Ordinary Differential Equations

Initial-Values Problems

1

Initial-Values Problems for ODE

$$\frac{dy}{dx} = f(x,y) \quad , a \leq x \leq b , \qquad y(a) = \alpha .$$

- Euler's Method
- Higher – Order Taylor Methods
- Runge – Kutta Methods

2

Euler's Method

$$\frac{dy}{dx} = f(x, y) \quad , a \le x \le b \, , \qquad y(a) = \alpha$$

- Selecting the mesh points:
  $x_i = a + ih \qquad$ for each $i = 0, 1, 2, \ldots \, N$.
- Step size: $h = \frac{b-a}{N} = x_{i+1} - x_i$
- Taylor's Theorem:

$$y(x_{i+1}) = y(x_i) + (x_{i+1} - x_i) y'(x_i) + \frac{(x_{i+1} - x_i)^2}{2} y''(\vartheta_i) \, ,$$
$$\vartheta_i \in (x_i, x_{i+1})$$

$$\rightarrow y(x_{i+1}) = y(x_i) + h f(x_i, y(x_i)) + \frac{h^2}{2} y''(\vartheta_i)$$

3

Euler's Method

$$y(x_{i+1}) \approx y(x_i) + h f(x_i, y(x_i))$$

- $y_i = y(x_i)$
- $y_0 = \alpha$
- $y_{i+1} = y_i + h f(x_i, y_i), \qquad$ for each $i = 0, 1, 2, \ldots \, N$

4

Euler's Method - Algorithm

$$y(x_{i+1}) = y(x_i) + hf(x_i, y(x_i)) + \frac{h^2}{2} y''(\vartheta_i)$$

- Approximating the solution of the initial-value problem
$$y' = f(x,y) \quad , a \le x \le b \, , \qquad y(a) = \alpha$$
  at $N + 1$ equally spaced numbers in the interval $[a, b]$:
- INPUT: a, b; integer N; initial condition α
- OUTPUT: y at the $(N + 1)$ values of x.

5

Euler's Method - Algorithm

INPUT: a, b; integer N; initial condition α
OUTPUT: y at the $(N + 1)$ values of x.

- Step 1 Set $h = (b - a)/N;$
  $x = a;$
  $y = \alpha;$
  OUTPUT (x, y).
- Step 2 For $i = 1, 2, \ldots, N$ do Steps 3, 4.
  Step 3 Set $y = y + hf\,(x, y);$ (*Compute* y_i)
  $x = a + ih.$ (*Compute* x_i)
  Step 4 OUTPUT (x, y).
- *Step 5* STOP

6

Thực hành nhỏ

- PT vi phân:

$$y' = y - x^2 + 1\,, 0 \le x \le 2\,, y(0) = 0.5$$

- Dùng EM để giải PT trên. Vẽ kết quả.
- Biết nghiệm giải tích: $y(x) = (x+1)^2 - 0.5e^x$. Hãy vẽ đồng thời kết quả số và kết quả từ nghiệm giải tích trên cùng đồ thị.


```
# Euler method Python code
# function
def f(x,y):
    return x+y
# or
# f = lambda x: x+y
# Euler method
def euler(x,y,b,N):
  h = (b-x)/N
  for i in range(N):
      y = y + h * f(x, y)
      x = x+h
      print('%.4f\t%.4f'% (x,y) )
      print('------------------------')
```

```
# Input
print('Enter initial conditions:')
a = float(input('a = '))
y0 = float(input('y0 = '))
print('Enter final point: ')
b = float(input('b = '))
print('Enter number of steps:')
N = int(input('N = ‘))
# Euler method call
euler(a,y0,b,N)
```

```
# Euler method Python code
# function
def f(x,y):
    return x+y    # your f(x,y)
# or
# f = lambda x: x+y
# Euler method
def euler(x,y,h,N):
for i in range(N):
        y = y + h * f(x, y)
        x = x+h
        print('%.4f\t%.4f'% (x,y) )
        print('-----------------------')
```

```
# Input
a = 0
y0 = 1.0
b = 1.0
N = 100
h = (b-a)/N
# Euler method call
euler(a,y0,h,N)
```

10

Higher-Order Taylor Methods

- Khai triển Taylor đến các số hạng bậc cao cho
- $y(x_{i+1}) = y(x_i) + hF^{(n)}\big(x_i, y(x_i)\big)$
- $F^{(n)}\big(x_i, y(x_i)\big) = f(x_i, y(x_i)) + \frac{h}{2} f'(x_i, y(x_i)) + \cdots + \frac{h^{n-1}}{n!} f^{(n-1)}(x_i, y(x_i))$
- Hoặc đặt $y(x_i) \equiv y_i$
- $y_{i+1} = y_i + hF^{(n)}(x_i, y_i)$
- $F^{(n)}(x_i, y_i) = f(x_i, y_i) + \frac{h}{2} f'(x_i, y_i) + \cdots + \frac{h^{n-1}}{n!} f^{(n-1)}(x_i, y_i)$
- Phải tính đạo hàm bậc cao!
- Euler method ứng với $n = 1$

12

Runge – Kutta Methods

$$\frac{dy}{dx} = f(x, y) \quad , a \leq x \leq b \,, \qquad y(a) = \alpha$$

- Midpoint Method / Runge-Kutta Methods of Order Two [RK2]
- Modified Euler Method
- Heun Method / Runge-Kutta Methods of Order Three [RK3]
- Higher-Order Runge-Kutta Methods: RK4

13

Runge – Kutta Methods

Midpoint Method / RK2:

- $y_i \equiv y(x_i)$
- $y_0 = \alpha$
- $y_{i+1} = y_i + hf\left(x_i + \frac{h}{2}, y_i + \frac{h}{2} f(x_i, y_i)\right)$, for each $i = 0, 1, 2, \ldots\ N - 1$

Modified Euler Method:

- $y_0 = \alpha$
- $y_{i+1} = y_i + \frac{h}{2}[f(x_i, y_i) + f(x_{i+1}, y_i + hf(x_i, y_i))]$,
  for each $i = 0, 1, 2, \ldots\ N - 1$

14

Runge – Kutta Methods

Heun Method [RK3] $\left(O\left(h^3\right)\right)$

- $y_0 = \alpha$

$$y_{i+1} = y_i + \frac{h}{4}\left[f(x_i, y_i) + 3f\left(x_i + \frac{2h}{3}, y_i + \frac{2h}{3} f\left(x_i + \frac{h}{3}, y_i + \frac{h}{3} f(x_i, y_i)\right)\right)\right]$$

for each $i = 0, 1, 2, \ldots\ N - 1$

$$k_1 = hf(x_i, y_i)$$
$$k_2 = hf\left(x_i + \frac{h}{3}, y_i + \frac{1}{3}k_1\right)$$
$$k_3 = hf\left(x_i + \frac{2h}{3}, y_i + \frac{2}{3}k_2\right)$$
$$y_{i+1} = y_i + \frac{1}{4}[k_1 + 3k_3]$$

16

Runge – Kutta Methods

Runge – Kutta Order Four [RK4] $\left(O\left(h^4\right)\right)$

- $y_0 = \alpha$
- $k_1 = hf(x_i, y_i)$
- $k_2 = hf\left(x_i + \frac{h}{2}, y_i + \frac{1}{2}k_1\right)$
- $k_3 = hf\left(x_i + \frac{h}{2}, y_i + \frac{1}{2}k_2\right)$
- $k_4 = hf(x_{i+1}, y_i + k_3)$

$y_{i+1} = y_i + \frac{1}{6}[k_1 + 2k_2 + 2k_3 + k_4]$ for each $i = 0, 1, 2, \ldots\ N - 1$

17

RK4 - Algorithm

INPUT: a, b; integer N; initial condition α
OUTPUT: y at the $(N + 1)$ values of x

- Step 1 Set $h = (b - a)/N$;
  $x = a$;
  $y = \alpha$;
  OUTPUT (x, y).
- Step 2 For $i = 1, 2, \ldots, N$ do Steps 3 – 5
- *Step 3* $k_1 = hf(x_i, y_i)$
  $k_2 = hf\left(x_i + \frac{h}{2}, y_i + \frac{1}{2}k_1\right)$
  $k_3 = hf\left(x_i + \frac{h}{2}, y_i + \frac{1}{2}k_2\right)$
  $k_4 = hf(x_{i+1}, y_i + k_3)$
- *Step 4 Set* $y = y + \frac{1}{6}[k_1 + 2k_2 + 2k_3 + k_4]$ (*Compute* y_i)
  $x = a + ih$ (*Compute* x_i)
  Step 5 OUTPUT (x, y).
- *Step 6* STOP

18

Thực hành

- Cho PT vi phân:
  $$y' = y - x^2 + 1\,, 0 \le x \le 2\,, y(0) = 0.5$$
- Dùng PP Midpoint, Modified Euler, RK3, RK4 để giải PT trên. Xuất dữ liệu ra file theo kiểu bảng. Vẽ kết quả.
- [Nghiệm giải tích: $y(x) = (x+1)^2 - 0.5e^x$]

20

Higher-Order Differential Equations [HODE] & Systems of Differential Equations [SoDE]

22

Systems of Differential Equations [SoDE]

$$\begin{cases} \dfrac{dy_1}{dx} = f_1(x, y_1, \dots, y_m) \\ \dfrac{dy_2}{dx} = f_2(x, y_1, \dots, y_m) \\ \vdots \\ \dfrac{dy_m}{dx} = f_m(x, y_1, \dots, y_m) \end{cases}$$

$$a \leq x \leq b\,, y_j(a) = \alpha_j \ \ , j = 1, \dots, m$$

23

Euler Method for SoDE

- $y_{0,j} = \alpha_j$ for $j = 1, \dots, m$ [j : chỉ số chạy cho số phương trình của hệ (m)]

  $y_{i+1,j} = y_{i,j} + hf_j(x_i, y_{i,1}, y_{i,2}, \dots, y_{i,m})$ for each $j = 1, \dots, m$ [$*$]
- for each $i = 0, 1, 2, \dots N-1$ [i: chỉ số chạy theo khoảng chia thời gian]

24

Euler's Method - Algorithm

PT vi phân

$$y_{i+1} = y_i + hf(x_i, y_i)$$

INPUT: a, b; integer N; initial condition α
OUTPUT: y at the $(N + 1)$ values of x.

- Step 1 Set $h = (b-a)/N$;
  $x = a$;
  $y = \alpha$;
  OUTPUT (x, y).
- Step 2 For $i = 1, 2, \dots, N$ do Steps 3, 4.
  Step 3 Set $y = y + hf(x, y)$; (Compute y_i)
  $x = a + ih$. (Compute x_i)
  Step 4 OUTPUT (x, y).
- Step 5 STOP

Hệ PT vi phân (m phương trình)

$$y_{i+1,j} = y_{i,j} + hf_j(x_i, y_{i,1}, \dots, y_{i,m}); j = 1, 2, \dots, m$$

INPUT: a, b; integer N; initial condition α_j
OUTPUT: y_j at the $(N + 1)$ values of x.

- Step 1 Set $h = (b-a)/N$;
  $x = a$;
  For $j = 1, \dots, m$: $y_j = \alpha_j$;
  OUTPUT $(x, y_1, \dots, y_m)$.
- Step 2 For $i = 1, 2, \dots, N$ do Steps 3 – 4
  Step 3 For $j = 1, \dots, m$: Set $y_j = y_j + hf_j(x_i, y_{i,1}, \dots, y_{i,m})$
  $x = a + ih$
  Step 4 OUTPUT $(x, y_1, \dots, y_m)$.
- Step 5 STOP

25

"SoDE" dạng ma trận/ mảng

- Có thể viết hệ PT vi phân dưới dạng ma trận như sau

$$\frac{d\boldsymbol{y}}{dx} = \boldsymbol{f}(x, \boldsymbol{y}) \quad , \qquad a \le x \le b\,, \boldsymbol{y}(a) = \boldsymbol{\alpha}$$

- Cũng rất nên "ma trận hoá", tức tạo mảng / array cho y và f trong code
- Gọi $\boldsymbol{y}$ và $\boldsymbol{f}$ là mảng [trong trường hợp này là mảng 1 chiều, tương tự 1 vector hoặc như 1 ma trận cột] với các thành phần $\{y_j\}$ (viết đầy đủ là $y_j(x)$, hoặc $y_j(x_i)$ nếu xét đến bước thời gian thứ i) và $\{f_j\}$ một cách tương ứng. PT [*] có thể được viết gọn thành

$$\boldsymbol{y}_{i+1} = \boldsymbol{y}_i + h\boldsymbol{f}(x_i, \boldsymbol{y}_i) \quad \text{or} \quad \boldsymbol{y} = \boldsymbol{y} + h\boldsymbol{f}(x, \boldsymbol{y})$$

26

Euler's Method - Algorithm

PT vi phân

$$y_{i+1} = y_i + hf(x_i, y_i)$$

INPUT: a, b; integer N; initial condition α
OUTPUT: y at the $(N + 1)$ values of x.

- Step 1 Set $h = (b - a)/N$;
  $x = a$;
  $y = \alpha$;
  OUTPUT (x, y).
- Step 2 For $i = 1, 2, \ldots, N$ do Steps 3, 4.
  Step 3 Set $y = y + hf(x, y)$; (Compute y_i)
  $x = a + ih$. (Compute x_i)
  Step 4 OUTPUT (x, y).
- Step 5 STOP

Hệ PT vi phân (m phương trình) [dạng mảng]

$$\boldsymbol{y}_{i+1} = \boldsymbol{y}_i + h\boldsymbol{f}(x_i, \boldsymbol{y}_i)$$

INPUT: a, b; integer N; initial condition $\boldsymbol{\alpha}$
OUTPUT: $\boldsymbol{y}$ at the $(N + 1)$ values of x.

- Step 1 Set $h = (b - a)/N$;
  $x = a$;
  $\boldsymbol{y} = \boldsymbol{\alpha}$;
  OUTPUT $(x, \boldsymbol{y})$.
- Step 2 For $i = 1, 2, \ldots, N$ do Steps 3 – 4
  Step 3 Set $\boldsymbol{y} = \boldsymbol{y} + h\boldsymbol{f}(x, \boldsymbol{y})$
  $x = a + ih$
  Step 4 OUTPUT $(x, \boldsymbol{y})$.
- Step 5 STOP

Dùng ma trận/ mảng [màu đậm] cho $\boldsymbol{y}$ và $\boldsymbol{f}$ → code của SoDE có cấu trúc cơ bản giống hệt trường hợp 01 PT vi phân!

27

RK4 for SoDE

- $y_{0,j} = \alpha_j$ for $j = 1, \ldots, m$
- $k_{1,j} = hf_j\left(x_i, y_{i,1}, y_{i,2}, \ldots, y_{i,m}\right)$ for each $j = 1, \ldots, m$
- $k_{2,j} = hf_j\left(x_i + \frac{h}{2}, y_{i,1} + \frac{1}{2}k_{1,1}, y_{i,2} + \frac{1}{2}k_{1,2}, \ldots, y_{i,m} + \frac{1}{2}k_{1,m}\right)$ for each $j = 1, \ldots, m$
- $k_{3,j} = hf_j\left(x_i + \frac{h}{2}, y_{i,1} + \frac{1}{2}k_{2,1}, y_{i,2} + \frac{1}{2}k_{2,2}, \ldots, y_{i,m} + \frac{1}{2}k_{2,m}\right)$ for each $j = 1, \ldots, m$
- $k_{4,j} = hf_j\left(x_{i+1}, y_{i,1} + k_{3,1}, y_{i,1} + k_{3,2}, \ldots, y_{i,m} + k_{3,m}\right)$ for each $j = 1, \ldots, m$

$$y_{i+1,j} = y_{i,j} + \frac{1}{6}\left[k_{1,j} + 2k_{2,j} + 2k_{3,j} + k_{4,j}\right] \text{ for each } j = 1, \ldots, m$$

for each $i = 0, 1, 2, \ldots N - 1$

RK4 – Algorithm SoDE

INPUT: a, b; integer N; initial conditions α_j
OUTPUT: y_j at the $(N + 1)$ values of x

- Step 1 Set $h = (b - a)/N$;
  $x = a$;
  For $j = 1, 2, \ldots, m$: $y_j = \alpha_j$;
  OUTPUT (x, y_j).
- Step 2 For $i = 1, 2, \ldots, N$ do Steps 3 – 5
- *Step 3* For $j = 1, \ldots, m$: $k_{1,j} = hf_j\left(x_i, y_{i,1}, y_{i,2}, \ldots, y_{i,m}\right)$
  For $j = 1, \ldots, m$: $k_{2,j} = hf_j\left(x_i + \frac{h}{2}, y_{i,1} + \frac{1}{2}k_{1,1}, y_{i,2} + \frac{1}{2}k_{1,2}, \ldots, y_{i,m} + \frac{1}{2}k_{1,m}\right)$
  For $j = 1, \ldots, m$: $k_{3,j} = hf_j\left(x_i + \frac{h}{2}, y_{i,1} + \frac{1}{2}k_{2,1}, y_{i,2} + \frac{1}{2}k_{2,2}, \ldots, y_{i,m} + \frac{1}{2}k_{2,m}\right)$
  For $j = 1, \ldots, m$: $k_{4,j} = hf_j\left(x_{i+1}, y_{i,1} + k_{3,1}, y_{i,2} + k_{3,2}, \ldots, y_{i,m} + k_{3,m}\right)$
- *Step 4* For $j = 1, \ldots, m$: *Set* $y_j = y_j + \frac{1}{6}\left[k_{1,j} + 2k_{2,j} + 2k_{3,j} + k_{4,j}\right]$ *(Compute $y_{i,j}$)*
  $x = a + ih$ *(Compute x_i)*
  Step 5 OUTPUT (x, y_j).
- *Step 6* STOP

RK4 for SoDE [dạng ma trận/ mảng]

- $\boldsymbol{y}_0 = \boldsymbol{\alpha}$;
- $\boldsymbol{k}_1 = h\boldsymbol{f}(x_i, \boldsymbol{y}_i)$
- $\boldsymbol{k}_2 = h\boldsymbol{f}\left(x_i + \frac{h}{2}, \boldsymbol{y}_i + \frac{1}{2}\boldsymbol{k}_1\right)$
- $\boldsymbol{k}_3 = h\boldsymbol{f}\left(x_i + \frac{h}{2}, \boldsymbol{y}_i + \frac{1}{2}\boldsymbol{k}_2\right)$
- $\boldsymbol{k}_4 = h\boldsymbol{f}(x_i + h, \boldsymbol{y}_i + \boldsymbol{k}_3)$

$$\boldsymbol{y}_{i+1} = \boldsymbol{y}_i + \frac{1}{6}[\boldsymbol{k}_1 + 2\boldsymbol{k}_2 + 2\boldsymbol{k}_3 + \boldsymbol{k}_4]$$

for each $i = 0, 1, 2, \ldots\ N - 1$

Hoặc kiểu viết khác $\Leftrightarrow$

- $\boldsymbol{y} = \boldsymbol{y}_0 = \boldsymbol{\alpha}$;
- $\boldsymbol{k}_1 = h\boldsymbol{f}(x, \boldsymbol{y})$
- $\boldsymbol{k}_2 = h\boldsymbol{f}\left(x + \frac{h}{2}, \boldsymbol{y} + \frac{1}{2}\boldsymbol{k}_1\right)$
- $\boldsymbol{k}_3 = h\boldsymbol{f}\left(x + \frac{h}{2}, \boldsymbol{y} + \frac{1}{2}\boldsymbol{k}_2\right)$
- $\boldsymbol{k}_4 = h\boldsymbol{f}(x + h, \boldsymbol{y} + \boldsymbol{k}_3)$

$$\boldsymbol{y} = \boldsymbol{y} + \frac{1}{6}[\boldsymbol{k}_1 + 2\boldsymbol{k}_2 + 2\boldsymbol{k}_3 + \boldsymbol{k}_4]$$

for each $i = 0, 1, 2, \ldots\ N - 1$

31

RK4 – Algorithm SoDE [dùng ma trận/ mảng]

INPUT: a, b; integer N; initial conditions $\boldsymbol{\alpha}$

OUTPUT: $\boldsymbol{y}$ at the $(N + 1)$ values of x

- Step 1 Set $h = (b - a)/N$;
  $x = a$;
  $\boldsymbol{y} = \boldsymbol{\alpha}$;
  OUTPUT $(x, \boldsymbol{y})$.
- Step 2 For $i = 1, 2, \ldots, N$ do Steps 3 – 5
- Step 3 $\boldsymbol{k}_1 = h\boldsymbol{f}(x, \boldsymbol{y})$
  $\boldsymbol{k}_2 = h\boldsymbol{f}\left(x + \frac{h}{2}, \boldsymbol{y} + \frac{1}{2}\boldsymbol{k}_1\right)$
  $\boldsymbol{k}_3 = h\boldsymbol{f}\left(x + \frac{h}{2}, \boldsymbol{y} + \frac{1}{2}\boldsymbol{k}_2\right)$
  $\boldsymbol{k}_4 = h\boldsymbol{f}(x + h, \boldsymbol{y} + \boldsymbol{k}_3)$
- Step 4 *Set* $\boldsymbol{y} = \boldsymbol{y} + \frac{1}{6}[\boldsymbol{k}_1 + 2\boldsymbol{k}_2 + 2\boldsymbol{k}_3 + \boldsymbol{k}_4]$ (*Compute* $\boldsymbol{y}$)
  $x = a + ih$ (*Compute* x_i)
  Step 5 OUTPUT $(x, \boldsymbol{y})$.
- Step 6 STOP

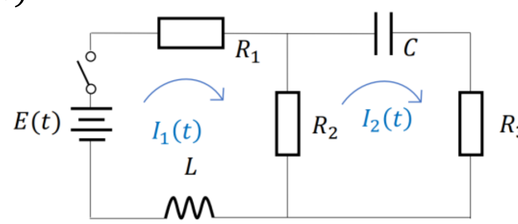
Nếu sử dụng ma trận/ mảng [màu đậm] cho $\boldsymbol{y}$ và $\boldsymbol{f}$, tức là cần khai báo mảng khi lập trình, thì code RK4 của SoDE có cấu trúc cơ bản giống y trường hợp 01 PT vi phân!

32

"Kirchhoff's Law" cho mạch điện

- "Kirchhoff's Law" cho mạch điện kín: Dòng điện $I(t)$ trong mạch đóng gồm trở R, tụ C, cuộn dây L và nguồn thế $E(t)$ được cho bởi:

$$LI'(t) + RI(t) + \frac{1}{C}\int I(t)dt = E(t)$$



- Áp dụng Kirchhoff's Law cho mạch điện

$$E(t) = 12\text{V}, L = 2\text{H}, C = 0.5\text{F},$$
$$R_1 = 2\Omega, R_2 = 6\Omega, R_3 = 4\Omega.$$

- Ta được hệ phương trình vi tích phân:

$$2I_1'(t) + 2I_1(t) + 6[I_1(t) - I_2(t)] = 12$$

$$\frac{1}{0.5}\int I_2(t)dt + 4I_2(t) + 6[I_2(t) - I_1(t)] = 0$$

34

"Kirchhoff's Law" cho mạch điện

$$2I_1'(t) + 2I_1(t) + 6[I_1(t) - I_2(t)] = 12$$

$$\frac{1}{0.5}\int I_2(t)dt + 4I_2(t) + 6[I_2(t) - I_1(t)] = 0$$

- Giả sử mạch điện đóng mạch tại thời điểm $t = 0$.
- Khi đó, $I_1(0) = I_2(0) = 0$. Xét $0 \leq t \leq 1$.
- Có thể biến đổi hệ PT vi tích phân thành hệ phương trình vi phân:

$$I_1'(t) = f_1(t, I_1, I_2) = -4I_1(t) + 3I_2(t) + 6$$
$$I_2'(t) = f_2(t, I_1, I_2) = -2.4I_1(t) + 1.6I_2(t) + 3.6$$

- Hãy giải hệ phương trình trên bằng phương pháp RK4 và MEM. Vẽ đồ thị $I_1(t)$ và $I_2(t)$.

36

"Kirchhoff's Law" cho mạch điện

- Hãy giải hệ phương trình trên bằng phương pháp RK4 và MEM. Vẽ đồ thị $I_1(t)$ và $I_2(t)$.
- So sánh (trên cùng đồ thị) với nghiệm giải tích

$$I_1(t) = -3.375e^{-2t} + 1.875e^{-0.4t} + 1.5$$
$$I_2(t) = -2.25e^{-2t} + 2.25e^{-0.4t}$$

38

Higher-Order Differential Equations

- Xét bài toán PT vi phân bậc cao

$$y^{(m)}(x) = f\left(x, y', y'', \dots, y^{(m-1)}\right) \ [*]$$

- với $a \le x \le b$ và các điều kiện ban đầu

$$y(a) = \alpha_1, y'(a) = \alpha_2, \dots, y^{(m-1)}(a) = \alpha_m$$

- Đặt $u_1(x) = y(x), u_2(x) = y'(x), \dots, u_m(x) = y^{(m-1)}(x)$. Khi đó PT [*] thành

$$\frac{du_1}{dx} = \frac{dy}{dx} = u_2(x), \frac{du_2}{dx} = \frac{dy'}{dx} = u_3(x), \dots, \frac{du_{m-1}}{dx} = \frac{dy^{(m-2)}}{dx} = u_m(x)$$
$$\frac{du_m}{dx} = \frac{dy^{(m-1)}}{dx} = f(x, u_1, u_2, \dots, u_m)$$

- với điều kiện ban đầu: $u_1(a) = \alpha_1, u_2(a) = \alpha_2, \dots, u_m(a) = \alpha_m$.

39

HODE → SoDE

PT vi phân bậc cao [HODE]

$$y^{(m)}(x) = f\left(x, y', y'', \ldots, y^{(m-1)}\right)$$

với $a \le x \le b$ và đk ban đầu

$$y(a) = \alpha_1,$$
$$y'(a) = \alpha_2, \ldots,$$
$$y^{(m-1)}(a) = \alpha_m$$

$$u_1(x) \equiv y(x),$$
$$u_2(x) \equiv y'(x),$$
$$\ldots$$
$$u_m(x) \equiv y^{(m-1)}(x)$$

Hệ PT vi phân [SoDE]

$$\frac{du_1}{dx} = u_2(x),$$
$$\frac{du_2}{dx} = u_3(x),$$
$$\ldots$$
$$\frac{du_{m-1}}{dx} = u_m(x),$$
$$\frac{du_m}{dx} = f(x, u_1, u_2, \ldots, u_m)$$

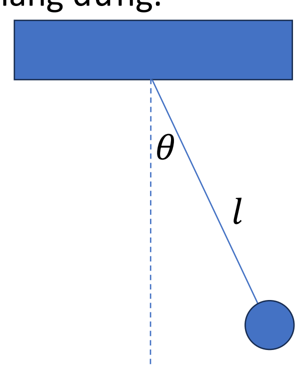
với $u_1(a) = \alpha_1, u_2(a) = \alpha_2, \ldots,$
$u_m(a) = \alpha_m$. $a \le x \le b$.

40

Thực hành: Bài toán Con lắc [Pendulum]

- Con lắc có dây dài l dao động tạo góc θ so với phương thẳng đứng.
- PT chuyển động của con lắc [một cách gần đúng]:

$$\frac{d^2\theta}{dt^2} + \frac{g}{l}\sin\theta = 0 \quad [1]$$



- Tại thời điểm ban đầu $\theta(t_0) = \theta_0$
  và vận tốc góc là $\theta'(t_0) = \theta_0'$.
- Đối với góc quay đủ nhỏ, PT [1] có dạng

$$\frac{d^2\theta}{dt^2} + \frac{g}{l}\theta = 0 \quad [2]$$

- Áp dụng RK4 để giải PT [1], [2]. Giả sử $\theta_0 = \pi/6$, $\theta_0' = 0$; $l = 60cm$.
  So sánh [vẽ đồ thị $\theta(t)$] các kết quả.

41

Thực hành: Bài toán “đạn pháo/ ném xiên”

- Hãy xét bài toán đạn pháo/ném xiên quen thuộc bạn đã biết. Chuyển đổi bài toán thành hệ phương trình vi phân. Viết ra hệ phương trình vi phân này.
- Áp dụng RK4 để giải hệ PT. Hãy tự chọn các điều kiện ban đầu một cách hợp lý.
- Vẽ quỹ đạo của đạn pháo theo các góc bắn khác nhau.
- Hãy xét thêm lực cản [ma sát/ frictional] $\boldsymbol{f}$. Giả sử $\boldsymbol{f}$ [đậm≡vector] có dạng:

$$\boldsymbol{f} = -km|v|^n \frac{\boldsymbol{v}}{|v|}$$

- $-\frac{\boldsymbol{v}}{|v|}$ nghĩa là lực ma sát này luôn ngược chiều với [vector] vận tốc. $n = 1$ cho trường hợp vận tốc nhỏ; $n = 3/2$ cho vận tốc trung bình; và $n = 2$ vận tốc cao.
- Dẫn ra hệ PT vi phân cho đạn pháo với lực cản này. Giải PT và vẽ đồ thị. Xem xét 03 trường hợp của n. Giả sử $k = 0.8$.

42