Math 685.

Lecture 13.

Summary of num. methods for ODE: $y' = f(t,y)$

Explicit

① Forward Euler

$$y_{k+1} = y_k + hf(t_k, y_k)$$

Ex. $y'' + 5y = 0$

$$\begin{pmatrix} x_1 = y \\ x_2 = y' \end{pmatrix} \Rightarrow \begin{cases} x_1' = x_2 \\ x_2' = -5x_1 \end{cases}$$

$$y_0 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad h = 0.1 \quad t_0 = 0$$

$$y_1 = y_0 + 0.1 \cdot f(0, y_0)$$

$$f(0, y_0) = f\left(0, \begin{pmatrix} 1 \\ 1 \end{pmatrix}\right) = \begin{pmatrix} 1 \\ -5 \end{pmatrix}$$

$$y_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + 0.1 \cdot \begin{pmatrix} 1 \\ -5 \end{pmatrix} = \begin{pmatrix} 1.1 \\ 0.5 \end{pmatrix}$$

If applied to $y' = \lambda y$, $\lambda < 0$

Conv. region is $|1 + h\lambda| < 1$

so h has to be relatively small

(conditionally stable)

Implicit

Backward Euler

$$y_{k+1} = y_k + hf(t_{k+1}, y_{k+1})$$

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}' = f\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \quad f(\vec{x}) = \begin{pmatrix} x_2 \\ -5x_1 \end{pmatrix}$$

$$y_1 = \left(y_1^{(1)}, y_1^{(2)}\right)^T$$

$$y_1 = y_0 + 0.1 \cdot f(0.1, y_1) = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + 0.1 \cdot \begin{pmatrix} y_1^{(2)} \\ -5y_1^{(1)} \end{pmatrix}$$

$$\begin{cases} y_1^{(1)} = 1 + 0.1 \cdot y_1^{(2)} \\ y_1^{(2)} = 1 - 0.5 \cdot y_1^{(1)} \end{cases}$$

$\Rightarrow$ solve for $y_1^{(1)}$ & $y_1^{(2)}$

requires more work,

but it's much

more stable

no restriction on h

(unconditionally stable)

② Trapezoidal method: average of BE & FE

implicit

Heun's method – 2 step method

$$y_{k+1} = y_k + \frac{h_k}{2}\left(f(t_k, y_k) + h_k \cdot k_1\right)$$

$$k_1 = y_k + h_k f(t_k, y_k)$$

③ Taylor series methods of higher order

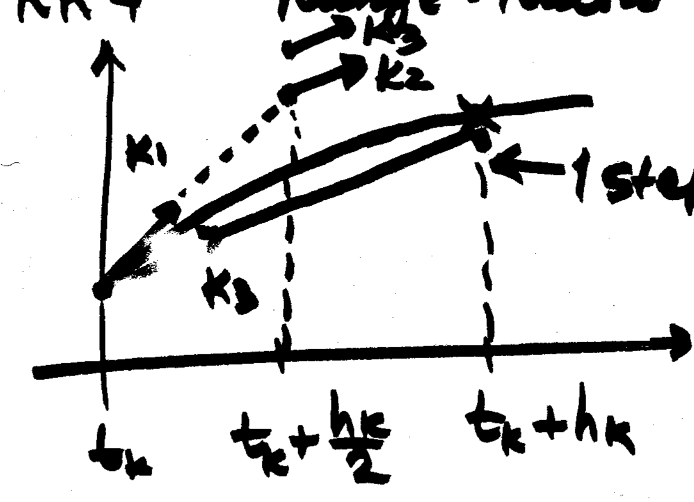
$$y_{k+1} = y_k + h_k f + \frac{h_k^2}{2} y'' \leftarrow O(h_k^2)$$

truncation error

$$y' = f \xrightarrow{\text{chain rule}} y'' = f_t + f_x \cdot f$$

more accurate, but requires more computational effort (f_x, f_t etc. need to be computed)

④ RK4 – Runge-Kutta method



$$y_{k+1} = y_k + \frac{h_k}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$k_1 = f(t_k, y_k)$$

$$k_2 = f\left(t_k + \frac{h_k}{2}, y_k + \frac{h_k}{2}k_1\right)$$

$$k_3 = f\left(t_k + \frac{h_k}{2}, y_k + \frac{h_k}{2}k_2\right)$$

$$k_4 = f(t_k + h_k, y_k + h_k \cdot k_3)$$

Ex. $y'' + 5y = 0$.

$$x_1 = y, \quad x_2 = y' \Rightarrow \vec{x}' = \begin{pmatrix} x_2 \\ -5x_1 \end{pmatrix} = \vec{f}(\vec{x}) \qquad \begin{cases} x_1' = x_2 \\ x_2' = -5x_1 \end{cases}$$

$$t_0 = 0, \quad h = 0.1 \quad x_0 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} = x(t_0)$$

$$k_1 = f\left(0.0, \begin{pmatrix} 1 \\ 1 \end{pmatrix}\right) = \begin{pmatrix} 1 \\ -5 \end{pmatrix}$$

$$k_2 = f\left(\frac{0.1}{2}, \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \frac{0.1}{2}\begin{pmatrix} 1 \\ -5 \end{pmatrix}\right) = f\left(\frac{0.1}{2}, \begin{pmatrix} 1.05 \\ 0.75 \end{pmatrix}\right) = \begin{pmatrix} 0.75 \\ -5.25 \end{pmatrix}$$

$$k_3 = f\left(\frac{0.1}{2}, \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \frac{0.1}{2}\begin{pmatrix} 0.75 \\ -5.25 \end{pmatrix}\right) = f\left(\frac{0.1}{2}, \begin{pmatrix} 1.0375 \\ 0.7375 \end{pmatrix}\right) = \begin{pmatrix} 0.7375 \\ -5 \cdot 1.0375 \end{pmatrix}$$

$$k_4 = f\left(0.1, \begin{pmatrix} 1 \\ 1 \end{pmatrix} + 0.1\begin{pmatrix} 0.7375 \\ -5.1875 \end{pmatrix}\right) = f\left(0.1, \begin{pmatrix} 1 + 0.1 \cdot 0.7375 \\ 1 - 0.1 \cdot 5.1875 \end{pmatrix}\right) = \begin{pmatrix} 1 - 0.51875 \\ -5 \cdot 1.07375 \end{pmatrix}$$

$$x_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \frac{0.1}{6}\left[\begin{pmatrix} 1 \\ -5 \end{pmatrix} + 2 \cdot \begin{pmatrix} 0.75 \\ -5.25 \end{pmatrix} + 2 \cdot \begin{pmatrix} 0.7375 \\ -5.1875 \end{pmatrix} + \begin{pmatrix} 0.49.. \\ -5.35.. \end{pmatrix}\right]$$

⊕ RK methods are very accurate (RK4 is 4-th order)

⊖ Rigorous convergence analysis is not available, step size control is lacking.

⑤ Extrapolation methods.

$(t_j, y_j) \leftarrow$ fit a poly to this data $\hat{y}(h)$
$j = 1 \dots k$

$y_{k+1} \approx \hat{y}(0)$.

⊕ very accurate solution

⊖ high interp. cost, no flexibility in choice of nodes.

⑥ Multistep methods.

PECE $\quad y_{k+1} = y_k + \frac{h}{2}(3y_k' - y_{k-1}') \quad (*)$

ex: $\quad y_{k+1} = y_k + \frac{h}{2}(y_{k+1}' + y_k') \quad (**)$

1. compute y_0, y_1 somehow (any method)
2. compute predictor $(*)$: y_2
3. Compute corrector $(**)$: $y_2 = y_1 + \frac{h}{2}(\underset{\uparrow \text{ predictor}}{y_2'} + y_1')$

Iterate.

Stability threshold: $-a \le h\lambda \le b$.

changing stepsize may be difficult

(7) Multivalue methods.

$$\vec{y}_k = \begin{bmatrix} y_k \\ h y_k' \\ \frac{h^2}{2} y_k'' \\ \frac{h^3}{6} y_k''' \end{bmatrix} \leftarrow \text{4 pieces of info at } y_k$$

$$\widehat{\vec{y}_{k+1}} = B y_k$$

$$\widehat{\vec{y}_{k+1}} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y_k \\ h y_k' \\ \frac{h^2}{2} y_k'' \\ \frac{h^3}{6} y_k''' \end{bmatrix}$$

$$y_{k+1}' = y' + h y'' + \frac{h^2}{2} y'''$$

$$y_{k+1}' = f(t_{k+1}, y_{k+1})$$

$$\vec{y}_{k+1} = \widehat{\vec{y}_{k+1}} + \alpha \vec{r}$$

$\uparrow$ line search. $\alpha_2 = 1.$