Numerical Analysis.

HW 17 and 18 before you leave
HW 19 assigned.

Today: Multistep Methods. – use more than one past value of the numerical soln.

Ex $$y_{n+1} = y_n + \frac{h}{2}\left[3f(x_n, y_n) - f(x_{n-1}, y_{n-1})\right]$$

this is explicit, stable, and second order... truncation error $O(h^3)$

Lets see how these methods are derived using Numerical Interpolation and Numerical Integration.

Start w/ our ODE $\quad y' = f(x, y)$

integrate over $[x_n, x_{n+1}]$ $$y(n+1) = y(n) + \int_{x_n}^{x_{n+1}} f(x, y)\, dx$$

approximate $f(x, y)$ using a polynomial interpolant.

interpolate @ x_n, x_{n-1}

LINEAR INTERPOLATION or TRAPEZOID RULE.

$$P_1 = \frac{(x_n - x) f(x_{n-1}, y_{n-1})}{(x_n - x_{n-1})} + \frac{(x - x_{n-1}) f(x_n, y_n)}{(x_n - x_{n-1})}$$

$$= \frac{1}{h}\left[(x_n - x) f(x_{n-1}, y_{n-1}) + (x - x_{n-1}) f(x_n, y_n)\right]$$

so $$y(n+1) = y(n) + \int_{x_n}^{x_{n+1}} \frac{1}{h}\left[(x_n - x) f(x_{n-1}, y_{n-1}) + (x - x_{n-1}) f(x_n, y_n)\right] dx$$

Evaluate the integral.

$$= y(n) + \frac{1}{h}\left[-\frac{1}{2}(x_n - x)^2 f(x_{n-1}, y_{n-1}) + \frac{1}{2}(x - x_{n-1})^2 f(x_n, y_n)\right]\Bigg|_{x_n}^{x_{n+1}}$$

$$= y(n) + \frac{1}{h}\left[-\frac{1}{2}h^2 f(x_{n-1}, y_{n-1}) + \frac{1}{2}(4h^2) f(x_n, y_n) - \frac{1}{2}h^2 f(x_n, y_n)\right]$$

$$= y(n) + \left[\frac{3h}{2} f(x_n, y_n) - \frac{h}{2} f(x_{n-1}, y_{n-1})\right]$$

this gives our method:

$$y(n+1) = y(n) + \frac{h}{2}\left[3f(x_n, y_n) - f(x_{n-1}, y_{n-1})\right]$$

What is the error associated with this method?

$$\text{Error} = \int_0^h f(x,y)\,dx - \frac{h}{2}\left[3f(0,y(0)) - f(-h,y(-h))\right]$$

for simplicity write $f(0) \Rightarrow f(0,y(0))$
$f(-h) \Rightarrow f(-h,y(-h))$

$$= \int_0^h f(x,y)\,dx - \frac{h}{2}\left[3f(0) - f(-h)\right]$$

Expand $f(x,y)$ in a Taylor Polynomial about $x=0$

$$f(x,y) = f(0) + xf'(0) + \frac{x^2}{2}f''(\xi_0) \quad \text{REMAINDER}$$

where $-h < \xi_0 < h$

Then $f(-h) = f(0) - hf'(0) + \frac{h^2}{2}f''(\xi_1)$

Plug these in ...

$$\text{Error} = \int_0^h f(0) + xf'(0) + \frac{x^2}{2}f''(\xi)\,dx - \frac{h}{2}\left[3f(0) - \left[f(0) - hf'(0) + \frac{h^2}{2}f''(\xi)\right]\right]$$

Integrate ...

$$hf(0) + \frac{h^2}{2}f'(0) + \frac{h^3}{6}f''(\xi) - \frac{h}{2}\left[2f(0) + hf'(0) - \frac{h^2}{2}f''(\xi)\right]$$

so $$\text{Error} = \frac{h^3 f''(\xi)}{6} + \frac{h^3}{4}f''(\xi) = \underbrace{\frac{5h^3 f''(\xi)}{12}}_{\text{we truncate the approximation.}}$$

Our error is $O(h^3)$
so our method is second order

NOTES ON ERROR AND APPROXIMATING ERROR ...

- If we know the REAL soln then $\text{ERROR}_h = \text{REAL} - \text{APPROX}_h$
$= y(x_n) - y_h(x_n)$

- Often in real life we do not know the real soln so we approximate the error using RICHARDSONS.
$\downarrow$
Always

HOW DO WE DERIVE RICHARDSONS ...

1. Determine the order of the method: SECOND

2. With standard assumptions on differentiability of $f(x,y)$ and y, we can write

$$y(x_n) - y_h(x_n) = D(x_n)h^2 + O(h^3)$$

error decreases as h^2 and there is a function $D(x)$ that can predict that...

for $D(x)$ a continuous <u>unknown</u> function.

<u>NOTE</u> ORDER ONE: $D(x_n)h + O(h^2)$

we just can't know $D(x)$

3. Write down an approximate error for h and $2h$...

$$y(x_n) - y_h(x_n) \approx D(x_n)h^2$$
$$y(x_n) - y_{2h}(x_n) \approx D(x_n)4h^2$$

4. Make the $D(x_n)$ disappear by combining these
   - Here we do 4 time the top minus the bottom.

$$3y(x_n) - 4y_h(x_n) + y_{2h}(x_n) \approx 0$$

5. Rearrange to get $y(x_n) - y_h(x_n)$ on LHS.

$$3[y(x_n) - y_h(x_n)] - y_h(x_n) + y_{2h}(x_n) \approx 0$$

$$y(x_n) - y_h(x_n) \approx \underbrace{\frac{1}{3}[y_h(x_n) - y_{2h}(x_n)]}_{\text{We can use this to estimate the error in } y_h(x_n).}$$

In practice we make sure the estimate $\frac{1}{3}[y_h(x_n) - y_{2h}(x_n)]$ is well below our error tolerance. Choose an h so that this error is very small.

PSUEDO CODE ... what do we need to do to implement this?

$$y(n+1) = y(n) + \frac{h}{2}[3f(x_n\, y_n) - f(x_{n-1}\, y_{n-1})]$$

note
- The first step must use a one step method! Usually EULER or RUNGE KUTTA. depending on accuracy needed.

This approximation is an example of an ADAMS-BASHFORTH method.

- Considered to be less computationally expensive than RUNGE-KUTTA
  - we can save past evaluations of $f(x,y)$ so that we only evaluate it once per step.

General ADAMS-BASHFORTH Methods:

$$y(n+1) = y(n) + \int_{x_n}^{x_{n+1}} f(x,y)\,dx$$

before we used P_1 but we could always pick a higher order interpolating polynomial. using previous grid points to interpolate.

$q+1$ points: $\{x_n, x_{n-1}, x_{n-2}, \dots, x_{n-q}\}$

then $$P_q(x) = \sum_{j=0}^{q} L_{j,n}(x) f[x_{n-j}, y(x_{n-j})]$$

$$= \sum_{j=0}^{q} L_{jn}(x)\, y'(x_{n-j})$$

because $y' = f(x,y)$

here $L_{jn}(x)$ are the Lagrange Basis Functions.

so

$$y(n+1) = y(n) + \int_{x_n}^{x_{n+1}} P_q(x)\,dx$$

$$= y(n) + h\sum_{j=0}^{q} w_{jq} f(x_{n-j}, y(x_{n-j})) + \underbrace{E_n}_{\text{Truncation Error.}}$$

where $$hw_{jq} = \int_{x_n}^{x_{n+1}} L_{jn}(x)\,dx$$

we must integrate each of the basis functions individually

so $$y_{n+1} = y_n + h\sum_{j=0}^{q} w_{jq} f(x_{n-j}, y(x_{n-j}))$$

is a $(q+1)$-step, explicit method, with a truncation error of

$O(h^{q+2})$

$\underline{q = 0}$:

$$
\begin{aligned}
y_{n+1} &= y_n + hf(x_n, y_n) \qquad \text{[Euler's method]} \\
E_n &= \frac{1}{2} h^2 Y''(\xi_n)
\end{aligned}
$$

$\underline{q = 1}$: Use the notation $y'_k = f(x_k, y_k)$.

$$
\begin{aligned}
y_{n+1} &= y_n + \frac{h}{2}\left[3y'_n - y'_{n-1}\right], \quad n \geq 1 \\
E_n &= \frac{5}{12} h^3 Y'''(\xi_n)
\end{aligned}
$$

$\underline{q = 2}$:

$$
\begin{aligned}
y_{n+1} &= y_n + \frac{h}{12}[23y'_n - 16y'_{n-1} + 5y'_{n-2}] \\
E_n &= \frac{3}{8} h^4 Y^{(4)}(\xi_n)
\end{aligned}
$$

$\underline{q = 3}$:

$$
\begin{aligned}
y_{n+1} &= y_n + \frac{h}{24}[55y'_n - 59y'_{n-1} + 37y'_{n-2} - 9y'_{n-3}] \\
E_n &= \frac{251}{720} h^5 Y^{(5)}(\xi_n)
\end{aligned}
$$

EXAMPLES OF THESE (Hand out).

- NOTE ... The $q+1$ Initial values must be computed with other methods...

  Runge Kutta or
  a LOWER ORDER Adams-Bashforth.

- CONVERGENCE... Similar to Euler methods we can show that convergence depends on higher derivatives.

From the examples we see that

$$\max_{x_0 \le x \le b} |E_n| \le d\, h^{q+2} \max_{x_0 \le x \le b} \left| y^{(q+2)}(x) \right|$$

- STABILITY ... not absolutely stable but we can choose an h that works.

- IN practice: we do a "convergence test"

Save all this data →

1. Start with a "large" h value
2. Solve the system
3. cut h in half
4. Solve

continue doing this untill h is quite small

Then Analyze:

A. As h decreases the solutions should be approaching the REAL soln.

B. The ratio of Richardson Errors should be decreasing with same order as the METHOD.

C. A you reduce h more and more the numerical solutions should vary less and less.

Recall the notation $y'_k = f(x_k, y_k)$, $k \geq 0$.

$\underline{q = 0}$: It is the backward Euler method,

$$\begin{aligned} y_{n+1} &= y_n + h y'_{n+1} \\ \widehat{E}_n &= -\frac{1}{2} h^2 Y''(\xi_n) \end{aligned}$$

$q = 1$: It is the trapezoidal method,

$$\begin{aligned} y_{n+1} &= y_n + \frac{h}{2}[y'_{n+1} + y'_n] \\ \widehat{E}_n &= -\frac{1}{12} h^3 Y^{(3)}(\xi_n) \end{aligned}$$

$q = 2$:

$$\begin{aligned} y_{n+1} &= y_n + \frac{h}{12}[5y'_{n+1} + 8y'_n - y'_{n-1}] \\ \widehat{E}_n &= -\frac{1}{24} h^4 Y^{(4)}(\xi_n) \end{aligned}$$

$q = 3$:

$$\begin{aligned} y_{n+1} &= y_n + \frac{h}{24}[9y'_{n+1} + 19y'_n - 5y'_{n-1} + y'_{n-2}] \\ \widehat{E}_n &= -\frac{19}{720} h^5 Y^{(5)}(\xi_n) \end{aligned}$$

ADAMS - MOULTON METHODS...

only difference $y(n+1) = y(n) + \int_{x_n}^{x_{n+1}} f(x,y)\,dx$

approximate using polynomial interpolants at points starting with x_{n+1}

$$\{x_{n+1}, x_n, \ldots, x_{n-q+1}\}$$

This means the method is always IMPLICIT.

EXAMPLES OF THESE...

general form

$$y_{n+1} = y_n + h \sum_{j=-1}^{q-1} v_{jq}\, f(x_{n-j}, y_{n-j})$$

where $hv_{jq} = \int_{x_n}^{x_{n+1}} \hat{L}_{jn}(x)\,dx$

the lagrange basis using the new points.

NOTES.

1 Higher order Adams-Moulton methods are not absolutely stable - but they are often more desirable for highly unstable or poorly conditioned problems.

2. To use these methods we:

A. Make an initial guess using an explicit method

B. use fixed point iteration to solve the implicit eqn.

often we use an Adams-Bashforth method with comparable accuracy.