$, CONVERGE FOR SOME $P_0 \in [a, b]$

9/22/2015

CONVERGENCE RATES:

- DEFINITION: $\lim_{n \to \infty} \frac{|\varepsilon_{N+1}|}{|\varepsilon_N|^{\alpha}} = \lambda$ 'ASYMPTOTIC STATEMENT'

α IS THE ORDER OF CONVERGENCE

$\varepsilon_{N+1} = \lambda \varepsilon_N^{\alpha} \rightarrow$ $\alpha = 1$ (LINEAR), $\alpha = 2$ (QUADRADIC), $\alpha = 3$ (CUBIC) } DOES NOT HAVE TO BE AN INTEGER

ASYMPTOTIC ERROR CONSTANT, WANT TO BE SMALL

GENERAL FIXED POINT ITERATION

$P_{n+1} = g(P_N)$ COMPUTE

$P = g(P)$ EXACT

$\varepsilon_{N+1} = |g(P_N) - g(P)|$

↳ TAYLOR EXPAND $g(P_N)$ ABOUT P

$$g(P_N) = \left| g(P) + g'(P)(P-P_N) + \frac{g''(P)(P-P_N)^2}{2!} + \frac{g^{(3)}(P)(P-P_N)^3}{3!} \cdots \right|$$

$= \varepsilon_N$

$$\varepsilon_{N+1} = \left| g'(P)(P-P_N) + \frac{g''(P)(P-P_N)^2}{2!} + \frac{g^{(3)}(P)(P-P_N)^3}{3!} + \cdots \right|$$

$$\varepsilon_{N+1} = \left| g'(P)\varepsilon_N + \frac{g''(P)\varepsilon_N^2}{2!} + \frac{g'''(P)\varepsilon_N^3}{3!} + \cdots \right|$$

$$\lim_{n \to \infty} \varepsilon_{N+1} = |g'(P)|\varepsilon_N \rightarrow \alpha = 1 \quad \lambda = |g'(P)|$$

LINEAR CONVERGENCE

$(2)(.2)^2$

9/22/2015

NEWTON'S METHOD

$$\text{IF} \quad g(x) = x - \underbrace{\phi(x)}_{\text{SCALING FUNCTION}} \overbrace{f(x)}^{\text{FUNCTION FOR WHICH WE WANT ROOT}}$$

$$\text{THEN} \quad g(p) = p - \phi(p) f(p) \rightarrow g'(x) = 1 - \phi'(x) f(x) - \phi(x) f'(x)$$

$$g(p) = p \qquad g'(p) = 1 - \phi'(p) \underbrace{f(p)}_{=0,\text{ IT'S A ROOT!}} - \phi(p) f'(p)$$

$$g'(p) = 1 - \phi(p) f'(p) = 0$$

↳ FORCE QUADR. CONVERGE

$$\phi(p) = \frac{1}{f'(p)}$$

$$\phi(x) = \frac{1}{f'(x)}$$

$g(x) = x - f(x)/f'(x)$ FIXED POINT FUNCTION FOR NEWTON'S METHOD

$$\boxed{p_{N+1} = p_N - \frac{f(p_N)}{f'(p_N)}}$$

NEWTON'S METHOD

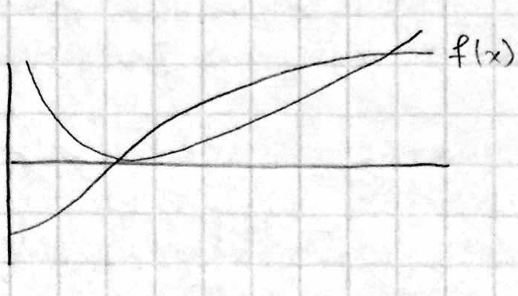
GUARENTEED QUADRADIC CONVERGENCE FOR $p_0 \in [a, b]$ PROVIDED $f'(p) \neq 0$

9/23/2015

NEWTON'S METHOD

$$p_{N+1} = p_N - \frac{f(p_N)}{f'(p_N)} \quad \text{FOR} \quad f'(p) \neq 0$$

WHAT IF $f'(p) = 0$



9/23/2015

"DERIVED" NEWTON'S METHOD

$$g(x) = x - \frac{f(x)}{f'(x)}$$

$$g'(x) = \frac{d}{dx}\left[x - f(x)/f'(x) \right] = 1 - \frac{f'(x)}{f'(x)} + \frac{f(x)f''(x)}{(f'(x))^2}$$

$$g'(x) = \frac{f(x)f''(x)}{[f'(x)]^2} \Rightarrow g(p) = \frac{f(p)f''(p)}{[f'(p)]^2} = \frac{0}{0}$$

$\rightarrow$ L'HOPITALS RULE $\leftarrow$

$$\frac{f'(p)f''(p) + f(p)f'''(p)}{2f'(p)f''(p)} = \frac{0}{0} \rightarrow \text{DO AGAIN}$$

$$\frac{[f''(p)]^2 + f'(p)f'''\cancel{(p)} + f'\cancel{(p)}f'''(p) + f\cancel{(p)}f''(p)}{2[f''(p)]^2 + 2f'\cancel{(p)}f'''(p)} = \frac{1}{2} \neq 0$$

- STILL CONVERGES, BUT $\alpha = 1$ (LINEARLY) WHEN $f'(p) = 0$
  ↳ SLOPE OF f @ p

GENERALIZE THE ROOT OF MULTIPLICITY m

$$f(x) = (x-p)^m q(x) \quad , \quad \lim_{x \to p} q(x) = 0$$

$f(x) = (x-p)^m q(x)$ WHERE WHICH ENSURES $q(x)$ CONTAINS NO ROOTS

9/23/2015

* MODIFIED NEWTON'S METHOD (FOR $m \geq 2$)

- FOR $m = 2$; $f(p) = 0$; $f'(p) = 0$; BUT $f''(p) \neq 0$

DEFINE $u(x) = \dfrac{f(x)}{f'(x)}$

i.) $u(p) = \dfrac{f(p)}{f'(p)} = 0 \rightarrow$ L'HOPITALS $\rightarrow \dfrac{f'(p)}{f''(p)} = 0$

- SO $u(x)$ HAS SAME ROOT AS $f(x)$

ii.) $u'(p) = 1 - \left.\dfrac{ff''}{[f']^2}\right|_{x=p} = \dfrac{1}{2}$ (SIMPLE ROOT) WHICH MEANS QUADRATIC CONVERGENCE,

iii.) APPLY NEWTON'S METHOD TO $u(x)$

$$\boxed{p_{n+1} = p_n - \frac{u(p_n)}{u'(p_n)}}$$ ← 'MODIFIED' NEWTON'S METHOD

$g(x) = x - \dfrac{u(x)}{u'(x)}$, $g'(p) = 0 \rightarrow \alpha = 2$

$\lambda = u''(p) = \dfrac{-f'''(p)}{6 f''(p)} \neq 0$, TEXT SHOWS THIS METHOD

FOR $u'(p) = \frac{1}{m}$, FOR LARGE m $u'(p) \rightarrow 0$

* SECANT METHOD

- MAKE APPROX. TO DERIVATIVE OF $f'(x)$ IN NEWTON'S METHOD

- $f'(x)$ CAN BE DIFFICULT TO COMPUTE

- "DISCRETE VERSION OF NEWTON'S METHOD

9/23/2015

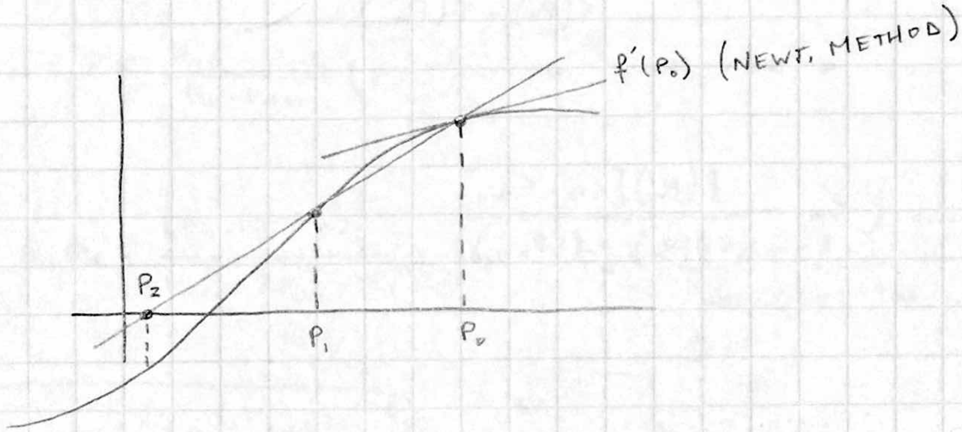
$$f'(p_n) \equiv \lim_{x \to p_n} \frac{f(x) - f(p_N)}{x - p_n} \rightarrow \text{USE } x \approx P_{N-1}$$

$$\rightarrow \quad f'(p_n) \simeq \frac{f(p_{n-1}) - f(p_n)}{p_{n-1} - p_n}$$

$$P_{N+1} = P_n - \frac{f(p_n)(p_n - p_{n-1})}{f(p_n) - f(p_{n-1})}$$

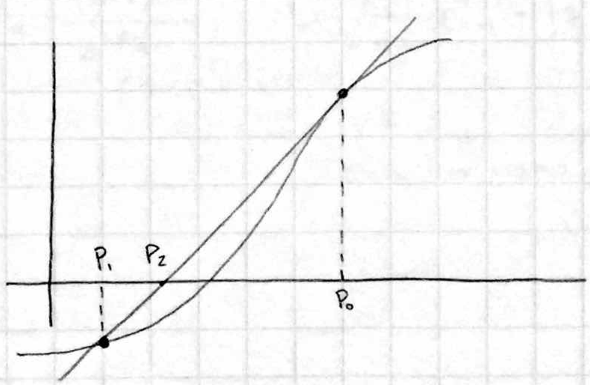
$\rightarrow$ NEED TWO STARTING VALUES. DO NOT NEED TO BRACKET THE ROOT

- SLOWER CONVERGENCE
- BUT DO NOT HAVE TO EVALUATE $f'(p_n)$



* METHOD OF FALSE POSITION

- SECANT METHOD BUT ALWAYS BRACKET THE ROOT



9/23/2015

$$P_{N+1} = P_n - \frac{f(P_n)(P_n - P_{n+1})}{f(P_n) - f(P_{n+1})} = g(P_n, P_{n-1})$$

WANT $\quad \varepsilon_{n+1} = f(\varepsilon_n)$

$$\underbrace{|P_{n+1} - P|}_{\varepsilon_{n+1}} = |g(P_n, P_{n+1}) - g(P)| = P_n - \frac{f(P_n)(P_n - P_{n-1})}{f(P_n) - f(P_{n-1})} - P$$

$$\varepsilon_{n+1} = P_n - P - \frac{f(P_n)(P_n - P - P_{n-1} + P)}{f(P_n) - f(P_{n-1})}$$

$$\varepsilon_{n+1} = \varepsilon_n - \frac{f(P_n)[\varepsilon_n - \varepsilon_{n-1}]}{f(P_n) - f(P_{n-1})}$$

$\vdots$

$$\lim_{n \to \infty} \varepsilon_{n+1} = \frac{1}{2} \frac{f''(P)}{f'(P)} \varepsilon_n \varepsilon_{n-1} \quad \cdots \quad \varepsilon_{n+1} = \lambda \varepsilon_n^{\frac{\sqrt{5}+1}{2}}$$

$$\alpha = \frac{\sqrt{5}+1}{2} \rightarrow \text{"SUPER LINEAR"}$$

9/25/2015

* ACCELARATING LINEARLY CONVERGING PROCESSES

MOTIVATION: CONSTRUCT A NEW SET OF ITERATES $\{\tilde{P_N}\}$ THAT CONVERGE MORE RAPIDLY TO P THEN DOES $\{P_N\}$

$$\lim_{n\to\infty} \varepsilon_{N+1} = g'(p)\varepsilon_N \,, \quad \text{LINEAR} \quad \alpha = 1$$
$$\lambda = g'(p)$$

$$\lim_{n\to\infty} P_{N+1} - P = g'(p)(P_N - P)$$

$$\lim_{n\to\infty} P_{N+2} - P = g'(P)(P_{N+1} - P)$$

$$P_{N+1} - P_{N+2} = g'(p)(P_N - P_{N+1}) \rightarrow g'(p) \cong \frac{(P_{N+1} - P_{N+2})}{(P_N - P_{N+1})}$$ FOR LARGE n (ESTIMATE)

$$P_{N+1} - P \cong \frac{P_{N+1} - P_{N+2}}{P_N - P_{N+1}}(P_N - P) \rightarrow \text{SOLVE FOR } P$$

$$P \cong P_N - \frac{(P_{N+1} - P_N)^2}{P_{N+2} - 2P_{N+1} + P_N} \rightarrow P_N = \frac{-(P_{N+1} - P_N)^2}{P_{N+2} - 2P_{N+1} + P_N}$$

$$\boxed{\tilde{P} = P_N - \frac{(P_{N+1} - P_N)^2}{P_{N+2} - 2P_{N+1} + P_N}}$$. AITKENS Δ^2 METHOD

$$\Delta P_N \equiv P_{N+1} - P_N$$
$$\Delta^2(P_N) = \Delta(\Delta P_N) = P_{N+2} - 2P_{N+1} + P_N$$

$$\tilde{P}_n = P_N - \frac{(\Delta P_N)^2}{\Delta^2 P_N} \,, \quad \lim_{n\to\infty} \tilde{\varepsilon}_{N+1} = |g'(p)|^2 \varepsilon_N$$

P_0
$P_1 = g(P_0) = \tilde{P}_0$
$P_2 = g(P_1) = \tilde{P}_1$
$P_3 = g(P_2) = \tilde{P}_2$
$P_4 = g(P_3)$

LOOK AT VIDEO

9/25/2015

* CHAPTER 10 - MULTIVARIATE NON-LINEAR EQUATIONS

$f_1(x_1, x_2, x_3 \dots x_N) = 0$

$f_2(x_1, x_2, x_3, \dots x_N) = 0 \quad \rightarrow \quad \underline{F}(\underline{x}) = 0$ — BAR INDICATES VECTOR

$\vdots \qquad \vdots \qquad \vdots \qquad\qquad \underline{x} = (x_1, x_2, x_3 \dots x_N)$

$f_N(x_1, x_2, x_3 \dots x_N) = 0$

f IS A FIXED POINT VECTOR $(P_1, P_2, \dots P_N)$

IF $\underline{G}(\underline{P}) = \underline{P}$

$g_i(P_1, P_2, \dots P_N) = P_i$ FOR $i = 1{:}N$

THE EXISTENCE OF f IS GUARENTEED BY

i. $\underline{G}(\underline{x})$ IS CONT. ON $a_i \le x_i \le b_i$

ii. $\underline{G}(\underline{x})$ IS BOUNDED BY $a_i \le g(x_i) \le b_i$

iii $\sum_{j=1}^{N} \left| \frac{\partial g_i}{\partial x_i} \right| < 1$

- LINEAR $\frac{\partial g_i}{\partial x_i}(f) \neq 0$
- QUADRATIC $\frac{\partial g_i}{\partial x_i} = 0$
- $\frac{\partial^2 g_i}{\partial x_j \partial x_i}(f) = 0$

* NEWTON'S METHOD FOR SYSTEMS OF NONLINEAR EQUATIONS

$\underline{g}(\underline{x}) = \underline{x} - \underline{\underline{A}}^{-1}(x) \cdot \underline{f}(\underline{x})$

COLUMN OF FUNCTION

INVERSE OF MATRIX

9/25/2015

$$g_i(\underline{x}) = x_i - \sum_{j=1}^{N} b_{ij}(\underline{x})\, f_i(\underline{x})$$

$$\underline{\underline{A}} = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & & b_{2n} \\ \vdots & & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nn} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix}$$

$$\underline{\underline{A}}^{-1}$$

FOR NEWTON'S METHOD $\dfrac{\partial g_i(f)}{\partial x_i} = 0$ FOR EACH i, K

- IF $i = K$

$$\frac{\partial g_i}{\partial x_K} = 1 - \left[\sum_{j=1}^{N} b_{ij} \frac{\partial f_j}{\partial x_K} + \underbrace{\sum_{j=1}^{N} \frac{\partial b_{ij}}{\partial x_K} f_j}_{=0} \right]\Bigg|_{x=f} = 0$$

- IF $i \neq K$

$$\frac{\partial g_i}{\partial x_K} = 0 - \left[\text{SAME } \uparrow \right] = 0$$

$$\sum_{j=1}^{N} b_{ij} \frac{\partial f_i}{\partial x_K} = \begin{Bmatrix} 1 & i = K \\ 0 & i \neq K \end{Bmatrix}$$

$$\sum_{j=1}^{n} b_{ij} f'_{jK} = \begin{Bmatrix} 1 & i = K \\ 0 & i \neq K \end{Bmatrix}$$

$$\underline{\underline{A}}^{-1} \cdot \underline{\underline{A}} = \underline{\underline{I}} \quad \rightarrow \quad \underline{\underline{A}} = \begin{array}{c} \begin{array}{ccc} X_1 & \cdots & X_N \end{array} \\ \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \cdots & \frac{\partial f_1}{\partial x_N} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial x_N} \\ \cdots & & \frac{\partial f_N}{\partial x_N} \end{bmatrix} \end{array}$$

$$\boxed{x_{N+1} = x_N - \underline{\underline{I}}^{-1}(\underline{x_N})\, \underline{f}(x_N)}$$

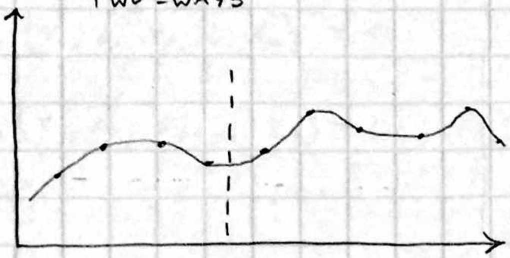
NEWTON'S METHOD FOR N-L SYS. OF EQUATIONS

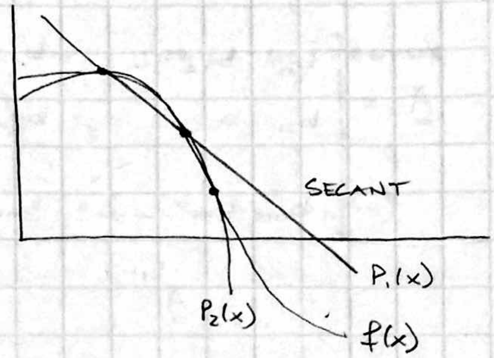
L2 NORM FOR STOPPING CRITERIA

9/28/2015

* POLYNOMIAL INTERPOLATION

TWO-WAYS





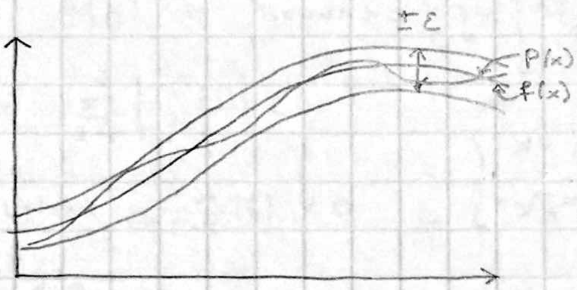
- APPEAL OF POLYNOMIALS:
  - EASY TO EVALUATE
    INTEGRATE
    DIFFERENTIATE
    STORE

W/ MORE POINTS → HIGHER ORDER FIT POSSIBLE

- APPROXIMATIONS BASED ON WEIERSTRAUSS APPROXIMATION THEORUM

IF $f(x)$ IS $C \in [a, b]$ AND $\varepsilon > 0$, THEN A POLYNOMIAL $P(x)$ EXISTS ON $[a, b]$ SUCH THAT $|f(x) - P(x)| < \varepsilon$

→ DOESN'T SAY ANYTHING ABOUT HOW TO GET $P(x)$



ONE TYPE OF POLYNOMIAL IS A TAYLOR POLYNOMIAL

i. GREAT B/C TRUNCATION ERROR IS EXPLICIT

$$\frac{f(\xi)(x - x_0)^{N+1}}{(N+1)!} \quad \xi \in [x, x_0]$$

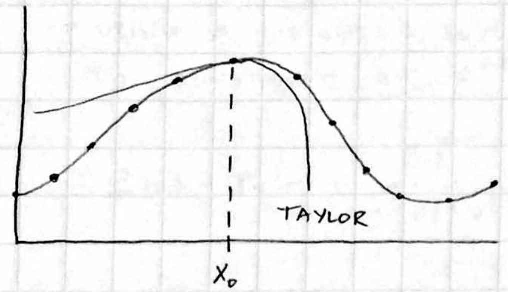
ii. ALL INFORMATION ABOUT $f(x)$ IS AT x_0

$$f(x_0)^{(k)}, \quad k = 0, 1, .. N$$

iii. "LOCALLY" GOOD BUT GETS WORSE AS $|x - x_0|$ GROWS

9/28/2015

- A BETTER POLYNOMIAL FOR INTERPOLATION $P_N(x)$ MATCHES $f(x)$ AT A FINITE # OF VALUES



- N^{TH} ORDER HAS N+1 COEFFICIENTS

$$P_N(x) = a_N x^N + a_{N-1} x^{N-1} + \dots a_1 x + a_0$$

IF $f(x_i) = P_N(x_i)$ THEN WE NEED N+1 x_i'S TO FIND A UNIQUE SET OF COEFF'S $\{a_i\}$

EG.

x_1, x_2, x_3 ; $f(x_1), f(x_2), f(x_3)$

$P_2(x) = ax^2 + bx + c$ → WANT $P_2(x_i) = f(x_i)$

$$\begin{aligned} ax_1^2 + bx_1 + c &= f(x_1) \\ ax_2^2 + bx_2 + c &= f(x_2) \\ ax_3^2 + bx_3 + c &= f(x_3) \end{aligned} \rightarrow \begin{bmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ x_3^2 & x_3 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} f(x_1) \\ f(x_2) \\ f(x_3) \end{bmatrix}$$

$A \quad x = B$

$\underline{\underline{X}}\, \underline{A} = \underline{F} \rightarrow \underline{A} = \underline{\underline{X}}^{-1} \underline{F}$ $\qquad x = A^{-1} B$

* KEY TO POLYNOMIAL INTERPOLATION
- GIVEN N+1 DISTINCT SAMPLES OF $f(x)$, CAN ALWAYS FIND A UNIQUE POLYNOMIAL $P_N(x)$ OF ORDER N SUCH THAT $P_N(x_i) = f(x_i)$, $i = 0, 1 .. N$

- COMLICATED HOWEVER, INSTEAD USE A 'BASIS' POLYNOMIAL

$$P_N(x) = \sum_{j=0}^{N} \alpha_j b_j(x)$$ WHERE $b_j(x) \equiv$ BASIS POLYNOMIAL (KNOWN FUNCTION OF x)

NOTE: IF $b_j(x) = x^j$ THEN $\{\alpha_j\} = \{a_j\}$, $P_N(x) = a_0 x^N + a_{N-1} x^{N-1} + \dots a_1 x + a_0$

9/28/2015

BY CLEVERLY CHOOSING $\{b_j(x)\}$ WE CAN EASILY IDENTIFY $\{\alpha_j\}$

IF $b_j(x_i) = \delta_{ij} = \begin{cases} 1 & i=j \\ 0 & i \neq j \end{cases}$ THEN $P_N(x_i) = \sum_{j=0}^{N} \alpha_i \delta_{ij} = \alpha_i = f(x_i)$

$$\boxed{P_N(x) = \sum_{j=0}^{N} f(x_j)\, b_j(x)}$$

LAGRANGE POLYNOMIAL

$$L_{N,j}(x) = \prod_{\substack{k=0 \\ k \neq j}}^{N} \frac{x - x_k}{x_j - x_k} = \underbrace{\frac{x-x_0}{x_j - x_0} \cdot \frac{x - x_1}{x_j - x_1} \cdot \frac{x - x_2}{x_j - x_2} \cdots \frac{x - x_N}{x_j - x_N}}_{N \text{ TERMS}}$$

EX.

$$L_{N,j}(x_i) = \frac{x_i - x_0}{x_j - x_0} \cdot \frac{x_i - x_1}{x_j - x_1} \cdots \frac{x_i - x_N}{x_j - x_N} \quad \begin{array}{l} \text{IF } i = j = 1 \\ \text{IF } i \neq j = 0 \end{array}$$

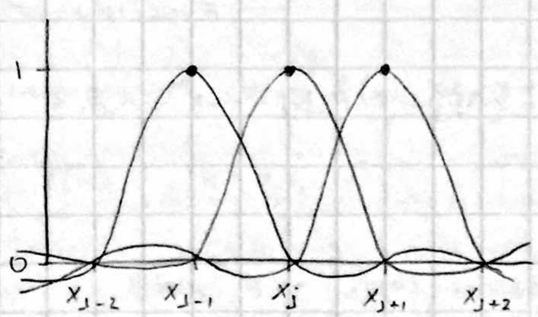
EX.

$$P_3(x) = \sum_{j=0}^{3} f(x_j)\, L_{3,j}(x)$$

$$= f(x_0) L_{3,0}(x) + f(x_1) L_{3,1}(x) + f(x_2) L_{3,2}(x) + f(x_3) L_{3,3}(x)$$

AND

$$L_{3,0}(x) = \frac{x - x_1}{x_0 - x_1} \cdot \frac{x - x_2}{x_0 - x_2} \cdot \frac{x - x_3}{x_0 - x_3} \rightarrow \begin{array}{l} L_{3,0}(x_0) = 1 \\ L_{3,0}(x_1) = 0 \\ L_{3,0}(x_2) = 0 \\ L_{3,0}(x_3) = 0 \end{array}$$



9/28/2015

EASY TO WRITE DOWN, BUT SOMEWHAT INEFFICIENT TO USE

$$P_N(x) = \sum_{j=0}^{N} f(x_j)\, L_{N,j}(x)$$

- USING A WEIGHTED SUM OF N+1 N^{TH} ORDER POLYNOMIALS TO CONSTRUCT AN N^{TH} ORDER POLYNOMIAL $P_N(x)$

$$f(x) - P_N(x) = \frac{f^{(N+1)}(\xi)}{(N+1)!} \prod_{i=0}^{N} (x - x_i)$$

9/29/2015

CHEBYSHEV POLYNOMIALS, OPTIMAL POINTS (SECT. 8.3)

- TAYLOR: ALL INFORMATION @ x_0
- LAGRANGE: PASS THROUGH $x_0, \ldots x_N$ (SPREAD OUT ERROR)

DOES THERE EXIST A SET OF POINTS $\{x_i\}$ FOR WHICH THE 'OPTIMAL' INTERPOLATION IS ACHIEVED?

"OPTIMAL" DEFINED AS MINIMIZING THE MAXIMUM ERROR ON $[a,b]$

$$f(x) - P_N(x) = \underbrace{\frac{f^{(N+1)}(\xi)}{(N+1)!}}_{\text{FIXED}} \underbrace{\prod_{i=0}^{N} (x - x_i)}_{\text{MINIMIZE THE MAX OF THIS}}, \quad \xi \in [x_0, x_N]$$

- CAN PROVE THAT A MINIMAL MAXIMUM EXISTS ON THE INTERVAL $[-1,1]$ IF WE USE ROOTS OF CHEBYSHEV POLYNOMIALS FOR $\{x_i\}$

$$x_i = \cos\left[\frac{(2i+1)\pi}{2(N+1)}\right] \qquad i = 0, 1, 2 \ldots N$$

FOR A GENERAL INTERVAL $[a,b]$

$$x_i = \frac{a+b}{2} + \frac{b-a}{2}\bar{x}_i$$

CAN BE USED TO DESIGN EXPERIMENTS (INTERVAL SPACING)

9/29/2015

* ITERATIVE INTERPOLATION (SEC. 3.1)

- PRODUCE HIGHER ORDER POLYNOMIALS FROM COMBINATIONS OF LOWER ORDER POLYNOMIALS

$$P_N(x) = \sum_{j=0}^{N} f(x_i)\, b_j(x) \qquad \text{(GENERAL)}$$

$$b_j(x) = L_{N_{ij}}(x) \prod_{\substack{k=0 \\ k \neq j}}^{N} \frac{x - x_k}{x_j - x_k} \qquad \text{(LAGRANGE)}$$

- SOME TERMS:

$$P_{(0)}(x) = f(x_0)\, b_0(x) = f(x_0)$$

NOT THE ORDER, BUT THE POINT IT PASSES THROUGH

$$P_{(1)}(x) = f(x_1)\, b_1(x) = f(x_1) \qquad \text{(1 TERM)}$$

$$P_{(0,1)}(x) = \overset{= P_{(0)}}{f(x_0)}\, \frac{x - x_1}{x_0 - x_1} + \overset{P_{(1)}}{f(x_1)}\, \frac{x - x_0}{x_1 - x_0} \qquad \text{(2 TERMS)}$$

$$P_{(0,1)}(x) = \frac{P_{(1)}(x)(x - x_0) - P_{(0)}(x)(x - x_1)}{x_1 - x_0}$$

N=2

$$\underbrace{P_{(0,1,2)}}_{N=2}(x) = f(x_0)\, b_0(x) + f(x_1)\, b_1(x) + f(x_2)\, b_2(x)$$

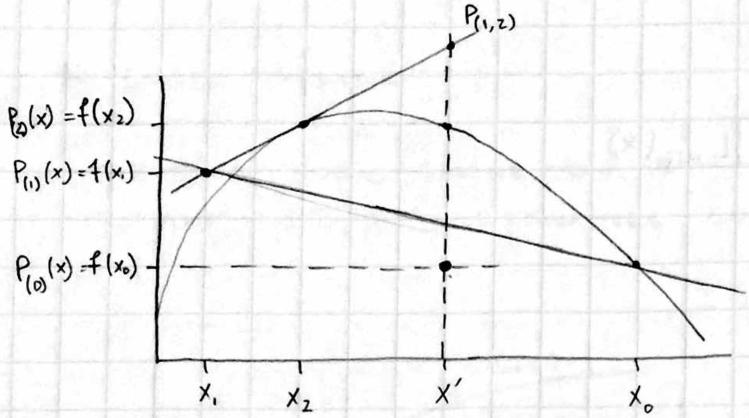
$$= \frac{f(x_0)(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} + \frac{f(x_1)(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} + \frac{f(x_2)(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}$$

$$= \frac{P_{(1,2)}(x)(x - x_0) - P_{(0,1)}(x)(x - x_2)}{x_2 - x_0} \qquad \text{"NEVILLE'S METHOD"}$$

- NEVILLE'S TABLE

$$\begin{array}{lllll}
x_0 & P_{(0)} & & & \\
 & & P_{(0,1)} & & \\
x_1 & P_{(1)} & & P_{(0,1,2)} & \\
 & & P_{(1,2)} & & P_{(0,1,2,3)} \\
x_2 & P_{(2)} & & P_{(1,2,3)} & \\
 & & P_{(2,3)} & & \\
x_3 & P_{(3)} & & &
\end{array}$$

9|29|2015



$x_0 \quad P_{(0)}(x')$

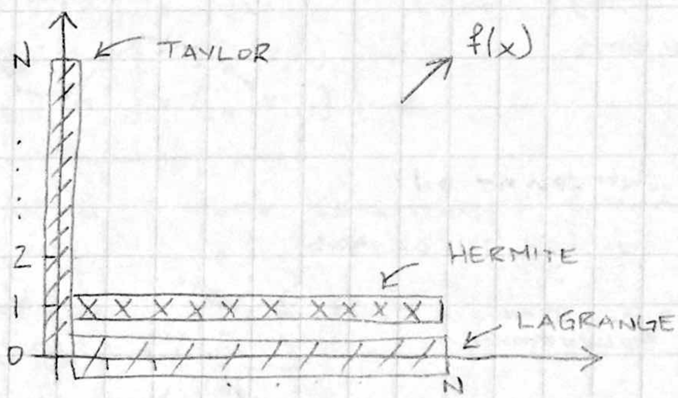
$x_1 \quad P_{(1)}(x') = P_{(0,1)}(x')$

$x_2 \quad P_{(2)}(x') = P_{(1,2)}(x')$

9|30|2015

* HERMITE POLYNOMIAL

- MATCHES f AND f' AT $N+1$ POINTS $\{x_i\}$



- $2N+2$ CONDITIONS:

$$P(x_j) = f(x_j) \text{ AND } P'(x_j) = f'(x_j) \quad \text{FOR } j = 0, 1, \ldots N$$

- UNIQUE POLYNOMIAL OF ORDER $2n+1$: $P_{2N+1}(x)$

INTERIMS OF BASIS FUNCTIONS

$$P_{2N+1}(x) = \sum_{J=0}^{N} f(x_j) H_{2N+1}(x) + \sum_{J=0}^{N} f'(x_j) \hat{H}_{2N+1}(x)$$

W/ PROPERTIES:

$H_{2N+1}(x_i) = \delta_{ij}$ $\qquad \hat{H}_{2N+1}(x_i) = 0$ $\qquad P_{2N+1}(x_i) = f(x_i)$

$\hat{H}_{2N+1}(x_i) = 0$ $\qquad \hat{H}'_{2N+1}(x_i) = f'(x_i)$ $\qquad P'_{2N+1}(x_i) = f'(x_i)$

9/30/2015

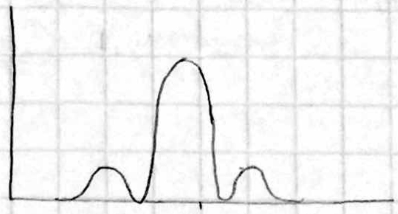
→ FROM TEXT ←

$$H_{2N+1}(x) = [1 - 2(x - x_j) L'_{N_{ij}}(x_j)] L^2_{N_{ij}}(x)$$

$$\hat{H}_{2N+1}(x) = (x - x_j) L^2_{N_{ij}}(x)$$

ERROR TERM:

$$\frac{f^{(2N+2)}(\xi)}{(2n+2)!} \prod_{j=0}^{N} (x - x_j)^2$$



- UP TO THIS POINT, GLOBAL POLYNOMIAL INTERPOLATION

(+) SINGLE FUNCTION (POLYNOMIAL)
(−) 'RINGING' CAN BE SERIOUS

↳ PIECEWISE POLYNOMIAL INTERPOLATION

- LOWER ORDER OVER SMALLER INTERVALS

EG. LINEAR PIECEWISE POLYNOMIAL

$$|P_1(x) - f(x)| \sim O(h^2) \quad \text{WHERE} \quad h \equiv x_{j+1} - x_j \,, \quad \frac{f^{(2)}(\xi_i)}{2!} \underbrace{(x - x_0)}_{h} \underbrace{(x - x_1)}_{h}$$

* HERMITE CUBIC FOR LOCAL INTERPOLATION

x_j $\quad x_{j+1}$ $\quad f(x_j)$ $\quad f(x_{j+1})$ $\quad f'(x_j)$ $\quad f'(x_{j+1})$

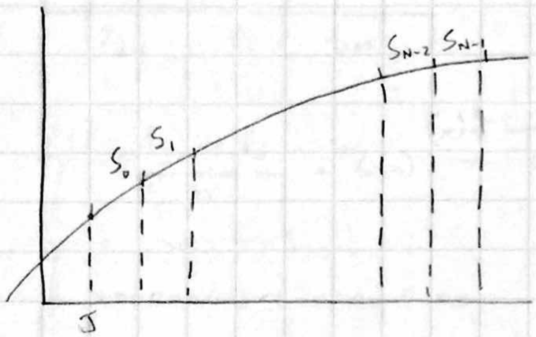
+ CONT. FIRST DERIVATIVE
− NEED TO KNOW $f'(x_j)$

- INSTEAD, USE A SPLINE, ONLY REQUIRES DERIVATIVE TO BE CONTINUOUS.

9/30/2015

* CUBIC SPLINE

- USE A CUBIC SPLINE ON EACH INTERVAL
- MAKE 1^{ST}, 2^{ND} DERIVATIVES CONTINUOUS ACROSS INTERVAL



N+1 POINTS

N SEGMENTS

$S_j \equiv a_j + b_j(x - x_j) + c_j(x - x_j)^2 \ldots$

4N CONDITIONS NEEDED FOR UNIQUE SOLUTION

CONSTRAINTS:

i.) $S_j(x_j) = f(x_j)$: RIGHT
ii.) $S_j(x_{j+1}) = f(x_{j+1})$: LEFT
} CONTINUITY

iii.) $S_j'(x_{j+1}) = S_{j+1}'(x_{j+1})$
iv.) $S_j''(x_{j+1}) = S_{j+1}''(x_{j+1})$
} ?

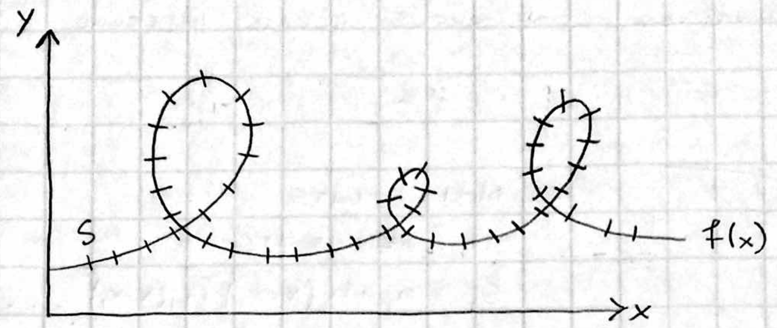
→ NEED 2 MORE CONDITIONS : COMMON STRATEGIES

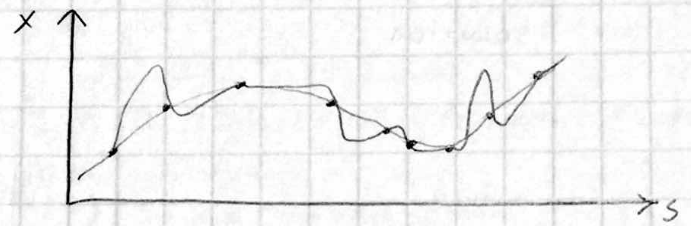
$S_0'(x_0) = f'(x_0)$
$S_{N-1}(x_N) = f'(x_N)$
} MOST ACCURATE, BUT REQUIRES KNOWING DERIVATIVES

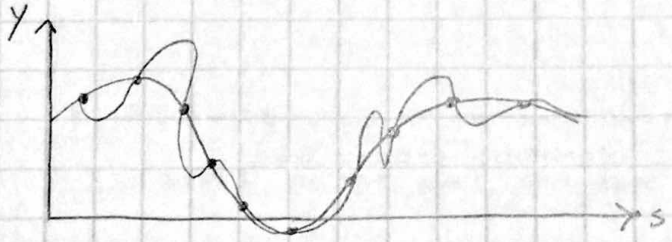
$S_0''(x_0) = S_{N-1}''(x_N) = 0$: 'NATURAL' SPLINE

10/2/2015

* PARAMETRIC INTERPOLATION







* CHAPTER 4.1: NUMERICAL DIFFERENTIATION

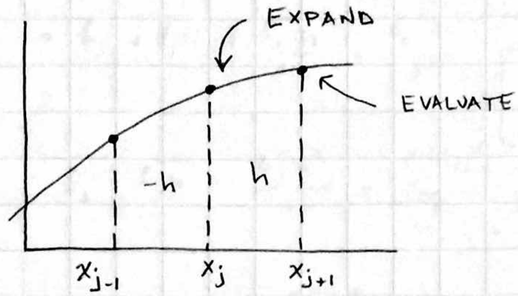
- LOOKING FOR WAYS TO APPROX. FUNCTION DERIVATIVES W/ COMBINATIONS OF FUNCTION EVALUATIONS @ DISCRETE POINTS.

- TEXT: CONSTRUCT A POLYNOMIAL APPROXIMATION TO $f(x)$ THROUGH EVENLY SPACED SAMPLES OF $f(x_i)$, THEN DIFFERENTIATE

EXPAND $f(x)$ ABOUT x_j – THE POINT WHERE WE WANT THE DERIVATIVE

$$f(x) = f(x_j) + f'(x_j)(x - x_j) + \frac{f''(x_j)}{2!}(x - x_j)^2 \ldots$$

10/2/2015



$$f_{j+1} = f_j + hf_i' + \frac{h^2}{2!} f_i'' \ldots$$

$$f_j' = \frac{f_{j+1} - f_j}{h} - \frac{h}{2!} f_j'' + \ldots$$

$$\underbrace{f_j' = \frac{f_{j+1} - f_j}{h}} + O(h) \rightarrow \text{ERROR} \sim \frac{f_j h}{2}$$

APPROXIMATE TO FIRST
DERIVATIVE TO ORDER h ACCURACY.

$$f_j' = \frac{\Delta f_j}{h}$$ FIRST FORWARD DIFFERENCE APPROX. TO $O(h)$

$$f_j' = \frac{\nabla f_j}{h}$$ WHERE $\nabla f_j = f_j - f_{j-1}$

CALCULATE f_j''

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

$$-2\left[f_{j+1} = f_j + hf_j + \frac{h^2}{2!} f_j'' + \frac{h^3}{3!} f_i'''\right]$$

$$= f_{j+2} - 2f_{j+1} + f_j = h^2 f_j'' + h^3 f_j''' + \ldots$$ SOLVE FOR,

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

NOTE: $\Delta^2 f_j = \Delta(\Delta f_j) = \Delta(f_{j+1} - f_j) = \Delta f_{j+1} - \Delta f_j = f_{j+2} - 2f_{j+1} + f_j$

$$f_j'' = \frac{\Delta^2 f_j}{h^2} + O(h) \quad \text{(FORWARD)}$$

$$f_j'' = \frac{\nabla^2 f_j}{h^2} + O(h) \quad \text{(BACKWARD)}$$

10/2/2015

WANT f_j' TO $O(h^2)$

$$-4\left[f_{j+1} = f_j + hf_j + \frac{h^2}{2!} f_j'' + \frac{h^3}{3!} f_j''' \right]$$

$$f_{j+2} = f_j + 2hf_j + \frac{4h^2}{2!} f_j'' + \frac{8h^3}{3!} f_j''' + \dots$$

$$f_{j+2} - 4f_{j+1} + 3f_j = -2h f_j' + \frac{4h^3}{6} f_j''' +$$

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

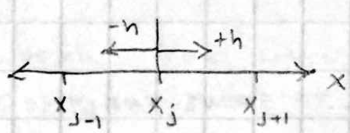
10/5/2015

* NUMERICAL DIFFERENTIATION

N^{TH} ORDER DERIVATIVE TO ACCURACY $O(h^m)$ REQUIRES $n+m$ POINTS INVOLVING $n+m-1$ TAYLOR SERIES

CENTRAL DIFFERENCE APPROXIMATIONS

- COMBINE FWD / BWD APPROXIMATIONS TO ELIMINATE HIGHER ORDER TERMS, RESULTING IN A SYMMETRIC SAMPLING AROUND x_j



$$f_{j+1} = f_j + f_j' h + \frac{f_j''}{2!} h^2 + \frac{f_j'''}{3!} h^3 \dots \quad \text{(FORWARD)}$$

$$+ \quad f_{j-1} = f_j - f_j' h + \frac{f_j'' h^2}{2!} - \frac{f_j''' h^3}{3!}$$

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

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

10/5/2015

$$f_{j+1} = f_j + f_j' h + \frac{f_j'' h^2}{2!} + \frac{f_j''' h^3}{3!} + \frac{f^{(4)} h^4}{4!} + \dots$$

$-$

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

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

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

EXTRA POINT GIVES $O(h^2)$ ACCURACY INSTEAD OF $O(h)$ W/ TWO POINTS

INTERPRETATION OF $O(h^m)$

IF E_h IS THE ERROR FOR $O(h^m)$, THEN IF $h \to h/2$

$$E_{h/2} = O\left(\frac{h}{2}\right)^m = E_h \left(1/2\right)^m$$

- NOTICE h IS IN DENOMONATOR FOR ALL DERIVATIVES, HOWEVER IT CAN DESTROY SOLUTION VIA ROUND OFF ERROR. HOWEVER IT'S BETTER FOR TRUNCATION ERROR

* COMPUTER ERRORS WHEN NUMERICALLY DIFFERENTIATING

$$f_j' = \frac{f_{j+1} - f_{j-1}}{2h} - \frac{f'''(\xi) h^2}{6} \rightarrow \underline{\text{EXACT}}$$

COMPUTE, EXACT, ROUND OFF

$$\hat{f}_j = f_j + \varepsilon_j$$

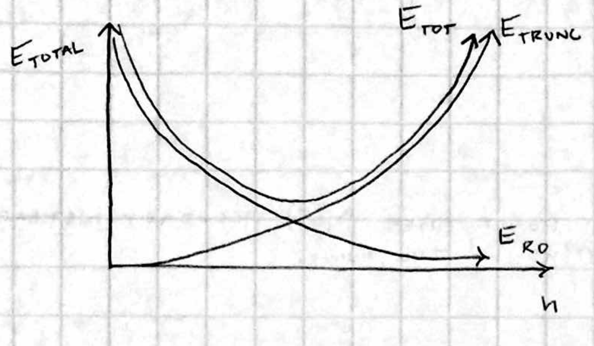
$$\hat{f}_j' = \frac{\hat{f}_{j+1} - \hat{f}_{j-1}}{2h} = \overbrace{\frac{f_{j+1} - f_{j-1}}{2h}}^{f_j' + E_{TRUNC}} + \frac{\varepsilon_{j+1} - \varepsilon_{j-1}}{2h}$$

$$\left| \hat{f}_j' - f_j' \right| = \left| E_{TRUNC} - E_{ROUND\text{-}OFF} \right| \leq \left| E_{TRUNC} \right| + \left| E_{ROUND\ OFF} \right|$$

10/5/2015

ASSUME $f'''(\xi)$ IS BOUNDED BY SAME NUMBER, m

$$E_{TOT}(h) = \frac{mh^2}{6} + \frac{\varepsilon}{h}$$ → RESULTS IN AN ERROR TRADE OFF



$$\frac{\partial E_{TOT}}{\partial h} = 0$$

$$\frac{mh}{3} = \frac{\varepsilon}{h^2}$$

* CURVE FITTING:

- SO FAR ALL INTERPOLANTS PASS EXACTLY THROUGH DATA POINTS $\{x_i\}, \{y_i\}$

WANT A 'BEST FIT' SOLUTION, NOT NECESSARILY AN INTERPOLATION THROUGH POINTS

EVALUATE ERROR BY MINIMIZING THE SUM OF THE SQUARED ERRORS

TOTAL ERROR ⇒ $\varepsilon = \sum_{j=0}^{N} \left[f(x_j) - \tilde{P}(x_j) \right]^2$, GOAL IS TO FIND $\tilde{P}(x_j)$ WHICH MINIMIZES ε

EG. STRAIGHT LINE THROUGH DATA "LINEAR LEAST SQUARES" FIT

$$\tilde{P}(x) = P_1(x) = a_0 + a_1 x, \quad \varepsilon = \sum_{j=0}^{N} \left[f(x_j) - (a_0 + a_1 x_j) \right]^2$$

$$\frac{\partial \varepsilon}{\partial a_0} = \frac{\partial \varepsilon}{\partial a_1} = 0 \rightarrow \frac{\partial \varepsilon}{\partial a_0} = 0 = -2 \sum_{j=0}^{N} \left[f(x_j) - (a_0 + a_1 x) \right]$$

$$\frac{\partial \varepsilon}{\partial a_1} = 0 = -2 \sum_{j=0}^{N} \left[f(x_j) - (a_0 + a_1 x) \right] (-x_j)$$

WHY?

10/5/2015

FOR POLYNOMIALS $0, \ldots N$ $\sum_{J=0}^{N} \rightarrow N+1$

FOR CURVE FITTING $1, \ldots m$ $\sum_{J=1}^{N} \rightarrow m$ # OF DATA POINTS

NORMAL EQUATIONS

$$a_0 m + a_1 \sum_{J=1}^{m} x_j = \sum_{J=1}^{m} y_i$$

$$a_0 \sum_{J=1}^{m} x_j + a_1 \sum_{J=1}^{m} x_j^2 = \sum_{J=1}^{m} x_j y_j$$

$$\begin{bmatrix} m & \sum x_j \\ \sum y_j & \sum x_j^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum x_j y_j \end{bmatrix}$$

$$a_0 = \frac{\sum x_j^2 \sum y_j - \sum x_j y_j \sum x_j}{m \sum x_j^2 - (\sum x_j)^2}$$

$$a_1 = \frac{m \sum x_j y_j - \sum x_j \sum y_i}{m \sum x_j^2 - (\sum x_j)^2}$$

ALL KNOWN INFORMATION

10/7/2015

LEAST SQUARES CURVE FITTING:

IF $\tilde{P}(x) = P_o(x) = a_n x^n + a_{n-1} x^{n-1} + \dots a_1 x + a_0$

THEN MINIMIZE

$$E = \sum_{i=1}^{N} \left[f(x_i) - P_n(x_i) \right]^2 = \sum_{i=1}^{m} \left[f(x_i) - \sum_{k=0}^{N} a_k x_i^k \right]^2$$

$$\frac{\partial E}{\partial a_j} = 0 \quad \text{FOR } j = 0, 1, \dots N$$

$$\frac{\partial E}{\partial a_j} = 2 \sum_{j=1}^{m} \left[f(x_i) - \sum_{k=0}^{N} a_k x_i^k \right] (-x_i^j) = 0$$

$$= -2 \sum_{i=1}^{m} y_i x_i + 2 \sum_{k=0}^{N} a_k \sum_{i=1}^{m} x_i^{j+k} = 0$$

* N+1 NORMAL EQUATIONS:

$$\sum_{k=0}^{N} a_k \sum_{i=1}^{m} x_i^{j+k} = \sum_{i=1}^{m} y_i x_i^j, \quad j = 0, 1, \dots N$$

EXPAND

$$j=0: \quad a_0 \sum_{i=1}^{m} x_i^0 + a_1 \sum_{i=1}^{m} x_i^1 + \dots a_n \sum_{i=1}^{n} x_i^N = \sum_{i=1}^{m} y_i x_i^0$$

$$j=1: \quad a_0 \sum_{i=1}^{m} x_i^1 + a_1 \sum_{i=1}^{m} x_i^2 + \dots a_n \sum_{i=1}^{m} x_i^{N+1} = \sum_{i=1}^{m} y_i x_i^1$$

$$j=N: \quad a_0 \sum_{i=1}^{m} x_i^N + a_1 \sum_{i=1}^{m} x_i^{N+1} + \dots a_n \sum_{i=1}^{m} x_i^{2N} = \sum_{i=1}^{m} y_i x_i^N$$

$$\underline{\underline{X}}\, \underline{A} = \underline{b} \qquad \underline{A} = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}, \quad b =$$

X IS INVERTIBLE AS LONG AS x_i's ARE UNIQUE

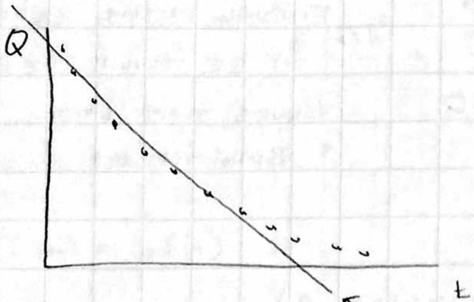
$$\underline{\underline{X}} = \begin{bmatrix} m \\ \sum x_i^2 \\ \sum x_i^3 \\ \vdots \\ \sum x_i^N \end{bmatrix}$$

LOOK AT VIDEO ↓

10/7/2015

WHAT IF $\tilde{P}(x)$ IS NOT A POLYNOMIAL?

- E.G. DATA DOES NOT FIT A POLYNOMIAL



GOES NEGATIVE, NOT A GOOD APPROXIMATION

$$\tilde{P}(x) = be^{ax}$$

$$E = \sum_{i=1}^{m} \left[f(x_i) - be^{ax_i} \right]^2$$

N.L — ROOT

LINEARIZE

$$\frac{\partial E}{\partial b} = 0 = 2\sum_{i=1}^{m} \left[y_i - be^{ax_i} \right]\left[-e^{ax_i} \right]$$

$$\frac{\partial E}{\partial a} = 0 = 2\sum_{i=1}^{m} \left[y_i - be^{ax_i} \right]\left(-bx_i e^{ax_i} \right)$$

$$\rightarrow b\sum_{i=1}^{m} e^{2ax_i} = \sum_{i=1}^{m} y_i e^{ax_i} \quad ; \quad b\sum_{i=1}^{m-1} x_i e^{2ax_i} = \sum x_i y_i e^{ax_i}$$

↳ PROBLEM: CANNOT WRITE AS $\underline{\underline{X}} \begin{bmatrix} a \\ b \end{bmatrix} = \underline{b}$, THERE IS NO EXACT SOLUTION

TWO SOLUTIONS:

1.) LINEARIZE

$$\{P\} = mx + b$$

$$\ln\left\{ \tilde{P}(x) = y = be^{ax} \right\} \rightarrow \ln y = \ln b + ax \rightarrow \underbrace{\ln y}_{\text{KNOWN}} = \underbrace{a}_{a_1}x + \underbrace{\ln b}_{a_0}$$

$$Y = B + ax \quad \text{(LINEAR IN THESE VARIABLES)}$$

→ APPLY LEAST SQUARES TO THIS EQ. & FIND $a + B$ WHICH MINIMIZE E IN TERMS OF $\{x_i\}$ + $\{Y_i\}$

10/7/2015

ii.) WRITE AS A SYS. OF N-L EQUATIONS AND USE A ROOT FINDING TECHNIQUE

$$f_1(a,b) = b\sum_{i=1}^{m} e^{2ax_i} - \sum_{i=1}^{m} y_i e^{ax_i} = 0$$

$$f_2(a,b) = b\sum_{i=1}^{m} x_i e^{2ax_i} - \sum_{i=1}^{m} x_i y_i e^{ax_i} = 0$$

FINDING INITIAL GUESS CAN BE TOUGH. TRY PLOTTING VALUES THAT LOOK "GOOD". & USING THOSE.

* NUMERICAL INTEGRATION, "QUADRATURE" (CHAPTER 8)

- NO 'h's' IN DENOMONATOR, SO INTEGRATION IS MORE STABLE THAN DIFFERENTIATION.

APPROXIMATE $f(x) \simeq \sum_{i=0}^{N} f_i L_{N,i}(x)$

$$\int_a^b f(x)\,dx \simeq \sum_{i=0}^{N} f_i \underbrace{\int_a^b L_{N,i}(x)\,dx}_{a_i \;\rightarrow\; \text{THIS IS A NUMBER}} = \sum_{i=0}^{N} a_i f_i$$

WEIGHTED SUM OF FUNCTION EVALUATIONS

$$\int_a^b f(x)\,dx = \underbrace{\sum_{i=0}^{N} a_i f_i}_{\text{QUADRATURE (NUMERICAL METHOD)}} + \text{ERROR TERM}$$

← QUANTIFY (NUMERICAL ANALYSIS)

TEXT:

$$\int_a^b f(x)\,dx = \sum_{i=0}^{N} a_i f_i + \int_a^b \prod_{i=0}^{N} \frac{(x-x_i)}{(N+1)!} f^{(N+1)}(\xi)\,dx$$

HORNBECK: TAYLOR SERIES

$$I(x) = \int_a^x f(\tilde{x})\,d\tilde{x}$$ → EXPAND $I(x)$ AROUND $x=a$

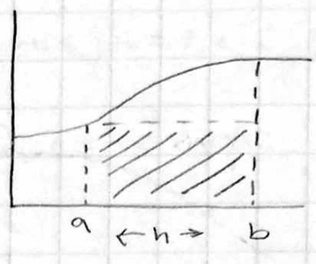
10/7/2015

$$I(x) = I(a) + I(a)(x-a)$$
$$= 0 + f(a)(x-a)$$
$$= f(a) + \frac{f'(a)}{2}(x-a)^2 + \frac{f''(a)}{3!}(x-a)^3$$

$$I(b) = f(a)h + \frac{f'(a)h^2}{2} + \frac{f''(a)h^3}{3!} + \ldots$$

SIMPLEST SOLUTION,

$$I(b) = hf(a)$$



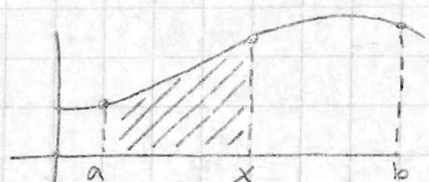
GOOD WHEN h IS SMALL

10/9/2015

NUMERICAL QUADRATURE

$$\int_a^b f(x)\,dx = \sum_{i=0}^{N} a_i f_i + \text{ERROR TERM}$$

$$I(x) = \int_a^b f(\hat{x})\,d\hat{x} \rightarrow$$



$$I(x) = I(a) + I'(a)(x-a) + \frac{I''(a)}{2!}(x-a)^2 + \ldots$$

$$= f(a)(x-a) + \frac{f'(a)}{2!}(x-a)^2 + \ldots$$

$$I(b) = f(a)h + \frac{f'(a)h^2}{2!} + \frac{f''(a)h^3}{3!} + \ldots \rightarrow$$ KEEP ANOTHER TERM

ii.) $I(b) = hf_a + \frac{h^2}{2}f_a' + \frac{h^3}{3!}f_a + \ldots$

↳ USE FORWARD DIFFERENCE, $f_a' = \frac{f_b - f_a}{h} - \frac{h}{2}f''(\xi_0)$

$$I(b) = hf_a + \frac{h^2}{2}\left(\frac{f_b - f_a}{h}\right) - \frac{h^3}{4}f''(\xi) + \frac{h^3}{6}$$

$$\boxed{I(b) = \underbrace{hf_a + \frac{h}{2}(f_b - f_a)}_{\text{TRAPAZOIDAL RULE}} - \frac{h^3}{12}f''(\xi)}$$

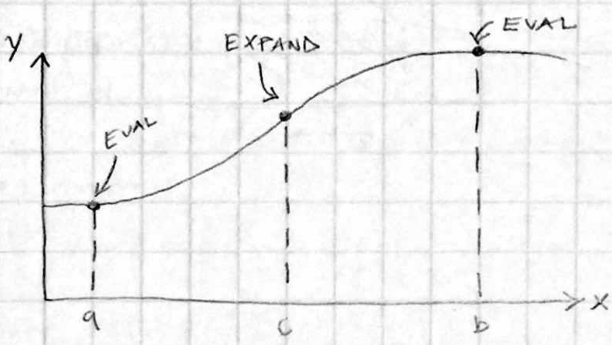
CAN'T CALCULATE THIS

10/9/2015

FOR NUMERICAL INTEGRATION, LOOK @ ORDER OF POLYNOMIAL THAT CAN BE INTEGRATED EXACTLY TO DETERMINE ORDER OF ERROR.

$$\text{I.E.} \quad P_N^{(N+1)}(x) = 0$$

INSTEAD OF USING A FWD DIFFERENCE, USE A CENTER DIFFERENCE



BEFORE: $b - a = h$

NOW: $b - a = 2h$

$$I(b) = I(c) + hf_c + \frac{h^2}{2!} f_c' + \frac{h^3}{3!} f'' + \ldots$$

$$I(a) = I(c) - hf_c + \frac{h^2}{2!} f_c' - \frac{h^3}{3!} f_c'' + \ldots$$

$$I(b) = 2hf_c + \frac{2h^3 f_c''}{3!} + \frac{2h^3}{3!} f_c^{(4)} + \ldots$$

↳ USE C.D. APPROX: $f_c'' = \dfrac{f_a - 2f_c + f_b}{h^2} - \dfrac{h^2}{12} f^{(4)}(\xi)$

$$\boxed{I(b) = \frac{h}{3}\left[f_a + 4f_c + f_b\right] - \frac{h^5}{90} f^{(4)}(\xi)}$$

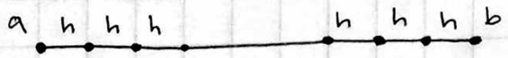
← CAN DO A CUBIC EXACTLY, 4TH DERIVATIVE OF A CUBIC = 0, SO ERROR → 0

* NEWTON COTES FORMULA'S

- CLOSED FORMULA'S INCLUDE END POINTS
- OPENS DO NOT

10/9/2015

* NEWTON-COTES OPEN FORMULA'S
  - DO NOT INCLUDE LIMITS OF INTEGRATION



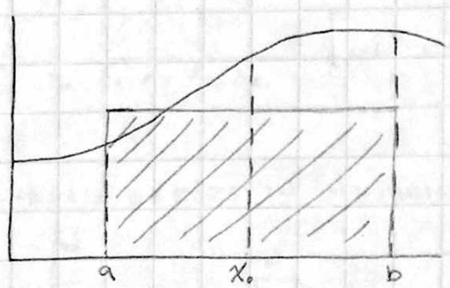
$b - a = (N+2)h$

$$I(b) = I(x_0) + hf_0 + \frac{h^2 f_0'}{2!} + \frac{h^3 f_0''}{3!} + \dots$$

$-$

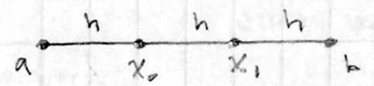
$$I(a) = I(x_0) - hf_0 + \frac{h^2 f_0'}{2!} + \frac{h^3 f_0''}{3!} + \dots$$

$$\boxed{I(b) = 2hf_0 + \frac{h^3 f''(\xi)}{3}}$$ MID POINT RULE



FOR LINEAR FUNCTION, ERROR CANCELS OUT

E.G. $N = 1$



EXPAND ABOUT x_0

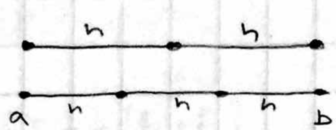
$$I(b) = I(x_0) + 2h_0 f_0 + \frac{(2h)^2}{2!} f_0' + \frac{(2h)^3 f_0''}{3!} + \dots$$

$-$

$$I(a) = I(x_0) - h_0 f_0 + \frac{h^3}{2!} f_0' - \frac{h^3}{3!} f_0'' + \dots$$

$$I(b) = 3hf_0 + \underbrace{\tfrac{3}{2}h^2 f_0'}_{\text{FORWARD DIFF.}} + \frac{9h^3}{6} f''(\xi)$$ → CAN STILL ONLY DO A LINEAR FUNCTION EXACTLY

- SOME ADVANTAGE B.C. h IS SMALLER THAN CLOSED CASE:



$F' \,\&$

I''

f'

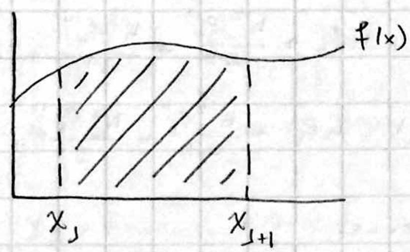
N EVEN: P_{N+1} CAN BE INTEGRATED EXACTLY

N ODD: P_N CAN BE INTEGRATED EXACTLY.

10/13/2015

* COMPOSITE INTEGRATION

- USE LOW ORDER N-C FORMULAS ON SUB INTERVALS



$$S_{j+1} = \int_{x_j}^{x_{j+1}} f(x)\,dx$$

- TRAPEZOIDAL RULE:

$$S_j = \frac{h}{2}\left[f_{j+1} + f_j\right] - \frac{h^3}{12} f''(\xi) \qquad \xi \in [x_j, x_{j+1}]$$

- COMPOSITE TRAPEZOIDAL RULE

$$I(b) = \sum_{j=1}^{N} \frac{h}{2}\left[f_j + f_{j-1}\right] - \sum_{j=1}^{N} \frac{h^3}{12} f''(\xi_j)$$

$$I(b) = \frac{h}{2}\left[2\sum_{i=1}^{N-1} f_i + f_0 + f_N\right]$$ COMPOSITE TRAPEZOIDAL RULE

INSIDE POINTS OUTSIDE POINTS

- ERROR $= \sum_{j=1}^{N} \frac{h^3}{12} f''(\xi) = \frac{h^3}{12} N \overline{f''}(\xi)$ WHERE $\overline{f''}$ IS THE AVERAGE VALUE OF $f''(\xi_i)$

$$b - a = Nh \rightarrow N = \frac{b-a}{h}$$

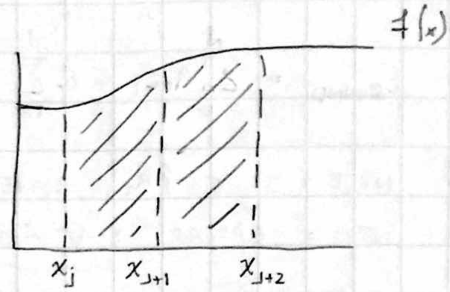
HIGHER ORDER $h \equiv$ BETTER/MORE ACCURATE THE SOLUTION

ERROR $= \frac{b-a}{12} h^2 \overline{f''}(\xi)$ WHICH IS $O(h^2)$

APPLICATION OF METHOD N TIMES REDUCES ORDER OF ERROR BY A FACTOR OF h

10/13/2015

EG. SIMPSON'S RULE

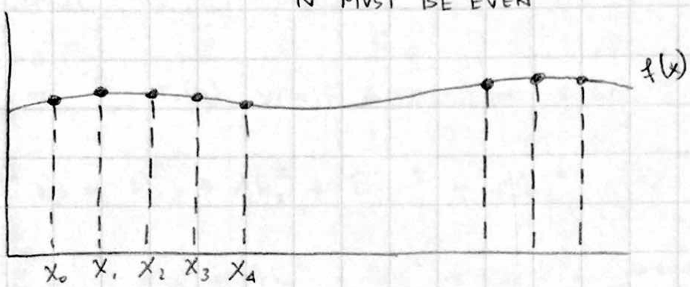


$$I(b) = \frac{h}{3}\left[f_a + 4f_c + f_b\right]$$

$$I(b) = \frac{h}{3}\left[f_0 + 2\sum_{j=1}^{N/2-1} f_{2j} + 4\sum_{j=1}^{N/2} + f_N\right]$$

$$- \frac{(b-a)h^4}{18} f''(\xi)$$

N MUST BE EVEN



ROUNDING ERRORS IN NUMERICAL INTEGRATION

$$\int_a^b f(x)dx = \underbrace{\sum_{i=0}^{N} a_i f_i}_{\text{QUADRATURE}} + E_{TRUNCATION},$$

TRUNCATION HAS h IN NUMERATOR SO $E_{TRUNCATION} \to 0$ AS $h \to 0$

ON COMPUTER:

$$\hat{f}_i = f_i + \varepsilon_i \rightarrow \underbrace{\sum_{i=0}^{N} a_i \hat{f}_i}_{\text{COMPUTED QUADRATURE}} = \underbrace{\sum_{i=0}^{N} a_i f_i}_{\text{EXACT QUADRATURE}} + \underbrace{\sum_{i=0}^{N} a_i \varepsilon_i}_{\text{ROUND-OFF}}$$

$$E_{TOTAL} = \left| \int_a^b f(x)dx - \sum_{i=0}^{N} a_i \hat{f}_i \right| = \left| \sum_{i=0}^{N} a_i f_i + E_{TRUNC} - \sum_{i=0}^{N} a_i f_i - E_{ROUND} \right|$$

$$= \left| E_{TRUNK} - E_{ROUND} \right| \leq \left| E_{TRUNC} \right| + \left| E_{ROUND} \right|$$

$E_{ROUNDOFF} = \sum_{i=0}^{N} a_i \varepsilon_i$, TRAPOZOIDAL RULE $\sum_{i=0}^{N} a_i f_i = \frac{h}{2}\left[f_0 + f_i\right]$

$a_0 = a_1 = \frac{h}{2}$

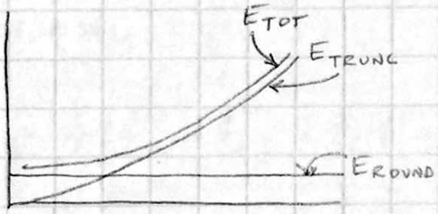
10/13/2015

$$E_{ROUND} = \sum_{i=0}^{N} a_i \varepsilon_i \leq \varepsilon \sum_{i=0}^{N} |a_i| \quad \text{WHERE } \varepsilon \equiv \text{MAX} |\varepsilon_i|$$

NOTE THAT $\{a_i\}$ HAVE A FACTOR OF h IN NUMERATOR, SO IT WOULD APPEAR THAT $\varepsilon_{ROUND} \to 0$ AS $h \to 0$

$$\int_a^b dx = b - a = \sum_{i=0}^{N} a_i(1) \,, \quad E_{ROUND} \leq \varepsilon(b-a)$$

I.E. INDEPENDENT OF h AND DOES NOT $\to 0$ FOR $h \to 0$



* ROMBERG INTEGRATION

- CAN BE USED ON ANY N-C FORMULA

IDEA: COMPUTE QUADRATURE RULE w/ TWO DIFFERENT h INTERVALS & COMBINE TO CANCEL THE LEADING ERROR TERM, GIVING A BETTER APPROXIMATION

10/14/2015

* ROMBERG INTEGRATION

→ START w/ COMPOSITE TRAPAZOIDAL RULE

$$I(b) = \frac{h}{2}\left[f_0 + 2\sum_{i=1}^{N-1} f_i + f_N\right] + \underbrace{Ah^2 + Bh^4 + Ch^6 \ldots}$$

$$A = \frac{f''(\xi)}{12}$$

→ DEFINE

$$R_{N,0} = \frac{h}{2}\left[f_0 + 2\sum_{i=1}^{N-1} f_i + f_N\right]$$ COMPOSITE TRAPEZOIDAL RULE ON N PANELS, SIZE h

LOOK @ $I(b)$ USING DIFFERENT PANEL SIZES, h_1, AND $h_2 = \frac{h_1}{2}$

h $\quad I(b) = R_{2N} + Ah_2^2 + Bh_2^4 + Ch_2^6 + \ldots$

h_2 $\quad I(b) = R_N + Ah_1^2 + Bh_1^4 + Ch_1^6 + \ldots$

$$\hookrightarrow = R_N + A(2h_2)^2 + B(2h_2)^4 + C(2h_2)^6 = R_N + 4h_2^2 + 4B$$

$$3I(b) = 4R_{2N} - R_N - 12B_2^4 - 60Ch_2^6$$

$$I(b) = \frac{4R_{2N} - R_N}{3} - 4Bh_2^4 - 20Ch_2^6$$

$R_N \to O(h_1^2)$

$R_{2N} \to O(h_2^2) = O\left(\frac{h_1^2}{4}\right)$, $O\left(\frac{h_1^4}{16}\right)$

EACH APPLICATION INCREASES ORDER OF ERROR TERM BY h_1^2 WHEREAS HALVING REDUCES ONLY BY A FACTOR OF 1/4. THIS HAS RESULTED IN SIMPSONS RULE

GENERALIZE: m = # OF PANEL DOUBLINGS

j = # OF ERROR REMOVALS

$$R_{m,j} = \frac{4^j R_{m,j-1} - R_{m-1,j-1}}{4^j - 1}$$

$$\begin{array}{lllll}
R_{0,0} & & & & \\
 & R_{1,1} & & & \\
R_{1,0} & & R_{2,2} & & \\
 & R_{2,1} & & R_{3,3} & \\
R_{2,0} & & R_{3,2} & & R_{4,4} \\
 & R_{3,1} & & R_{4,3} & \\
R_{3,0} & & R_{4,2} & & \\
 & R_{4,1} & & & \\
R_{4,0} & & & &
\end{array}$$

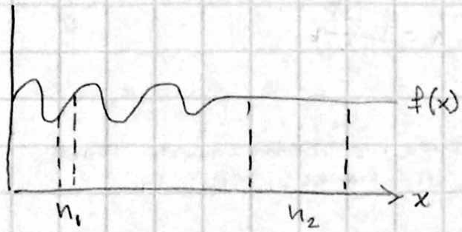
$O\left(\frac{h}{2^4}\right)^{10}$ $\quad O\left(\frac{h^{10}}{16^{10}}\right)$

$O\left(\frac{h}{2^m}\right)^{2m+2}$

10/14/2015

USE $R_{m,m}$; CONVERGES FASTER

* ADAPTIVE QUADRATURE



ERROR $\sim h^m f^{(n)}$

IF WANT CONSTANT ERROR OVER INTERVAL, $h_1 << h_2$

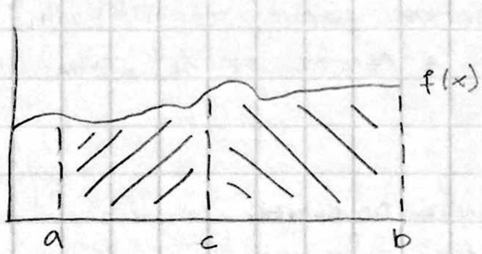
WANT TO UNIFORMLY DISTRIBUTE ERROR

SIMPSON'S RULE:

$$\int_a^b f(x)\,dx = \frac{h}{3}\left[f_a + 4f_c + f_c\right] - \frac{h^5 f^{iv}(\xi)}{90}$$

DEFINE: $S(a,b) \equiv$ SIMPSON'S ON $[a,b]$

$$\int_a^b f(x)dx = S(a,b) - \frac{h^5 f^{iv}(\xi)}{90}$$



$$\int_a^b f(x)dx = S(a,c) - \frac{h^5 f^{(4)}(\xi_1)}{90} + S(c,b) - \frac{h^5 f^{(4)}(\xi_2)}{90}$$

LET $f^{(4)}(\xi) \equiv \dfrac{f^{(4)}(\xi_1) + f^{(4)}(\xi_2)}{2}$ THEN,

$$\int_a^b f(x)dx = S(a,c) + S(b,c) - \frac{2}{90}\left(\frac{h}{2}\right)^5 f^{(4)}(\xi)$$

$$= S(a,c) + S(c,b) - \frac{1}{16}\left[\frac{h^5 f^{(4)}(\xi)}{90}\right]$$

$$S(a,b) = -\frac{h^5 f^{(4)}(\xi)}{90} = S(a,c) + S(c,b) - \frac{1}{16}\left[\frac{h^5 f^{(4)}(\xi)}{90}\right]$$

ASSUMPTION
$f^{(4)}(\xi) = \overline{f^{(4)}}(\xi)$

10/14/2015

$$\frac{15}{16}\left[\frac{h^5 f^{(4)}(d)}{90}\right] \simeq S(a,b) - S(a,c) - S(c,b)$$

$$\left| \int_c^b f(x)\,dx - S(a,c) - S(c,b) \right| \cong \frac{1}{15}\left| S(a,b) - S(a,c) - S(c,b) \right|$$

INTERPRETATION: $S(a,c) + S(c,b)$ APPROXIMATES $\int_a^b f(x)\,dx$ 15 TIMES BETTER THAN $S(a,b)$

MIDTERM STUDY GUIDE:

* TAYLORS THEOREM

IF $f(x)$ HAS $N+1$ DERIVATIVES ON $[a,b]$, THEN

$f(x) = P_N(x) + R_N(x)$

WHERE $P_N(x) = \sum_{K=0}^{N} \frac{f^{(K)}(x_0)}{K!}(x-x_0)^K$ BY DEFINITION, $P_N(x_0) = f(x_0)$

$$R_N(x) = \frac{f^{(N+1)}(\xi)}{(N+1)!}(x-x_0)^{N+1} \quad \text{WHERE } \xi \in [x, x_0]$$

* ERROR

LINEAR: $\varepsilon = KN\varepsilon_0$

EXPONENTIAL: $\varepsilon = K^N\varepsilon_0$

$P_N = \frac{1}{3}P_{N-1} \rightarrow P_N - \hat{P}_N = \frac{1}{3}(P_{N-1} - \hat{P}_{N-1}) = \varepsilon_N = \frac{1}{3}\varepsilon_{N-1}$

$n=1: \quad \varepsilon_1 = \frac{1}{3}\varepsilon_0$

$n=2: \quad \varepsilon_2 = \frac{1}{3}\varepsilon_1 \rightarrow \varepsilon_2 = (\frac{1}{3})^2\varepsilon_0 \rightarrow \varepsilon_N = (\frac{1}{3})^N\varepsilon_0$

$n=3: \quad \varepsilon_3 = \frac{1}{3}\varepsilon_2 \rightarrow \varepsilon_3 = (\frac{1}{3})^3\varepsilon_0$

$\lim_{n\to\infty} \varepsilon_N = \lim_{n\to\infty} (\frac{1}{3})^N\varepsilon_0 = 0$ (CANNOT ASSUME EXACT CONSTANTS)

$\rightarrow$ LET $\frac{1}{3} = \hat{\frac{1}{3}} + \delta \rightarrow \hat{\frac{1}{3}} = \frac{1}{3} - \delta$

$\hat{P}_N = (\frac{1}{3} - \delta)\hat{P}_{N-1}, \quad P_N = \frac{1}{3}P_{N-1}$

$\varepsilon_N = \frac{1}{3}P_{N-1} - (\frac{1}{3} - \delta)\hat{P}_{N-1} \rightarrow \varepsilon_N = \frac{1}{3}P_{N-1} - \frac{1}{3}\hat{P}_{N-1} + \delta\hat{P}_{N-1}$

$\varepsilon_N = \frac{1}{3}\varepsilon_{N-1} + \delta\hat{P}_{N-1}$

$n=1 \quad \varepsilon_1 = \frac{1}{3}\varepsilon_0 + \delta\hat{P}_0$

$n=2 \quad \varepsilon_2 = \frac{1}{3}\varepsilon_1 + \delta\hat{P}_1 = \frac{1}{3}(\frac{1}{3}\varepsilon_0 + \delta\hat{P}_0) + \delta\hat{P}_1 = (\frac{1}{3})^2\varepsilon_0 + \frac{1}{3}\delta\hat{P}_0 + \delta P_1$

$n=3 \quad \varepsilon_3 = \frac{1}{3}((\frac{1}{3})^2\varepsilon_0 + \frac{1}{3}\delta\hat{P}_0 + \delta P_1) + \delta\hat{P}_2 = (\frac{1}{3})^3\varepsilon_0 + (\frac{1}{3})^2\delta\hat{P}_0 + (\frac{1}{3})\hat{P}_1 + \delta\hat{P}_2$

$\varepsilon_N = (\frac{1}{3})^N\varepsilon_0 + \underbrace{n\delta\hat{P}_0}$ $\quad \varepsilon_N \le (\frac{1}{3})^N\varepsilon_0 + n\delta\hat{P}_0$

$\hookrightarrow$ ERRORS GROW LINEARLY

* FIXED POINT ROOT FINDING

WANT SOLUTION TO $f(x) = 0$, SOLVE BY SOLVING $p = g(p)$ INSTEAD (FIXED POINT)

EX. $x^3 - 7x + 2 = 0$ WHERE $f(x) = x^3 - 7x + 2$

$g(x)$ FORM $\rightarrow$ $x = (x^3+2)/7$ $\rightarrow$ $x_{N+1} = (x_N^3 + 2)/7$

$\rightarrow$ ITERATE UNTIL $|x_{N+1} - x_N| < \varepsilon$ $\rightarrow$ $x_{N+1} \equiv$ ROOT OF $f(x)$

CONVERGENCE CRITERIA:

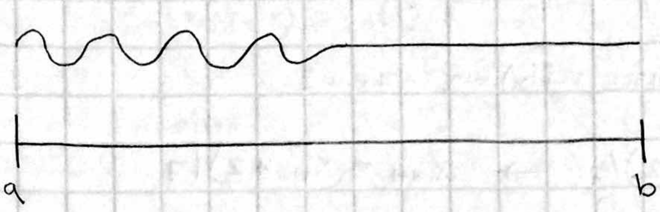
$P_N = g(P_N)$ $\rightarrow$ $P_N - P_{N-1} = g(P_N) - g(P_{N-1})$ $\rightarrow$ MVT $= g'(P_N) = \dfrac{g(P_N) - g(P_{N-1})}{P_N - P_{N-1}}$

$(P_N - P_{N-1})\, g'(P_N) = g(P_N) - g(P_{N-1})$

$\hookrightarrow$ $|g'(P_N)| < 1$ FOR CONVERGENCE, $\varepsilon_N = g'(P_N)(P_N - P_{N-1})$

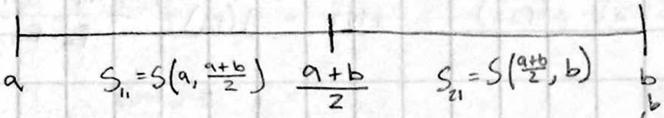
10/16/2015

* ADAPTIVE QUADRATURE

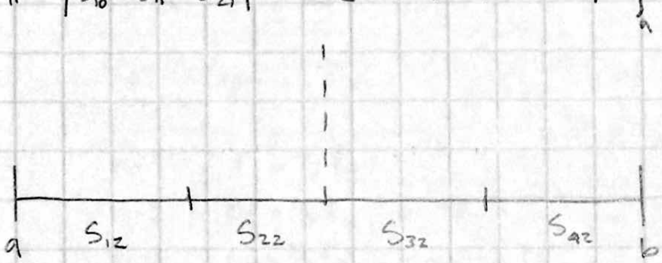


$S(a,b) \equiv S_{10}$

SECTION ↗ ↖ REFINEMENT



IF $\left|S_{10} - S_{11} - S_{21}\right| < 15\varepsilon$ STOP $\rightarrow \left|\int_a^b f(x)\,dx - S_{11} - S_{21}\right| < \varepsilon$



IF $\left|S_{11} - S_{12} - S_{22}\right| < 15\left(\frac{\varepsilon}{2}\right)$; IF $\left|S_{21} - S_{32} - S_{42}\right| < 15\left(\frac{\varepsilon}{2}\right)$

$\int_a^b f(x)\,dx \cong$ SUM OF PIECES

* GAUSSIAN QUADRATURE (SECTION 4.7)

- IS THERE AN OPTIMAL SET OF POINTS FOR APPROXIMATING $\int_a^b f(x)\,dx$
- OPTIMAL IN THE SENSE THAT WE CAN INTEGRATE THE MAX. ORDER POLYNOMIAL USING THE MINIMUM POSSIBLE POINTS.

$$\int_a^b f(x)\,dx \approx \sum_{i=1}^{N} a_i f(x_i)$$

"N" DEG. OF FREEDOM → N-1 ORDER POLYNOMIAL

2N DEG. OF 2N-1 ORDER POLYNOMIAL

THERE ARE 'SPECIAL' POINTS $\{x_i\}$ CALLED GAUSS POINTS WHICH ARE ROOTS OF THE LEGENDRE POLYNOMIALS

10/16/2015

BY USING THESE POINTS WE CAN INTEGRATE EXACTLY A POLYNOMIAL OF ORDER $2N+1$, WHERE N = # OF SAMPLE POINTS

TRAPEZOIDAL RULE: INTEGRATE A LINEAR FUNCTION EXACTLY

$$I(b) = \frac{h}{2}(f_1 + f_2) - \frac{h^2}{12} f''(\xi) \quad \xi \in [x_1, x_2]$$

$$I(b) = \sum_{i=1}^{N=2} a_i f(x_i) = a_1 f(x_1) + a_2 f(x_2)$$

WE SHOULD BE ABLE TO INTEGRATE A CUBIC EXACTLY: $P_{2N-1}(x) = P_3(x)$

↳ x^3, x^2, x, CONSTANT

$$\left.\begin{aligned}
x^3&: \int_{-1}^{1} x^3 dx = 0 = a_1 x_1^3 + a_2 x_2^3 \\
x^2&: \int_{-1}^{1} x^2 dx = \frac{x^3}{3}\Big|_{-1}^{1} = \frac{2}{3} = a_1 x_1^2 + a_2 x_2^2 \\
x&: \int_{-1}^{1} x\,dx = 0 = a_1 x_1 + a_2 x_2 \\
1&: \int_{1}^{1} dx = 2 = a_1 + a_2
\end{aligned}\right\}$$

4 EQUATIONS, 4 UNKNOWNS

$a_1 = a_2 = 1$ (WEIGHTS)

$-x_1 = x_2 = \sqrt{1/3}$ (GAUSS POINTS)

→ MAP $[-1, 1]$ TO $[a, b]$

$$\int_a^b f(y)dy \quad \text{WHERE } y = \frac{a+b}{2} + \frac{b-a}{2}x, \quad dy = \frac{b-a}{2}dx$$

$$\int_{-1}^{1} f(y)\frac{b-a}{2}dx \simeq \frac{b-a}{2}\sum_{i=1}^{N} a_i f(y_i) \quad \text{WHERE } y_i = \frac{a+b}{2} + \frac{b-a}{2}x_i$$

↑ GAUSS POINTS

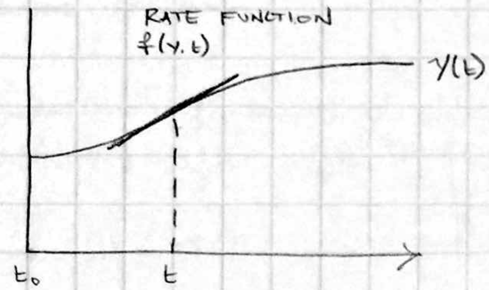
10/16/2015

* SOLUTION'S TO ODE'S

- ORDINARY: ONE IND. VARIABLE

"CLASSIC IVP"

$$\frac{d}{dt} y(t) = y'(t) = \underbrace{f(y,t)}_{\text{"RATE" FUNCTION}} \; ; \; y(t_0) = y_0 \;, \text{ WANT } y(t) \text{ GIVEN } f(y,t)$$



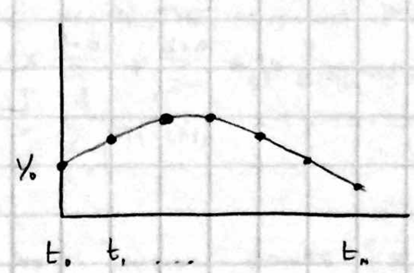
SOLUTION FAMILY IS AN INTRINSIC PROPERTY OF ODE

SIMPILEST CASE $f(y,t) = f(t)$

10/19/2015

- CLASSIC IVP: $\frac{dy(t)}{dt} = y' = f(y,t)$

FIND y(t)



10/19/2015

EULER'S METHOD:

– 3 DIFFERENT WAYS TO DERIVE

i.) SUBSTITUTE FINITE DIFFERENCE

$$y' = \left[\frac{y_{i+1} - y_i}{h} - f'(y,t)\right] = f(y,t) \qquad \text{WHERE} \quad y_i = y(t_i), \quad h \equiv t_{i+1} - t_i$$

$$y_{i+1} = y_i + hf_i + \underbrace{\frac{h^2}{2} f'(y(\xi_i), \xi_i)}_{\text{TRUNCATION}}$$

TRUNCATE & SWITCH TO $w_i \simeq y_i$

$$\boxed{w_{i+1} = w_i + hf_i} \quad \text{EULER'S METHOD}$$

ii.) INTEGRATE THE ODE

$$\int_{t_i}^{t_{i+1}} y' dt = \int_{t_i}^{t_{i+1}} f(y_i, \tau) d\tau \rightarrow y_{i+1} - y_i = \underbrace{\int_{t_i}^{t_{i+1}} f(y_i, \tau) d\tau}_{\substack{\text{TAYLOR EXPAND} \\ \text{\& EVALUATE @ } t_{i+1}}}$$

$$I(t_{i+1}) = \cancel{I(t_i)}^{0} + hf_i + \frac{h^2 f'(\xi)}{2}$$

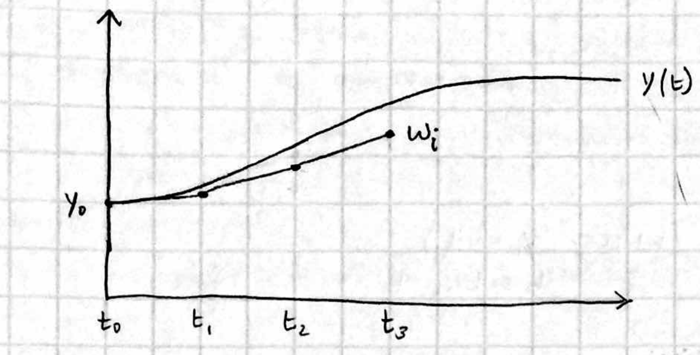
$$\boxed{y_{i+1} = y_i + hf_i + \frac{h^2 f'(\xi)}{2}}$$

iii.) TAYLOR EXPAND THE SOLUTION $y(t)$ AROUND t_i AND EVALUATE AT t_{i+1}

$$y(t) = y(t_i) + y'(t_i)(t - t_i) + \frac{y''(t_i)}{2!}(t - t_i)^2 + \dots$$

$$\boxed{y(t_{i+1}) = y(t_i) + hf_i + \frac{h^2}{2} f'(\xi)}$$

10/19/2015



$$w_{i+1} = w_i + h f_i$$

w_i & y_i RATE FUNCTIONS WILL ONLY BE EQUAL IF $y_i = f(t)$ ONLY

- LOCAL TRUNCATION ERROR IS $O(h^2)$, AS t $\underbrace{\text{ADVANCES} \rightarrow O(h)}_{\text{'GLOBAL' ERROR}}$

- ADDITIONAL ERROR COMPONENT DUE TO EVALUATION OF RATE FUNCTION AT WRONG PLACE

$$f(w_i, t_i) \quad \underline{\text{NOT}} \quad f(y_i, t_i)$$

DEFINE TOTAL ERROR:

$$\varepsilon_{i+1} = \left| y_{i+1} - w_{i+1} \right| \rightarrow y_{i+1} = y_i + h f_i + \frac{h^2}{2} f'(y(\xi_i), \xi_i)$$

EXACT: $f_i = f(y_i, t_i)$

COMP: $\bar{f}_i = f(w_i, t_i)$

$$y_{i+1} - w_{i+1} = y_i - w_i + h\left(f_i - \bar{f}_i\right) + \frac{h^2}{2} f'(y, \xi)$$

$$\varepsilon_{i+1} = \varepsilon_i \left[1 + \underbrace{\frac{h(f_i - \bar{f}_i)}{y_i - w_i}}_{} \right] + \frac{h^2}{2} f'(y, \xi_i)$$

ERROR COMPONENT FROM Y DEPENDENCE

$J_i = 0$ IF ONLY A FUNCTION OF t

$$J_i = \frac{\partial f}{\partial y}(\eta, t_i), \quad \eta_i \in [y_i, w_i] \rightarrow \text{JACOBIAN OF THE EQUATION}$$

10/19/2015

$$\underbrace{\varepsilon_{i+1} = \varepsilon_i\left(1 + h\overset{\text{DUE TO EVALUATION @ WRONG PLACE}}{J_i}\right)}_{\text{AMPLIFICATION FACTOR}} + \underbrace{\frac{h^2}{2} f'(y(\xi_i), \xi_i)}_{\text{TRUNCATION ERROR IN STEP } t_i \to t_{i+1} \text{ ASSUMING EXACT @ } t}$$

- ERROR PROPAGATES AS TIME ADVANCES
- WANT J_i SMALL AND NEGATIVE, CERTANLY NOT BIG.

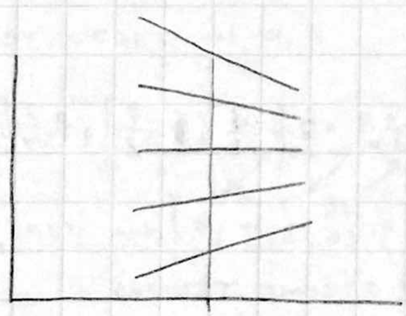
GE = GLOBAL ERROR

LTE = LOCAL TRUNCATION ERROR

$GE_{i+1} = (1 + hJ_i)\, GE_i + LTE_i$ → FOR EULER'S METHOD, WE NEED

$|1 + J_i h| < 1$ OTHERWISE GE BLOWS UP → $J_i < 0$

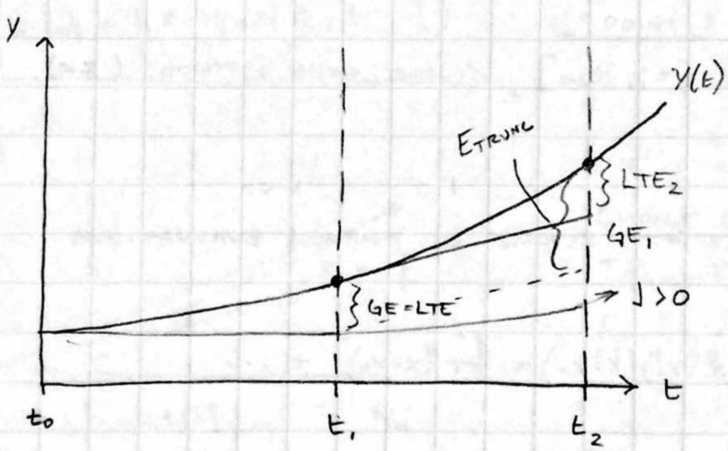
$\frac{\partial f}{\partial y} < 0$?



A STABLE ODE HAS $\frac{\partial f}{\partial y} < 0$

STABILITY CRITERION FOR EULER'S METHOD

$$\boxed{h < \frac{2}{|J|}}$$



FIRST STEP:

$$Y_1 = Y_0 + h f_0 + \underbrace{\frac{h^2}{2} f'(\xi)}_{LTE}$$

$$W_1 = W_0 + h f_0$$

2ND STEP:

$$Y_2 = Y_1 + h f(Y_1, t_1) + LTE_2$$

$$W_2 = W_1 + h f(W_1, t_1)$$

$$GE_{i+1} = (1 + hJ_1)\, GE + LTE_{i+1}$$

- E_{TRUNC} IS GROWING
- IF $J = 0$, $E_{SLOPE} = 0$

  $J < 0$, E_{SLOPE} CANCELS SOME OF E_{TRUNC}.

10/20/2015

EXAM:

- TESTED ON 'MATHEMATICAL' SIDE
- NO ODE'S

* EULER'S METHOD:
  - NOT VERY ACCURATE
  - STABILITY LIMITATIONS
  - SUSCEPTIBLE TO EXPONENTIAL ERROR GROWTH
  - WANT METHODS W/ HIGHER ORDER LTE

$$\underbrace{y_{i+1} = y_i + hf_i}_{} \quad \rightarrow \quad \underbrace{y_{i+1} = y_i + hf_i + \frac{h^2}{2} f_i'}_{}$$

EULER'S METHOD, $O(h^2)$ / STEP, $O(h)$ / GLOBAL

$O(h^3)$ / STEP, $O(h^2)$ / GLOBAL

→ BUT NEED $f_i' = \frac{d}{dt} f(y,t) = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial y}\frac{dy}{dt} = \frac{\partial f}{\partial t} + f\frac{\partial f}{\partial y}$

$$y_{i+1} = y_i + hf_i + \frac{h^2}{2} f_i' + \frac{h^3}{3!} f_i''$$

$$\hookrightarrow f_i'' = \frac{d}{dt}\left[f_i'\right] = \frac{\partial}{\partial t}\left[\frac{\partial f}{\partial t} + f\frac{\partial f}{\partial y}\right] + f\frac{\partial}{\partial y}\left[\frac{\partial f}{\partial t} + f\frac{\partial f}{\partial y}\right]$$

- HIGHERER ORDER TAYLOR METHOD'S QUICKLY BECOME TEDIOUS

- INSTEAD APPROXIMATE DERIVATIVES USING WEIGHTED FUNCTION EVALUATIONS OF f.

- RESTRICT EVALUATIONS TO $[t_i, t_{i+1}]$, SINGLE STEP METHODS (RK)

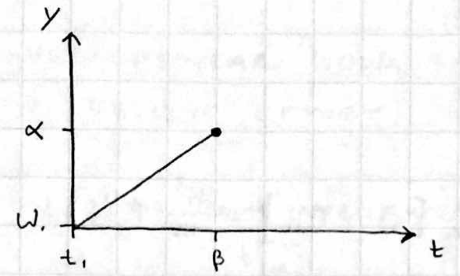
* RUNGE-KUTTA 2 (RK2)

$$w_{i+1} = w_i + hf_i + \frac{h^2}{2} f_i' \quad \leftarrow \text{REPLACE W/ FUNCTION EVALUATIONS}$$

- BEFORE: $f(x) = f(x_0) + f'(x_0)(x-x_0) + \frac{f''}{2!}(x-x_0)^2 + \ldots$

- NOW: $f(y,t) = ?$

10/20/2015



EXPAND $f(w,t)$ AROUND w_i, t_i
AND EVALUATE AT $w_i+\alpha$, $t_i+\beta$

$$f(w,t) = f(w_i,t_i) + \frac{\partial f_i}{\partial t}(t-t_i) + \frac{\partial f_i}{\partial y}(w-w_i) + \frac{\partial^2 f_i}{\partial t^2}\frac{(t-t_i)^2}{2} + \frac{\partial^2 f_i}{\partial y^2}\frac{(w-w_i)^2}{2}$$

$$+ \frac{\partial^2 f_i}{\partial t \partial y}(w-w_i)(t-t_i) \dots$$

→ EVALUATE @ α, β

$$f(w_i+\alpha, t_i+\beta) = f_i + \alpha\frac{\partial f_i}{\partial y} + \beta\frac{\partial f_i}{\partial t} + \frac{\beta^2}{2}\frac{\partial f_i}{\partial t^2} + \frac{\alpha^2}{2}\frac{\partial^2 f_i}{\partial y^2} + \alpha\beta\frac{\partial^2 f_i}{\partial t \partial y} + \dots$$

TO FIRST ORDER IN α, β

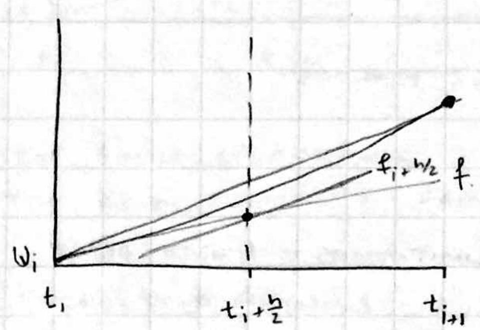
$$f(w_i+\alpha, t_i+\beta) = f_i + \beta\frac{\partial f_i}{\partial t} + \alpha\frac{\partial f_i}{\partial y}$$

$$w_{i+1} = w_i + h\left[f_i + \frac{h}{2}\frac{\partial f_i}{\partial t} + \frac{h}{2}\frac{\partial f_i}{\partial y}f_i\right]$$

IF WE LET $\beta = \frac{h}{2}$; $\alpha = \frac{h}{2}f_i$

$$\boxed{w_{i+1} = w_i + hf\left(w_i + \frac{h}{2}f_i, t_i + \frac{h}{2}\right)}$$

MIDPOINT RULE (RK2)



THE MIDPOINT RULE IS A PARTICULAR CHOICE OF A MORE GENERAL APPROX. TO

$$\left[f_i + \frac{h}{2}\left(\frac{\partial f}{\partial t} + \frac{\partial f}{\partial y}f\right)\right] = a_1 f_1 + a_2 f\left[w_i+\alpha, t_i+\beta\right]$$

→ MIDPOINT RULE:

$a_1 = 0$, $a_2 = 1$

$\alpha = \frac{h}{2}f_i$, $\beta = \frac{h}{2}$

↳ LEADS TO TRAPEZOIDAL RULE METHOD ; MODIFIED EULER'S METHOD.

10/20/2015

$$\int_{t_i}^{t_{i+1}} y'\,dt = \underbrace{\int_{t_i}^{t_{i+1}} f(y,\tau)\,d\tau}$$

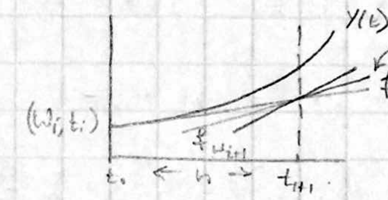
USE TRAPEZOIDAL RULE $\frac{h}{2}\left[f_i + f_{i+1}\right] - \frac{h^3}{12} f'(\xi)$

$$y_{i+1} - y_i = \frac{h}{2}\left[f_i + f_{i+1}\right] - \frac{h^3}{12} f'(\xi)$$

$$\boxed{w_{i+1} = w_i + \frac{h}{2}\left[f_i + f(w_{i+1}, t_{i+1})\right]}$$

10/21/2015

ODE SOLUTIONS



AVERAGE $\frac{f_i + f_{i+1}}{2}$

USE THIS SLOPE INSTEAD, OVER A FULL h STEP

FIND $y(t_i)$ GIVEN $y' = f(y,t)$; $y(t_0) = y_0$

$$\boxed{w_{i+1} = w_i + \frac{h}{2}\left[f_i + f(w_{i+1}, t_{i+1})\right]}$$ TRAPEZOIDAL RULE METHOD

AVERAGE OF SLOPE @ i & $i+1$

HOW TO GET w_{i+1} IN $f(w_{i+1}, t_{i+1})$

GENERAL APPROXIMATION TO:

$$\left[f_i + \frac{h}{2}\left(\frac{\partial f}{\partial t} + \frac{\partial f}{\partial y} f\right)\right] = a f_i + a_2 f(w_i + \alpha, t_i + \beta)$$

MODIFIED EULER'S METHOD

$$\boxed{w_{i+1} = w_i + \frac{h}{2}\left[f_i + f(w_i + hf_i, t_{i+1})\right]}$$

RK2 METHODS

HEUN'S METHOD

$$\boxed{w_{i+1} = w_i + \frac{h}{4}\left[f_i + 3f_{i+2/3}\right]}$$

4TH ORDER RUNGE-KUTTA BEGINS TO REACH A LIMITATIONS — HIGHER ORDER METHODS HAVE $\varepsilon_{TRUNC} \sim O(\varepsilon_{ROUND})$

10/21/2015

* RUNGE-KUTTA 4TH ORDER

- VERY POPULAR. GOOD TRADE-OFF BETWEEN TRUNC. ERROR REDUCTION & RELATIVE EFFORT.

- DERIVED FROM INTEGRATION OF ODE USING SIMPSON'S RULE

$$\int_{t_i}^{t_{i+1}} y_i \, d\tau = \int_{t_i}^{t_{i+1}} f(y,\tau)\, d\tau = \frac{h}{6}\left[f_i + 4f_{i+1/2} + f_{i+1}\right] + O(h^5)$$

RUNGE-KUTTA 4TH ORDER METHOD

$$\boxed{w_{i+1} = w_i + \frac{h}{6}\left[f_i + 4f_{i+1/2} + f_{i+1}\right]}$$

GENERAL FORMULA: WEIGHTED AVERAGE OF RATE FUNCTIONS EVALUATED AT t_i, $t_{i+1/2}$, t_{i+1}

- MANY POSSIBILITIES: MOST COMMON

$$w_{i+1} = w_i + \frac{h}{6}\left[f_1 + 2(f_2 + f_3) + f_4\right]$$

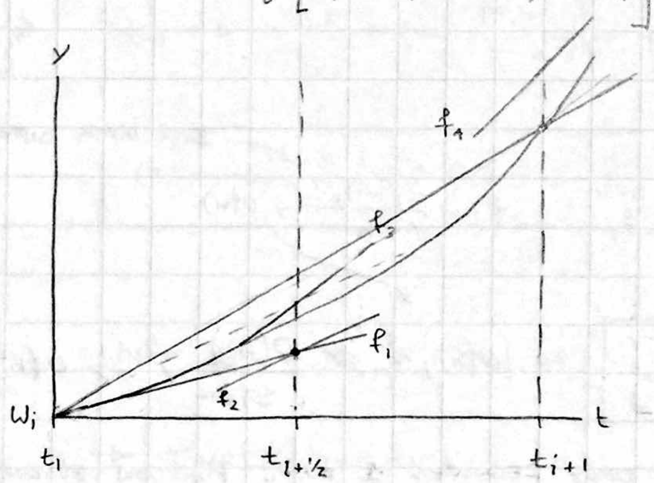
WHERE

$$f_1 = f_i$$

$$f_2 = f\left(w_i + \tfrac{h}{2} f_1,\, t_{i+1/2}\right)$$

$$f_3 = f\left(w_i + \tfrac{h}{2} f_2,\, t_{i+1/2}\right)$$

$$f_4 = f\left(w_i + h f_3,\, t_{i+1}\right)$$



$$f_5 = \frac{f_1 + 2f_2 + 2f_3 + f_4}{6}$$

SAFE & STABLE METHOD (FOR SINGLE STEP)

- FOR SINGLE STEP METHODS, WE DO NOT USE ANY INFORMATION ABOUT THE RATE FUNCTION @ PREVIOUS STEPS: t_{i-1}, t_{i-2}, ETC.

* MULTI-STEP METHODS:
  - USE PAST INFORMATION ABOUT RATE FUNCTION
  - MORE RISKY & LIKELY TO GO UNSTABLE

10/21/2015

INSTEAD USE f_i, f_{i-1}, f_{i-2} ETC. TO APPROX. f', f'', ETC.

IF WE USE f_{i+1} (EG. TRAPEZOIDAL METHOD) THEN WE HAVE TO FIGURE OUT w_{i+1} IN $f()$

THESE METHODS ARE <u>NOT</u> SELF STARTING → MUST USE A SINGLE STEP METHOD TO GET STARTED.

FOR N STEP METHOD, MUST TAKE N-1 STEPS W/ A SINGLE STEP METHOD

* <u>ADAMS METHODS</u>

- ADAMS - BASHFORTH METHODS: "OPEN" OR EXPLICIT METHODS DO NOT INCLUDE f_{i+1}

- ADAMS - MOULTON METHOD: "CLOSED" OR IMPLICIT METHODS <u>DO</u> INCLUDE f_{i+1}

$$y_{i+1} = y_i + hf_i + \frac{h^2}{2} f_i' + \frac{h^3}{3!} f_i'' + \dots$$

APPROXIMATE DERIVATIVES W/ $\underbrace{f_{i+1}, \overbrace{f_i, f_{i-1}, f_{i-2}}^{A\text{-}B}}_{A\text{-}M}$

- ADAMS - BASHWORTH 2-STEP

$$y_{i+1} = y_i + hf_i + \frac{h^2}{2} f_i' + \frac{h^3}{3!} f_i''$$

BACKWARDS DIFFERENCE

$$f_i' = \frac{f_i - f_{i-1}}{h} + O(h)$$

$$\boxed{w_{i+1} = w_i + h\left[\frac{3}{2} f_i - \frac{1}{2} f_{i-1}\right]}$$

$$[O(h)]\, h^2 \rightarrow \frac{O(h^3)}{\text{STEP}} \checkmark \rightarrow O(h^2) \text{ GLOBALY}$$

ONLY REQUIRES 1 RATE FUNCTION EVALUATION

$$w_{i+2} = w_{i+1} + h\left[\tfrac{3}{2} f(i+1) - \tfrac{1}{2} f(i)\right]$$

- A-B 3 STEP:

3 POINTS TO APPROXIMATE → $\left.\begin{matrix} f' \sim O(h^2) \\ f'' \sim O(h) \end{matrix}\right\}$ → EXPECT $\frac{O(h^4)}{\text{STEP}} \Rightarrow \frac{O(h^3)}{\text{GLOBAL}}$

$$\boxed{w_{i+1} = w_i + \frac{h}{12}\left[23 f_i - 16 f_{i-1} + 5 f_{i-2}\right]}$$

10/23/2015

$$y_{i+1} = y_i + hf_i + \frac{h^2}{2} f_i' + \frac{h^3}{3!} f'' + \ldots$$

- ADAM'S MOULTON METHODS INVOLVE f_{i+1}, SO WE CAN USE CENTERED DIFFERENCES TO APPROXIMATE THE DERIVATIVES

IE, f' TO $O(h^2)$ W/ ONLY TWO STEPS

- 2 STEP METHOD'S

t_{i-1} — t_i — t_{i+1}

WE CAN DO CENTER DIFF. APPROXIMATIONS TO f' & f''

$$\frac{h^2}{2} f_i' \rightarrow \frac{h^2}{2}\left[\frac{f_{i+1} - f_{i-1}}{2h} + O(h^2)\right]$$

$$\frac{h^3}{3!} f_i'' \rightarrow \frac{h^3}{6}\left[\frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} + O(h^2)\right]$$

EXPECT $O(h^4)$

$$w_{i+1} = w_i + hf_i + \frac{h^2}{2}\left[\frac{f_{i+1} - f_{i-1}}{2h}\right] + \frac{h^3}{6}\left[\frac{f_{i+1} - 2f_i + f_{i-1}}{h^2}\right]$$

ADAM'S MOULTON 2-STEP

$$\boxed{w_{i+1} = w_i + \frac{h}{12}\left[5f_{i+1} + 8f_i - f_{i-1}\right]}$$

$$\text{LTE} = \frac{-h^4}{24} f'''(y_i, \xi_i)$$
$$\xi \in [t_{i-1}, t_{i+1}]$$

WHAT ABOUT $f_{i+1} = f(w_{i+1}, t_{i+1})$?

→ TWO POSSIBILITIES:

1.) y APPEARS LINEARLY IN $f(y,t)$

I.E. $y' = -y + t + 1$ → THEN y_{i+1} IS NO PROBLEM

USING A A-M 2 STEP

$$w_{i+1} = w_i + \frac{h}{12}\left[5(-w_{i+1} + t_{i+1} + 1) + 8(-w_i + t_i + 1) + (-w_{i-1} + t_{i-1} + 1)\right]$$

SOLVE FOR w_{i+1}

10/23/2015

$$\left(1+\frac{5h}{12}\right)w_{i+1} = w_i + \frac{h}{12}\left[5\left(t_{i+1}+1\right) + 8f_i - f_{i-1}\right]$$

↳ EXPLICIT SOLUTION

2. Y APPEARS NON-LINEAR IN $f(y,t)$

I.E. $y' = \cos(y) + t$

$$w_{i+1} = w_i + \frac{h}{12}\left[5\left(\cos(w_{i+1}) + t_{i+1}\right) + 8f_i - f_{i-1}\right]$$

↳ NOT POSSIBLE TO COMBINE w_{i+1} TO GET EXPLICIT SOLUTION

INSTEAD WE MUST PREDICT w_{i+1} AND CORRECT W/ THE METHOD

WHICH LEADS TO THE "PREDICTOR / CORRECTOR" SCHEME

ACTUALLY DID THIS ALREADY W/ MODIFIED EULER METHOD.

$$w_{i+1} = w_i + \frac{h}{2}\left[f_i + f_{i+1}\right], \quad f_{i+1} = f(w_{i+1}, t_{i+1})$$

→ A-B/A-M 2 STEP P/C

ALL JUST WEIGHTED AVERAGES OF RATE FUNCTIONS

$$\hat{w}_{i+1} = w_i + \frac{h}{2}\left(3f_i - f_{i-1}\right) \quad \text{(PREDICTOR)} \rightarrow O(h^2) \text{ OVERALL}$$

$$w_{i+1} = w_i + \frac{h}{12}\left[5f(\hat{w}_{i+1}, t_{i+1}) + 8f_i - f_{i-1}\right]$$

USE A PREDICTOR NO MORE THAN ONE ORDER OF h LOWER

CAN CORRECT MORE THAN ONCE (ITERATE THE PROCESS)

ITERATION INDEX

$$w_{i+1}^{k+1} = g\left(w_{i+1}^{k}\right) \quad \text{w/ } w_{i+1}^{0} \text{ COMING FROM THE PREDICTOR}$$

$g(w_{i+1}^{k})$ IS THE RHS OF w_{i+1}^{k+1}: THE CORRECTOR

- IN MOST CASES, ITERATION IS BETTER SPENT ON SMALLER STEP SIZE.

10/23/2015

- IF WE DO ITERATE, THEN CONVERGENCE TO w_{i+1}

$$\left| g'(w_{i+1}) \right| < 1$$

EG. 2-STEP A-B/A-M P/C:

$$\left| g'(w_{i+1}) \right| = \left| \frac{\partial g}{\partial y} \right| = \left| \underbrace{\frac{5h}{12} \frac{\partial f}{\partial y}}_{J\, =\, \text{JACOBIAN}} \right|_{w_{i+1}} < 1$$

PROBLEMS CAN OCCUR WHEN J IS LARGE, REQUIRES SMALL h

* <u>KNOWN AS "STIFF" PROBLEMS.</u>

→ NEWTON'S METHOD:

- REWRITE $g(w_{i+1}^{k}) = w_{i+1}^{k+1}$ AS $F(w_{i+1}) = 0$

$$F(w_{i+1}) = w_{i+1} - \left\{ w_i + \frac{h}{12}\left[5f(w_{i+1}, t_{i+1}) + 8f_i - f_{i-1} \right] \right\} = 0$$

$F(z) = 0$ WHERE $z = w_{i+1}$

$$z^{k+1} = z^{k} - \frac{F(z^{k})}{F'(z^{k})}$$

$$w_{i+1}^{k+1} = w_{i+1}^{k} - \frac{w_{i+1}^{k} - \left\{ w_i + \frac{h}{12}\left[5f(w_{i+1}^{k}, t_{i+1}) + 8f_i - f_{i-1} \right] \right.}{1 - \frac{5h}{12} \frac{\partial f}{\partial y}\Big|_{w_{i+1}^{k}}}$$

↳ CONVERGES QUADRATICALLY TO w_{i+1}, BUT REQUIRES A GOOD GUESS

$$y' = -ky \;\rightarrow\; y = y_0 e^{-kt}, \quad \frac{\partial f}{\partial y} = -k$$

SYSTEM'S OF ODE'S

- m 1ST ORDER ODE'S IN m VARIABLES AND t

$$\frac{d\underline{z}}{dt} = \underline{f}(\underline{z}, t) \quad \text{WHERE } \underline{z} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix}$$

10/23/2015

$$f(\underline{z}, t) = \begin{bmatrix} f_1(z_1, z_2 \dots z_m, t) \\ f_2(z_1, z_2 \dots z_m, t) \\ \vdots \\ f_m(z_1, z_2, \dots z_n, t) \end{bmatrix}$$

M ELEMENT COLUMN VECT.

$$z_0 = \begin{bmatrix} z_1(0) \\ z_2(0) \\ \vdots \\ z_n(0) \end{bmatrix}$$

EXPLICIT METHOD'S ARE EASY

EG. FOR $i = 1:N$

$t_i = t_0 + i * h$

$w_i = w_{i-1} + h * f(w_{i-1}, t_{i-1})$

END.

FOR $j = 1:M$

$z_j(t_i) = z_j(t_{i-1}) + h * f_j(\underline{z}(t_{i-1}), t_{i-1})$

END

10/26/2015

* SYSTEMS OF ODE'S

- m 1^{st} ORDER ODE'S IN M VARIABLES & t

$\frac{d\underline{z}}{dt} = \underline{f}(\underline{z}, t)$; $\underline{z}_0$ $\rightarrow$ WANT $\underline{z}(t_i)$

→ EXPLICIT: EASY (LOOP)

→ IMPLICIT: PREDICT ALL THEN CORRECT ALL

HIGHER ORDER ODE'S:

$y'' = f(y', y, t)$ w/ $y'(0) = y_0'$; $y(0) = y_0$

↳ WRITE AS A SYS. OF FIRST ORDER ODE'S & SOLVE USING ANY METHOD.

10/26/2015

WRITE AS A SYSTEM OF FIRST ORDER ODES

$$z_1(t) = y(t) \qquad z_2(t) = y'(t)$$
$$z_1'(t) = y'(t) \qquad z_2'(t) = y''(t)$$
$$\vdots$$
$$z_m(t) = y^{(m+1)}(t) = z_{m-1}'(t) \rightarrow y^{(m)}(t) = z_m'(t) = f(z_1, z_2 \ldots z_{m-1}, t)$$

W/ I.C'S:

$$z_1(0) = y_0$$
$$z_2(0) = y_0'$$
$$\vdots$$
$$z_m(0) = y_0^{m+1}$$

→ IN VECTOR NOTATION:

$$\begin{bmatrix} z_1'(t) \\ z_2'(t) \\ \vdots \\ z_n'(t) \end{bmatrix} = \begin{bmatrix} z_2(t) \\ z_3(t) \\ \vdots \\ f(z_1, z_2, z_m, t) \end{bmatrix} \quad \text{W/ I.C. } z_0 = \begin{bmatrix} z_1(0) \\ z_2(0) \\ \vdots \\ z_m(0) \end{bmatrix}$$

EG. FIND $y(t_i)$

GIVEN $y'' + 4y' + 5y = 0$; $y(0) = 3$, $y'(0) = 5$

$$\left.\begin{array}{l} z_1(t) = y(t) \\ z_2(t) = y'(t) \end{array}\right\} \rightarrow \boxed{\begin{array}{l} z_1'(t) = z_2(t) \\ z_2'(t) = -4z_2(t) - 5z_1(t) \\ z_1(0) = 3; \; z_2(0) = 5 \end{array}}$$

← SOLVE USING EULER'S METHOD

TIME STEP

$$z_1^{i+1} = z_1^i + hf_1^i = z_1^i + hz_2^i$$

$$z_2^{i+1} = z_2^i + hf_2^i = z_2^i + h(-4z_2^i - 5z_1^i)$$

$$\boxed{\begin{array}{l} z_1^{i+1} = z_1^i + hz_2^i \\ z_2^{i+1} = z_2^i(1-4h) - 5hz_1^i \end{array}}$$

SOLUTION

10/26/2015

* STABILITY

- RECALL EULER'S METHOD:

$$\varepsilon_{i+1} = \underbrace{(1 + hJ_i)}\varepsilon_i + \frac{h^2}{2} f'(\xi)$$

STABILITY DEPENDS ON J_i (AMPLIFICATION FACTOR)

ERROR HAS BOTH A TRUNCATION COMPONENT & A PROPOGATING COMPONENT

- ASSUME ODE IS ANALYTICALLY STABLE (SECTION 5.1) → $J \leq 0$

- 3 IDEAS ABOUT SOLUTIONS

i. DOES THE DIFFERENCE EQUATION APPROACH THE ODE AS $h \to 0$? I.E. $\lim_{h \to 0} LTE = 0$ (CONSISTENCY)

   - FOR ODE METHODS STUDIED, ALL ARE BASED ON TAYLOR EXPANSIONS, SO THIS IS ALWAYS THE CASE (h IN NUMERATOR) OF LOCAL TRUNCATION ERROR.

ii. DO ERRORS ONCE INTRODUCED REMAIN BOUNDED OR GROW EXPONENTIALLY? I.E. $|\lambda| \leq 1$? (STABILITY)

iii. DOES THE NUMERICAL SOLUTION APPROACH THE EXACT SOLUTION AS $h \to 0$? (CONVERGENCE)
   - THIS IS REALLY WHAT WE WANT TO KNOW
   - HARD TO PROVE THEORETICALLY.

EQUIVALENCE THEORUM: A NECCESSARY & SUFFICIENT CONDITION FOR CONVERGENCE IS CONSISTENCY & STABILITY.

10/26/2015

- NOTE THAT STABILITY DOES NOT GUARENTEE ACCURACY.

- LOOK AT STABILITY OF VARIOUS METHODS
  - NOTE THAT STABILITY DEPENDS ON
    - STEP SIZE
    - SOLUTION FAMILY (J)
    - NUMERICAL METHOD (LTE)

BACKWARD EULER METHOD:

$w_{i+1} = w_i + h f_{i+1}$, COMES FROM EXPANDING $I(t) = \int_t^{t_{i+1}} f\, d\tau$ AROUND t_{i+1}

$\varepsilon_{i+1} = (1 - hJ_{i+1})^{-1} \varepsilon_i + O(h^2)$ (IMPLICIT)

- GIVEN AN ANALYTICALLY STABLE ODE ($J \leq 0$), THE AMPLIFICATION FACTOR IS ≤ 1 FOR ANY $h > 0$

  ↳ UNCONDITIONALLY STABLE METHOD (GREAT FOR STIFF PROBLEM)

10/27/2015

TRAPEZOIDAL RULE METHOD:

- CAME FROM

$$\int_{t_i}^{t_{i+1}} f\, d\tau \approx \frac{h}{2}\left[f_i + f_{i+1} \right] + O(h^3)$$

- CAN SHOW:

$$\varepsilon_{i+1} = \left(\frac{1 + \frac{h}{2} J_i}{1 - \frac{h}{2} J_{i+1}} \right) \varepsilon_i + O(h^3)$$

IF $J \leq 0 \rightarrow$ ANALYTICALLY STABLE ODE

THEN THE AMPLIFICATION FACTOR IS ≤ 1 FOR ANY $h > 0$

↳ UNCONDITIONALLY STABLE

10/27/2015

ORDER (LTE) IS BETTER THAN BACKWARD EULER

BUT FOR STIFF PROBLEMS, AMPLIFICATION $\to 1$

- FOR OTHER METHODS IT GETS MORE DIFFICULT TO EXPRESS ε_{i+1} IN TERMS OF J.

- INSTEAD, CHOOSE A PROTOTYPE PROBLEM THAT CAN BE ANALYZED AND EXTRAPOLATE THE FINDINGS TO THE GENERAL CASE

"PROTOTYPE"

$$y' = -ky \ , \quad y(0) = y_0 \quad \to \quad \text{SOLUTION: } y(t) = y_0 e^{-kt}$$

$$\frac{\partial f}{\partial y} = -k = J$$

- TRAPEZOIDAL RULE

$$w_{i+1} = w_i + \frac{h}{2}\left[f_i + \underbrace{f_{i+1}}_{\text{EULER'S METHOD: } f_{i+1} = f(w_i + hf_i,\, t_{i+1})} \right]$$

$$f_i = f(w_i, t_i) = -kw_i$$

$$f_{i+1} = f(w_{i+1}, t_{i+1}) = f(w_i + hf_i, t_{i+1}) = -k(w_i + hf_i) = -k(w_i + h(-kw_i))$$

$$w_{i+1} = w_i + \frac{h}{2}\left[-kw_i + -k(w_i + h(-kw_i)) \right] = w_i - \frac{hk}{2}w_i - \frac{hk}{2}w_i + \frac{h^2k^2}{2}w_i$$

$$w_{i+1} = w_i \underbrace{\left[1 - hk + \frac{(hk)^2}{2} \right]}$$

NEED THIS TO BE <1 SINCE $w_{i+1} \to 0$ AS $t_{i+1} \to \infty$

A.) TURNS OUT TO BE THE AMPLIFICATION FACTOR
  ↳ I.E. THIS ERROR PROPOGATES (BEHAVES) THE SAME WAY AS THE SOLUTION

B.) THIS TURNS OUT TO BE THE FIRST 3 TERMS IN THE T-EXPANSION OF THE EXACT SOLUTION TO THE PROTOTYPE PROBLEM. (RK 4 PRODUCES THE 1ST 5 TERMS)

10/27/2015

$$y_{i+1} = y_0 e^{-kt_{i+1}} = y_0 e^{-k(t_i+h)} = y e^{-kt_i} e^{-kh}$$

$$y_i = y_0 e^{-kt_i}$$

$$\frac{y_{i+1}}{y_i} = e^{-kh} = \underbrace{\underbrace{1 - kh + \frac{(kh)^2}{2}}_{RK2} - \frac{(kh)^3}{3!} + \frac{(kh)^4}{4!}}_{RK4} + \dots$$

$$y_{i+1} = y_i + hf_i + \frac{h^2}{2} f_i' + \frac{h^3}{3} f''(\xi)$$

$$= y_i + hf_i + \frac{h^2}{2}\left[\frac{\partial f_i}{\partial t} + \frac{\partial f_i}{\partial y} f_i\right] + \frac{h^3}{3!} f''(\xi)$$

→ PROTOTYPE PROBLEM ←

$$\frac{\partial f}{\partial t} = 0 \; ; \quad \frac{\partial f_i}{\partial y} = -k \; ; \quad f_i = -ky$$

$$y_{i+1} = y_i - hky_i + \frac{(hk)^2}{2} y_i + \frac{h^3}{3} f'''(\xi)$$

$$w_{i+1} = w_i - hkw_i + \frac{(hk)^2}{2} w_i \quad \rightarrow \quad \varepsilon_{i+1} = \varepsilon_i - hk\varepsilon_i + \frac{(hk)^2}{2}\varepsilon_i + LTE$$

STABILITY OF MULTI-STEP METHODS

- APPLY TO PROTOTYPE PROBLEM → DIFFERENCE EQUATION W/ CONST. COEFF.

EG. 4-STEP A-B

$$w_{i+1} = w_i + \frac{h}{24}\left[55 f_i - 59 f_{i-1} + 37 f_{i-2} - 9 f_{i-3}\right]$$

$f_i = -kw_i$, $f_{i-1} = -kw_{i-1}$ → GROUP TERMS:

$$w_{i+1} = -\left(1 - \frac{55hk}{24}\right) w_i - \frac{59hk}{24} w_{i-1} - \frac{37hk}{24} w_{i-2} + \frac{9hk}{24} w_{i-3} = 0$$

SINCE ε_i BEHAVES LIKE w_i, $w_i \sim \lambda^i$

$$\lambda^4 - \left(-\frac{55hk}{24}\right)\lambda^3 - \frac{39hk}{24}\lambda^2 - \frac{37hk}{24}\lambda + \frac{9hk}{24} = 0$$

↳ NEED ALL 4 ROOTS ≤ 1 FOR STABILITY

10/27/2015

FOR QUADRATICS,

$$A\lambda^2 + B\lambda + C = 0 \quad \text{w/ } A > 0$$

$$|\lambda| < 1 \quad \text{IFF} \quad \frac{C}{A} < 1 \quad \& \quad |B| < A + C$$

10/30/2015

BOUNDARY VALUE PROBLEMS

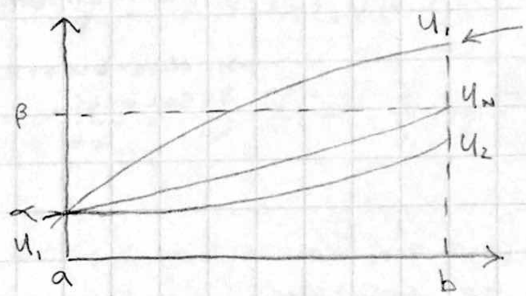
$y'' = f(x, y, y')$ → WANT $y(x_i)$ FOR $a \leq x_i \leq b$

- GIVEN B.C.'S: $\left.\begin{array}{l} y(a) = \alpha \\ y(b) = \beta \end{array}\right\}$ COULD SOLVE IF WE HAD $y'(a) = \gamma$

$\left.\begin{array}{l} z_1(x) = y \\ z_2(x) = y' \end{array}\right\}$ → $\begin{array}{l} z_1' = z_2 \\ z_2' = f(x, z_1, z_2) \end{array}$ W/ B.C'S $\begin{array}{l} z_1(a) = \alpha \; , \; z_1(b) = \beta \\ z_2(a) = ? \end{array}$

SHOOTING METHOD:

- GUESS THE BOUNDARY CONDITION: $z_2(a) = u$
- SOLVE FOR $z_1(x_i)$, $z_2(x_2)$ ON $a \leq x_i \leq b$
- SEE IF $z_1(b) = \beta$, IF IT IS, THEN WE'RE DONE. IF NOT, THEN GUESS AGAIN



HOW TO UPDATE GUESS OF $u = z_2(a) = y'(a)$?

- SIMPLE CASE

$f(x, y, y')$ IS LINEAR IN y, y' THEN WE ONLY NEED TWO GUESSES, WE CAN ALWAYS CONSTRUCT A LINEAR SOLUTION FROM A LINEAR COMBINATION OF LINEAR SOLUTIONS

$$z_1(x_1) = C_1 z_1(x_i, u_0) + C_2 z_1(x_i, u_1)$$

→ FIND $C_1 + C_2$ AT B.C'S.

$x = a$: $z_1(a) = \alpha = C_1 z_1(a, u_0) + C_2 z_1(a, u_1)$

$x = b$: $z_1(b) = \beta = C_1 z_1(\beta, u_0) + C_2 z_2(\beta, u_1)$

10/30/2015

$$C_1 = \frac{\beta - z_1(b,u_1)}{z_1(b,u_0) - z_1(b,u_1)}$$

AND

$$C_2 = \frac{z_1(b,u_0) - \beta}{z_1(b,u_0) - z_1(b,u_0)}$$

$$\sum F_y = m\vec{a}_y = -m\vec{g}$$

$$m\frac{d^2y}{dt^2} = -\cancel{m}g$$

LINEAR DIFF EQ

$$y'' = -g$$

NON-LINEAR SOLUTION

$$\frac{d^2y}{dt^2} = g \rightarrow \frac{dy}{dt} = gt + c_1$$

$$\boxed{y = gt^2 + c_1 t + c_2}$$

$$\sum F_y = F_D - F_g = m\vec{a}$$

$$F_D = \tfrac{1}{2}\rho A v^2$$

$$F_g = m\vec{g}$$

$$\tfrac{1}{2}\rho A \vec{v}^2 - m\vec{g} = m\vec{a}$$

$$\tfrac{1}{2}\rho A \left(\frac{dy}{dt}\right)^2 - m\vec{g} = m\frac{d^2y}{dt^2}$$

$$y'' = \tfrac{1}{2}\rho A y'^2 - mg$$

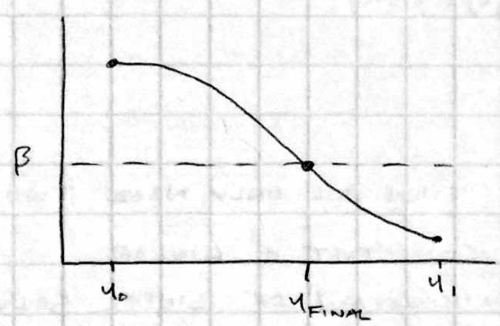
↳ NON-LINEAR

↳ NON-LINEAR SOLUTION

NOW LOOK AT $f(x,y,y')$ NON-LINEAR

WRITE $z_1 = y$ AS $z_1(x,u)$

LOOK AT $z_1(b,u)$; END POINT OF SOLUTION



THINK OF AS A ROOT FINDING PROBLEM

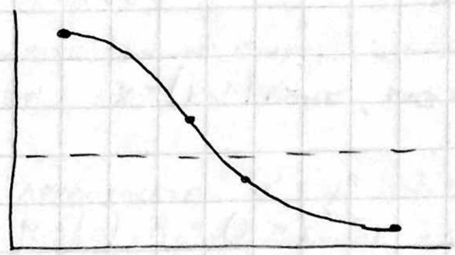
$$z_1(b,u) - \beta = 0$$

$$F(u) = 0$$

→ USE ANY OF THE ITERATIVE METHODS THAT WE KNOW TO FIND u_{FINAL}

10/30/2015

$u_1 - u_2 \nearrow$
$u_2 : u_3$



SECANT METHOD:

$$u_{N+1} = u_N - F(u_N)\left[\frac{u_N - u_{N-1}}{F(u_N) - F(u_{N-1})}\right]$$

- NEED TWO GUESSES, u_0, u_1
- CONVERGENCE IS SLOW, $\alpha = \frac{1+\sqrt{5}}{2}$
- FUNCTION EVALUATIONS COSTLY

NEWTON'S METHOD:

$$u_{N+1} = u_N - \frac{F(u_N)}{F'(u_N)} \quad \frac{\partial}{\partial u}, \qquad u_{N+1} = u_N - \frac{z_1(b, u_N) - \beta}{\frac{\partial z_1(b, u_N)}{\partial u}}$$

$\leftarrow$ NEED THIS TERM

$\rightarrow$ DIFFERENTIATE ODE:

$$y'' = f(x, \underbrace{y(x,u)}_{z_1}, \underbrace{y'(x,u)}_{z_2}) \rightarrow \frac{\partial}{\partial u}$$

$$\frac{\partial y''}{\partial u} = \frac{\partial f}{\partial x}\underbrace{\cancel{\frac{dx}{du}}}_{=0} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial u} + \frac{\partial f}{\partial y'}\frac{\partial y'}{\partial u}$$

$\rightarrow$ SWITCH ORDER OF DIFFERENTIATION

$$\left(\frac{\partial y}{\partial u}\right)'' = \frac{\partial f}{\partial y}\frac{\partial y}{\partial u} + \frac{\partial f}{\partial y'}\left(\frac{\partial y}{\partial u}\right)'$$

WHERE PRIME IS WITH RESPECT TO x

RECALL,

$$z_1(x, u) = y, \quad \text{SO,} \quad \frac{\partial y}{\partial u} = \frac{\partial z_1(x,u)}{\partial u}$$

$\leftarrow$ WANT AT $x = b$

$$g = g(x,u) \equiv \frac{\partial z_1(x,u)}{\partial u}, \quad \text{WE WANT} \quad g(b,u) = \frac{\partial z_1(b,u)}{\partial u}$$

$$g'' = \frac{\partial f}{\partial y} g + \frac{\partial f}{\partial y'} g'$$

10/30/2015

TO SOLVE FOR $g(b,u) = \dfrac{\partial z_1(b,u)}{\partial u}$ WE NEED $g(a)$; g' $\Rightarrow$ IVP

SO $g(a,u) = \dfrac{\partial z_1(a,u)}{\partial u} = 0$; $z_1(a,u) = \alpha$

$$g'(a,u) = \frac{\partial z_1'(a,u)}{\partial u} = 1 \; ; \quad z_1'(a,u) = z_2(a) = u$$

$$\left.\begin{aligned} V_1 &= g \\ V_2 &= g' \end{aligned}\right\} \Rightarrow \begin{aligned} V_1' &= V_2 \\ V_2' &= \frac{\partial f}{\partial y} V_1 + \frac{\partial f}{\partial y} V_2 \end{aligned} \qquad \begin{aligned} V_1(a) &= 0 \\ V_2(a) &= 1 \end{aligned}$$

- SOLVE SYSTEM FOR $V_1(x_i)$, $V_2(x_i)$
  - ↳ WANT $V_1(b) = g(b,u) = \dfrac{\partial z_1(b,u)}{\partial u}$

- PICK u_0
- SOLVE $y'' = f(x,y,y')$ w/ $y(a) = \alpha$, $y'(a) = u_0$
- SOLVE $g'' = \dfrac{\partial f}{\partial y} g + \dfrac{\partial f}{\partial y'} g'$ w/ $g(a) = 0$, $g'(a) = 0$
- REPEAT UNTIL $|z_1(b) - \beta| < \epsilon$

$$u_{N+1} = u_1 - \frac{z_1(b,u_N) - \beta}{V_1(b,u_N)}$$

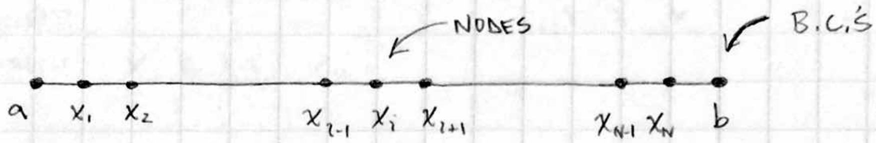
11/2/2015

* FINITE DIFFERENCE METHODS FOR BVP'S

IDEA: WRITE DOWN FINITE DIFFERENCE "APPROXIMATION" OF ODE AT SOME POINT x_i, WHERE $a \le x_i \le b$

→ APPROXIMATE y' & y'' W/ FINITE DIFFERENCES

→ MAKE APPROXIMATION AT AN INTERVAL OF NODES



LINEAR CASE:

$$y'' = f(x, y, y') = p(x)\,y'(x) + q(x)\,y(x) + r(x)$$

LINEAR

- WHERE $p(x), q(x), r(x)$ ARE <u>KNOWN</u>

- APPROXIMATE AT NODE i:

$$y_i'' = \frac{w_{i+1} - 2w_i + w_{i-1}}{h^2} \quad \left(\text{SECOND ORDER C.D. } O(h^2)\right)$$

$$\underbrace{\frac{w_{i+1} - 2w_i + w_{i-1}}{h^2}}_{y_i''} - \underbrace{p(x)\left[\frac{w_{i+1} - w_{i-1}}{2h}\right]}_{p(x)\,y_i'} - \underbrace{q(x)\,w_i}_{q(x)\,y_i} = r(x_i)$$

- GROUP TERMS:

$$\left[1 + \frac{h}{2}p(x_i)\right]w_{i-1} - \left[2 + h^2 q(x_i)\right]w_i + \left[1 - \frac{h}{2}p(x_i)\right]w_{i+1} = h^2 r(x_i)$$

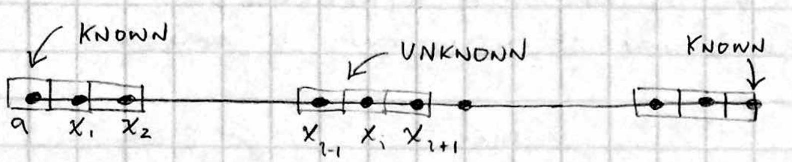
↳ ONE EQUATION, 3 UNKNOWNS → WRITE AS A FINITE DIFFERENCE MOLECULE

11/2/2015

- MOLECULE:

$$\boxed{1+\frac{h}{2}p(x)} - \boxed{-2-h^2 q(x_i)} - \boxed{1-\frac{h}{2}p(x_i)}$$

$$\quad i-1 \qquad\qquad i \qquad\qquad i+1$$



→ COUPLED SYSTEM OF N LINEAR EQUATIONS IN N UNKNOWNS

→ WRITE IN MATRIX FORM:

$$\begin{bmatrix} -(2+h^2q_1) & (1-\frac{h}{2}p_1) & 0 & 0 & 0 & \cdots & 0 \\ (1+\frac{h}{2}p_2) & -(2+h^2q_2) & (1+\frac{h}{2}p_2) & 0 & 0 & \cdots & 0 \\ 0 & (1+\frac{h}{2}p_3) & -(2+h^2q_3) & (1+\frac{h}{2}p_3) & 0 & \cdots & 0 \\ 0 & & & & & & \\ \vdots & & & & & & \\ 0 & & & & & (1+\frac{h}{2}p_N) & -2(+h^2q_N) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{N-1} \\ w_N \end{bmatrix} = h^2 \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_{N-1} \\ r_N \end{bmatrix} - \begin{bmatrix} 1+\frac{h}{2}p_1\,\alpha \\ 0 \\ \vdots \\ 0 \\ (1-\frac{h}{2}p_N)\,\beta \end{bmatrix}$$

↳ TRIDIAGONAL N×N SPARSE MATRIX

SOLVE $y = w = A^{-1}b \rightarrow O(n^3)$

$w = A\backslash b \rightarrow O(n)$

↳ NON-LINEAR CASE

11/2/2015

NON-LINEAR CASE:

- WRITE ODE AS $-y'' + f(x, y, y') = 0$

- APPROXIMATE:

$$-\left(\frac{w_{i+1} - 2w_i + w_{i-1}}{h^2}\right) + f\left(x_i, w_i, \frac{w_{i+1} - w_{i-1}}{2h}\right) = 0$$

FOR EACH $x_i \in [x_1, x_N]$

↳ SYSTEM OF NON-LINEAR EQUATIONS IN N UNKNOWNS

$$-2w_i - w_2 + h^2 f\left(x_1, w_1, \frac{w_2 - \alpha}{2h}\right) - \alpha = 0 \quad ; \quad F_1(x_1, \underline{w}) \approx 0$$

$$-w_1 + 2w_2 - w_3 + h^2 f\left(x_2, w_2, \frac{w_3 - w_1}{2h}\right) = 0 \quad ; \quad F_2(x_2, \underline{w}) \approx 0$$

$$\vdots \qquad \vdots \qquad \vdots \qquad \vdots$$

$$-w_{N-1} + 2w_N + h^2 f\left(x_N, w_N, \frac{\beta - w_{N-1}}{2h}\right) - \beta = 0 \quad ; \quad F_N(x_N, \underline{w}) = 0$$

↳ USE NEWTON'S METHOD FOR SYSTEMS OF EQUATIONS

- NEED $\underline{w_0}$
- AND $\underline{\underline{J}}(\underline{x}, \underline{w})$

* PARTIAL DIFFERENTIAL EQUATIONS:

- MORE THAN ONE VARIABLE (INDEPENDENT)

EQUATIONS:

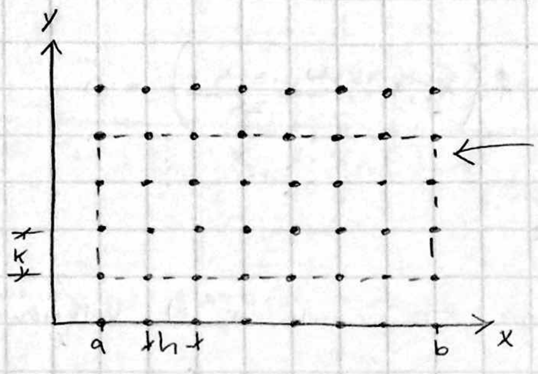
- $\nabla^2 u = f(x,y)$ ← SOURCE TERM ; POISSON'S EQUATION
- $\nabla^2 u = 0$: LAPLACE'S EQUATION

↳ IN 2D CARTESIAN $\nabla^2 = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}\right)$

11|2|2015

- AS W/ ODE'S, APPROXIMATE DERIVATIVES W/ FINITE DIFFERENCE'S

USE i, j INDEXING



- SECOND CENTER DIFFERENCE

$$\frac{\partial^2 u}{\partial x^2} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} - \frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}(\xi, y_j)$$

$$\frac{\partial^2 u}{\partial y^2} = \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{(k)^2} - \frac{k^2}{12}\frac{\partial^4 u}{\partial y^4}(\xi, \eta), \quad \eta \in [y_{j-1}, y_{j+1}]$$

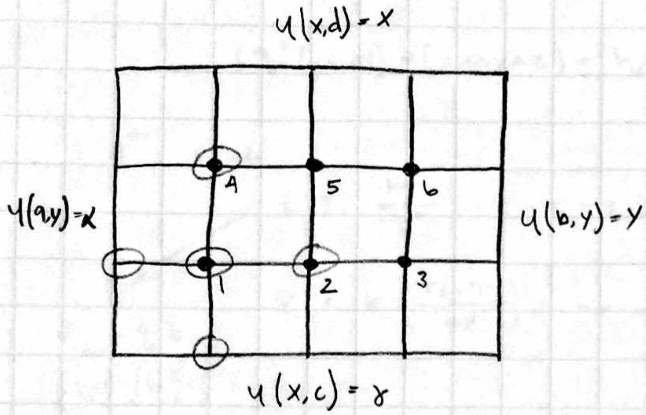
$$u_{i+1,j} + u_{i-1,j} + \beta(u_{i,j+1} + u_{i,j-1}) - (2+2\beta)u_{i,j} = 0 \quad \text{(LAPLACE EQ.)}$$

$$\beta = \frac{h^2}{k^2}$$

$$\frac{\partial^2 u}{\partial x^2} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} \quad ; \quad \frac{\partial^2 u}{\partial y^2} = \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{k^2}$$

$$\beta = \frac{h^2}{k^2}$$

$$u_{i+1,j} + u_{i-1,j} + \beta(u_{i,j+1}\ u_{i,j-1}) - 2(2+2\beta)u_{i,j} = 0 = h^2 f(x_i, y_i)$$



$$\begin{bmatrix} -4 & 1 & 0 & 1 & 0 & 0 \\ 1 & -4 & 1 & 0 & 1 & 0 \\ 0 & 1 & -4 & 0 & 0 & 1 \\ 1 & 0 & 0 & -4 & 1 & 0 \\ 0 & 1 & 0 & 1 & -4 & 1 \\ 0 & 0 & 1 & 0 & 1 & -4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} = \begin{bmatrix} -\alpha - \gamma \\ -\gamma \\ -\gamma - (c+k) \\ -\alpha - (a+h) \\ -(a+2h) \\ -(a+3h) - (c+2k \end{bmatrix}$$

TYPES OF PDE'S:

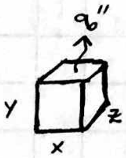
ELIPTIC: LAPLACE, POISSONS

PROPOGATING: PARABOLIC, HYPERBOLIC

- DIFFUSSION EQUATION

$$\frac{\partial u}{\partial t} = D \frac{\partial^2 u}{\partial x^2}$$ BVP w/ INITIAL CONDITIONS

$$\frac{D \Delta t}{h^2} \leq 1/2$$ STABILITY CRITERIA



$q'' A \cancel{m} t = \cancel{m} h_f \rightarrow q'' A t = h_f$ $\qquad \left\{ \frac{W}{kg} m^2 \right\} = \{$

$t q'' (\Delta x \Delta z) \rho (\Delta x \Delta y \Delta z) = h_f \rho (\Delta x \Delta y \Delta z) \rightarrow q'' \underbrace{\Delta x \Delta z}_{=1} t = h_f$

1^{ST} CELL:

$t = \frac{h_f}{q''}$ TIME TO FREEZE CELL

$Q_{out} = -k \left(\frac{T_2 - T_1}{\Delta y} \right) t \rightarrow T_2 = \frac{q'' \Delta y}{-kt} + T_1$ KNOWN (FREEZING TEMP)

$Q_{out} = q'' t$

$\{kJ\} = \{W\}\{t\}$

+1 2^{ND} CELL

$T_3 = \frac{q'' \Delta y}{-k} + T_2$, $T_2 =$

$Q_{COND} = \dot{q} t = h_f \rightarrow Q = \dot{q} t = -k \left(\frac{T_2 - T_1}{\Delta y} \right) t \rightarrow \underline{T_2 = \text{KNOWN}}$

$Q_{COND} = -k \left(\frac{T_2 - T_1}{\Delta y} \right)$, $t_{FREEZE} = \frac{h_f}{q_{COND}}$

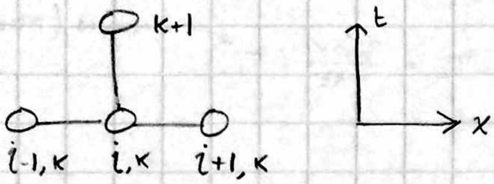
$Q_{COND} = \frac{-k(T_2 - T_3)}{\Delta y} \rightarrow T_3 = \text{KNOWN}$

11/6/2015

* PDE'S

ELLIPTIC : $\nabla^2 u = 0$

PARABOLIC: $\frac{\partial u}{\partial t} = D \frac{\partial^2 u}{\partial x^2}$ → FORWARD DIFFERENCE METHOD



$$u_i^{K+1} = u_i^K + \underbrace{\frac{D\Delta t}{h^2}}_{\lambda}\left(u_{i+1}^K - 2u_i^K + u_{i-1}^K\right)$$ → EXPLICIT SOLUTION FOR I.V.P

$\underline{u}^{K+1} = \underline{\underline{A}}\, u^K + b$ WHERE,

$$A = \begin{bmatrix} 1-2\lambda & \lambda & 0 & 0 & 0 & \cdots \\ \lambda & 1-2\lambda & \lambda & 0 & 0 & \cdots \\ 0 & \lambda & 1-2\lambda & 0 & & \cdots \\ \vdots & & & \ddots & \lambda & \\ 0 & 0 & 0 & \lambda & 1-2\lambda & \end{bmatrix}$$

↳ NOT AN IMPLICIT SOLUTION

DIFFERENT THAN LAPLACE EQ. SOLUTION

$$b = \begin{bmatrix} \lambda\alpha \\ 0 \\ 0 \\ 0 \\ \vdots \\ \lambda\beta \end{bmatrix}$$

STABILITY LIMIT:

$$\boxed{\lambda \equiv \frac{D\Delta t}{h^2} \leq \frac{1}{2}}$$

COMES FROM THE REQUIREMENT THAT THE SPECTRAL RADIUS OF $\underline{A}$ BE ≤ 1, OR NORM OF THE LARGEST EIGENVALUE OF $\underline{\underline{A}}$, $|\lambda| \leq 1$

11/6/2015

HOW ABOUT A BACKWARD DIFFERENCE IN TIME

↳ FROM BACKWARD EULER INTEGRATION

$$w_{i+1} = w_i + hf_{i+1}$$

- RESULTS IN A FLIPPED TEMPLATE:

$$y_i^{k+1} - y_i^{k} = \frac{D\Delta t}{h^2}\left(y_{i+1}^{k+1} - 2y_i^{k+1} + y_{i-1}^{k+1}\right)$$

$$\left.\begin{array}{ccccc} -\lambda & - & 1+2\lambda & - & \lambda \\ & & | & & \\ & & \lambda & & \end{array}\right\} \text{IMPLICIT SOLUTION}$$

$$\underline{\underline{A}}\,\underline{y}^{k+1} = -y^{k} + b$$

$\underline{\underline{A}}$ ← N x N TRIDIAGONAL MATRIX

- SOLUTION IS "TIED" TOGETHER IN SPACIAL COORDINATES
- UNCONDITIONALLY STABLE METHOD, BUT STILL ONLY $O(\Delta t + h^2)$
- COMPUTATIONALLY EXPENSIVE ⟶ MUST SOLVE A LINEAR SYS. @ EACH TIME STEP.

TO MAKE TIME STEP MORE ACCURATE, TRY CENTER DIFF.

$$\frac{y_i^{k+1} - y_i^{k-1}}{2\Delta t} = \frac{D}{h^2}\left[y_{i+1}^{k} - 2y_i^{k} + y_{i-1}^{k}\right]$$

$$\begin{array}{ccccc|c} & & 1 & & & k+1 \\ & & | & & & \\ -2\lambda & - & 4\lambda & - & 2\lambda & k \\ & & | & & & \\ & & -1 & & & k-1 \\ \hline i-1 & & i & & i+1 & \end{array}$$

⟶ UNCONDITIONALLY UNSTABLE

- CALLED THE LEAPFROG SCHEME

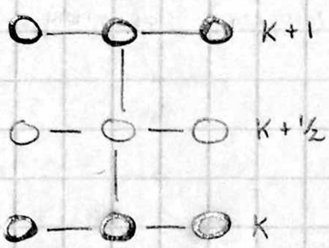
11/6/2015

INSTEAD APPROXIMATE PDE AT x_i, $t_{k+1/2}$: $(\text{PDE})^{k+1/2}$

$$\frac{u_i^{k+1} - u_i^{k}}{\Delta t} = D\,\frac{\left(u_{i+1}^{k+1/2} - 2u_i^{k+1/2} + u_{i+1}^{k+1/2}\right)}{h^2}$$

HOW TO GET $u^{k+1/2}$? $\rightarrow$ AVERAGE $\Rightarrow$ $u^{k+1/2} = \dfrac{u^{k} + u^{k+1}}{2}$

$$u_i^{k+1} - u_i^{k} = \frac{D\Delta t}{2h^2}\left\{\left[u_{i+1}^{k} - 2u_i^{k} + u_{i-1}^{k}\right] + \left[u_{i+1}^{k+1} - 2u_i^{k+1} + u_{i-1}^{k+1}\right]\right\}$$



CRANK - NICHOLSON SCHEME

$\rightarrow$ $Au^{k+1} = Bu^{k} + b$

$\hookrightarrow$ MATRIX SOLUTION @ EACH TIME STEP, BUT UNCOND. STABLE W/ $O\left((\Delta t)^2 + h^2\right)$

* BOUNDARY CONDITIONS

- TYPE 1, OR 'DIRCHLET' B.C'S. VALUE IS GIVEN $u(x) = \alpha$

- TYPE 2, OR 'NEUMANN' BC'S. DERIVATIVE IS GIVEN $\dfrac{\partial u(x)}{\partial x} = \alpha$

- TYPE 3, MIXED OR 'CAUCHEY'. LINEAR COMBO OF TYPE 1, 2'S

$$a u(x) + b\,\frac{\partial u}{\partial x} = \beta$$

- ELLIPTIC EQ'S
  - NEED ONE B.C (I, II, III) EVERYWHERE ON CLOSED BOUNDARY

    EG. $\nabla^2 = 0$

- PARABOLIC EQ
  - NEED ONE I.C, TWO B.C'S (I, II, III)

    EG. $\dfrac{\partial u}{\partial t} = D\nabla^2 u$

11/9/2015

* LINEAR ALGEBRA

CHAPTER 6, DIRECT SOLUTIONS TO LINEAR ALGEBRAIC SYSTEMS

LOOKING FOR DIRECT SOLUTIONS OF ALGEBRAIC SYSTEMS, EXACT ANSWERS (NOT ACCOUNTING FOR ROUND-OFF ERROR) IN A FINITE # OF STEPS

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2,n} \\ \vdots & & & \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

$$\underline{\underline{A}} \quad \underline{x} = \underline{b}$$

SOLUTION STRATEGY: MAKE AN UPPER TRIANGULAR MATRIX THEN SOLVE USING BACKSUBSTITUTION TO SOLVE FOR x.

$$\begin{bmatrix} \tilde{a}_{11} & \tilde{a}_{12} & \cdots & \tilde{a}_{1n} \\ 0 & \tilde{a}_{22} & \cdots & \tilde{a}_{2n} \\ 0 & 0 & \tilde{a}_{33} \; \cdot & \tilde{a}_{3n} \\ \vdots & \vdots & & \\ 0 & 0 & & \tilde{a}_{n,n} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} \tilde{b}_1 \\ \tilde{b}_2 \\ \tilde{b}_3 \\ \\ \tilde{b}_n \end{bmatrix}$$

EQUIVALENT SYSTEM OF EQUATIONS

$$x_n = \frac{\tilde{b}_n}{\tilde{a}_n}$$

JUST SOLVED FOR!

$$x_{n-1} = \frac{\left(\tilde{b}_{n-1} - \tilde{a}_{n-1,n}\, x_n\right)}{\tilde{a}_{n-1,n-1}}$$

$$x_i = \left(b_i - \sum_{j=i+1}^{n} \tilde{a}_{ij}\, x_j\right) \Big/ a_{ii}$$ FOR EACH $i = n-1, n-2 \ldots$

11/9/2015

GAUSSIAN ELIMINATION W/ BACK SUBSTITUTION

STEP 1: ELIMINATION

STEP 2: BACK SUBSTITUTION

ON COMPUTERS:

$T_{MULT} \sim T_{DIV} \gg T_{ADD}, T_{SUB}$

$1 + 2 + 3 + 4 + \dots n$

$(n+1)(n+1) \rightarrow \frac{n}{2}(n+1) \rightarrow O(n^2)$

- DEFINE $R_j \equiv$ ROW j (E_j)

$$R_j - \frac{a_{ji}}{a_{ii}} R_i \rightarrow R_j$$

"PIVOT ELEMENT"

FOR EACH $i = 1,2,3 \dots N-1$ (EACH COLUMN)

$j = i+1, i+2 \dots n$ (EACH ROW BELOW MASTER ROW)

j = ROW

i = COLUMN

↳ UPPER TRIANGULAR MATRIX (ASSUMING ALL PIVOTS $a_{ii} \neq 0$)

$i = 1$: COLUMN 1

ROW, COL

$j=2$: $R_2 - \frac{a_{21}}{a_{11}} R_1 \rightarrow R_2$: $a_{21} = 0$

$j=3$: $R_3 - \frac{a_{31}}{a_{11}} R_1 \rightarrow R_3$: $a_{31} = 0$

$\underline{\underline{A}} \rightarrow \underline{\underline{\tilde{A}}}$ WHERE $\underline{\underline{\tilde{A}}}$ IS THE UPPER TRIANGULAR MATRIX.

EXAMPLE

$$2x_1 - x_2 + x_3 = -1$$

$$3x_1 - 3x_2 + 9x_3 = 0$$

$$3x_1 + 3x_2 + 5x_3 = 4$$

$$\begin{bmatrix} 2 & -1 & 1 \\ 3 & 3 & 9 \\ 3 & 3 & 5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \\ 4 \end{bmatrix}$$

$$\underline{\underline{A}} \quad \underline{x} = \underline{b}$$

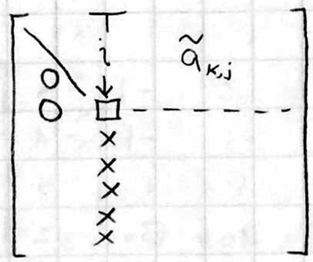
↳ FORM AUGMENTED MATRIX

11/9/2015

$$\left[\begin{array}{ccc|c} 2 & -1 & 1 & -1 \\ 3 & 3 & 9 & 0 \\ 3 & 3 & 5 & 4 \end{array}\right]$$

$\underline{A}$ $\underline{b}$

ESTIMATE # OF OPERATIONS FOR GAUSSIAN ELIMINATION



$$R_j = \frac{a_{ji}}{a_{ii}} R_i \to R_j \qquad \text{FOR } j = i+1 \cdots N$$

NEED: $(n-i+1)$ MULTIPLIES PER m_j FOR $n-i$

$$\sum_{i=1}^{n-1} (n-i)(n-i+2) = \frac{2n^3+3n^2-5n}{6} \to O(n^3)$$

* PIVOTING STRATEGIES IN GAUSSIAN ELIMINATION

- IF THE PIVOT ELEMENT IS ZERO, THE ALGORITHM FAILS

  $m_{ji} = \infty$

- THE 'FIX' IS TO INTERCHANGE "MASTER" ROW W/ ANOTHER THAT DOESN'T HAVE $a_{ij} = 0$

$$\left[\begin{array}{cccc|c} 1 & -1 & 2 & -1 & -8 \\ 2 & -2 & 3 & -3 & -20 \\ 1 & 1 & 1 & 0 & -2 \\ 1 & 3 & 4 & 7 & -4 \end{array}\right] \xrightarrow{\text{AFTER } 1^{ST} \text{ COL.}} \left[\begin{array}{cccc|c} 1 & -1 & 2 & -1 & -8 \\ 0 & 0 & -1 & -1 & -4 \\ 0 & 2 & -1 & 1 & 6 \\ 0 & 4 & 2 & 8 & 12 \end{array}\right] \updownarrow \text{SWAP}$$

ALWAYS PICK A ROW BELOW

IN FINITE PRECISION COMPUTING, UNLIKELY TO HIT 0 EXACTLY, SO WE WATCH FOR SMALL NUMBERS

$$R_j - \frac{a_{ij}}{a_{ii}} R_i \to R_j$$

$\uparrow$ SMALL

HOW TO DECIDE ON WHICH ROW TO EXCHANGE?

11/9/2015

* PRACTICAL PIVOTING — TAKE MAXIMUM

— COEFFICIENT IN COLUMN AS THE NEW POINT

$$\left[\begin{array}{cccc|c} 1 & -1 & 2 & -1 & -8 \\ 0 & \boxed{0} & -1 & -1 & -4 \\ 0 & 2 & -1 & 1 & 6 \\ 0 & \boxed{4} & 2 & 8 & 12 \end{array}\right]$$

ENSURES ALL $|m_{ji}| < 1$ → NO ERROR AMPLIFICATION

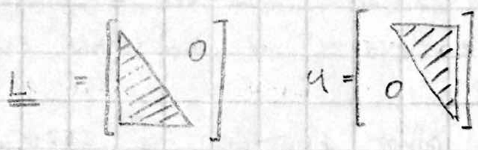
$O(n^2)$

11/11/2015

* MATRIX FACTORIZATION

$N \times N$ → $\underline{\underline{A}}\,\underline{x} = \underline{b}$

$A = \underline{\underline{L}} \cdot \underline{\underline{U}}$

$\underline{\underline{L}} =$ 

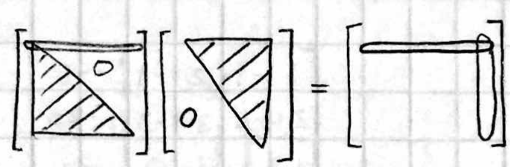
$\underline{\underline{L}}\,\underbrace{\underline{\underline{U}}\,x}_{\underline{y}} = \underline{b}$

L U DECOMPOSITION

$\underline{y}$ → $\underline{\underline{L}}\,\underline{y} = \underline{b}$ F.S. $O(n^2)$

$\underline{\underline{U}}\,\underline{x} = \underline{y}$ B.S $O(n^2)$

$$\sum_{k=1}^{\min(i,j)} l_{i,k}\, u_{k,j} = a_{ij}$$



$\underline{\underline{L}}$ $\underline{U}$ $\underline{\underline{A}}$

11/11/2015

* CROUT LU DECOMPOSITION

$$\begin{bmatrix} \times & & & & \\ \times & \ddots & & \emptyset & \\ \times & & \ddots & & \\ \times & & & \ddots & \\ \times & & & & \ddots \end{bmatrix} \begin{bmatrix} 1 & & & & \\ 0 & 1 & & & \\ 0 & & 1 & & \\ 0 & \emptyset & & 1 & \\ 0 & & & & 1 \end{bmatrix} = \begin{bmatrix} \times & & & & \\ \times & & & & \\ \times & & & & \\ \times & & & & \\ \times & & & & \end{bmatrix}$$

START W/ FIRST COLUMN OF $\underline{\underline{L}}$, EVERY ROW OF $\underline{\underline{L}}$ BY COLUMN 1 OF $\underline{\underline{U}}$ $\rightarrow$ $l_{i1} = a_{i1}$

THEN FIRST ROW OF $\underline{\underline{U}}$ * FIRST ROW OF $\underline{\underline{L}}$ TIMES COLUMNS THROUGH N OF $\underline{\underline{U}}$

$$u_{1j} = \frac{a_{1j}}{l_{11}}$$

SECOND COLUMN OF $\underline{\underline{L}}$:
- ROWS 2-N OF $\underline{\underline{L}}$ TIMES COLUMN 2 OF $\underline{\underline{U}}$
- $l_{i2} = a_{i2} - l_{i1} u_{12}$

SECOND ROW OF $\underline{\underline{U}}$
- ROW 2 OF $\underline{\underline{L}}$ BY COLUMNS 3-N OF $\underline{\underline{U}}$

$$\rightarrow u_{2j} = (a_{2j} - l_{21} u_{1j}) / l_{22}$$

$\underline{\underline{L}}_{\text{DOOLITTLE}} = \underline{\underline{U}}^T_{\text{CROUT}}$, $\underline{\underline{U}}_{\text{DOOLITTLE}} = \underline{\underline{L}}^T_{\text{CROUT}}$

* COMPUTING AN INVERSE OF A MATRIX: A^{-1}

TO $\underline{A}\underline{x} = \underline{b}$, $\underline{x} = \underline{A}^{-1}\underline{b}$ $\rightarrow$ DO NOT DO THIS IF ONLY WANT SOLUTION

IN PRACTICE, A^{-1} IS DETERMINED BY SOLVING $\underline{A}\underline{x} = \underline{b}$ FOR MULTIPLE b's, WHERE b's ARE COLUMNS OF $\underline{\underline{I}}$

11|11|2015

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 1 & 0 \\ -1 & 1 & 2 \end{bmatrix} ; \quad \text{SOLVE} \quad \underline{\underline{A}}\underline{x} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \underline{\underline{A}}\underline{y} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \underline{\underline{A}}\underline{z} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\underline{\underline{A}}^{-1} = \left[\begin{bmatrix} \underline{x} \end{bmatrix} \begin{bmatrix} \underline{y} \end{bmatrix} \begin{bmatrix} \underline{z} \end{bmatrix} \right] \rightarrow \underline{\underline{A}}\underline{y} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} ; \quad \underline{y} = A^{-1} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} x \\ x \\ x \end{bmatrix}$$

BETTER TO USE LU DECOMPOSITION, ONLY DECOMPOSE <u>ONCE</u>, $O(n^3)$

* COMPUTING DETERMINATE OF A MATRIX

$$\text{DET}(\underline{\underline{A}}) = \text{DET}(\underline{\underline{L}}\,\underline{\underline{U}}) = \text{DET}(\underline{\underline{L}}) \cdot \text{DET}(\underline{\underline{U}})$$

FOR UPPER & LOWER TRIANGULAR MATRICES THE DET IS PRODUCT OF ELEMENTS ALONG THE DIAGONAL

$$\text{DET}(\underline{\underline{A}}) = \text{DET}(\underline{\underline{L}}) \cdot \text{DET}(\underline{\underline{U}}) = \prod_{i=1}^{N} l_{ii} \prod_{j=1}^{N} u_{jj}$$

$$= \prod_{i=1}^{N} l_{i} \quad \text{CROUT}$$

$$= \prod_{i=1}^{N} u_{ij} \quad \text{DOOLITTLE}$$

$$= \prod_{i=1}^{N} u_{ij}^{2} \quad \text{CHOLESKI}$$

* BANDED MATRICES

N.B. = "HALF BANDWIDTH, EXCLUDES MAIN DIAGONAL

$(NB+1) \times (NB+1)$

$$\begin{bmatrix} x & x & x & 0 & 0 & 0 & 0 & \cdots \\ x & x & x & x & 0 & 0 & 0 & \cdots \\ x & x & x & x & x & 0 & 0 & \cdots \\ x & x & x & x & x & x & 0 & \\ x & x & x & x & x & x & x & \cdots \\ 0 & x & x & x & x & x & x & \\ 0 & 0 & & & & & & \end{bmatrix}$$

MOST SITUATIONS LEAD TO BANDED MATRICES

- CUBIC SPLINE
- BVP, ETC.

11/11/2015

DIAGONALLY DOMINANT

$$|a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^{N} |a_{ij}| \longrightarrow \text{NO ZERO PIVOTS}$$

* STORAGE FOR BANDED MATRICES

- NO SENSE IN STORING ALL THE ZERO'S
- INSTEAD PACK INTO A $N \times (2NB+1)$ MATRIX

FOR $NB \ll N$

$$\begin{array}{c}
\overbrace{}^{NB} \\
\begin{bmatrix}
O & O & O & X & X & X & X \\
O & O & X & X & X & X & X \\
O & X & X & X & & & \\
X & X & X & X & & & \\
 & & & X & & &
\end{bmatrix}
\end{array}
\quad N \text{ ROWS}, \quad 2NB+1 \text{ COLUMNS}$$

* TRIDIAGONAL MATRIX - NB-1

$\hookrightarrow$ LOOK AT VIDEO

1.) GET ALL NON-ZERO OFF DIAGONAL TERMS IN $\underline{\underline{L}}$
   - MULTIPLY ROW i IN $\underline{\underline{L}}$ BY COLUMN $i-1$ IN $\underline{\underline{U}}$

   $\rightarrow l_{i,\cdot} \rightarrow$ VIDEO

- IN A BANDED MATRIX, THE # OF NON-ZERO BANDS OFF (AND EXCLUDING) THE MAIN DIAGONAL IS CALLED THE <u>HALF BANDWIDTH</u>

11/13/2015

* ITERATIVE MATRIX SOLUTIONS

- VERY USEFUL FOR SPARCE MATRICES
- IDEA: TO SOLVE $Ax = b$
  - GUESS SOLUTION $\underline{x}^0$
  - UPDATE: $x^{k+1} = f(x^k)$
  - CONTINUE UNTIL $x^k \rightarrow x$
- ISSUES:
  - HOW TO UPDATE?
  - DOES PROCESS CONVERGE?
  - STOP CRITERIA
- TWO CLASSIC METHODS:
  - JACOBIAN ITERATION
  - GAUSS - SEIDEL ITERATION

* JACOBIAN ITERATION:

- SOLVE EQ. (OR ROW) i FOR x_i AND USE PREVIOUS GUESS OF SOLUTION TO COMPUTE UPDATED SOLUTION

$$x^{k+1} = \left(b_1 - \left[a_{12} x_2^k + a_{13} x_3^k + \ldots a_{1N} x_N^k\right]\right) / a_{11}$$

$$x^{N+1} = \left(b_2 - \left[a_{21} x_1^k + a_{23} x_3^k + \ldots a_{2N} x_N^k\right]\right) / a_{22}$$

JACOBI ITERATION →

$$\boxed{x_i^{k+1} = \left(b_i - \sum_{\substack{j=1 \\ j \neq i}}^{N} a_{ij} x_j^k\right) / a_{ii}}$$

11/13/2015

* GAUSS SEIDEL ITERATION

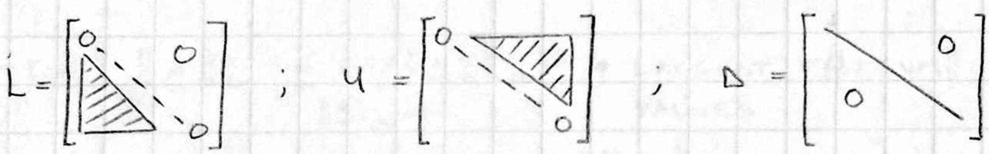
$$x_3^{k+1} = \left(b_3 - \left[a_{31} x_1^{k+1} + a_{32} x_2^{k+1} + a_{34} x_4^{k+1} + \dots a_{3N} x_N^{k}\right]\right) / a_{33}$$

GAUSS-SEIDEL

$$\boxed{x_i^{k+1} = \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{k+1} - \sum_{j=i+1}^{N} a_{ij} x_j^{k}\right) / a_{ii}}$$

* ANALYSIS OF ITERATIVE SCHEMES

DEFINE: $\underline{\underline{A}} = \underline{\underline{L}} + \underline{\underline{U}} + \underline{\underline{D}}$



- IN MATRIX FORM:

- JACOBI: $\underline{\underline{D}}\, \underline{x}^{k+1} = b - \left(\underline{\underline{L}} + \underline{\underline{U}}\right) x^k$

$$\rightarrow x^{k+1} = \underbrace{-\underline{\underline{D}}^{-1}\left(\underline{\underline{L}} + \underline{\underline{U}}\right)}_{\underline{\underline{T}}_J} x^k + \underbrace{\underline{\underline{D}}^{-1} b}_{\underline{C}_3}$$

$\hookrightarrow$ "ITERATION MATRIX FOR JACOBI"

$$\underline{x}^{k+1} = \underline{\underline{T}}_J\, x^k + C_3$$

- GAUSS-SEIDEL

$$\underline{\underline{D}}\, \underline{x}^{k+1} = \left(b - \underline{\underline{L}}\, \underline{x}^{k+1} - \underline{\underline{U}}\, x^k\right) \rightarrow \left(\underline{\underline{D}} + \underline{\underline{L}}\right) \underline{x}^{k+1} = \left(\underline{b} - \underline{\underline{U}}\, x^k\right)$$

$$x^{k+1} = \underbrace{-\left(\underline{\underline{D}} + \underline{\underline{L}}\right)^{-1} \underline{\underline{U}}}_{\underline{\underline{T}}_G} \underline{x}^k + \underbrace{\left(\underline{\underline{D}} + \underline{\underline{L}}\right)^{-1} \underline{b}}_{\underline{C}_g} \rightarrow x^{k+1} = \underline{\underline{T}}_{GS}\, x^k + C_{GS}$$

11/13/2015

BOTH METHODS HAVE FORM

$$\underline{x}^{k+1} = \underline{\underline{T}}\, x^{k} + \underline{c}$$

$$- \quad \underline{x} = \underline{\underline{T}}\,\underline{x} + c \qquad \text{(EXACT SOLUTION)}$$

$$\underline{x}^{k+1} - \underline{x} = \underline{\underline{T}}(\underline{x}^{k} - \underline{x}) \rightarrow \underline{e}^{k+1} = \underline{\underline{T}}\,\underline{e}^{k}$$

↳ WE WANT $\underline{e}^{k+1} = 0$

$\uparrow$ VECTOR $\quad \uparrow$ SCALAR

- NEED THE CONCEPT OF A "NORM"
  - A MEASURE OF THE SIZE OF $\underline{e}$ & $\underline{\underline{T}}$

* NORM $\| \quad \|$

- VECTOR NORMS $\|x\|$
- PROPERTIES
  - $\|x\| \geq 0$
  - $\|x\| = 0$, IFF $x = 0$, $i = 1, \dots N$
  - $\|\alpha x\| = |\alpha| \|x\|$
  - $\|\underline{x} + \underline{y}\| \leq \|\underline{x}\| + \|\underline{y}\|$

- MOST USEFUL L NORM:

$$L_2 \text{ NORM} = \|x\|_2 = \left[\sum_{j=1}^{N} x_j^2 \right]^{1/2}$$

$$L_\infty \text{ NORM} = \|\underline{x}\|_\infty = \max_{1 \leq j \leq N} |x_j|$$

IN GENERAL

$$\|\underline{x_p}\| \equiv \left[\sum_{j=1}^{N} |x_j|^p \right]^{1/p} \qquad \text{(P-NORM)}$$

11 | 13 | 2015

- RELATIVE NORM:

$$\frac{\|\underline{x}^{k+1} - \underline{x}^{k}\|}{\|\underline{x}^{k+1}\|}$$

- MATRIX NORM $\|A\| \equiv \max \|\underline{\underline{A}}\, \underline{x}\|$

  - TAKE ALL THE VECTORS $\underline{x}$ W NORM OF 1, MULTIPLY BY $\underline{\underline{A}}$ TO GET NEW VECTORS $\underline{\underline{A}}\underline{x}$ THEN PICK LARGEST IN TERMS OF THE NORM

  - EG. $\|\underline{\underline{A}}\|_2 = \max\limits_{\|x\|_2 = 1} \|\underline{\underline{A}}\underline{x}\|_2$
    - $\underline{x}$ IS INPUT VECTOR
    - FIND LONGEST $\underline{\underline{A}}\underline{x}$

  - EG $\|\underline{\underline{A}}\|_\infty = \max\limits_{\|\underline{x}\|_\infty = 1} \|\underline{\underline{A}}\underline{x}\|_\infty$
    - LARGEST ROW SUM OF ABSOLUTE VALUES

    $$= \max_{1 \le j \le N} \sum_{j=1}^{N} |a_{ij}|$$

WE HAVE,

$$e^{k+1} = \underline{\underline{T}}\, e^{k} \rightarrow \|\underline{e}^{k+1}\| = \|\underline{\underline{T}}\, \underline{e}^{k}\| \le \|\underline{\underline{T}}\| \|\underline{e}^{k}\|$$

FOR ITERATION TO CONVERGE, $\|T\| < 1$

- IT TURNS OUT THAT THE MATRIX NORM IS NOT VERY PRACTICAL

- INSTEAD USE THE "SPECTRAL RADIUS"

FOR ITERATION TO CONVERGE

$\rho(\underline{\underline{T}}) < 1$, $\rho(\underline{\underline{T}})$ IS THE SPECTRAL RADIUS OF $\underline{\underline{T}}$

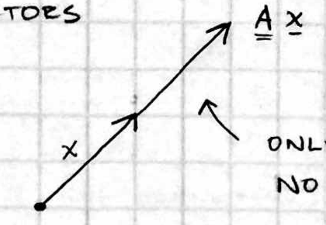
↳ = THE LARGEST EIGENVALUE OF $\underline{\underline{T}}$

$\Rightarrow \max |\lambda_i| < 1$, $\{\lambda_i\}$ = EIGENVALUES OF T

11/13/2015

* EIGENVALUES / EIGENVECTORS

$$\underline{\underline{A}}\underline{x} = \lambda \underline{x}$$



ONLY A SCALAR TRANSFORMATION, NO ROTATION

"PROOF"

$$\begin{aligned} \underline{x}^{k+1} &= \underline{\underline{T}}\,\underline{x}^{k} + c \\ - \quad \underline{x}^{k} &= \underline{\underline{T}}\,\underline{x}^{k-1} + c \end{aligned} \quad \rightarrow \quad \underline{x}^{k+1} - \underline{x}^{k} = \underline{\underline{T}}\left(\underline{x}^{k} - \underline{x}^{k-1}\right)$$

$$\underline{\delta}^{k+1} = \underline{\underline{T}}\,\underline{\delta}^{k}$$

WRITE AS A LINEAR COMBONATION OF N LINEARLY INDEPT. EIGENVECTORS OF $\underline{\underline{T}}$

$$\underline{\underline{T}}\,\underline{v}_i = \lambda_i \underline{v}_i$$

$$\underline{\delta}^{1} = \sum_{i=1}^{N} b_i \underline{v}_i$$

$$\underline{\delta}^{2} = \underline{\underline{T}}\,\underline{\delta}^{1} = \underline{\underline{T}} \sum_{i=1}^{N} b_i \underline{v}_i = \sum_{i=1}^{N} b_i \underline{\underline{T}}\,\underline{v}_i$$

$$= \sum_{i=1}^{N} b_i \lambda_i \underline{v}_i$$

$$\underline{\delta}_3 = \underline{\underline{T}}\,\delta^2 = \sum_{i=1}^{N} b_i \lambda_i^2 v_i \quad \rightarrow \quad \delta^N = \underline{\underline{T}}\,\delta^{N-1} = \ldots \sum_{i=1}^{N} b_i \lambda_i^{N-1} \underline{v}_i$$

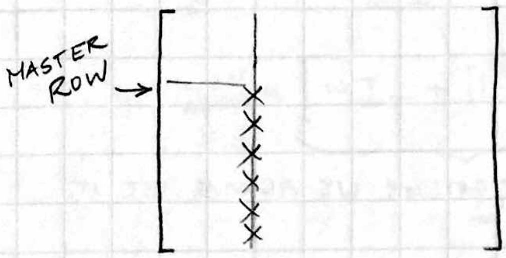
AS $n \rightarrow \infty$, FOR $\underline{\delta}^n \rightarrow 0$, IF ANY $|\lambda_i| > 1$, THEN $\underline{\delta}^N \rightarrow \infty$

$$|\lambda_i| \approx \frac{\|\delta^{k+1}\|}{\|\delta^{k}\|}$$

11/10/2015

*** PIVOTING STRATEGIES**

- PARTIAL PIVOTING, ORDER N^2 STEPS



$$R_j = \frac{a_{ji}}{a_{ii}} R_i \rightarrow R_j$$

$$\left[\begin{array}{cccc|c} 1 & -1 & 2 & -1 & -8 \\ 0 & 0 & -1 & -1 & -4 \\ 0 & 2 & -1 & 1 & 6 \\ 0 & 4 & 2 & 8 & 12 \end{array}\right]$$

- BETTER TO FIND THE PIVOT BASED ON ITS RELATIVE SIZE BASED ON OTHER ELEMENTS IN THE ROW

→ CALLED "SCALED" OR "COLUMN" PIVOTING.

- DETERMINE PIVOT ELEMENT BASED ON ITS SIZE RELATIVE TO OTHER COEFFICIENTS IN THE ROW.

- IN PRACTICE ITS TOO EXPENSIVE TO COMPUTE THE SCALE FACTORS FOR EVERY COLUMN

- INSTEAD, BASE SCALING ON SIZE RELATIVE TO ORIGINAL MATRIX COEFFICIENTS → GOES $O(n^2)$

→ LOOK AT ORIGINAL MATRIX, EQUAL TO MAX ABS. IN EACH ROW.

$$S_i = |a_{ij}| \rightarrow \underline{S}_i = \begin{bmatrix} 2 \\ 3 \\ 1 \\ 7 \end{bmatrix}$$ COMPUTE ONCE & USE AT EACH STEP.

11/10/2015

$$
\rightarrow \left[\begin{array}{cccc|c} 1 & -1 & 2 & -1 & -8 \\ 0 & \boxed{0} & -1 & -1 & -4 \\ 0 & 2 & -1 & 1 & 6 \\ 0 & 4 & 2 & 8 & 12 \end{array}\right] \begin{array}{l} \\ \leftarrow \text{MASTER ROW} \\ 1/2 \\ 7/4 \end{array} \qquad \underline{S}_i = \begin{bmatrix} 2 \\ 3 \\ 1 \\ 7 \end{bmatrix}
$$

CAN'T PROCEDE DUE TO ZERO → CALCULATE SCALE FACTORS

- IN PRACTICE, IF WE ARE USING A PIVOTING STRATEGY, WE ALWAYS USE IT.

23:09 IN VIDEO

11/16/2015 ITERATIVE MATRIX SOLUTIONS

$$\underline{x}^{k+1} = \underline{\underline{T}}\,\underline{x}^k + \underline{c}, \quad \rightarrow \quad |\lambda_i| \simeq \frac{\|\delta^{k+1}\|}{\|\delta^k\|}$$

BOTH JACOBI & GAUSS SEIDEL, GAURENTEED TO CONVERGE FOR $\underline{x}^0$

- IN GENERAL, $\rho(\underline{\underline{T}}) \rightarrow 1$ AS RANK (SIZE) OF $\underline{\underline{T}}$ GETS LARGE

- CONVERGENCE SLOWS $\rho(\underline{\underline{T}}) \rightarrow 1$

- STRATEGY: INTRODUCE A FACTOR, WHICH WE CONTROL, TO ALTER SPECTRAL RADIUS

  ↳ LEADS TO RELAXATION METHODS

IN GENERAL, $\tilde{\underline{x}}^{k+1} = \underline{\underline{T}}\,\underline{x}^k + \underline{c}$

$$\underline{x}^{k+1} = w\,\tilde{\underline{x}}^{k+1} + (1-w)\,\underline{x}^k$$

$0 < w < 1$ → "UNDER RELAXATION": MORE WEIGHT ON OLD VALUE USED TO MAKE A NON-CONVERGENT PROCESS CONVERGE

↳ MORE "WEIGHT" ON NEW VALUE, SPEEDS CONVERGENCE

11/16/2015

LOOK @ JACOBI ITERATION

→ SUBSTITUTE FOR $\tilde{x}^{k+1}$

$$\underline{x}^{k+1} = \omega \underline{\underline{T}}\, \underline{x}^{k} + (1-\omega)x^{k} + \omega \underline{c}$$

$$\underline{x}^{k+1} = \underbrace{\left[\omega \underline{\underline{T}}_J + (1-\omega)\underline{\underline{I}}\right]}_{= \underline{\underline{T}}_\omega} x^{k} + \omega \underline{c}$$

$$\underline{x}^{k+1} = \underline{\underline{T}}_\omega \underline{x}^{k} + \omega c$$

- SUCCESSIVE OVER RELATION (SOR)

$$x_i^{k+1} = (1-\omega)x_i^{k} + \omega \tilde{x}_i^{k+1}$$

↑ COMPUTE USING GAUSS-SEIDEL

$$x_i^{k+1} = (1-\omega)x_i^{k} + \omega\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{k+1} - \sum_{j=i+1}^{N} a_{ij}x_j^{k}\right)/a_{ii}$$

- IN MATRIX FORM:

$$(\underline{\underline{D}} + \omega \underline{\underline{L}})x^{k+1} = \left[(1-\omega)\underline{\underline{D}} - \omega \underline{\underline{U}}\right]x^{k} + \omega \underline{b}$$

$$\hookrightarrow \quad x^{k+1} = \underbrace{(\underline{\underline{D}} + \omega \underline{\underline{L}})^{-1}\left[(1-\omega)\underline{\underline{D}} - \omega \underline{\underline{U}}\right]}_{I_{SOR}} \underline{x}^{k} + \omega(\underline{\underline{D}} + \omega \underline{\underline{L}})^{-1} b$$

* EIGENVALUES & EIGENVECTORS (CHAPTER 9.1 - 9.4)

$$\underline{\underline{A}}\,\underline{v}_i = \lambda_i \underline{v}_i \quad \rightarrow \quad (A - \lambda \underline{\underline{I}})\underline{v} = 0$$

↳ SOLUTION REQUIRES $\det(\underline{A} - \lambda \underline{\underline{I}}) = 0$

- FOR $N \times N$; $\underline{\underline{A}}$, $\det(A - \lambda \underline{\underline{I}})$ IS AN N^{TH} DEG. POLYNOMIAL

11/16/2015

* POWER METHOD

- CONSIDER THE ITERATION

$$y^{(k+1)} = \underline{\underline{A}}\, \underline{y}^{(k)}$$

EXPAND $\underline{y}^{(k)}$ IN TERMS OF EIGENVECTORS OF $\underline{\underline{A}}$

$$\underline{y}^{0} = \sum_{i=1}^{N} b_i \underline{v}_i$$

$$\underline{y}^{1} = \underline{\underline{A}}\, \underline{y}^{(0)} = \underline{\underline{A}} \sum_{i=1}^{N} b_i \underline{v}_i = \sum_{i=1}^{N} b_i \underline{\underline{A}}\, \underline{v}_i$$

$$y^{2} = \underline{\underline{A}} y^{(1)} = \ldots \sum_{i=1}^{N} b_i \lambda_i^{2} \underline{v}_i$$

$$\vdots$$

$$y^{(k)} = \underline{\underline{A}} y^{(k-1)} = \sum_{i=1}^{N} b_i \lambda_i^{k} \underline{v}_i \quad \rightarrow \quad \text{ASSUME } |\lambda_1| > |\lambda_2| > \ldots |\lambda_N|$$

THEN, $$y^{(k)} = \lambda_1^{k} \left\{ b_1 \underline{v}_1 + \sum_{i=2}^{N} b_i \underbrace{\left(\frac{\lambda_i}{\lambda_1}\right)}_{<1}^{k} \underline{v}_i \right.$$

$$\lim_{k \to \infty} \underline{y}^{(k)} = b_1 \lambda_1^{k} \underline{v}_1 = \lambda_1 \underbrace{\left(b_1 \lambda^{k-1} \underline{v}_1\right)}_{y^{(k-1)}}$$

$$\underline{y}^{(k)} = \lambda_1 \underline{y}^{(k-1)}$$

IF $|\lambda_1| > 1$, THEN $y^{(k)} \rightarrow \infty$ $\rightarrow$ TO KEEP $y^{(k)}$ FINITE & BOUNDED
IF $|\lambda_1| < 1$, THEN $y^{(k)} \rightarrow 0$ NORMALIZE AT EACH STEP

$$\hookrightarrow \|\underline{y}^{(k)}\| = \|\lambda_1 y^{(k-1)}\|$$

$$\|\underline{y}^{(k)}\| \cong |\lambda_1| \underbrace{\|\underline{y}^{(k-1)}\|}_{\equiv 1} \quad \rightarrow \quad |\lambda_1| \cong \|y^{(k)}\|$$

"POWER METHOD" FINDS THE LARGEST EIGENVALUE & CORRESPONDING EIGENVECTOR, $\underline{y}^{(k)}$ OF $\underline{\underline{A}}$

11/16/2015

- SHIFTING :

$$\underline{\underline{A}}\,\underline{x} = \lambda \underline{x} - s\underline{x} + s\underline{x} \rightarrow (\underline{\underline{A}} - s\underline{\underline{I}})\,\underline{x} = (\lambda - s)\,\underline{x}$$

$$\bar{\underline{\underline{A}}}\,\underline{x} = \bar{\lambda}\,\underline{x}$$

- INVERSE :

$$\underline{\underline{A}}\,\underline{x} = \lambda \underline{x} \rightarrow x = \underline{\underline{A}}^{-1} \lambda \underline{x} \rightarrow \frac{1}{\lambda} \underline{x} = \underline{\underline{A}}^{-1} \underline{x}$$