Introduction to Numerical Analysis for Engineers

	Mathews
• Ordinary Differential Equations	9
– Initial Value Problems	9.1
• Euler's Method	9.2
• Taylor Series Methods	9.4
– Error analysis	
• Runge-Kutta Methods	9.5
– Systems of differential equations	9.7
– Boundary Value Problems	9.8
• Shooting method	9.8
• Direct Finite Difference methods	9.9

Ordinary Differential Equations
Initial Value Problems

Differential Equation

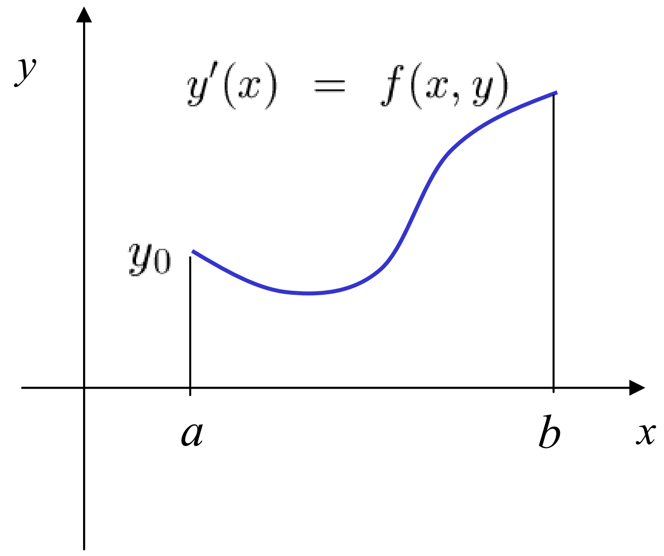
$$y'(x) = f(x,y)\,, \quad x \in [a,b]$$
$$y(x_0) = y_0$$

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

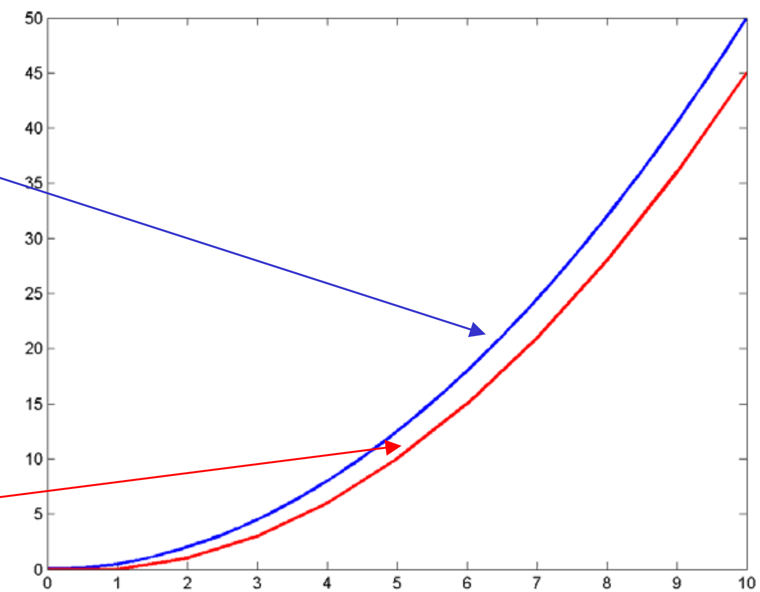
$$x_n = nh$$

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

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

$$\begin{aligned}
y' = f(x,y) \quad &\Rightarrow \quad y'(x_0) = f(x_0, y_0) \\
y'' = \frac{df(x,y)}{dx} \quad &= \quad f_x + f_y y' = f_x + f_y f \\
y''' = \frac{d^2 f(x,y)}{dx^2} \quad &= \quad f_{xx} + f_{xy} f + f_{yx} f + f_{yy} f^2 + f_y f_x + f_y^2 f \\
&= \quad f_{xx} + 2f_{xy} + f_{yy} f^2 + f_x f_y + f_y^2 f
\end{aligned}$$

Partial Derivatives

$$\begin{aligned}
f_x &= \frac{\partial}{\partial x} \\
f_y &= \frac{\partial}{\partial y}
\end{aligned}$$

Truncate series to k terms

$$\begin{aligned}
y_1 = y(x_1) &= y_0 + h y'(x_0) + \frac{h^2}{2!} y''(x_0) + \cdots + \frac{h^k}{k!} y^{(k)}(x_0) \\
y_2 = y(x_2) &= y_1 + h y'(x_1) + \frac{h^2}{2!} y''(x_1) + \cdots + \frac{h^k}{k!} y^{(k)}(x_1) \\
&\cdot \\
y_n = y(x_n) &= y_{n-1} + h y'(x_{n-1}) + \frac{h^2}{2!} y''(x_{n-1}) + \cdots + \frac{h^k}{k!} y^{(k)}(x_{n-1})
\end{aligned}$$

Choose Step Size h

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

Discretization

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

Recursion Algorithm

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

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

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

$$y(0.01) \simeq y1 = y_0 + hf(x_0, y_0) = 1 + 0.01 \cdot 1 = 1.01$$

$$y(0.02) \simeq y2 = y_1 + hf(x_1, y_1) = 1.01 + 0.01 \cdot 1.01 = 1.021$$

$$y(0.03) \simeq y3 = y_2 + hf(x_2, y_2) = 1.021 + 0.01 \cdot 1.021 = 1.03121$$

$$y(0.03) = 1.0305$$

Error Analysis?

Initial Value Problems
Taylor Series Methods
Error Analysis

Error Estimates and Convergence

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

Euler's Method

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

$$x_n = x_0 + nh$$

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

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

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

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

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

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

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

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

Derivative Bounds

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

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

$$\eta_{n+1} = (1 + hL)\eta_n + \frac{h^2}{2}Y\ ,\quad y_0 = 0$$

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

$$\eta_n \le |e_n|$$

$$n = 0\ :\ \xi_0 = 0\ ,\quad e_0 = 0$$

$$n = k\ :\quad \eta_k \le |e_n|$$

$$\eta_{k+1} = (1 + hL)\eta_k + \frac{h^2}{2}Y \le (1 + hL)|e_k| + \frac{h^2}{2}Y \le |e_{k+1}|$$

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

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

O(h)

Error Analysis

Example – Euler's Method

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

Exact solution

$$y = e^x$$

Derivative Bounds

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

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

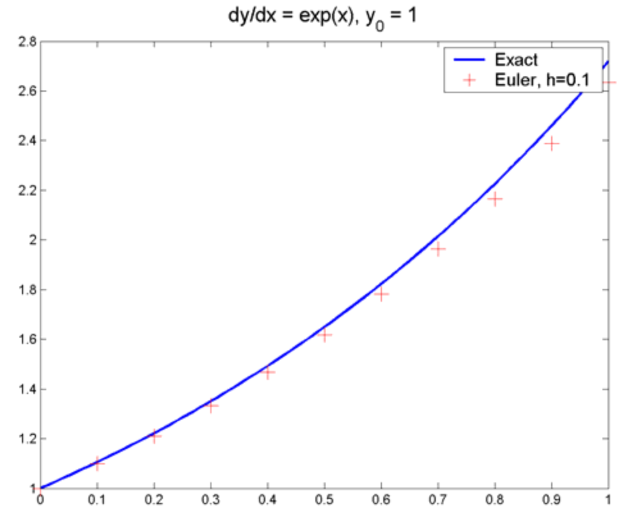
Error Bound

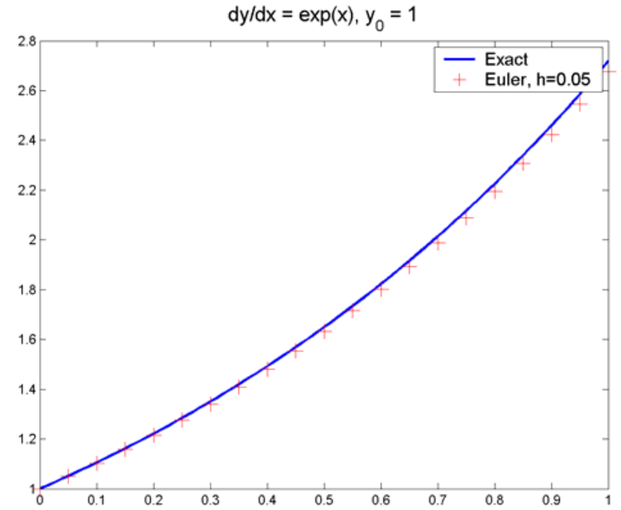
$$|e_n| \leq \frac{he}{2}(e - 1)$$

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

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

$$\begin{aligned} y_{11} &= 2.5937 \\ y(1) &= 2.71828 \\ e_{11} &= 0.1246 < 0.24 \end{aligned}$$





Initial Value Problems
Runge-Kutta Methods

Taylor Series Recursion

$$y(x_{n+1}) = y(x_n) + hf(x_n, y_n) + \frac{h^2}{2}(f_x + ff_y)_n$$
$$+\frac{h^3}{6}(f_{xx} + 2ff_{xy} + f_{yy}f^2 + f_xf_y + f_y^2f)_n + O(h^4)$$

Runge-Kutta Recursion

$$y_{n+1} = y_n + ak_1 + bk_2$$
$$k_1 = hf(x_n, y_n)$$
$$k_2 = hf(x_n + \alpha h, y_n + \beta k_1)$$

Match ***a***,***b***,α,β to match Taylor series amap.

$$\frac{k_2}{h} = f(x_n + \alpha h, y_n + \beta k_1)$$
$$= f(x_n, y_n) + \alpha h f_x + \beta k_1 f_y$$
$$+\frac{\alpha^2 h^2}{2} f_{xx} + \alpha h \beta k_1 f_{xy} + \frac{\beta^2}{2} f^2 f_{yy}) + O(h^4)$$

Substitute k_2 in Runge Kutta

$$y_{n+1} = y_n + (a+b)hf + bh^2(\alpha f_x + \beta f f_y)$$
$$+bh^3(\frac{\alpha^2}{2} f_{xx} + \alpha\beta f f_{xy} + \frac{\beta^2}{2} f^2 f_{yy}) + O(h^4)$$

Match **2nd order Taylor series**

$$\left.\begin{aligned} a+b &= 1 \\ b\alpha &= 1/2 \\ b\beta &= 1/2 \end{aligned}\right\} \Leftarrow a = b = 0.5\,, \quad \alpha = \beta = 1$$

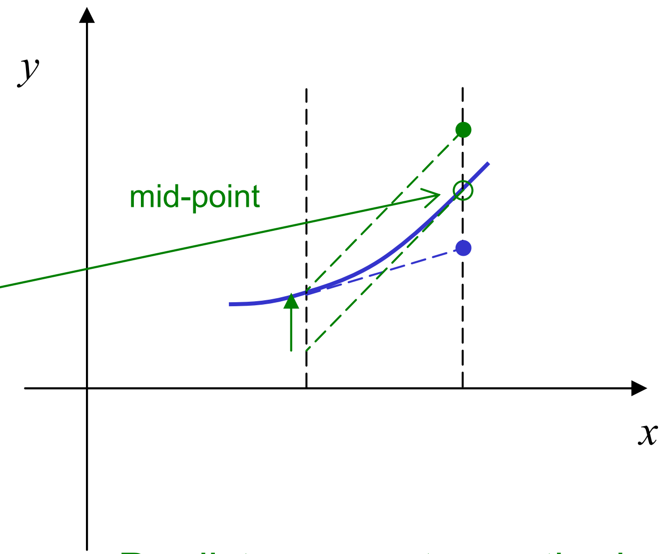
Initial Value Problems
Runge-Kutta Methods

Initial Value Problem

$$y' = f(x,y)$$
$$y(x_0) = y_0$$
$$x_n = x_0 + nh$$

2nd Order Runge-Kutta

$$y_{n+1} = y_n + \frac{1}{2}(k_1 + k_2)$$
$$k_1 = hf(x_n, y_n)$$
$$k_2 = hf(x_n + h, y_n + k_1)$$



Predictor-corrector method

4th Order Runge-Kutta

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$
$$k_1 = hf(x_n, y_n)$$
$$k_2 = hf(x_n + \frac{h}{2}, y_n + \frac{k_1}{2})$$
$$k_3 = hf(x_n + \frac{h}{2}, y_n + \frac{k_2}{2})$$
$$k_4 = hf(x_n + h, y_n + k_3)$$

Initial Value Problems
Runge-Kutta Methods

Euler's Method

$$x_n = nh$$

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

Recurrence

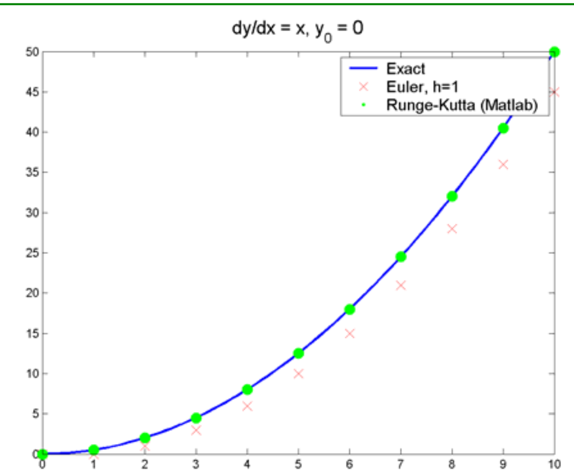
$$y_{n+1} = y_n + hf(nh, y_n)$$

4th Order Runge-Kutta

$$\begin{aligned}
y_{n+1} &= y_n + \tfrac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\
k_1 &= hf(x_n, y_n) \\
k_2 &= hf(x_n + \tfrac{h}{2}, y_n + \tfrac{k_1}{2}) \\
k_3 &= hf(x_n + \tfrac{h}{2}, y_n + \tfrac{k_2}{2}) \\
k_4 &= hf(x_n + h, y_n + k_3)
\end{aligned}$$

rk.m

```
h=1.0;
x=[0:0.1*h:10];
y0=0;
y=0.5*x.^2+y0;
figure(1); hold off
a=plot(x,y,'b'); set(a,'Linewidth',2);
% Euler's method, forward finite difference
xt=[0:h:10]; N=length(xt);
yt=zeros(N,1); yt(1)=y0;
for n=2:N
    yt(n)=yt(n-1)+h*xt(n-1);
end
hold on; a=plot(xt,yt,'xr'); set(a,'MarkerSize',12);
% Runge Kutta
fxy='x'; f=inline(fxy,'x','y');
[xrk,yrk]=ode45(f,xt,y0);
a=plot(xrk,yrk,'.g'); set(a,'MarkerSize',30);
a=title(['dy/dx = ' fxy ', y_0 = ' num2str(y0)])
set(a,'FontSize',16);
b=legend('Exact',['Euler, h=' num2str(h)],
'Runge-Kutta (Matlab)'); set(b,'FontSize',14);
```



Matlab ode45 automatically ensures convergence

Matlab inefficient for large problems –> Convergence Analysis