Numerical Marine Hydrodynamiscs

- Numerical Differentiation
  - Newton Interpolation
  - Finite Differences
- Ordinary Differential Equations
  - Initial Value Problems
    - Euler's Method
    - Taylor Series Methods
      - Error analysis
    - Runge-Kutta Methods
  - Systems of differential equations
  - Boundary Value Problems
    - Shooting method
    - Direct Finite Difference methods

Numerical Interpolation
Newton's Iteration Formula

Standard triangular family of polynomials

$$
\begin{aligned}
f(x) &= p(x) + r(x) \\
&= c_0 + c_1(x - x_0) \cdots + c_n(x - x_0) \cdots (x - x_{n-1}) \\
&\quad + \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0) \cdots (x - x_n)
\end{aligned}
$$

Divided Differences

$$f(x_0) = c_0 \Rightarrow c_0 = f(x_0)$$

$$f(x_1) = c_0 + c_1(x_1 - x_0) \Rightarrow c_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} = f[x_0, x_1]$$

$$f(x_2) = c_0 + c_1(x_2 - x_0) + c_2(x_2 - x_0)(x_2 - x_1)$$

$$c_2 = \frac{\frac{f(x_2)-f(x_1)}{x_2-x_1} - \frac{f(x_1)-f(x_0)}{x_1-x_0}}{x_2 - x_0} = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0} = f[x_0, x_1, x_2]$$

$$c_n = f[x_0, x_1, \ldots, x_n] = \frac{f[x_1, \ldots, x_n] - f[x_0, \ldots x_{n-1}]}{x_n - x_0}$$

Newton's Computational Scheme

c_0 c_1 c_2 c_3

x_0 $f(x_0)$

$f[x_0, x_1]$

x_1 $f(x_1)$ $f[x_0, x_1, x_2]$

$f[x_1, x_2]$ $f[x_0, x_1, x_2, x_3]$

x_2 $f(x_2)$ $f[x_1, x_2, x_3]$

$f[x_2, x_3]$

x_3 $f(x_3)$

Numerical Interpolation
Equidistant Newton Interpolation

Equidistant Sampling

$$x_i = x_0 + ih$$

$$f[x_0, x_1] = \frac{f(x_1) - f(x_0)}{x_1 - x_0} = \frac{1}{h}(f_1 - f_0) = \frac{1}{h}\Delta f_0$$

$$f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0} = \frac{1}{1 \cdot 2 \cdot h^2}(f_2 - 2f_1 + f_0) = \frac{1}{2!h^2}\Delta^2 f_0$$

$$f[x_0, x_1, x_2, x_3] = \frac{1}{3! \cdot h^3}(f_3 - 3f_2 + 3f_1 - f_0) = \frac{1}{3!h^3}\Delta^3 f_0$$

Divided Differences
Stepsize Implied

$x_0 \quad f_0$

$f_1 - f_0 = \Delta f_0$

$x_1 \quad f_1$

$f_2 - 2f_1 + f_0 = \Delta^2 f_0$

$f_2 - f_1 = \Delta f_1$

$f_3 - 3f_2 + 3f_1 - f_0 - = \Delta^3 f_0$

$x_2 \quad f_2$

$f_3 - 2f_2 + f_1 = \Delta^2 f_1$

$f_3 - f_2 = \Delta f_2$

$x_3 \quad f_3$

$$c_0 = f_0$$

$$c_1 = \frac{1}{h}\Delta f_0$$

$$c_2 = \frac{1}{2!h^2}\Delta^2 f_0$$

$$c_3 = \frac{1}{3!h^3}\Delta^3 f_0$$

Triangular Family of Polynomials
Equidistant Sampling

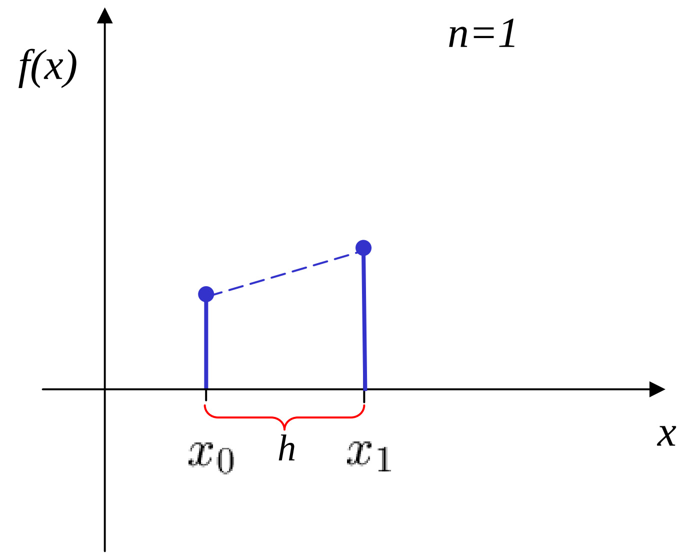
$$f(x) = f_0 + \frac{\Delta f_0}{h}(x - x_0) + \frac{\Delta^2 f_0}{2!h^2}(x - x_0)(x - x_1) + \cdots$$

$$+ \frac{\Delta^n f_0}{n!h^n}(x - x_0)(x - x_1)\cdots(x - x_{n-1}) + \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0)\cdots(x - x_n)$$

Numerical Differentiation

Triangular Family of Polynomials
Equidistant Sampling

$$
\begin{aligned}
f(x) &= f_0 + \frac{\Delta f_0}{h}(x - x_0) + \frac{\Delta^2 f_0}{2! h^2}(x - x_0)(x - x_1) + \cdots \\
&+ \frac{\Delta^n f_0}{n! h^n}(x - x_0)(x - x_1)\cdots(x - x_{n-1}) + \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0)\cdots(x - x_n)
\end{aligned}
$$



First order

$$n = 1$$

$$f(x) = f_0 + \frac{\Delta f_0}{h}(x - x_0) + \frac{f''(\xi)}{2!}(x - x_0)(x - x_1)$$

$$\boxed{f'(x) = \frac{\Delta f_0}{h} + O(h) = \frac{1}{h}(f_1 - f_0) + O(h)}$$

Numerical Differentiation

Second order

$n = 2$

$$f(x) = f_0 + \frac{\Delta f_0}{h}(x-x_0) + \frac{\Delta^2 f_0}{2!h^2}(x-x_0)(x-x_1) + \frac{f'''(\xi)}{3!}(x-x_0)(x-x_1)(x-x_2) + \cdots$$

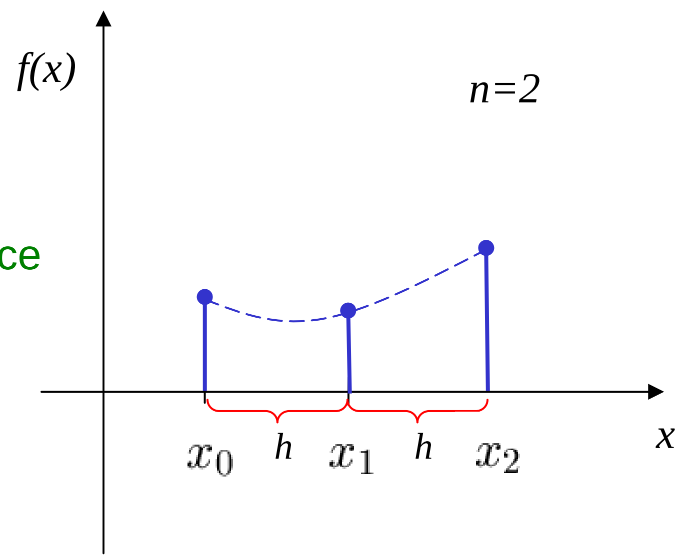
$$f'(x) = \frac{\Delta f_0}{h} + \frac{\Delta^2 f_0}{h}(x-x_0) + \frac{\Delta^2 f_0}{h}(x-x_1) + O(h^2)$$

$$\begin{aligned} f'(x_0) &= \frac{f_1 - f_0}{h} - \frac{1}{2h}(f_2 - 2f_1 + f_0) + O(h^2) \\ &= \frac{2f_1 - 2f_0 - f_2 + 2f_1 - f_0}{2h} + O(h^2) \\ &= \boxed{\frac{1}{h}(-\frac{3}{2}f_0 + 2f_1 - \frac{1}{2}f_2) + O(h^2)} \end{aligned}$$

Forward Difference

$$\begin{aligned} f'(x_1) &= \frac{f_1 - f_0}{h} + \frac{1}{2h}(f_2 - 2f_1 + f_0) + O(h^2) \\ &= \boxed{\frac{1}{2h}(f_2 - f_0) + O(h^2)} \end{aligned}$$

Central Difference

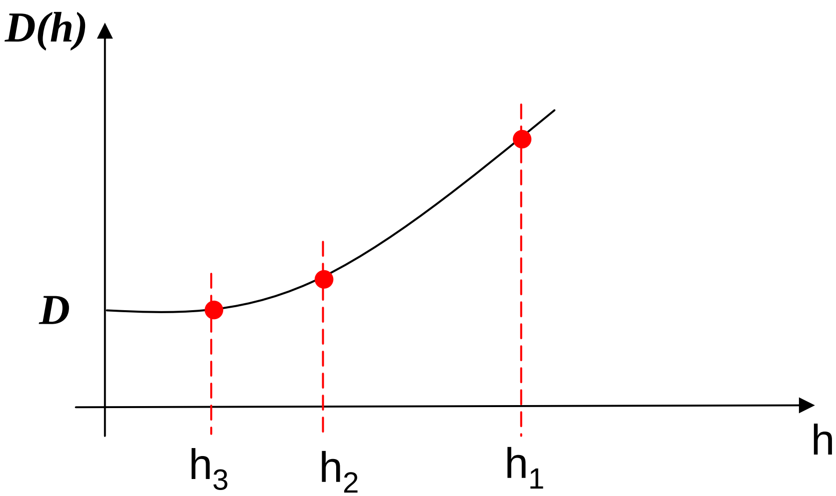


Second Derivatives

n=2 $$f''(x_0) = \frac{\Delta^2 f_0}{h^2} + O(h) = \boxed{\frac{1}{h^2}(f_0 - 2f_1 + f_2) + O(h)}$$ Forward Difference

n=3 $$f''(x_1) = \boxed{\frac{1}{h^2}(f_0 - 2f_1 + f_2) + O(h^2)}$$ Central Difference

Numerical Differentiation
Richardson Extrapolation



'Romberg' Differentiation Algorithm

$$\boldsymbol{D}_{j,k} \simeq \frac{4^{k-1}\boldsymbol{D}_{j+1,k-1} - \boldsymbol{D}_{j,k-1}}{4^{k-1}-1}$$

Order 2

$$k = 2, j = 1$$

$$\boldsymbol{D}_{1,2} \simeq \frac{4\boldsymbol{D}_{2,1} - \boldsymbol{D}_{1,1}}{3}$$

k →

		1: $O(h^2)$	2: $O(h^4)$	3: $O(h^6)$	4: $O(h^8)$
a.		0.172800	1.367467		
		1.068800			
b,.		0.172800	1.367467	1.640533	
		1.068800	1.623467		
		1.484800			
c.	j ↓	0.172800	1.367467	1.640533	1.640533
		1.068800	1.623467	1.640533	
		1.484800	1.639467		
		1.600800			

Ordinary Differential Equations
Initial Value Problems

Differential Equation

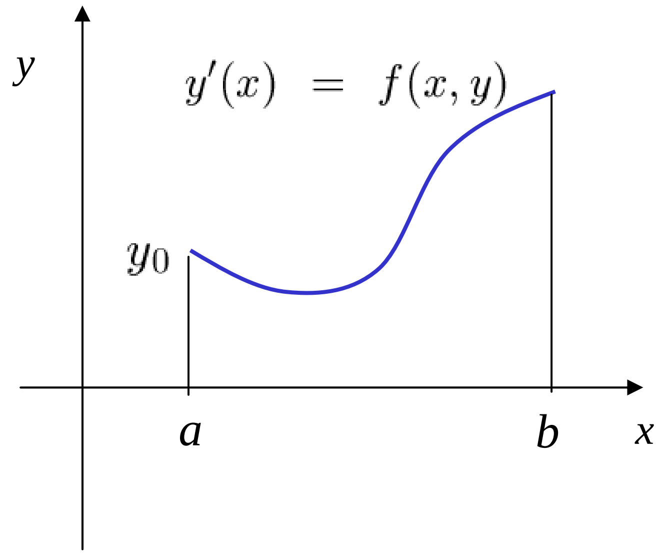
$$y'(x) = f(x,y)\,, \quad x \in [a,b]$$
$$y(x_0) = y_0$$

Linear Differential Equation

$$f(x,y) = -p(x)y + q(x)$$

Non-Linear Differential Equation

$f(x,y)$ non-linear in y



Linear differential equations can often be solved analytically

Non-linear equations require numerical solution

Ordinary Differential Equations
Initial Value Problems

Euler's Method

Differential Equation

$$\frac{dy}{dx} = f(x,y)\ ,\ y_0 = p$$

Example

$$f(x,y) = x\ (y = x^2/2 + p)$$

Discretization

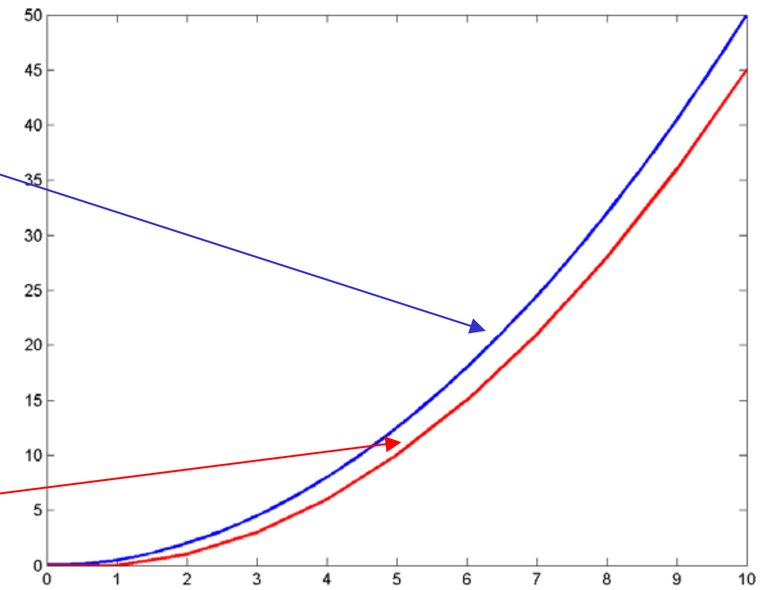
$$x_n = nh$$

Finite Difference (forward)

$$\frac{dy}{dx}\Big|_{x=x_n} \simeq \frac{y_{n+1} - y_n}{h}$$

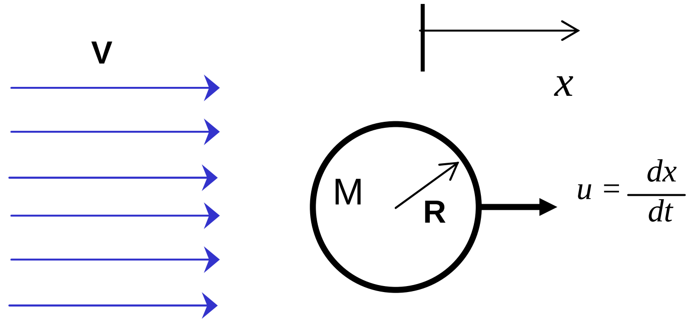
Recurrence

$$y_{n+1} = y_n + hf(nh, y_n)$$



euler.m

Sphere Motion in Fluid Flow



Equation of Motion – 2nd Order Differential Equation

$$M \frac{d^2x}{dt^2} = 1/2 \rho C_d \pi R^2 \left(V - \frac{dx}{dt} \right)^2$$

$$\frac{dx}{dt} = u$$

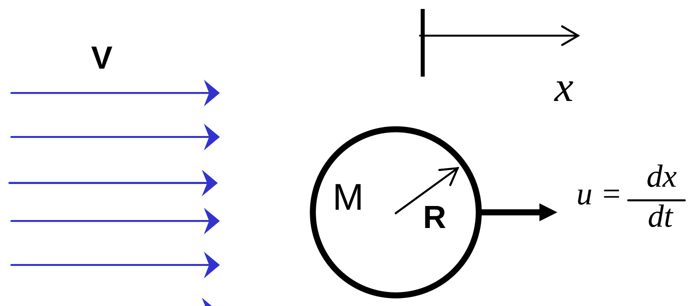
$$\frac{du}{dt} = \frac{\rho C_d \pi R^2}{2M} (V^2 - 2uV + u^2)$$

Euler's Method

$$u_{i+1} = u_i + \left(\frac{du}{dt} \right)_i \Delta t, \quad u(0) = 0$$

$$x_{i+1} = x_i + \left(\frac{dx}{dt} \right)_i \Delta t, \quad x(0) = 0$$

Sphere Motion in Fluid Flow
MATLAB Solutions



dudt.m

```
function [f] = dudt(t,u)
% u(1) = u
% u(2) = x
% f(2) = dx/dt = u
% f(1) = du/dt=rho*Cd*pi*r/(2m)*(v^2-2uv+u^2)
[rho,Cd,m,r,v] = sph_param();
fac=rho*Cd*pi*r^2/(2*m);

f(1)=fac*(v^2-2*u(1)+u(1)^2);
f(2)=u(1);
f=f';
```

sph_drag_2.m

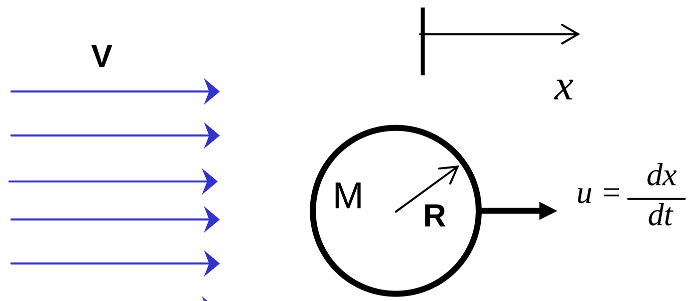
```
%step size
h=1.0;
% Euler's method, forward finite difference
t=[0:h:10];
N=length(t);
u_e=zeros(N,1);
x_e=zeros(N,1);
u_e(1)=0;
x_e(1)=0;
[rho,Cd,m,r,v] = sph_pa
fac=rho*Cd*pi*r^2/(2*m),
for n=2:N
    u_e(n)=u_e(n-1)+h*fac*(v^2-2*v*u_e(n-1)+u_e(n-1)^2);
    x_e(n)=x_e(n-1)+h*u_e(n-1);
end
% Runge Kutta
u0=[0 0]';
[tt,u]=ode45(@dudt,t,u0);

figure(1)
hold off
a=plot(t,u_e,'+b');
hold on
a=plot(tt,u(:,1),'.g');
a=plot(tt,abs(u(:,1)-u_e),'+r');
...
figure(2)
hold off
a=plot(t,x_e,'+b');
hold on
a=plot(tt,u(:,2),'.g');
a=plot(tt,abs(u(:,2)-x_e),'xr');
...
```

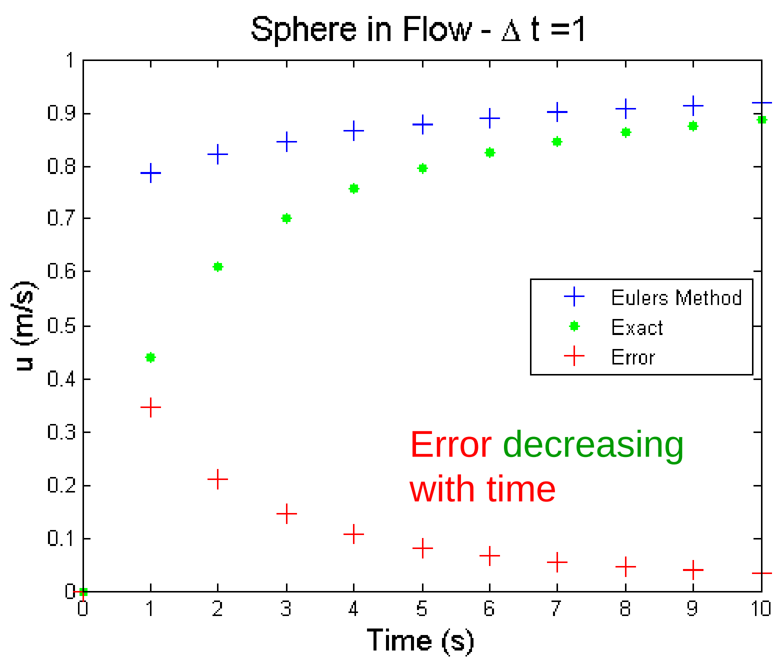
$$u_{i+1} = u_i + \left(\frac{du}{dt}\right)_i \Delta t, \quad u(0) = 0$$

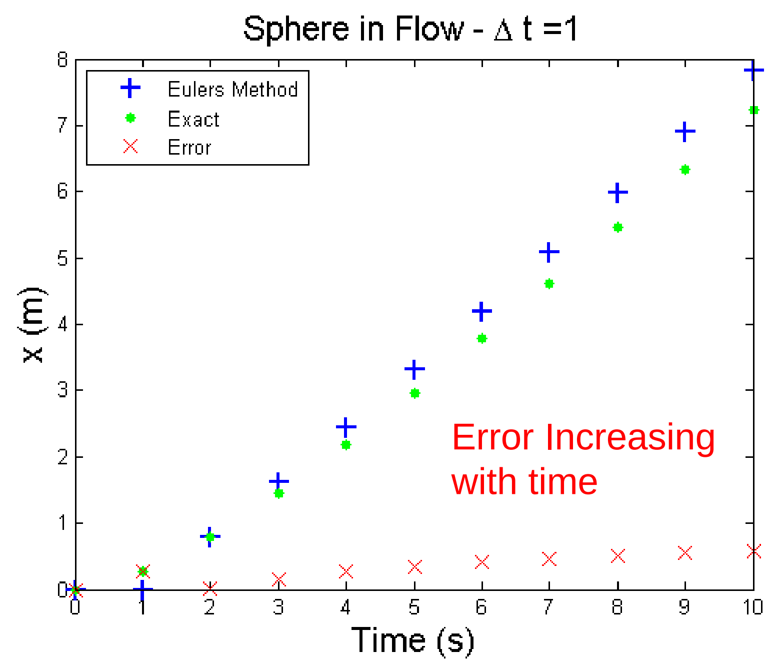
$$x_{i+1} = x_i + \left(\frac{dx}{dt}\right)_i \Delta t, \quad x(0) = 0$$

Sphere Motion in Fluid Flow

Error Propagation







Initial Value Problems
Taylor Series Methods

Initial Value Problem

$$y' = f(x,y)\ ,\quad y(x_0) = y_0$$

Taylor Series

$$y(x) = y_0 + (x - x_0)y'(x_0) + \frac{(x-x_0)^2}{2}y'' + \cdots$$

Derivatives

$$y' = f(x,y) \;\Rightarrow\; y'(x_0) = f(x_0, y_0)$$

$$y'' = \frac{df(x,y)}{dx} = f_x + f_y y' = f_x + f_y f$$

$$y''' = \frac{d^2 f(x,y)}{dx^2} = f_{xx} + f_{xy} f + f_{yx} f + f_{yy} f^2 + f_y f_x + f_y^2 f$$

$$= f_{xx} + 2f_{xy} + f_{yy} f^2 + f_x f_y + f_y^2 f$$

Partial Derivatives

$$f_x = \frac{\partial}{\partial x}$$

$$f_y = \frac{\partial}{\partial y}$$

Truncate series to k terms

$$y_1 = y(x_1) = y_0 + hy'(x_0) + \frac{h^2}{2!}y''(x_0) + \cdots + \frac{h^k}{k!}y^{(k)}(x_0)$$

$$y_2 = y(x_2) = y_1 + hy'(x_1) + \frac{h^2}{2!}y''(x_1) + \cdots + \frac{h^k}{k!}y^{(k)}(x_1)$$

.

$$y_n = y(x_n) = y_{n-1} + hy'(x_{n-1}) + \frac{h^2}{2!}y''(x_{n-1}) + \cdots + \frac{h^k}{k!}y^{(k)}(x_{n-1})$$

Choose Step Size h

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

Discretization

$$x_n = a + nh\ ,\quad n = 0, 1, \ldots N$$

Recursion Algorithm

$$y(x_{n+1}) = y_{n+1} = y_n + hT_k(x_n, y_n) + \frac{h^{k+1}}{(k+1)!}y^{(k+1)}(\xi)$$

with

$$T_k(x_n, y_n) = f(x_n, y_n) + \frac{h}{2!}f'(x_n, y_n) + \cdots \frac{h^{k-1}}{k!}f^{(k-1)}(x_n, y_n)$$

Local Error

$$E = \frac{h^{k+1} f^{(k)}(\xi, y(\xi))}{(k+1)!} = \frac{h^{k+1} y^{(k+1)}(\xi)}{(k+1)!}\ ,\quad x_n < \xi < x_n + h$$

Initial Value Problems
Taylor Series Methods

General Taylor Series Method

$$x_n = a + nh\ , \quad n = 0, 1, \ldots N$$

$$y(x_{n+1}) = y_{n+1} = y_n + hT_k(x_n, y_n) + \frac{h^{k+1}}{(k+1)!} y^{(k+1)}(\xi)$$

$$T_k(x_n, y_n) = f(x_n, y_n) + \frac{h}{2!} f'(x_n, y_n) + \cdots \frac{h^{k-1}}{k!} f^{(k-1)}(x_n, y_n)$$

$$E = \frac{h^{k+1} f^{(k)}(\xi, y(\xi))}{(k+1)!} = \frac{h^{k+1} y^{(k+1)}(\xi)}{(k+1)!}\ , \quad x_n < \xi x_n + h$$

Example

$$k = 1$$

Euler's Method

$$y_{n+1} = y_n + hf(x_n, y_n)$$

$$E = \frac{h^2}{2!} y''(\xi)$$

Example – Euler's Method

$$y' = y\ , \quad y(0) = 1\ , \quad y = e^x$$

$$\begin{aligned} y(0.01) \simeq y1 &= y_0 + hf(x_0, y_0) = 1 + 0.01 \cdot 1 = 1.01 \\ y(0.02) \simeq y2 &= y_1 + hf(x_1, y_1) = 1.01 + 0.01 \cdot 1.01 = 1.021 \\ y(0.03) \simeq y3 &= y_2 + hf(x_2, y_2) = 1.021 + 0.01 \cdot 1.021 = 1.03121 \end{aligned}$$

$$y(0.03) = 1.0305$$

Error Analysis?

Initial Value Problems
Taylor Series Methods
Error Analysis

Error Estimates and Convergence

$$y' = f(x,y)\ ,\ \ y(x_0) = y_0$$

Euler's Method

Estimate $$y_{n+1} = y_n + hf(x_n, y_n)\ ,\ \ n = 0, 1, \cdots$$

$$x_n = x_0 + nh$$

$$e_n = y(x_n) - y_n$$

Exact $$y(x_{n+1}) = y(x_n) + hy'(x_n) + \frac{h^2}{2}y''(\xi_n)\ ,\ \ x_n < \xi_n < x_{n+1}$$

$$e_\ell = \frac{h^2}{2}y''(\xi^n)$$

$$e_{n+1} = y(x_{n+1}) - y_{n+1} = y(x_n) + hy'(x_n) + \frac{h^2}{2}y''(\xi_n) - y_n - hf(x_n, y_n)$$

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

$$f(x_n, y(x_n)) - f(x_n, y_n) = \frac{\partial f(x_n, y_n)}{\partial y}(y(x_n) - y_n) = f_y(x_n, y_n)e_n$$

$$e_{n+1} = e_n + hf_y(x_n, y_n)e_n + \frac{h^2}{2}y''(\xi_n)$$

)
)

$$|e_{n+1}| \le |e_n| + h|f_y(x_n, y_n)e_n| + \frac{h^2}{2}|y''(\xi_n)|$$

Derivative Bounds

$$|f_y(x_n, y_n)| \le L\ ,\ \ |y''(\xi_n)| \le Y$$

$$|e_{n+1}| \le (1 + hL)|e_n| + \frac{h^2}{2}Y$$

$$\eta_{n+1} = (1 + hL)\eta_n + \frac{h^2}{2}Y\ ,\ \ \eta_0 = 0$$

$$\eta_n = \frac{hY}{2L}[(1 + hL)^n - 1]$$

Error Bounds

$$\begin{aligned} |e_n| \le \eta_n &= \frac{hY}{2L}[(1 + hl)^n - 1] \\ &\le \frac{hY}{2L}[(e^{hL})^n - 1] \\ &= \frac{hY}{2L}[e^{hLn} - 1] \\ &\Rightarrow \end{aligned}$$

$$|e_n| \le \frac{hY}{2L}[(e^{(x_n - x_0)L} - 1]$$

O(h)

Initial Value Problems
Taylor Series Methods
Error Analysis

Example – Euler's Method

$$y' = y \ , \quad y(0) = 1 \ , \quad x \in [0,1]$$

Exact solution

$$y = e^x$$

Derivative Bounds

$$f_y = 1 \Rightarrow L = 1$$

$$y''(x) = e^x \Rightarrow Y = e$$

Error Bound $x=nh=1$

$$|e_n| \leq \frac{he}{2}(e-1)$$

$$h = 0.1 \Rightarrow |e_n| \leq 0.24$$

Euler's Method

$$y_{n+1} = y_n + hf(x_n, y_n) = (1+h)y_n$$

$$y_{11} = 2.5937$$

$$y(x_{11}) = 2.71828$$

$$e_{11} = 0.1246 < 0.24$$