MATLAB ỨNG DỤNG

TS. NGUYỄN HÒAI SƠN

KHOA XÂY DỰNG & CƠ HỌC ỨNG DỤNG

2006

Chương 1

MATLAB CĂN BẢN

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

I. BIỂU THỨC (EXPRESSION)

- Biến số (variables)
- Số (Numbers)
- Toán tử (Operaters)
- Hàm (Functions)

Biến (Variables)

- tối đa 19 ký tự có nghĩa
- phân biệt giữa chữ hoa và chữ thường.
- bắt đầu bằng một từ theo sau là từ hay số hoặc dấu (_).
- biến toàn cục (global) tác dụng trong tòan chương trình.
- biến cục bộ (local) tác dụng trong nội tại hàm (function)
- một số biến đặc biệt: pi, ans,...

❖ Kiểm tra biến (who và whos)
❖ Xóa biến (clear và clear all)

Ví dụ

```
>> clear a
>> clear b degree
>> a
undefined function or variable
```

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

1. Số (Numbers)

Tất cả những con số đều được lưu kiểu định dạng (*format*)
Dùng hàm format để định dạng kiểu số:

format (định dạng)

```
>> b=3/26;
>> format long; b
b =
   0.11538461538462
>> format short e; b
b =
  1.1538e-001
>> format bank; b
b =
          0.12
>> format short eng; b
b =
   115.3846e-003
>> format hex; b
b =
   3fbd89d89d89d89e
```

```
>> format +; b
b =
+
>> format rat; b
b =
       3/26
>> format short; b
b =
    0.1154
>> format long eng; b
b =
    115.384615384615e-003>>
```

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

2. Toán tử (operaters) *(+, -, *, /, \,^,')*

```
>> 2*4+2
ans =
10
>> (2+sqrt(-1))^2
ans =
3.0000 + 4.0000i
```

MATLAB

- Các biến không cần khai báo trước.
- Các ký tự thường và in là phân biệt.
- Kết thúc câu lệnh với ";" không hiển thị kết qủa câu lệnh.
- Biến mặc nhiên "ans"

```
>> rayon = 1e-1;
>> surface = pi * rayon * rayon
surface =
0.0314
```

```
>> volume= 4*pi*rayon^3/3;
volume =
0.0042
```

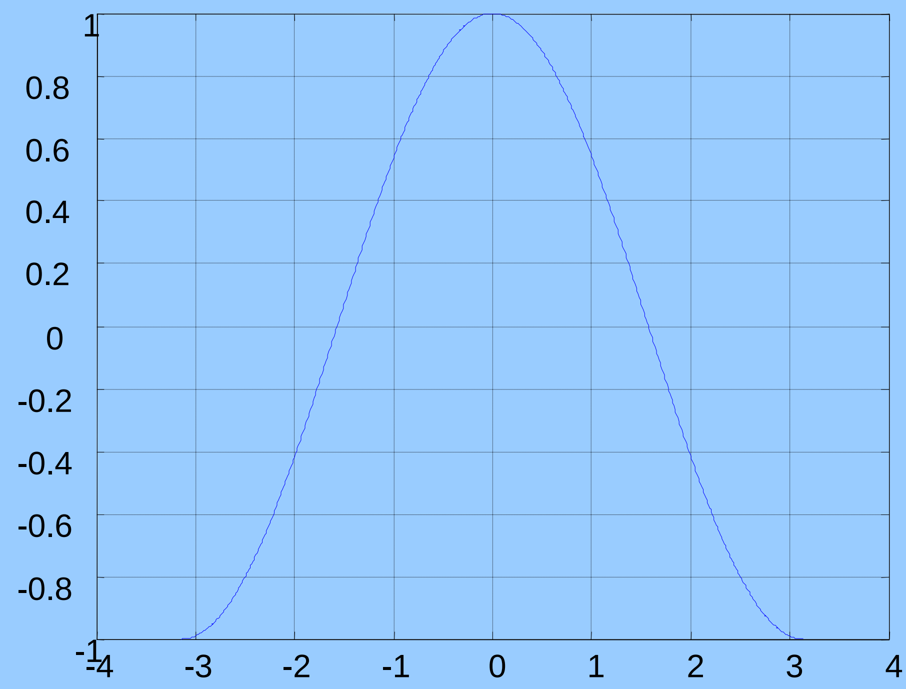
MATLAB CĂN BẢN

3. Hàm cơ bản (basis functions) abs, sqrt, exp, sin,...

$$\cos(x+iy) = \cos(x)\cosh(y) - i\sin(x)\sinh(y)$$

$$\cos(z) = \frac{e^{iz} + e^{-iz}}{2}$$

```
>> x=-pi:0.01:pi;
>> plot(x,cos(x); grid on
```



$$z = x + i*y \rightarrow \log(z) = \log(abs(z)) + a\tan 2(y,x)*i$$

```
>> abs(log(-1))
ans
3.1416
```

$$z = x + i*y \rightarrow r = abs(z);\ theta = a\tan 2(y,x) = a\tan(y/x)$$

```
>> z = 4 + 3i;
>> r = abs(z)
>> theta = atan2(imag(z),real(z))
r =
    5
theta =
   0.6435
```

```
>> z=r*exp(theta*i)
z=
4.0000+3.0000i
```

TS Nguyễn Hoài Sơn

4. Ưu tiên các phép toán

```
>> a=2; b=3; c=4;
>> a*b^c
ans =
162
>> (a*b)^c
ans =
1296
```

5. Tạo , lưu và mở tập tin (fprintf, save, fscanf, load, fopen, fclose...)

```
x = 0:.1:1; y = [x; exp(x)];
fid = fopen('exp.txt','w');
fprintf(fid,'%6.2f  %12.8f\n',y);
fclose(fid);
```

→

```
0.00    1.00000000
0.10    1.10517092
           ...
1.00    2.71828183
```

Chương trình chính

```
clear all; clc
file_dulieu
load dulieu, A
```

Chương trình con

```
function file_dulieu
A=[1 2 3;4 5 6;7 8 9];
save dulieu A
```

→

```
A =
   1   2   3
   4   5   6
   7   8   9
```

TS Nguyễn Hoài Sơn

6. Hàm xử lý số (fix, floor, ceil, round, sign, sort...)

- fix: làm tròn về 0

>> a=[1.25,-4.54,6.5,-7.1];
>> fix(a)
ans =
1 -4 6 -7

- floor: làm tròn về âm vô cùng

>> a=[1.25,-4.54,6.5,-7.1];
>> floor(a)
ans =
1 -5 6 -8

- ceil: làm tròn về dương vô cùng

>> a=[1.25,-4.54,6.5,-7.1];
>> ceil(a)
ans =
2 -4 7 -7

- round: làm tròn

>> a=[1.25,-4.54,6.5,-7.1];
>> round(a)
ans =
1 -5 7 -7

- **sign: hàm dấu với giá trị đơn vị**

>> a=[1.25,-4.54,6.5,-7.1];
>> sign(a)
ans =
1 -1 1 -1

- sort: sắp xếp từ nhỏ đến lớn

>> a=[1.25,-4.54,6.5,-7.1];
>> sort(a)
ans =
-7.1000 -4.5400 1.2500 6.5000

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

II. MA TRẬN VÀ VECTƠ "[...;...;...]"

```
>> A = [ 1, 2, 3; 4, 5, 6; 7, 8, 10]
    A  =
          1 2 3
          4 5 6
          7 8 10
>> A(3,3) = 17
    A =
          1 2 3
          4 5 6
          7 8 17
>> vec = [10; 0; 1000]
    vec =
        10
        0
        1000
>> A'
    ans =
          1 4 7
          2 5 8
          3 6 17
```

- ";" có nghĩa là chuyển sang hàng kế tiếp.
- "," hay " " phân cách giữa các phần tử.

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

```
>> t = 1:5
t =
1 2 3 4 5
>> row = A(1,:)
row =
1 2 3
>> col = A(:,1)
col =
1
4
7
>> 1:0.3:2
ans =
1 1.3000 1.6000 1.9000
>> tt = t(:)
tt =
1
2
3
4
5
```

- “:” có nghĩa là tất cả.
- “:” từ giá trị này tới giá trị khác.
- “:” từ giá trị này tới giá trị khác bước bao nhiêu.

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

- Ma trận phức.

```
>> b = [4; 5-15*i; -5;2+i];
>> abs(b)
   ans =
   4.0000
   15.8114
    5.0000
    2.2361
>> conj(b)
ans =
   4.0000
   5.0000 +15.0000i
  -5.0000
   2.0000 - 1.0000i
>> real(b)
ans =
     4
     5
    -5
     2
```

```
>> imag(b)
ans =
     0
   -15
     0
     1
>> angle(b)
ans =
         0
   -1.2490
    3.1416
    0.4636
>> atan2(imag(b),real(b))
ans =
         0
   -1.2490
    3.1416
    0.4636
```

MATLAB CĂN BẢN

- Hàm tạo ma trận đặc biệt.
- zeros(n)
- zeros(m,n)
- zeros([m n])
- zeros(size(A))
- ones(n)
- ones(m,n)
- ones([m n])
- ones(size(A))
- eye(n)
- eye(m,n)
- eye(size(A))
- pascal
- magic
- numel(A)
- length(A)
- rand(m,n)
- diag(v,k), diag(v)
- tril, triu
- linspace(a,b), linspace(a,b,n)
- logspace(a,b,n)

```
>> A=zeros(3)
A =
    0    0    0
    0    0    0
    0    0    0
>> B=zeros(2,3)
B =
    0    0    0
    0    0    0
>> size(A)
ans =
    3    3
>> zeros(size(B))
ans =
    0    0    0
    0    0    0
>> numel(B)
ans =
    6
>> length(B)
ans =
    3
>> rand(3,2)
ans =
   0.9501    0.4860
   0.2311    0.8913
   0.6068    0.7621
```

```
>> C=ones(3)
C =
    1    1    1
    1    1    1
    1    1    1
>> D=eye(3)
D =
    1    0    0
    0    1    0
    0    0    1
>> eye(3,2)
ans =
    1    0
    0    1
    0    0
>> pascal(3)
ans =
    1    1    1
    1    2    3
    1    3    6
>> magic(3)
ans =
    8    1    6
    3    5    7
    4    9    2
```

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

```
>> diag([2 1 2],1)
ans =
     0     2     0     0
     0     0     1     0
     0     0     0     2
     0     0     0     0
>> diag(A)
ans =
     1
     5
     9
>> triu(A)
ans =
     1     2     3
     0     5     6
     0     0     9
>> tril(A)
ans =
     1     0     0
     4     5     0
     7     8     9
>> linspace(1,2,4)
ans =
    1.0000    1.3333    1.6667    2.0000
>> logspace(1,2,4)
ans =
   10.0000   21.5443   46.4159  100.0000
```

```
>> A=[1 2 3;4 5 6;7 8 9]
A =
     1     2     3
     4     5     6
     7     8     9
```

TS Nguyễn Hoài Sơn

III. CÁC PHÉP TÓAN TRÊN MA TRẬN VÀ VECTƠ

Phép tính	Chú thích
+, -	Cộng hoặc trừ hai ma trận cùng kích thước
A*B	Nhân hai ma trận A và B
A/B	Chia hai ma trận (chia phải) A và B
A\B	Chia trái hai ma trận B và A
A.*B	Nhân từng phần tử của hai ma trận A và B
A./B	Chia từng phần tử của hai ma trận A và B
A.\B	Chia từng phần tử của hai ma trận B và A
.^	Mũ cho từng phần tử của mảng

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

```
>> A=[1 2 3;4 5 6;7 8 9]
A =
     1     2     3
     4     5     6
     7     8     9
>> A(2,3)=10
A =
     1     2     3
     4     5    10
     7     8     9
>> B=A(2,1)
B =
     4
>> C=[-4 2 3;1 2 1;2 5 6]
C =
    -4     2     3
     1     2     1
     2     5     6
```

```
>> D=[A C]
D =
     1     2     3    -4     2     3
     4     5    10     1     2     1
     7     8     9     2     5     6
>> D(5)
ans =
     5
>> D(4,5)
??? Index exceeds matrix dimensions.
>> X=D
X =
     1     2     3    -4     2     3
     4     5    10     1     2     1
     7     8     9     2     5     6
>> X(2,6)
ans =
     1
>> X(2,:)
ans =
     4     5    10     1     2     1
```

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

```
X =
     1     2     3    -4     2     3
     4     5    10     1     2     1
     7     8     9     2     5     6
```

```
>> X(:,1)
ans =
     1
     4
     7
>> 1:5
ans =
     1     2     3     4     5
>> 30:-4:15
ans =
    30    26    22    18
>> X(2:3,:)
ans =
     4     5    10     1     2     1
     7     8     9     2     5     6
```

```
>> X(:,end)
ans =
     3
     1
     6
>> E=X([2 3],[1 3])
E =
     4    10
     7     9
>> X(2,end)
ans =
     1.
>> X(3,:)=[ ]
X =
     1     2     3    -4     2     3
     4     5    10     1     2     1
>> X(:,5)=[3 4]
X =
     1     2     3    -4     3     3
     4     5    10     1     4     1
>> X(2,:)
ans =
     4     5    10     1     2     1
```

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

```
>> C(4,:)=[8 4 6]
C =
   -4    2    3
    1    2    1
    2    5    6
    8    4    6
>> C(:,4)=[8 4 6 1]'
C =
   -4    2    3    8
    1    2    1    4
    2    5    6    6
    8    4    6    1
>> C=[C ones(4);zeros(4) eye(4)]
C =
   -4    2    3    8    1    1    1    1
    1    2    1    4    1    1    1    1
    2    5    6    6    1    1    1    1
    8    4    6    1    1    1    1    1
    0    0    0    0    1    0    0    0
    0    0    0    0    0    1    0    0
    0    0    0    0    0    0    1    0
    0    0    0    0    0    0    0    1
```

```
C =
   -4    2    3
    1    2    1
    2    5    6
```

- Ma trận phức.

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

- Hàm xử lý ma trận và vectơ (size, median, max, min, mean, sum, length,...)

```
>> size(C)
ans =
   3    3
>> mean(B)
ans =
   2.6667
>> sum(B)
ans =
   16
>> min(C)
ans =
  -4    2    1
>> sort(C)
ans =
  -4    2    1
   1    2    3
   2    5    6
```

```
C =
  -4    2    3
   1    2    1
   2    5    6
B =
   1    5    6   -5    7    2
```

II. Giải hệ phương trình tuyến tính và phi tuyến bằng hàm thư viện Matlab: solve

1. Hệ đại số tuyến tính A*x=b

```
>>clear all
>>clc
>>A=[1 3 6;2 7 8;0 3 9];
>>b=[10;9;8];
>>x=inv(A)*b        %(x=A\b)

 x =
    7.8571
   -3.1905
    1.9524
```

$$x_1 + 3x_2 + 6x_3 = 10$$
$$2x_1 + 7x_2 + 8x_3 = 9$$
$$0x_1 + 3x_2 + 9x_3 = 8$$

2. Hệ đại số tuyến tính A*x=b , solve

```
>>S=solve('x+3*y+6*z=10','2*x+7*y+8*z=9','3*y+9*z=8')
S =
   x: [1x1 sym]
   y: [1x1 sym]
   z: [1x1 sym]
>> eval([S.x S.y S.z])
ans =
   7.8571   -3.1905    1.9524
```

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

3. Hệ đại số tuyến tính A*x=b, LU decomposition

```
>> clear all
>> clc
>> [L,U]=lu(A)
L =
   0.5000   -0.1667    1.0000
   1.0000         0         0
        0    1.0000         0
U =
   2.0000    7.0000    8.0000
        0    3.0000    9.0000
        0         0    3.5000
>> x=U\(L\b)
x =
   7.8571
  -3.1905
   1.9524
>> x=inv(U)*inv(L)*b
x =
   7.8571
  -3.1905
   1.9524
```

$$
\begin{aligned}
x_1 + 3x_2 + 6x_3 &= 10\\
2x_1 + 7x_2 + 8x_3 &= 9\\
0x_1 + 3x_2 + 9x_3 &= 8
\end{aligned}
$$

TS Nguyễn Hoài Sơn

7. CÁC PHÉP TÓAN TRÊN ĐA THỨC $a_k x^k + \ldots + a_2 x^2 + a_1 x + a_0$

❖ *Tính giá trị đa thức*

```
> pol=[1,2,3,4]
pol =
1 2 3 4
> polyval(pol,-1)
ans =
2
```

❖ *Tìm nghiệm đa thức*

```
> pol=[1,2,3,4]
pol =
1 2 3 4
> roots(pol)
ans =
-1.6506+ 0.0000j
-0.1747+ 1.5469j
-0.1747- 1.5469j
```

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

❖ ***Nhân và chia đa thức***

Cho hai đa thức: $f_1 = s^2 + 7s + 12$ $f_2 = s^2 + 9$

Hãy tính $f_3 = f_1 * f_2$

```
> f1=[1 7 12];
> f2=[1 0 9];
>f3=conv(f1,f2)
f3= 1  7  21  63  108
```

$\rightarrow$ $\boxed{f_3 = s^4 + 7s^3 + 21s^2 + 63s + 108}$

Cho hai đa thức: $f_4 = s^4 + 9s^3 + 37s^2 + 81s + 52$ $f_5 = s^2 + 4s + 13$

Hãy tính $f_6 = f_4 / f_5$

```
> f4=[1 9 37 81 52];
> f5=[1 4 13];
>[f6 r]=deconv(f4,f5)
f6= 1 5 4
r= 0  0  0  0  0
```

$\rightarrow$ $\boxed{f_6 = s^2 + 5s + 4}$

r là phần dư của phép chia

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

❖ ***Tính đạo hàm đa thức: polyder(p)***

```
>> p=[2 0 9 1];
>> polyder(p);
```

ans =

6   0   9

❖ ***Phân rã đa thức***

$$\frac{b(s)}{a(s)} = \frac{r_1}{s-p_1} + \frac{r_2}{s-p_2} + \ldots + \frac{r_n}{s-p_n} + k_s$$

Phân rã đa thức: $F(s) = \dfrac{2s^3 + 9s + 1}{s^3 + s^2 + 4s + 4}$

```
> a=[2 0 9 1];
> b=[1 1 4 4];
> [r,p,k]=residue(a,b)
```

⟷ ***[b,a]=residue(r,p,k)***

```
r =
        0.0000   -0.2500i
        0.0000   +0.2500i
       -2.0000
p =
        0.0000   +2.0000i
        0.0000   -2.0000i
       -1.0000
K =
        2.0000
```

→

$$2 + \frac{-2}{s+1} + \frac{j0.25}{s+j2} + \frac{-j0.25}{s-j2} = 2 + \frac{-2}{s+1} + \frac{1}{s^2+4}$$

TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

$$\frac{-4+8s^{-1}}{1+6s^{-1}+8s^{-2}}$$

```
b = [-4 8];
a = [1 6 8];
[r,p,k] = residue(b,a)

r =
   -12
     8


p =
    -4
    -2


k =
     []
```

Biểu thức phân rã ?

❖ ***Phương pháp bình phương tối thiểu trong xử lý số liệu thực nghiệm***

> x=[1 3 10];
> y=[10 18 37];
> polyfit(x,y,1)
ans =
2.92537 8.01493

$\boxed{y = 2.92537\,x + 8.01493}$

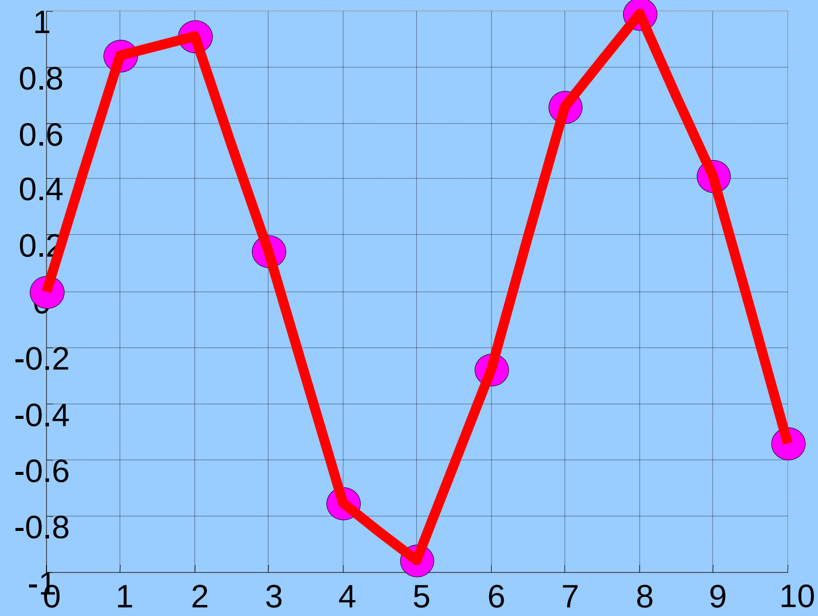
TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

8. Nội suy

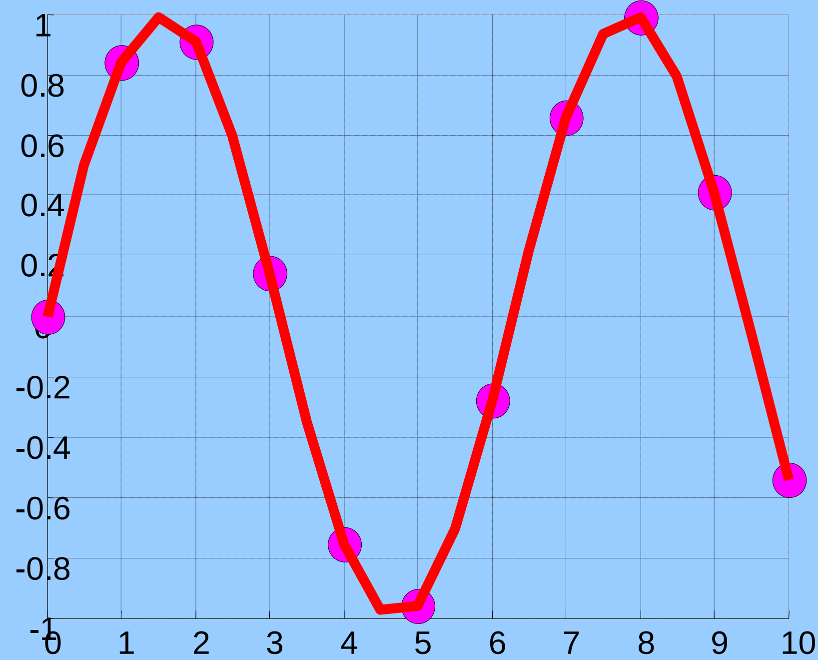
❖ ***Nội suy dữ liệu một chiều : interp1(x,y,xi)***

> x= 0 : 10 ;
> y = sin(x);
> xi= 0 : .5 : 10;
> yi= interp1(x,y,xi);



❖ ***Nội suy dữ liệu một chiều đa thức bậc ba : spline(x,y,xi)***

> x= 0 : 10 ;
> y = sin(x);
> xi= 0 : .5 : 10;
> yi= spline(x,y,xi);

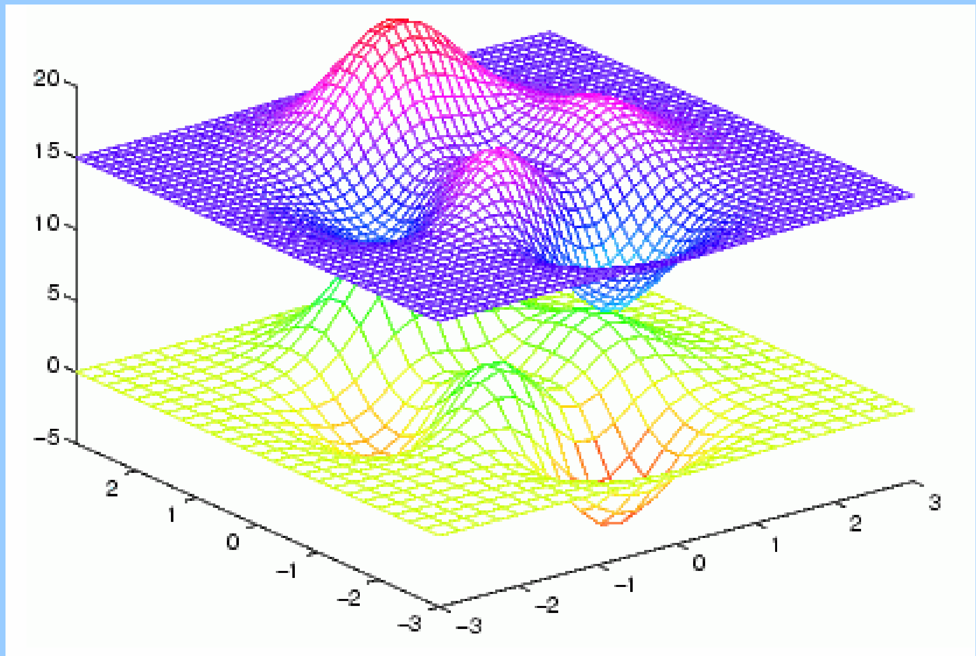


TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

❖ ***Nội suy dữ liệu hai chiều : interp2(x,y,z,xi,yi)***

> [x,y]= messhgrid(-3 : .25 : 3) ;
> z = peaks(x,y);
> [xi, yi]= messhgrid(-3 : .125 : 3) ;
> zi= interp2(x,y,z,xi,yi)
> hold on
> mesh(x,y,z), mesh(xi,yi,zi)



TS Nguyễn Hoài Sơn

9. Giải phương trình, hệ phương trình vi phân thường

❖ ***Hàm : dsolve(eq1,eq2,...,cond1,cond2,...,v)***

Ví dụ	Kết quả
dsolve('Dy = a*y')	exp(a*t)*C1
dsolve('Df = f + sin(t)')	-1/2*cos(t)-1/2*sin(t)+exp(t)*C1
dsolve('(Dy)^2 + y^2 = 1','s')	-sin(-s+C1)
dsolve('Dy = a*y', 'y(0) = b')	exp(a*t)*b
dsolve('D2y = -a^2*y',... 'y(0) = 1', 'Dy(pi/a) = 0')	cos(a*t)
dsolve('Dx = y', 'Dy = -x')	x = cos(t)*C1+sin(t)*C2 y = -sin(t)*C1+cos(t)*C2

TS Nguyễn Hoài Sơn

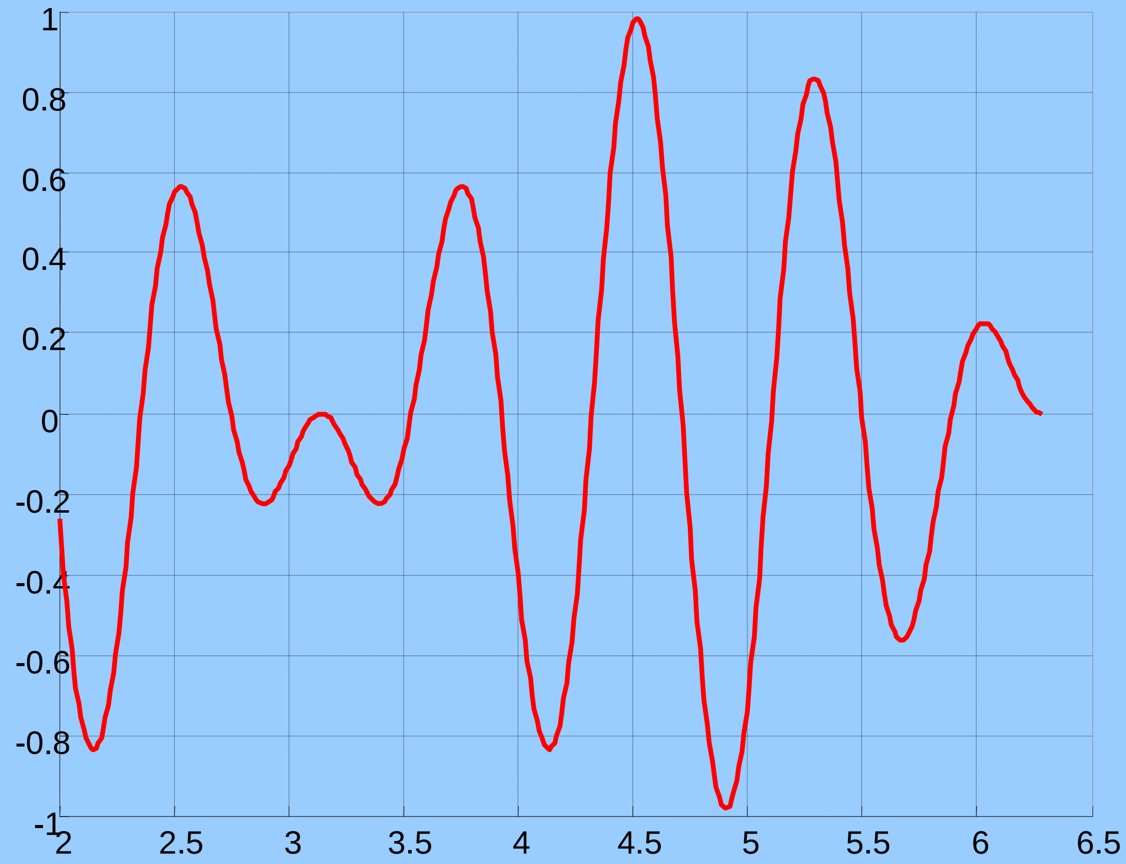
MATLAB CĂN BẢN

❖ ***Hàm : dsolve(eq1,eq2,…,cond1,cond2,…,v)***

Ví du: giải phương trình vi phân cấp hai $\ddot{y} + 81y = 16\cos(7t)$

Với điều kiện đầu $y(0) = 0,\ y^{'}(0) = 0$

>y= dsolve('D2y+81*y=16*cos(7*t)','y(0)=0','Dy(0)=0','t') ;
> t = linspace(0,2*pi,400);
>y= subs(y,t) ;
> plot(t,y)



TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

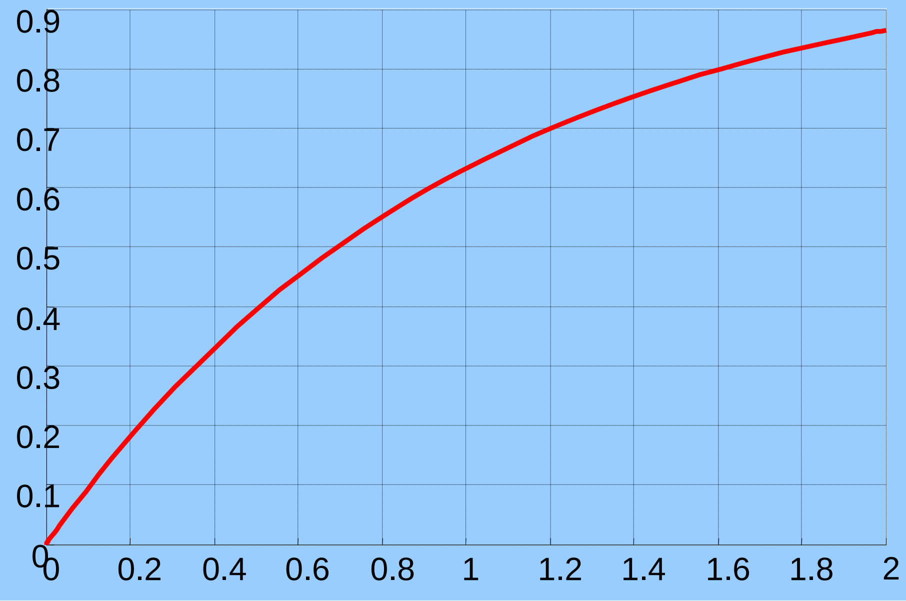
❖ ***Hàm : dsolve(eq1,eq2,...,cond1,cond2,...,v)***

Với solver tương ứng với ode45, ode32, ode113, ode15s, ode23s, ode23t, ode23tb

Cú pháp	[T,Y] = solver(odefun,tspan,y0)
Chú thích	**odefun** là hàm bên vế phải của phương trình $y' = f(t,y)$ **tspan** là khoảng lấy tích phân [t0 tf] để có được nghiệm tại những thời điểm xác định. tspan = [t0,t1,...,tf]. **y0** là vector điều kiện đầu.

Ví du: giải phương trình vi phân thường $y'(t) + y(t) = 1$ với $y(0) = 0$

```
> f=inline('1-y','t','y')
> [t, y]= ode45(f, [0  2],0) ;
> plot(t,y) ;
```



TS Nguyễn Hoài Sơn

MATLAB CĂN BẢN

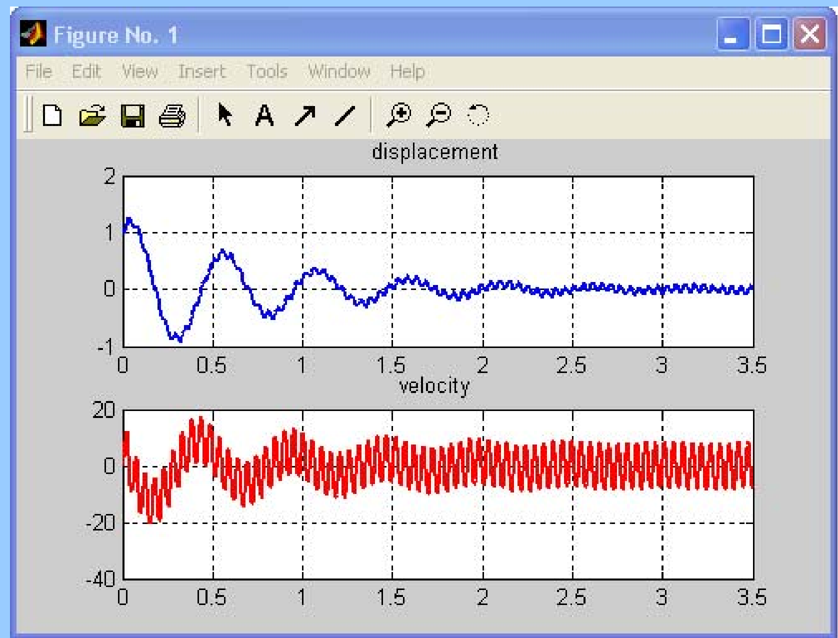
❖ ***Hàm : dsolve(eq1,eq2,...,cond1,cond2,...,v)***

Ví du: giải phương trình vi phân cấp hai $\ddot{y}(t)+B\dot{y}(t)+\Omega^2 y(t)=A_0\sin(\omega t)$

Đưa phương trình vi phân cấp hai về hệ hai phương trình vi phân cấp một

$$\begin{cases} \dot{y}_1 = y_2 \\ \dot{y}_2 = A_0\sin(\omega t) - By_2 - \Omega^2 y_1 \end{cases}$$

```
> y0=[1 0];
> tspan=[0 3.5];
> B=2.5; OME=150; ome=122; A0=1000;
> [t,y]=ode45('f',tspan,y0,[],B,OME,A0,ome)
> subplot(2,1,1), plot(t,y(:,1))
> subplot(2,1,2), plot(t,y(:,2))
> %%%%%%%%%%%%%%%%%%%%%
> function dy=f(t,y,flag,B,OME,A0,ome)
> dy= zeros(2,1) ;
> dy(1)=y(2);
> dy(2)=-B*y(2)-OME*y(1)+A0*sin(ome*t) ;
```



8. Lập trình với Matlab

Matlab cho phép lập trình theo hai hình thức: SCRIPTS và function

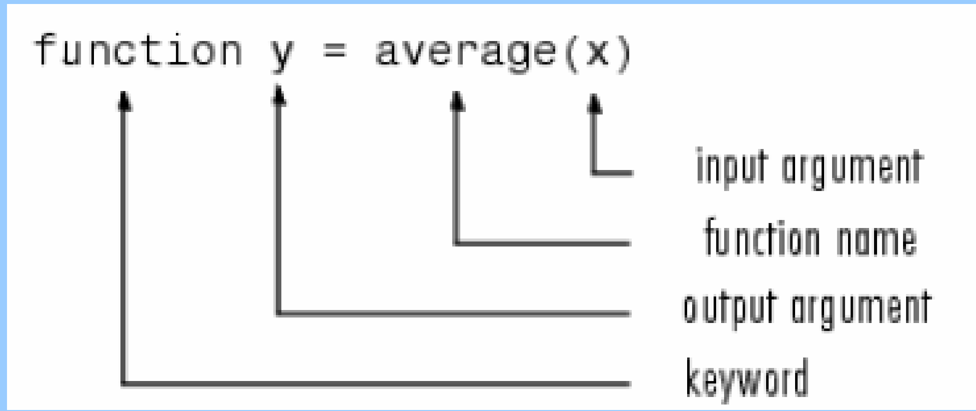
Scripts	Là hình thức đơn giản nhất của M-file, nó không có thông số vào và ra. Là tập hợp các lệnh và hàm của Matlab. Tất cả các biến tạo ra trong Scripts đều có thể sử dụng sau khi Scripts kết thúc.
	-- 	
	M-file: vidu.m
	x= 0:0.01:2*pi;
	y=sin(x);
	plot(x,y);

function	Là Scripts tuy nhiên có thêm đối số vào (input arguments) và đối số đầu ra (output argument). Tất cả các biến hoạt động trong một Workspace riêng. Biến trong function chỉ là biến cục bộ. -- M-file: doido.m function rad = doido(do) rad=do*pi/180;

TS Nguyễn Hoài Sơn

8. Lập trình với Matlab

Hình thức khai báo hàm



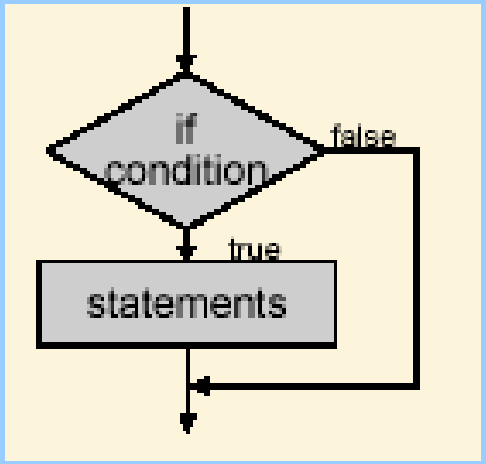
- Từ khoá function bắt buộc phải khai báo.
- Thông số đầu ra: nếu có nhiều giá trị trả về, các thông số này được đặt trong dấu “[ ]”. Nếu không có giá trị trả về ta có thể để trống hay để dấu [].
- Tên hàm
-Thông số đầu vào được khai báo trong dấu ()
- Biến toàn cục và địa phương

TS Nguyễn Hoài Sơn

8. Cấu trúc điều kiện

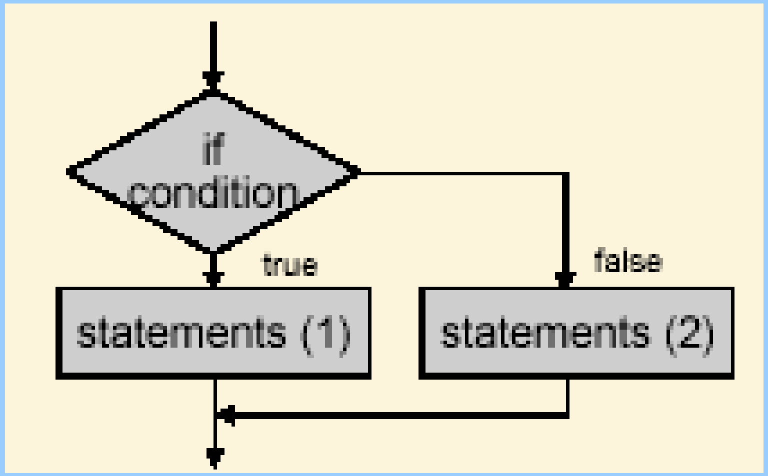
❖ Cấu trúc điều kiện: if

```
if    (biểu thức logic)
            nhóm lệnh
end
```



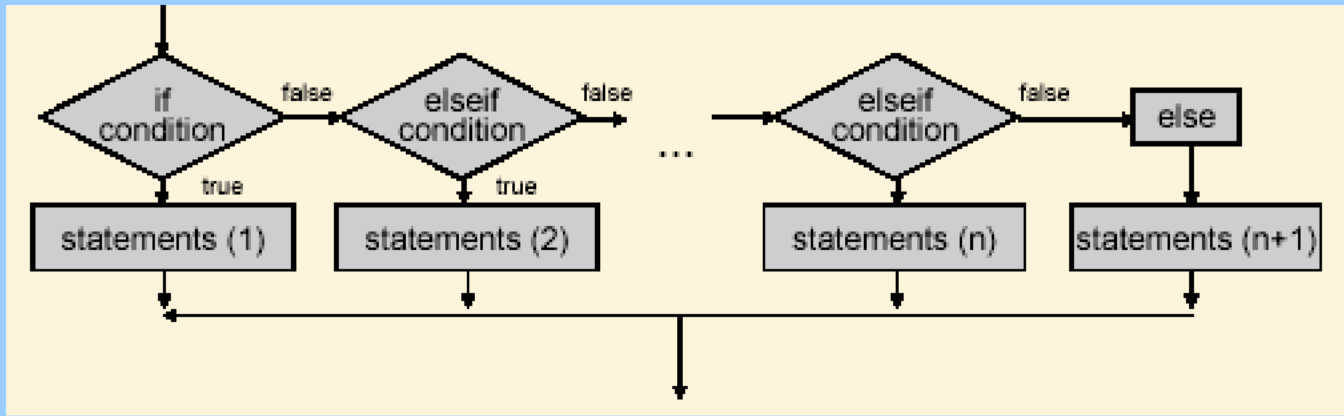
Toán tử	Ý nghĩa
<	Nhỏ hơn
<=	Nhỏ hơn hoặc bằng
>	Lớn hơn
>=	Lớn hơn hoặc bằng
==	Bằng nhau
~=	Không bằng

```
if    (biểu thức logic)
            nhóm lệnh 1
else
            nhóm lệnh 2
end
```



8. Cấu trúc điều kiện

❖ Cấu trúc điều kiện: if...end




```
if      (biểu thức logic)
            nhóm lệnh 1
elseif
            nhóm lệnh 2
else
            nhóm lệnh 3
end
```

Bài tập

h=(a-b)/n và xi = a+i*h tính tích phân của hàm f=cos(x)+sin(x) cho a=0,b=pi/3

Giải thuật

$$\int_a^b f(x)\,dx \approx \frac{h}{2}\left[f(x_0) + 2f(x_1) + 2f(x_2) + \cdots + 2f(x_{n-1}) + f(x_n)\right]$$

TS Nguyễn Hoài Sơn

8. Cấu trúc điều kiện

❖ Cấu trúc điều kiện: switch … case

```
switch       (biểu thức điều kiện)
case         (giá trị 1 biểu thức)
                 nhóm lệnh 1
 otherwise
                 nhóm lệnh 2
end
```

Ví dụ: tạo một menu lựa chọn

```
chon = input('Nhap vao lua chon cua ban, chon=  ')
Switch  chon
case 1
            disp('menu ve do thi ');
case 2
            disp('menu noi suy da thuc ');
otherwise
            disp('thoat khoi chuong trinh ');
end
```

Ví dụ: tạo một menu lựa chọn

```
fprintf(' \n');
fprintf('Select a case:\n');
fprintf('==============\n');
fprintf(' 1 - pi\n');
fprintf(' 2 - e \n');
fprintf(' 3 - i \n');
fprintf('==============\n');
n = input('');
switch n
case 1
disp('Pi = ');disp(pi);
case 2
disp('e = ');disp(exp(1));
case 3
disp('i = ');disp(i);
otherwise
disp('Nothing to display');
end
```

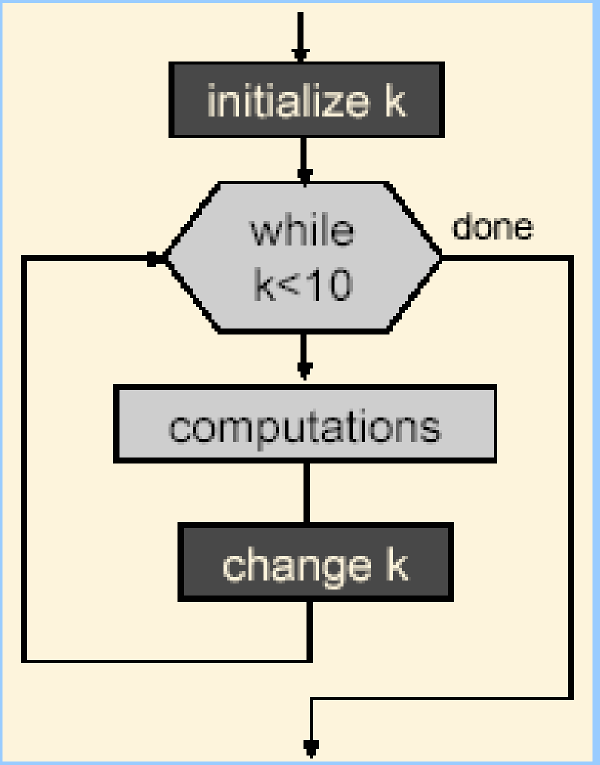
```
Select a case:
==============
 1 - pi
 2 - e
 3 - i
==============
1
Pi =
   3.1416
```

TS Nguyễn Hoài Sơn

8. Cấu trúc lặp có điều kiện

❖ Cấu trúc lặp có điều kiện: while

```
while (biểu thức điều kiện)
        nhóm lệnh
end
```



Ví dụ: yêu cầu nhập vào giá trị cho biến x.
việc nhập chỉ kết thúc khi x có giá dương

```
a= input('Nhap vao gia tri a:  ')
while a<=0
    disp('a lon hon khong ');
    a= input('Nhap vao gia tri a:  ')
end
```

Bài tập

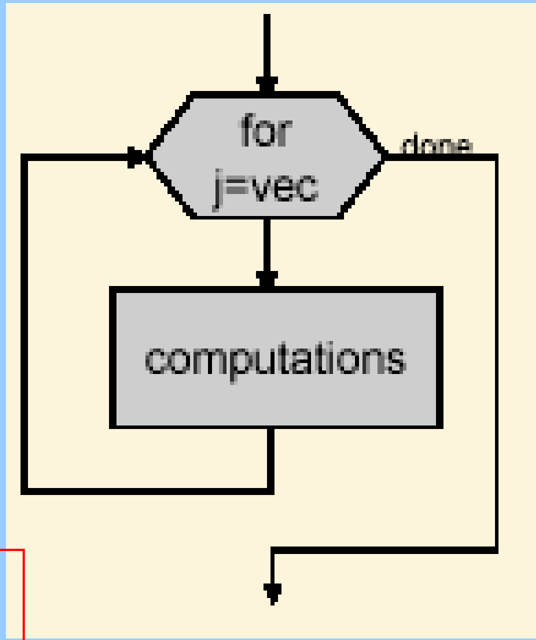
Tính tổng của chuỗi: $S(n) = \sum_{k=0}^{n} \frac{1}{k^2+1}$

TS Nguyễn Hoài Sơn

9. Cấu trúc lặp

❖ Cấu trúc lặp: for

```
for   biến =  biểu thức
            nhóm lệnh
end
```



Ví dụ: viết chương trình nhập vào mười giá trị cho biến A

```
for  i = 1 : 10
   tb=strcat(‘Nhap gia tri cho A(’,num2str(i),’) = ’);
   A(i)= input(‘’)
end
A
```

Bài tập

Viết hàm tính giá trị trung bình và độ lệch chuẩn của dữ liệu chứa trong vec tơ hàng x=[x1 x2 . . . xn] được định nghĩa theo công thức sau

$$x = \frac{1}{n}\sum_{i=1}^{n} x_i \;,\; s = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n-1}}$$

TS Nguyễn Hoài Sơn

PHƯƠNG TRÌNH VI PHÂN THƯỜNG

TS Nguyễn Hoài Sơn

NỘI DUNG:

Bài toán giá trị đầu :

- **Ví dụ định luật 2 Newton**
- **Phương pháp Euler**
- **Phương pháp điểm giữa**
- **Phương pháp Runge-Kutta**

Bài toán giá trị biên :

- **Phương trình vi phân cấp 2 :**
- **Phương trình vi phân cấp 4**

TS Nguyễn Hoài Sơn

Ví dụ định luật 2 Newton

1.1 Ví dụ định luật 2 Newton :

$$\vec{F} = m\vec{a}$$

Gia tốc là đạo hàm bậc 1 của vận tốc theo thời gian, do đó :

$$\frac{d\vec{v}}{dt} = \vec{a}$$

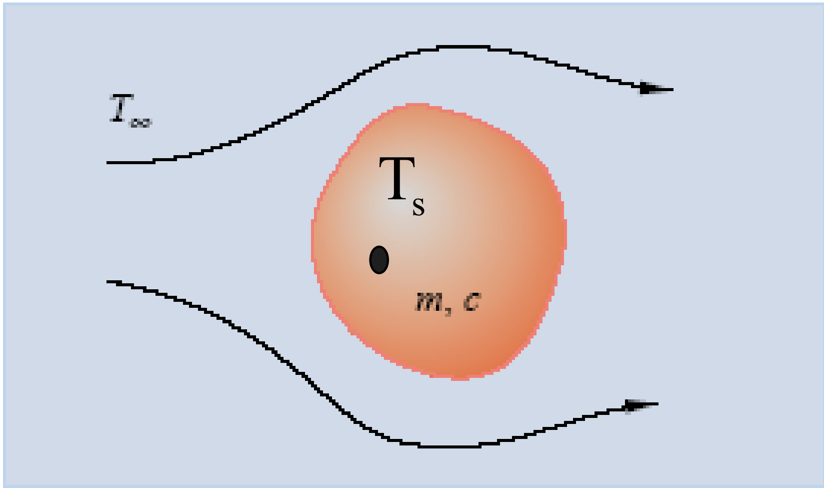
và

$$\frac{d\vec{v}}{dt} = \frac{\vec{F}}{m}$$

Minh họa:

Định luật 2 Newton cho một vật nóng bỏ vào trong môi trường chất lỏng. Sự thay đổi nhiệt độ theo thời gian của vật được mô tả bởi phương trình vi phân cân bằng năng lượng.

$$mc\frac{dT}{dt} = -Q$$



TS Nguyễn Hoài Sơn

Ví dụ định luật 2 Newton

Với nhiệt năng do làm lạnh:

$$Q = hA(T_s - T_\infty)$$

Giả sử vật liệu có tính cách nhiệt cao : => Ts = T

$$mc\frac{dT}{dt} = -hA(T - T_\infty) \qquad \text{hoặc} \qquad \frac{dT}{dt} = -\frac{hA}{mc}(T - T_\infty)$$

Ví dụ 1:

$$\frac{dy}{dt} = -y \qquad y(0) = y_0$$

Phương trình này có thể tích phân trực tiếp :

$$\frac{dy}{y} = -dt \qquad \ln\frac{y}{C_2} = -t$$

ln y = -t + C

ln y – $\ln C_2$ = -t

y = C2e-t

y = y0e–-t

TS Nguyễn Hoài Sơn

Ví dụ định luật 2 Newton

Tích phân số của các phương trình vi phân

Cho :

$$\frac{dy}{dt} = f(t, y); \qquad y(0) = y_0$$

Tìm kết quả chính xác tại giá trị t bất kì :

$$t_j = t_0 + jh$$

Với h là bước thời gian.

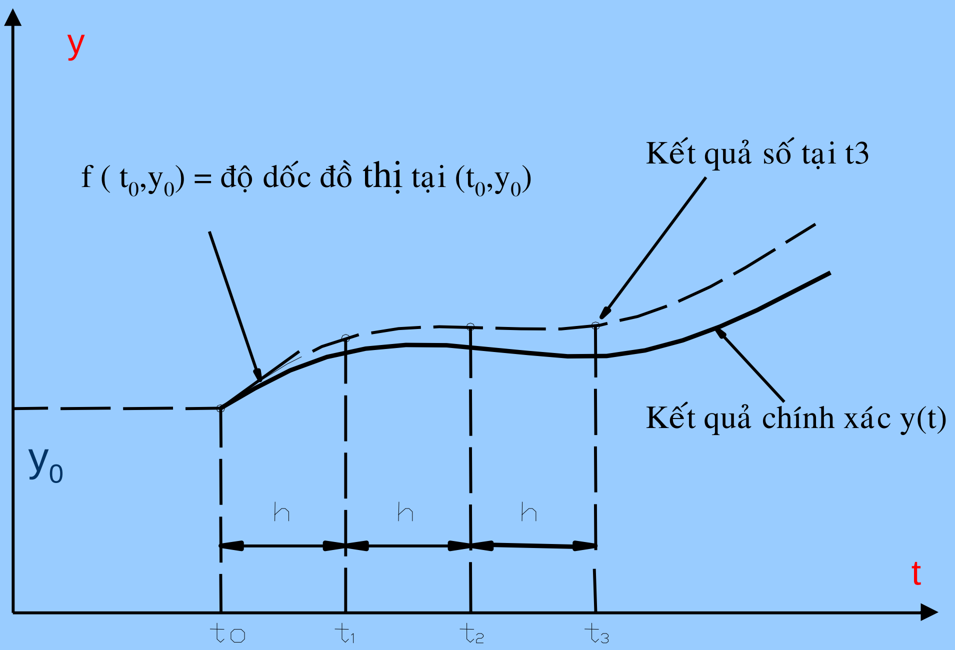
Gọi:

y(t) = kết quảchính xác

y(tj)= kết quả chính xác tại tj

yj = kết quả gần đúng tại tj

f(tj , yj) = kết quả gần đúng của hàm về phía phải tại t



TS Nguyễn Hoài Sơn

Phương pháp Euler

Cho h = t1 – t0 và điều kiện ban đầu, y = y(t0), tính :

$$y_1 = y_0 + hf(t_0, y_0)$$

$$y_2 = y_1 + hf(t_1, y_1)$$

$$\dots$$

$$y_{j+1} = y_j + hf(t_j, y_j)$$

Hoặc

$$y_j = y_{j-1} + hf(t_{j-1}, y_{j-1})$$

Ví dụ 2: Sử dụng phương pháp Euler để tính

$$\frac{dy}{dt} = t - 2y \qquad \text{y(0) = 1}$$

Kết quả chính xác là :

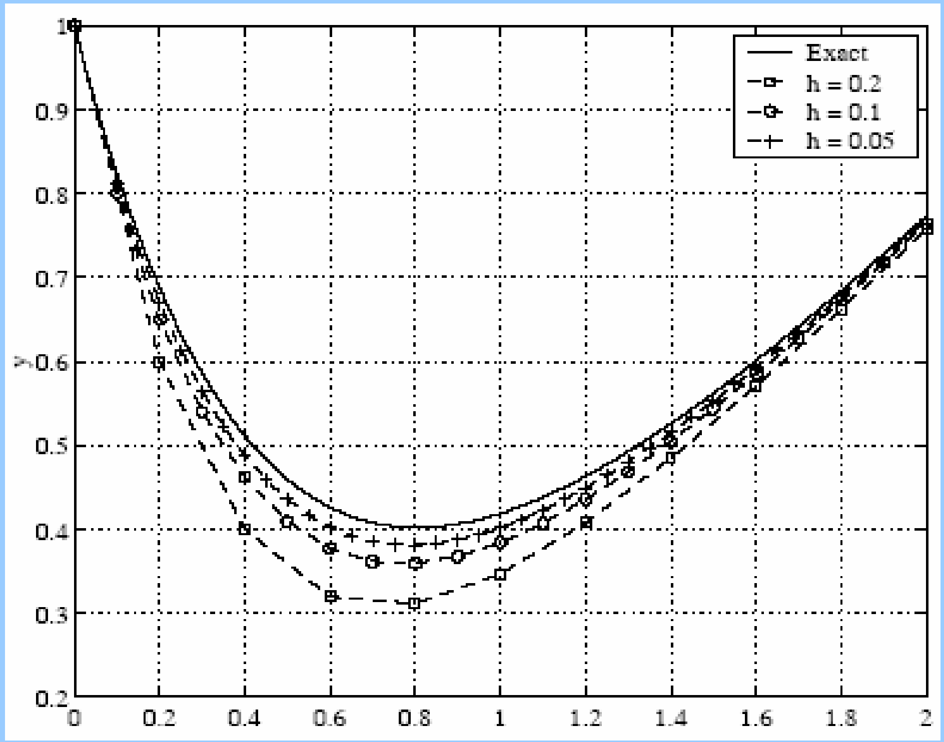
$$y_j = y_{j-1} + hf(t_{j-1}, y_{j-1}) \qquad y = \frac{1}{4}[2t - 1 + 5e^{-2t}]$$

TS Nguyễn Hoài Sơn

Phương pháp Euler

j	t_j	$f(t_{j-1}, y_{j-1})$	Euler $y_j = y_j + hf(t_{j-1}, y_{j-1})$	C.xác $y(t_j)$	Sai số $y_j - y(t_j)$
0	0.0	NaN	(Đk ban đầu) 1.000	1.000	0
1	0.2	0-(2)(1) = - 2.000	1.0 + (0.2)(-2.0) = 0.60	0.688	-0.0879
2	0.4	0.2 – (2)(0.6) = -1.000	0.6 + (0.2)(-1.0) = 0.40	0.512	-0.1117
3	0.6	0.4 – (2)(0.4) = -0.4	0.4 + (0.2)(- 0.4) = 0.32	0.427	-0.1065

So sánh với đồ thị :



Đối với h đã biết, sai số lớn nhất trong kết quả số đó là sai số rời rạc toàn cục

$$\max \left(\sum_j \left|(y_j - y(t_j))\right|\right)$$

TS Nguyễn Hoài Sơn

Phương pháp Euler

❖ **Đánh giá sai số :**

Sai số địa phương tại mỗi bước là:

$$e_j = y_j - y(t_j)$$

với y(tj) là kết quả chính xác tại tj

GDE = max(ej) j = 1, ...

h	max(e_j)
0.200	0.1117
0.100	0.0502
0.050	0.0240
0.025	0.0117

```
>> rhs = inline(‘cos(t)’,’t’,’y’) ;
>> [t,Y] = odeEuler(rhs,2*pi,0.01, 0) ;
>> plot(t,Y,’o’) ;
```

Giải bằng Matlab:

```
function [t,y] = odeEuler(diffeq,tn,h,y0)]
t = (0:h:tn)’;
n = length(t);
y = y0 + ones(n , 1);
for j = 2 : n
y(j) = y(j – 1) + h* feval(diffeq,t(j -1),y(j-1));
end
```

Phương pháp điểm giữa

Tăng mức độ chính xác bằng cách tính độ nghiêng 2 lần trong mỗi bước của h:

$$k_1 = f(t_j, y_j)$$

- Tính một giá trị của y tại điểm giữa :

$$y_{j+1/2} = y_j + \frac{h}{2} f\left(t_j, y_j\right)$$

- Đánh giá lại độ nghiêng

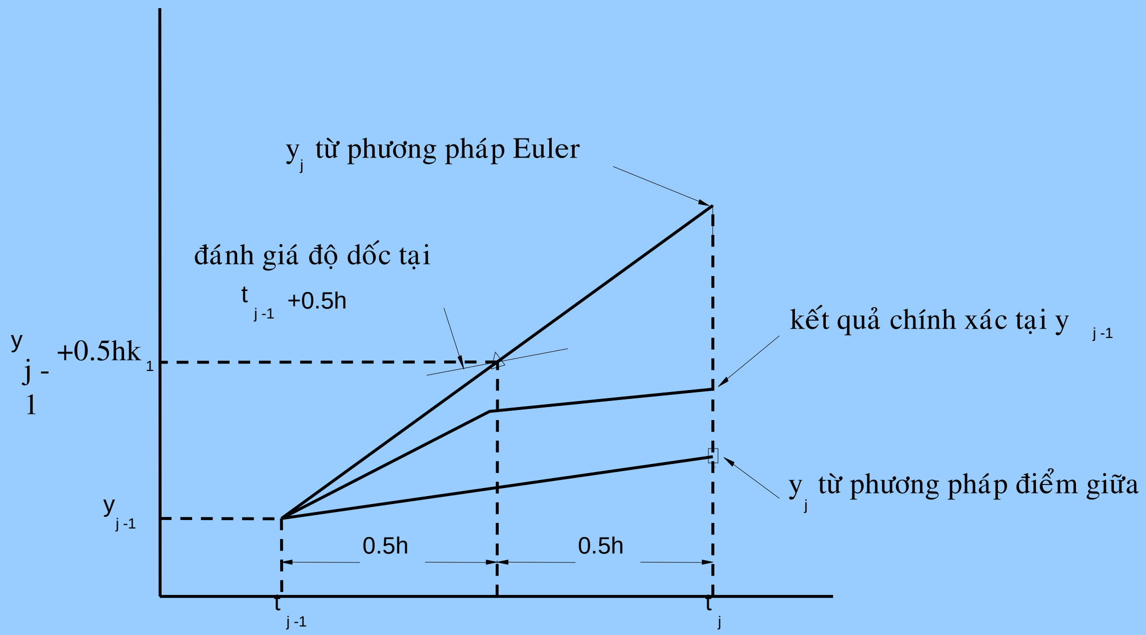
$$k_2 = f(t_j + \frac{h}{2}, y_j + \frac{h}{2} k_1)$$

- Tính giá trị cuối cùng của y

$$y_{j+1} = y_j + hk_2$$

TS Nguyễn Hoài Sơn

Phương pháp điểm giữa




```
>> rhs = inline('cos(t)','t','y') ;
>> [t,Y] = odeEuler(rhs,2*pi,0.01, 0) ;
>> plot(t,Y,'o') ;
```

Giải bằng Matlab:

```
function [t,y] = odeMidpt(diffeq,tn,h,y0)]
        t = (0:h:tn)';
        n = length(t) ;
        y = y0 + ones(n , 1) ;
        h2= h /2 ;
        for j = 2 : n
            k1 = feval(diffeq,t(j -1),y(j-1)) ;
            k2 = feval(diffeq,t(j -1)+h2,y(j-1)+h2*k1) ;
                        y(j) = y(j – 1) + h* k2 ;
        end
```

TS Nguyễn Hoài Sơn

Phương pháp điểm giữa

- So sánh phương pháp Midpoint với phương pháp Euler

Giải:

$$\frac{dy}{dt} = -y \qquad y(0) = 1 \qquad 0 \le t \le 1$$

- Kết quả chính xác là : y = e-t

h	flopeE	errE	flopeH	errH
0.20000	31	4.02e-02	57	2.86e-03
0.10000	61	1.92e-02	112	6.62e-04
0.05000	121	9.39e-03	222	1.59e-04
0.02500	241	4.66e-03	442	3.90e-05
0.01250	481	2.31e-03	882	9.67e-06
0.00625	961	1.15e-03	1762	2.41e-06

TS Nguyễn Hoài Sơn

Phương pháp Runge-Kutta

Tính độ dốc ở 4 vị trí ứng với mỗi bước lặp:

$$k_1 = f(t_j, y_j)$$

$$k_2 = f(t_j + \frac{h}{2}, y_j + \frac{h}{2}k_1)$$

$$k_3 = f(t_j + \frac{h}{2}, y_j + \frac{h}{2}k_2)$$

$$k_4 = f(t_j + h, y_j + hk_3)$$

Ta tính được y_{j+1}

$$y_{j+1} = y_j + h\left(\frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6}\right)$$

TS Nguyễn Hoài Sơn

Phương pháp Runge-Kutta

Giải bằng Matlab

```
>> rhs = inline('cos(t)','t','y') ;
>> [t,Y] = odeRK4(rhs,2*pi,0.01, 0) ;
>> plot(t,Y,'o') ;
```

Hàm thư viện Matlab

[t,Y] = ode45(diffep,tn,y0)

```
>> rhs = inline('cos(t)','t','y') ;
>> [t,Y] = ode45(rhs,[0 2*pi], 0) ;
>> plot(t,Y,'r','linewidth',2) ;
```

```
function [t,y] = odeRK4(diffeq,tn,h,y0)]
t = (0:h:tn)';
n = length(t) ;
y = y0 + ones(n , 1) ;
h2= h /2 ; h3= h /3 ; h6= h /6 ;
for j = 2 : n
k1 = feval(diffeq, t(j -1), y(j-1))  ;
k2 = feval(diffeq , t(j -1)+h2, y(j-1)+h2*k1 ) ;
k3 = feval(diffeq , t(j -1)+h2, y(j-1)+h2*k2 ) ;
k4 = feval(diffeq , t(j -1)+h  , y(j-1)+h*k3)    ;
y(j) = y(j – 1) + h6* (k1+k4) + h3*(k2+k3);
end
```

TS Nguyễn Hoài Sơn

Phương pháp Runge-Kutta

So sánh Euler, Midpoint và RK4:

Giải: $$\frac{dy}{dt} = -y \qquad y(0) = 1 \qquad 0 \le t \le 1$$

h	flope E	errE	flopeM	errM	flope4	err4
0.20000	31	4.02e-02	57	2.86e-3	129	5.80e-6
0.10000	61	1.92e-02	112	6.62e-4	254	3.33e-7
0.05000	121	9.39e-03	222	1.59e-4	504	2.00e-8
0.02500	241	4.66e-03	442	3.90e-5	1004	1.22e-9
0.01250	481	2.31e-03	882	9.67e-6	2004	7.56e-11
0.00625	961	1.15e-03	1762	2.41e-6	4004	4.70e-12

Sử dụng hàm của Matlab:

Sử dụng ode45

Cú pháp :

[t,Y] = ode45(diffep,tn,y0)
[t,Y] = ode45(diffep,[t0 tn],y0)
[t,Y] = ode45(diffep,[t0 tn],y0,options)
[t,Y] = ode45(diffep,[t0 tn],y0,options,arg1,arg2,...)

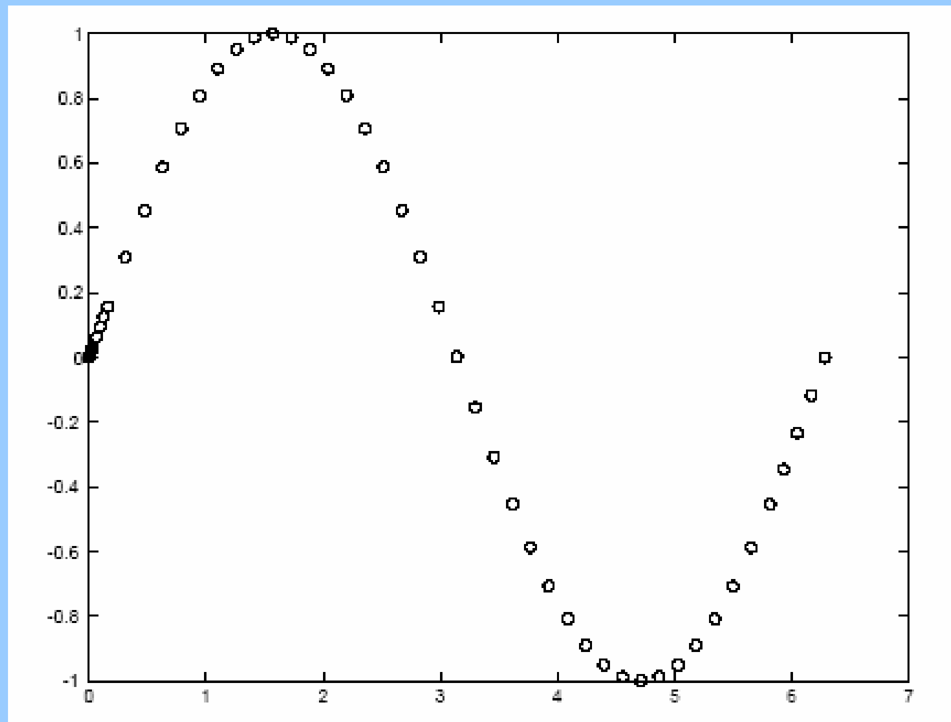
TS Nguyễn Hoài Sơn

Phương pháp Runge-Kutta

Ví dụ

$$\frac{dy}{dt} = \cos(t) \qquad y(0) = 0$$

```
>> rhs = inline('cos(t)','t','y') ;
>> [t,Y] = ode45(rhs,[0  2*pi], 0) ;
>> plot(t,Y,'o') ;
```



TS Nguyễn Hoài Sơn

Bài toán giá trị biên :

Phương trình vi phân cấp 2 :

Ứng dụng cho các bài toán về thanh , truyền nhiệt ,vv...

Dạng : ay''(x)+by'(x)+cy(x)=f(x) $0 < x < 1$ (7.10)

Điều kiện biên :

a/ $y(x=0) = y_0$ *

$y(x=L) = \mathbf{y_L}$ *

b/ $y'(x=0) = y'_o$ *

$y(x=L) = y_L$ *

c/ $y(x=0) = y_0$ *

$y'(x=L) = \mathbf{y'_L}$ *

Xấp xỉ (7.10) bằng lưới đều sai phân trung tâm :$h_o = \Delta x$

$$y_i' = \frac{y_{i+1} - y_{i-1}}{2h} + 0(h^2) \qquad \text{với } O(h^2) = -\frac{1}{6} h^2 f_i''' \qquad \mathbf{(7.11)}$$

$$y_i'' = \frac{y_{i+1} - 2f_i + y_{i-1}}{h^2} + 0(h^2) \qquad \text{với } 0(h^2) = -\frac{1}{12} h^2 f_i''' \qquad (7.12)$$

TS Nguyễn Hoài Sơn

(7.10), (7.11) và(7.12) cho ta phương trình sai phân

$$a\left[\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2}\right] + b\left[\frac{y_{i+1} - y_{i-1}}{2h}\right] + cy_i = f(x) \qquad (7.13)$$

$$(2a + bh)y_{i+1} + (2ch^2 - 4a)y_i + (2a - bh)y_{i-1} = 2h^2 f(x) \qquad (7.14)$$

i=1 => $(2a + bh)y_2 + (2ch^2 - 4a)y_1 + (2a - bh)y_o = 2h^2 f(x)$

i=2 => $(2a + bh)y_3 + (2ch^2 - 4a)y_2 + (2a - bh)y_1 = 2h^2 f(x)$

i=3 => $(2a + bh)y_4 + (2ch^2 - 4a)y_3 + (2a - bh)y_2 = 2h^2 f(x)$

i=4 => $(2a + bh)y_5 + (2ch^2 - 4a)y_4 + (2a - bh)y_3 = 2h^2 f(x)$

i=5 => $(2a + bh)y_6 + (2ch^2 - 4a)y_5 + (2a - bh)y_4 = 2h^2 f(x)$

TS Nguyễn Hoài Sơn

Đặt :

$A=2a + bh$

$B=2ch^2 - 4a$

$C=2a - bh$

Đưa hệ 5 phương trình trên về dạng ma trận :

a/

$$
\begin{aligned}
By_1 + Ay_2 &= 2h^2f(x) - Cy_0^* \\
Cy_1 + By_2 + Ay_3 &= 2h^2f(x) \\
Cy_2 + By_3 + Ay_4 &= 2h^2f(x) \\
Cy_3 + By_4 + Ay_5 &= 2h^2f(x) \\
Cy_4 + By_5 &= 2h^2f(x) - Ay_L^*
\end{aligned}
$$

Hay dạng ma trận :

$$
\begin{bmatrix} B & A & 0 & 0 & 0 \\ C & B & A & 0 & 0 \\ 0 & C & B & A & 0 \\ 0 & 0 & C & B & A \\ 0 & 0 & 0 & C & B \end{bmatrix}
\begin{pmatrix} y1 \\ y2 \\ y3 \\ y4 \\ y5 \end{pmatrix}
=
\begin{bmatrix} 2h^2 f(x) - Cy_0\ ^* \\ 2h^2 f(x) \\ 2h^2 f(x) \\ 2h^2 f(x) \\ 2h^2 f(x) - Ay_L\ ^* \end{bmatrix}
$$

TS Nguyễn Hoài Sơn

b/ $y'_1 = \dfrac{y_2 = y_0}{2h} = y'_0{}^*$ (Biết) $\Rightarrow$ $y_0 = y_2 - 2hy_0'{}^*$

$$(5) \Rightarrow \begin{bmatrix} B & A+C & 0 & 0 & 0 \\ C & B & A & 0 & 0 \\ 0 & C & B & A & 0 \\ 0 & 0 & C & B & A \\ 0 & 0 & 0 & C & B \end{bmatrix} \begin{pmatrix} y1 \\ y2 \\ y3 \\ y4 \\ y5 \end{pmatrix} = \begin{bmatrix} 2hCy_0{}'^* + 2h^2 f(x) \\ 2h^2 f(x) \\ 2h^2 f(x) \\ 2h^2 f(x) \\ 2h^2 f(x) - Ay_L{}^* \end{bmatrix}$$

c/ $y'_5 = \dfrac{y_6 - y_4}{2h} = y_L{}'^*$ $\Rightarrow$ $y_6 = y_4 + 2hy_L{}'^*$

$$\begin{bmatrix} B & A & 0 & 0 & 0 \\ C & B & A & 0 & 0 \\ 0 & C & B & A & 0 \\ 0 & 0 & C & B & A \\ 0 & 0 & 0 & A+C & B \end{bmatrix} \begin{pmatrix} y1 \\ y2 \\ y3 \\ y4 \\ y5 \end{pmatrix} = \begin{bmatrix} 2h^2 f(x) - Cy_0{}^* \\ 2h^2 f(x) \\ 2h^2 f(x) \\ 2h^2 f(x) \\ 2h^2 f(x) - 2hAy_L{}'^* \end{bmatrix}$$

TS Nguyễn Hoài Sơn

I. Phương pháp số: Chia đôi khoảng, Newton-Raphson, Dây cung

1. Chia đôi khoảng (Bisection Method)

$$(1): a, b, tol$$

$$(2): k = 1,2,..$$

$$(3): x_m = \frac{a+b}{2}$$

$$(4): f(x_m) \neq 0$$

$$(5): f(a).f(x_m) < 0$$

$$(6): b = x_m$$

$$(7): a = x_m$$

$$(8): |f(x_m)| \leq tol$$

$$(9): x_m = \frac{a+b}{2}$$

$$(10): in\ x$$

Matlab code.

```
clear all
clc
a=3;b=4;tol=0.0001
for k=1:10
    x=(a+b)/2;
    if sign(f(x))==sign(f(a))
        a=x;
    else
        b=x;
    end
    if abs(f(x)>tol)
        break
    end
end
function gg=f(x)
gg=x-x.^(1/3)-2;
```

TS Nguyễn Hoài Sơn

2. Newton-Raphson

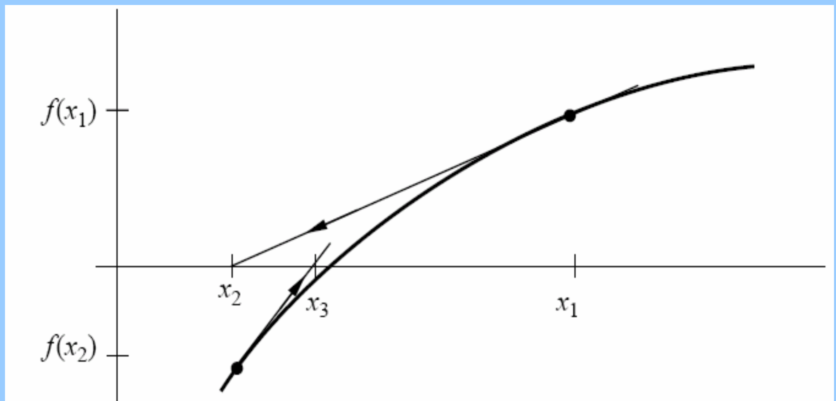
$(1): x_0, tol$

$(2): k = 1,2,...$

$(3): x_k = x_{k-1} - \dfrac{f(x_{k-1})}{f'(x_{k-1})}$

$(4): \left| x_k - x_{k-1} \right| < tol$

$(5): in\, x$




```
clear all
clc
format long
x=10; tol=1e-10; itemax=20; itein=0;
while abs(f(x))>tol
    itein=itein+1;
    if itein>itemax break
    end
    Dx=-f(x)/df(x);
    sprintf ('itein = %d   x= %20.10f  f(x) =
%20.10f  Dx= %20.10f\n',...
              itein,x,f(x),Dx);
     x=x+Dx;
end
sprintf ('solution %20.10f, f(x)=…
%20.10f\n',x, f(x))
%--------------------------------------
function q=f(x)
q=x-x.^1/3-2;
function q=df(x)
q=1-1/3*1/(x.^(2/3));
```

TS Nguyễn Hoài Sơn

3. Dây cung (Secant Method):

$(1): x_1 , x_2 , tol.$

$(2): k = 2, 3, ...$

$$(3): x_{k+1} = x_k - f(x_k)\frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})}$$

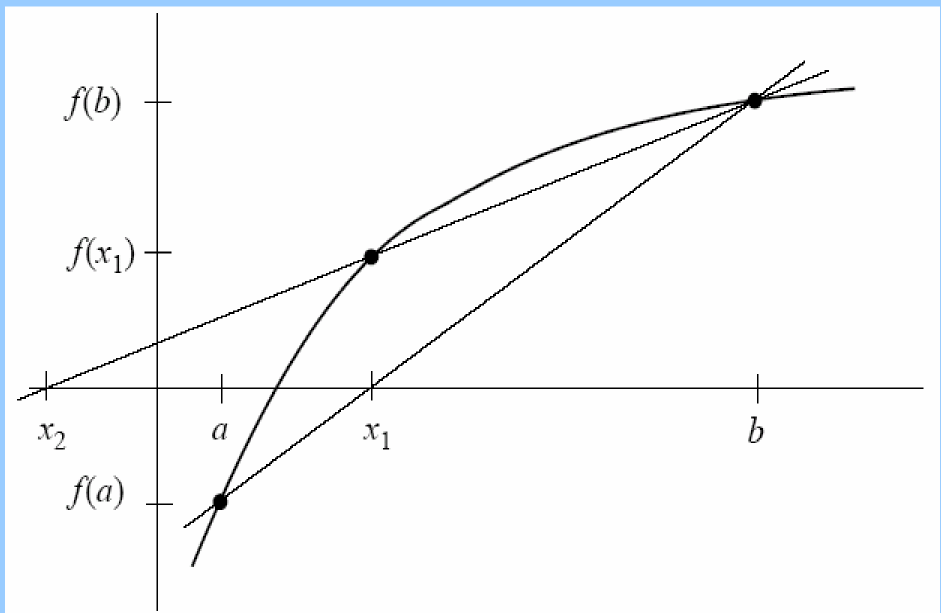
$(4): f(x_k) f(x_{k-1}) < 0$

$(5): x_{k-1} = x_{k+1}$

$(6): x_k = x_{k+1}$

$(7): |f(x_{k+1})| \le tol$

$(8): In \quad x$




```
clear all
clc
syms x
format long
x1=3;
x2=4;
tol=1e-6
while abs(f(x2))>tol
    xk=x2-f(x2)*(x2-x1)/(f(x2)-f(x1));
    if f(x1)*f(x2)<0
       x2=xk;
    else
       x1=xk;
    end
end
nghiem=x2
%---------------------------------------
function g=f(x)
g=x-x^(1/3)-2;
```

Kết Qủa và so sánh

1. Chia đôi khoảng

k	a	b	x_{mid}	$f(x_{mid})$
0	3	4		
1	3	4	3.5	-0.01829449
2	3.5	4	3.75	0.19638375
3	3.5	3.75	3.625	0.08884159
4	3.5	3.625	3.5625	0.03522131
5	3.5	3.5625	3.53125	0.00845016
6	3.5	3.53125	3.515625	-0.00492550
7	3.51625	3.53125	3.5234375	0.00176150
8	3.51625	3.5234375	3.51953125	-0.00158221
9	3.51953125	3.5234375	3.52148438	0.00008959
10	3.51953125	3.52148438	3.52050781	-0.00074632

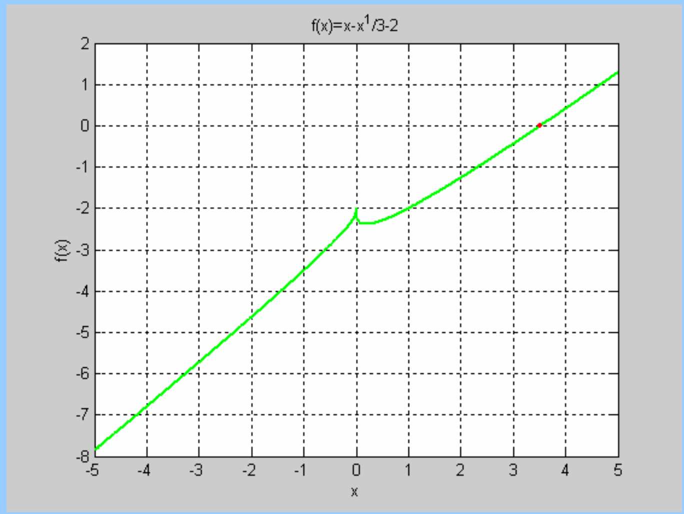
1. Newton-Raphson

k	x_k	$f'(x_k)$	$f(x)$
0	3	0.83975005	-0.44224957
1	3.52664429	0.85612976	0.00450679
2	3.52138015	0.85598641	3.771×10^{-7}
3	3.52137971	0.85598640	2.664×10^{-15}
4	3.52137971	0.85598640	0.0

1. Dây cung

k	x_{k-1}	x_k	$f(x_k)$
0	4	3	-0.44224957
1	3	3.51734262	-0.00345547
2	3.51734262	3.52141665	0.00003163
3	3.52141665	3.52137970	-2.034×10^{-9}
4	3.52137959	3.52137971	-1.332×10^{-15}
5	3.52137971	3.52137971	0.0

4. Đồ thị f(x)=x-x^1/3-2



TS Nguyễn Hoài Sơn

2. Phương pháp giải lặp hệ phương trình tuyến tính

a. Conjugate gradient method (CG):

$$\mathbf{x}^{(0)} = \mathbf{0},\ \mathbf{r}^{(0)} = \mathbf{b} - \mathbf{A}\mathbf{x}^{(0)},\ \mathbf{p}^{(0)} = \mathbf{r}^{(0)}$$

for k = 1,2,3,.....

$$\alpha^{(k)} = \frac{\left(\mathbf{r}^{(k-1)}\right)^T \left(\mathbf{r}^{(k-1)}\right)}{\left(\mathbf{p}^{(k-1)}\right)^T \mathbf{A} \left(\mathbf{p}^{(k-1)}\right)}$$ Kích thước bước

$$\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)} + \alpha^{(k)}\mathbf{p}^{(k-1)}$$ Nghiệm xấp xỉ

$$\mathbf{r}^{(k)} = \mathbf{r}^{(k-1)} - \alpha^{(k)}\mathbf{A}\mathbf{p}^{(k-1)}$$ Thặng dư

$$\beta^{(k)} = \frac{\left(\mathbf{r}^{(k)}\right)^T \left(\mathbf{r}^{(k)}\right)}{\left(\mathbf{r}^{(k-1)}\right)^T \left(\mathbf{r}^{(k-1)}\right)}$$ Cải tiến

$$\mathbf{p}^{(k)} = \mathbf{r}^{(k)} + \beta^{(k)}\mathbf{p}^{(k-1)}$$ Hướng tìm nghiệm

end

```
clear all
clc
a=[2 4 7; 4 5 6; 7 6 1];
b=[-1 2 5]';
x=[0 0 0]';
r=b-a*x;
p=r;
for i=1:10
    alpha=r'*r/(p'*a*p);
    x=x+alpha*p;
    r1=r;
    r1=r1-alpha*a*p;
    beta=r1'*r1/(r'*r);
    p=r1+beta*p ;
    r=r1;
end
r
```

b. Preconditioned Conjugate gradient method(PCG):

$$\mathbf{x}^{(0)} = \mathbf{0},\ \mathbf{r}^{(0)} = \mathbf{b} - \mathbf{A}\mathbf{x}^{(0)},\ \mathbf{h}^{(0)} = \mathbf{C}\mathbf{r}^{(0)},\ \mathbf{p}^{(0)} = \mathbf{h}^{(0)}$$

for k = 1,2,3,.....

$$\alpha^{(k)} = \frac{\left(\mathbf{r}^{(k-1)}\right)^T\left(\mathbf{p}^{(k-1)}\right)}{\left(\mathbf{p}^{(k-1)}\right)^T\mathbf{A}\left(\mathbf{p}^{(k-1)}\right)}$$

$$\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)} + \alpha^{(k)}\mathbf{p}^{(k-1)}$$

$$\mathbf{r}^{(k)} = \mathbf{r}^{(k-1)} - \alpha^{(k)}\mathbf{A}\mathbf{p}^{(k-1)}$$

$$\mathbf{h}^{(k)} = \mathbf{C}\mathbf{r}^{(k)}$$

Chỉnh lý

$$\beta^{(k)} = \frac{\left(\mathbf{r}^{(k)}\right)^T\left(\mathbf{h}^{(k)}\right)}{\left(\mathbf{r}^{(k-1)}\right)^T\left(\mathbf{h}^{(k-1)}\right)}$$

$$\mathbf{p}^{(k)} = \mathbf{h}^{(k)} + \beta^{(k)}\mathbf{p}^{(k-1)}$$

end

```
clear all
clc
a=[2 4 7; 4 5 6; 7 6 1];
b=[-1 2 5]';
x=[0 0 0]';
r=b-a*x;
h=0.5*r;
p=h;
for i=1:10
    alpha = r'*p/(p'*a*p);
    x = x + alpha*p;
    r1 = r;
    r1= r1-alpha*a*p;
    h1 = 0.5*r1;
    beta = r1'*h1/(r'*h);
    p=h1+beta*p;
    r=r1;
    h=h1;
end
r1
```

B. Hệ phương trình phi tuyến: Newton-Raphson

Giải thuật Newton.

$$Ax = b, \qquad A(n,n)$$

thặng dư :

$$f = Ax - b$$

Dạng tổng quát

$$f(x) = \begin{pmatrix} f_1(x_1, x_2, \ldots\ldots, x_n) \\ f_2(x_1, x_2, \ldots\ldots, x_n) \\ . \\ . \\ . \\ f_n(x_1, x_2, \ldots\ldots, x_n) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ . \\ . \\ . \\ 0 \end{pmatrix}$$

a) Chọn nghiệm đề nghị $x^{(k)}$ và số gia nghiệm ở bước lặp thứ k $\Delta x^{(k)}$ để:

$$x^{(k+1)} = x^{(k)} + \Delta x^{(k)} \rightarrow f(x^{(k+1)}) = 0$$

b) Khai triển Taylor hàm f:

$$f\left(x^{(k+1)}\right) = f\left(x^{(k)}\right) + \Delta x^{(k)} f'(x^{(k)}) + 0\left(\left\|\Delta x^{(k)}\right\|^2\right)$$

$$f'(x) \equiv J(x) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & . & . & . & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & . & . & . & \frac{\partial f_2}{\partial x_n} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & . & . & . & \frac{\partial f_n}{\partial x_n} \end{bmatrix}$$

b) Jacobian hàm f bỏ đi số hạng bậc cao

$$f\left(x^{(k+1)}\right) = f\left(x^{(k)}\right) + J(x^{(k)})\Delta x^{(k)}$$

c) Tìm $\Delta x^{(k)}$ từ:

$$f(x^{(k+1)}) = 0 \Leftrightarrow J(x^{(k)})\Delta x^{(k)} = -f(x^{(k)})$$

d) Bảy bước cho giải thuật:

* Đề nghị nghiệm ban đầu.
* Tính giá trị hàm f.
* Kiểm tra chuẩn $\|f\|$ đủ bé thì dừng.
* Tính giá trị Jacobian J.
* Giải $J.\Delta x = -f$
* Cập nhật nghiệm $x \leftarrow x + \Delta x$
* Trở về bước 2.

Ví dụ:

Giải hệ phương trình phi tuyến sau:

$$x_1 + x_2 = 2$$
$$x_1 x_3 + x_2 x_4 = 0$$
$$x_1 x_3^2 + x_2 x_4^2 = \frac{2}{3}$$
$$x_1 x_3^3 + x_2 x_4^3 = 0$$

Matlab program

```
clear all
clc
format long
x=zeros(4,1);
x(1)=0.7;x(2)=0.5;x(3)=-0.01;
x(4)=0.1;
tol=1e-10; itemax=100;
itein=0;
f=fnorm(x);
while norm(f)>tol
    itein=itein+1;
```

TS Nguyễn Hoài Sơn

```
    if itein>itemax break
    end
    jac=jacobian(x);
    dx=-jac\f;
    x=x+dx;
    f=fnorm(x);
    sprintf ('itein = %d   x1= %15.10f  x2= %15.10f  x3= %15.10f...
    x4= %15.10f  residual= %15.10f\n',itein,x,norm(f))
end
sprintf ('solution %20.10f, f(x)= %20.10f\n',x, f(x))
%------------------------------------------------------------------------------
function f=fnorm(x)
f=zeros(4,1);
f(1)=x(1)+x(2)-2;
f(2)=x(1).*x(3)+x(2).*x(4);
f(3)=x(1).*x(3).^2+x(2).*x(4).^2-2/3;
f(4)=x(1).*x(3).^3+x(2).*x(4).^3;
%------------------
function jac=jacobian(x)
jac=zeros(4,4);
jac(1,1)=1;jac(1,2)=1;jac(1,3)=0;jac(1,4)=0;jac(2,1)=x(3);jac(2,2)=x(4);
jac(2,3)=x(1); jac(2,4)=x(2); jac(3,1)=x(3).^2;jac(3,2)=x(4).^2;
jac(3,3)=2*x(1).*x(3); jac(3,4)=2*x(2).*x(4);jac(4,1)=x(3).^3;
jac(4,2)=x(4).^3;jac(4,3)=3*x(1).*x(3).^2;jac(4,4)=3*x(2).*x(4).^2;
```

Kết quả.

ans =

itein = 33   x1=1.0000000000 x2=1.0000000000 x3=0.5773502692

x4= -0.5773502692 residual=0.0000000000

I. Dùng phương pháp tính số :

1. Luật tuyến tính :

$$S_{xx} = \sum_{i=1}^{m} x_i x_i = 2.08 \qquad S_x = \sum_{i=1}^{m} x_i = 3.2$$

$$S_{xy} = \sum_{i=1}^{m} x_i y_i = 4.832 \qquad S_y = \sum_{i=1}^{m} y_i = 7.74$$

$$\alpha = \frac{1}{d}\left(S_x S_y - mS_{xy}\right) = 1.8857$$

$$\beta = \frac{1}{d}\left(S_x S_{xy} - S_{xx} S_y\right) = 0.2843$$

$$d = S_x^2 - mS_{xx} = -2.24$$

Ví dụ

Độ mòn bề mặt segment theo thời gian cho bảng dữ liệu sau: với m = 6.

x	0.1	0.4	0.5	0.6	0.7	0.9
y	0.61	0.92	0.99	1.52	1.67	2.03

Phương trình cần tìm: **y = 1.8857x+0.2843**

TS Nguyễn Hoài Sơn

2. Luật đa thức bậc 2:

$$f(x) = a_0 + a_1 x + a_2 x^2, \quad m = 6, \quad k = 0,1,2, \quad i = 1,2,\ldots,6$$

$$\begin{bmatrix} m & \sum_{i=1}^{m} x_i & \sum_{i=1}^{m} x_i^2 \\ \sum_{i=1}^{m} x_i & \sum_{i=1}^{m} x_i^2 & \sum_{i=1}^{m} x_i^3 \\ \sum_{i=1}^{m} x_i^2 & \sum_{i=1}^{m} x_i^3 & \sum_{i=1}^{m} x_i^4 \end{bmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{m} y_i \\ \sum_{i=1}^{m} x_i y_i \\ \sum_{i=1}^{m} x_i^2 y_i \end{pmatrix}$$

Phương trình cần tìm:

$y = 0.485 + 0.7845x + 1.1152x^2$

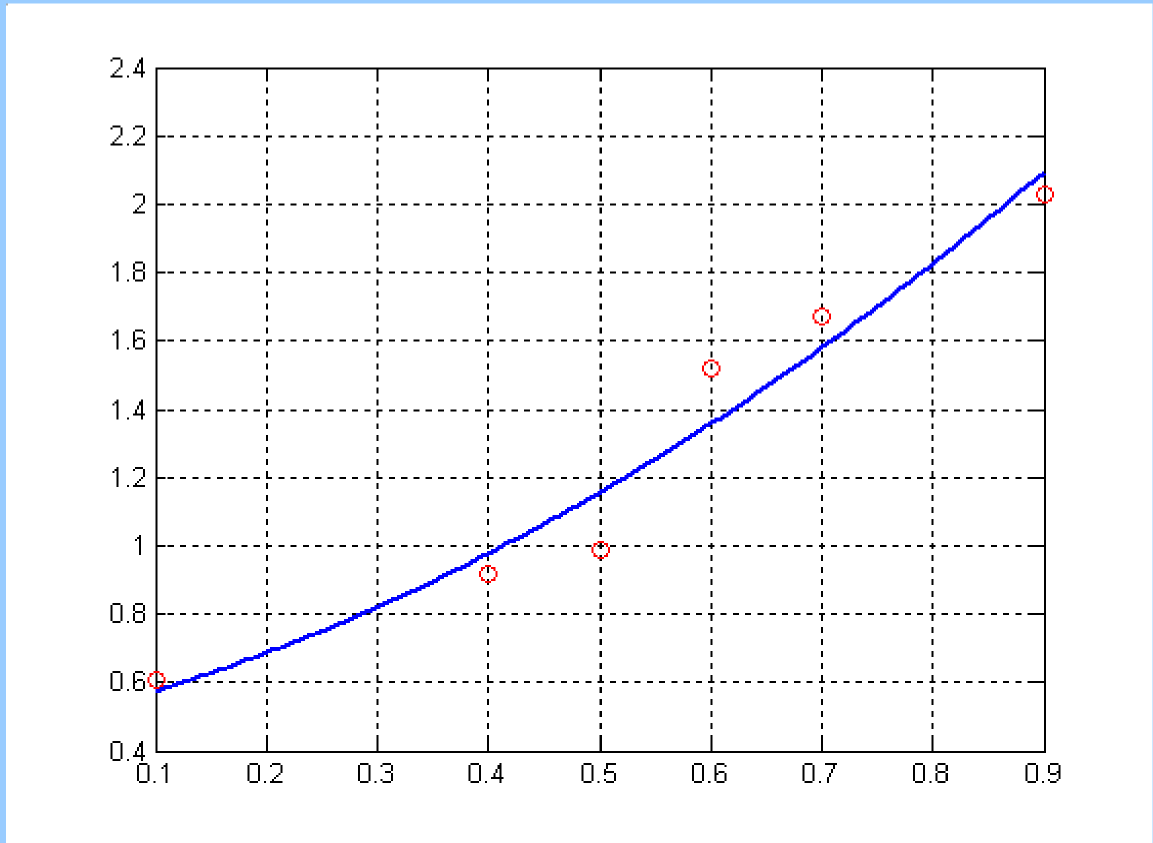
Matlab program

```
Clear all
clc
x=[0.1 0.4 0.5 0.6 0.7 0.9];
y=[0.61 0.92 0.99 1.52 1.67 2.03];
s1=0;s2=0;s3=0;s4=0;s5=0;s6=0;s7=0;
for   i=1:6
    s1=s1+x(i);s2=s2+(x(i))^2;
    s3=s3+(x(i))^3;s4=s4+(x(i))^4;
    s5=s5+y(i);s6=s6+x(i)*y(i);
    s7=s7+x(i)^2*y(i);
end
```

```
a=zeros(3,3);
b=zeros(3,1);
a(1,1)=6;
a(1,2)=s1;
a(1,3)=s2;
a(2,1)=s1;
a(2,2)=s2;
a(2,3)=s3;
a(3,1)=s2;
a(3,2)=s3;
a(3,3)=s4;
b(1,1)=s5;
b(2,1)=s6;
b(3,1)=s7;
c=LU(a,b);
% gọi hàm LU đã thực hiện ở
%chương trước để giải nghiệm
%giải bằng Matlab:
c0=0.485; c1=0.7845; c2=1.1152;
```

TS Nguyễn Hoài Sơn

$y=0.485 + 0.7845x + 1.1152x^2$



3. Luật phi tuyến :

$$y = c_1 e^{c_2 x} \rightarrow \ln y = \alpha x + \beta$$

$$y = c_1 x^{c_2} \rightarrow \ln y = \alpha \ln x + \beta$$

$$y = c_1 x e^{c_2 x} \rightarrow \ln(y/x) = \alpha x + \beta$$

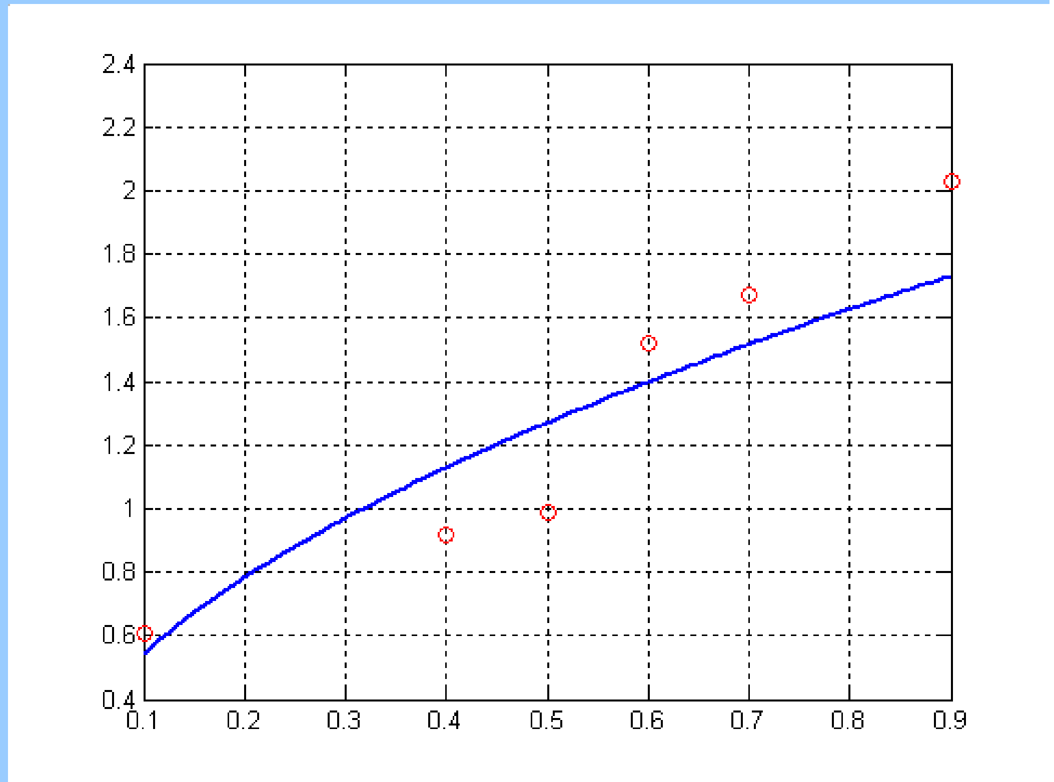
TS Nguyễn Hoài Sơn

Nội suy theo luật hàm luỹ thừa

Matlab program

```
clear all
clc
x=[0.1  0.4  0.5  0.6  0.7  0.9];
y=[0.61  0.92  0.99  1.52  1.67  2.03];
%================================
% Bảng số liệu đo đạc
%================================
xx=[];
yy=[];
for i=1:6
    xx=[xx log(x(i))];
    yy=[yy log(y(i))];
end
su=0;
suu=0;
sv=0;
suv=0;
for i=1:6
    su=su+xx(i);
    suu=suu+(xx(i)^2);
    sv=sv+yy(i);
    suv=suv+xx(i)*yy(i);
end
d=su^2-6*suu;
c2=(su*sv-6*suv)/d
b=(su*suv-suu*sv)/d
c1=exp(b)
```

$y = 1.8311\ x^{0.5227}$



4. Nội suy theo luật tổ hợp

$$f(x) = c_1 f_1(x) + c_2 f_2(x) + ... + c_n f_n(x)$$

$$f(x) = \sum_{i=1}^{n} c_i f_i(x)$$

Phương trình cần tìm:

$$y = \frac{0.0365}{x} + 2.2177x$$

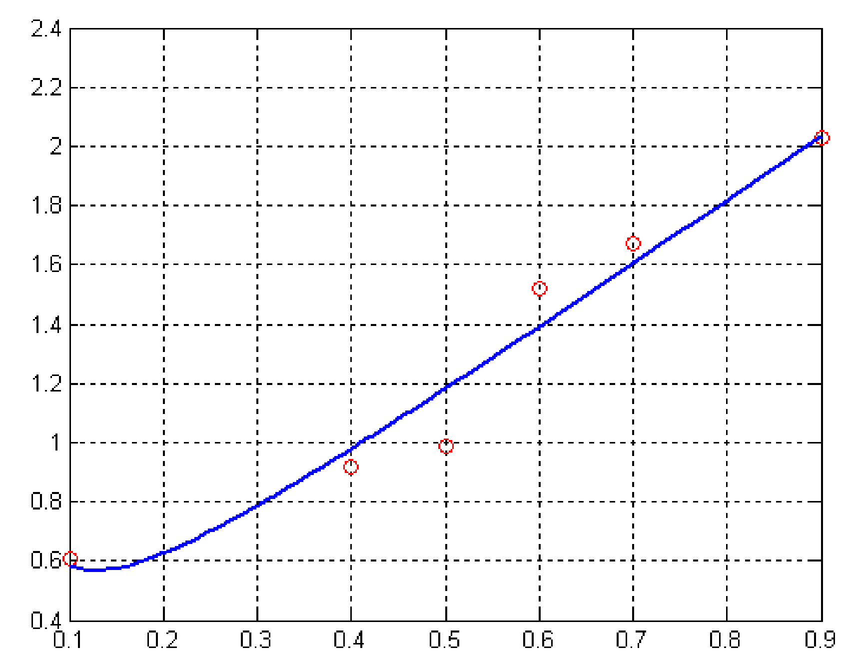
Matlab program

```
clear all
clc
x=[0.1  0.4  0.5  0.6  0.7  0.9];
y=[0.61  0.92  0.99  1.52  1.67  2.03];
A=zeros(6,2);
B=zeros(6,1);
for i=1:6
    A(i,1)=f1(x(i));
    A(i,2)=f2(x(i));
    B(i,1)=y(i);
end
c=(A'*A)\(A'*B)
```

```
function b=f2(x)
b=x;
```

```
function a=f1(x)
    a=1/x;
```

$$y = \frac{0.0365}{x} + 2.2177x$$



5. Nội suy theo luật đa thức dựa trên khai triển Taylor

Luật đa thức $y = a_n x^n + a_{n-1}x^{n-1} + ... + a_1 x + a_0$

$$y_1 = a_n x_1^n + a_{n-1}x_1^{n-1} + ... + a_1 x_1 + a_0$$
$$y_2 = a_n x_2^n + a_{n-1}x_2^{n-1} + ... + a_1 x_2 + a_0$$
$$\cdot$$
$$\cdot$$
$$y_n = a_n x_n^n + a_{n-1}x_n^{n-1} + ... + a_1 x_n + a_0$$

$$\Rightarrow \begin{bmatrix} x_1^n & x_1^{n-1} & ... & x_1 & 1 \\ x_2^n & x_2^{n-1} & ... & x_2 & 1 \\ . & . & . & . & . \\ . & . & . & . & . \\ x_n^n & x_n^{n-1} & ... & x_n & 1 \end{bmatrix} \begin{pmatrix} a_n \\ a_{n-1} \\ . \\ . \\ a_0 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ . \\ . \\ y_n \end{pmatrix}$$

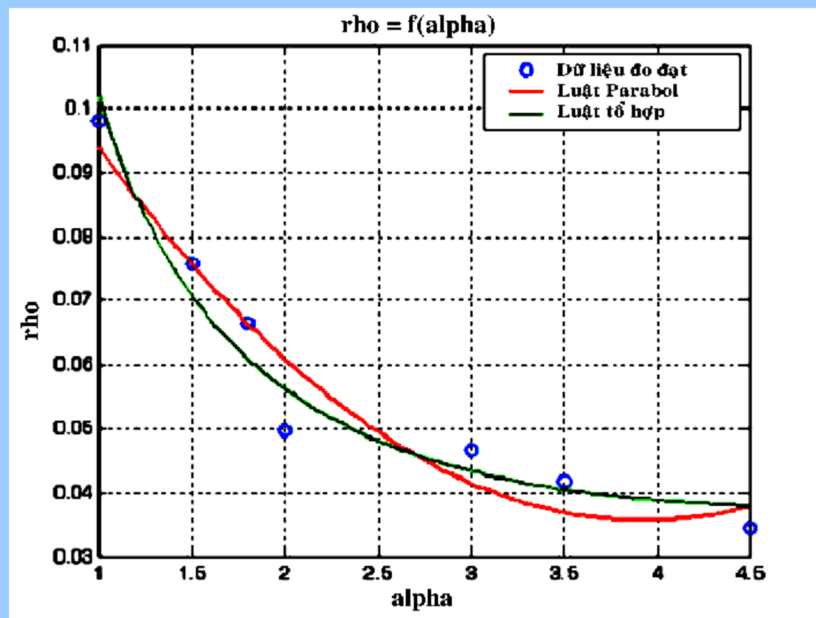
Ví dụ

Bảng dữ liệu đo đạc :

α	ρ_{max}
1.0	0.098158
1.5	0.075798
1.8	0.066604
2.0	0.049851
3.0	0.046624
3.5	0.041890
4.5	0.034597

a) Luật Parabol $c_1 x^2 + c_2 x + c_3$

b) Luật tổ hợp tuyến tính $\dfrac{c_1}{x} + c_2 x$



TS Nguyễn Hoài Sơn

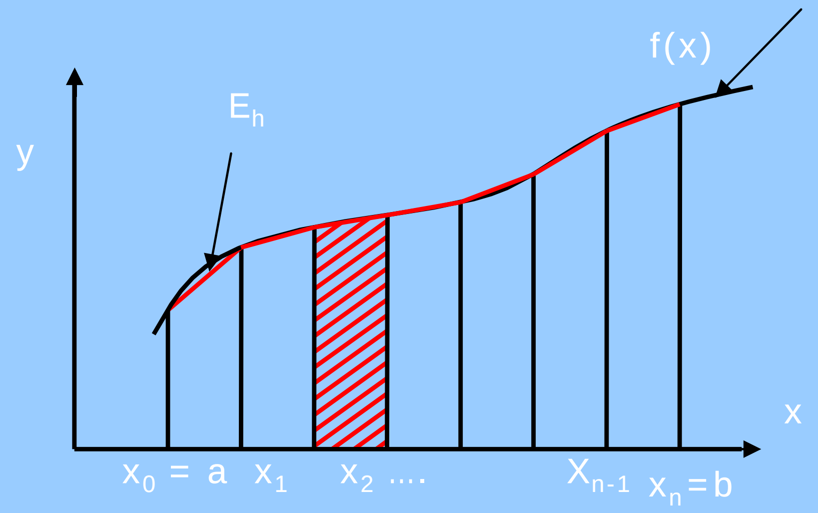
Matlab program

```
clear all
clc
alpha=[1 1.5 1.8 2.0 3.0 3.5 4.5]';
rho=  [0.098158 0.075798 0.066604 0.049851
0.046624 0.04189 0.0346]';
% luật parabol qua 7 điểm: c1x^2+c2x+c3
A=[alpha.^2 alpha ones(size(alpha))];
disp(A'*A)
disp(A'*rho)
c =(A'*A)\(A'*rho)
% vẽ đồ thị
xfit=linspace(min(alpha),max(alpha));
yfit1=c(1)*xfit.^2+c(2)*xfit+c(3);
% luật c1/x+c2x
A=[1./alpha alpha];
c=(A'*A)\(A'*rho);
xfit=linspace(min(alpha),max(alpha));
yfit2=c(1)./xfit+c(2)*xfit;
plot(alpha,rho,'o',xfit,yfit1,'r',xfit,yfit2,'c')
xlabel('alpha')
ylabel('rho')
title(‘rho=f(alpha)')
legend(‘ dữ liệu đo đạc','luật parabol',luật tổ hợp')
grid on
```

TS Nguyễn Hoài Sơn

II. Dùng tích phân số :

1. Luật hình thang (Trapzoidal Rule) :



$$\int_{x_{i-1}}^{x_i} f(x)dx \approx \frac{h_i}{2}(f(x_{i-1}) + f(x_i))$$

$$I_{trap} = \frac{h}{2}\begin{pmatrix} f(x_0) + 2f(x_1) + 2f(x_2) + \ldots.. \\ + 2f(x_{n-1}) + f(x_n) \end{pmatrix} + E$$

$$h = \frac{b-a}{N},\ \ x_i = a + i*h,\ \ x_0 = a,\ \ x_n = b$$

$$I_{trap} = \frac{h}{2}\left(f_0 + 2f_1 + 2f_2 + \ldots.. + 2f_{n-1} + f_n\right) + E$$

$$E \approx -\frac{1}{12}\frac{(b-a)^3}{N^3}\sum_{i=1}^{N} f''(\bar{x}_i), \qquad \bar{x}_i = \frac{x_{i-1} + x_{i+1}}{2}$$

Ví dụ

Tính tích phân: $S = \int_0^2 f(x)dx = \int_0^2 \pi\left(1 + \left(\frac{x}{2}\right)^2\right)^2 dx$

TS Nguyễn Hoài Sơn

Matlab program

```
clear all
clc
N=16;
a=0;
b=2;
h=(b-a)/N;
S=0;
for i=0:N
   x=a+i*h;
   if i==0 | i==N
      c=1;
   else
      c=2;
   end
   S=S+c*pi*(1+(x/2).^2).^2;
end
S=h*S/2
```

Kết qủa:

N	h	S_h	E_h
2	1.	12.7627	-1.0341
4	0.5	11.9895	-0.2609
8	0.25	11.7940	-0.0654
16	0.125	11.7449	-0.0163
32	0.0625	11.7326	-0.0040
64	0.03125	11.7296	-0.0010

2. Luật Simpson 1/3 (Simpson Rule) :

$$S = \int_a^b f(x)dx = \frac{h}{3}\left[f(a) + 4f(\bar{x}) + f(b)\right] + E$$

$$x_0 = a,\ x_2 = b,\ h = \frac{b-a}{2},\ \ \bar{x} = \frac{a+b}{2}$$

$$S = \int_a^b f(x)dx = \frac{h}{3}\left[f_0 + 4f_1 + f_2\right] + E$$

$$S_{simp} = \frac{h}{3}\left[f(a) + 4\sum_{i=1}^{N-1} f(a+ih) + 2\sum_{i=2}^{N-2} f(a+ih) + f(b)\right] + E$$

$$S_{simp} = \frac{h}{3}\begin{pmatrix} f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + \\ \ldots.. + 4f(x_{n-1}) + f(x_n) \end{pmatrix} + E$$

$$E \approx -\frac{N}{2}\frac{h^5}{90}\bar{f}'''', \qquad \bar{f}'''' = \sum_{i=1}^{N} f''''(\bar{x}_i)/N, \qquad \bar{x}_i = \frac{x_{i-1} + x_{i+1}}{2}$$

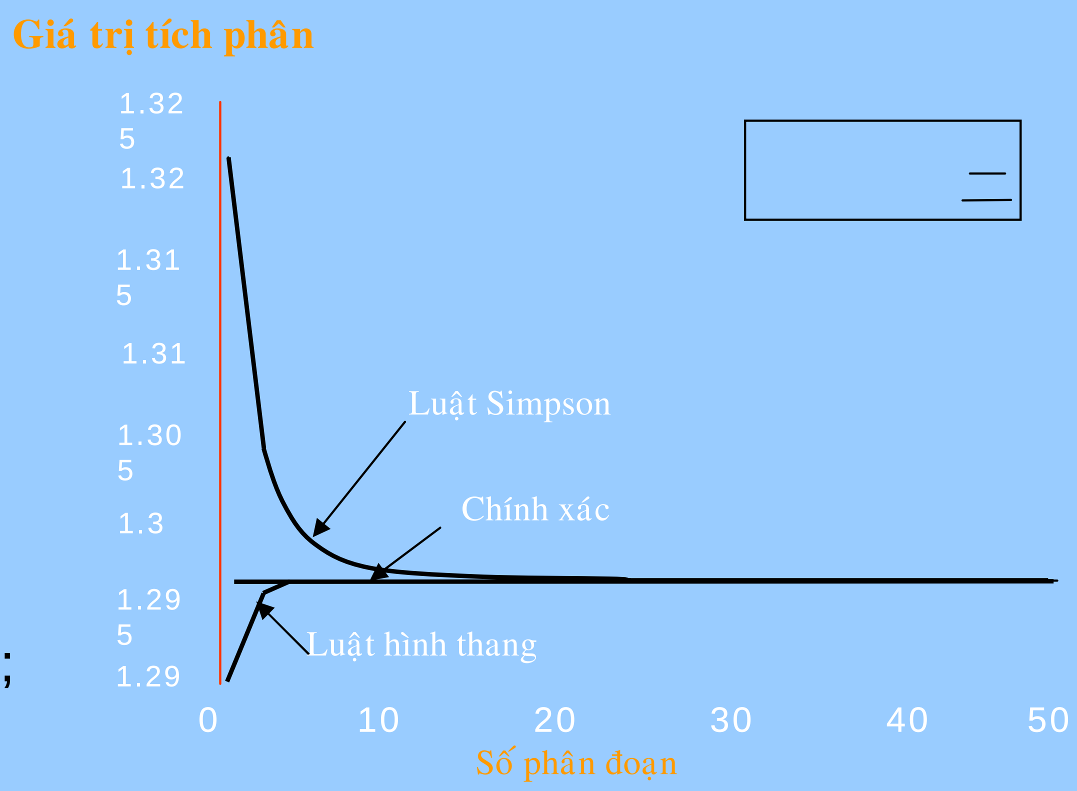
Matlab program

```
clear all
clc
N=16;
a=0;
b=2;
h=(b-a)/N;
S=0;
for i=0:N
   x=a+i*h;
  if i==0 | i==N
     c=1;

  elseif i==fix(i/2)*2+1
     c=4;
  else
     c=2;
   end
   S=S+c*pi*(1+(x/2).^2).^2;
end
S=h*S/3
```

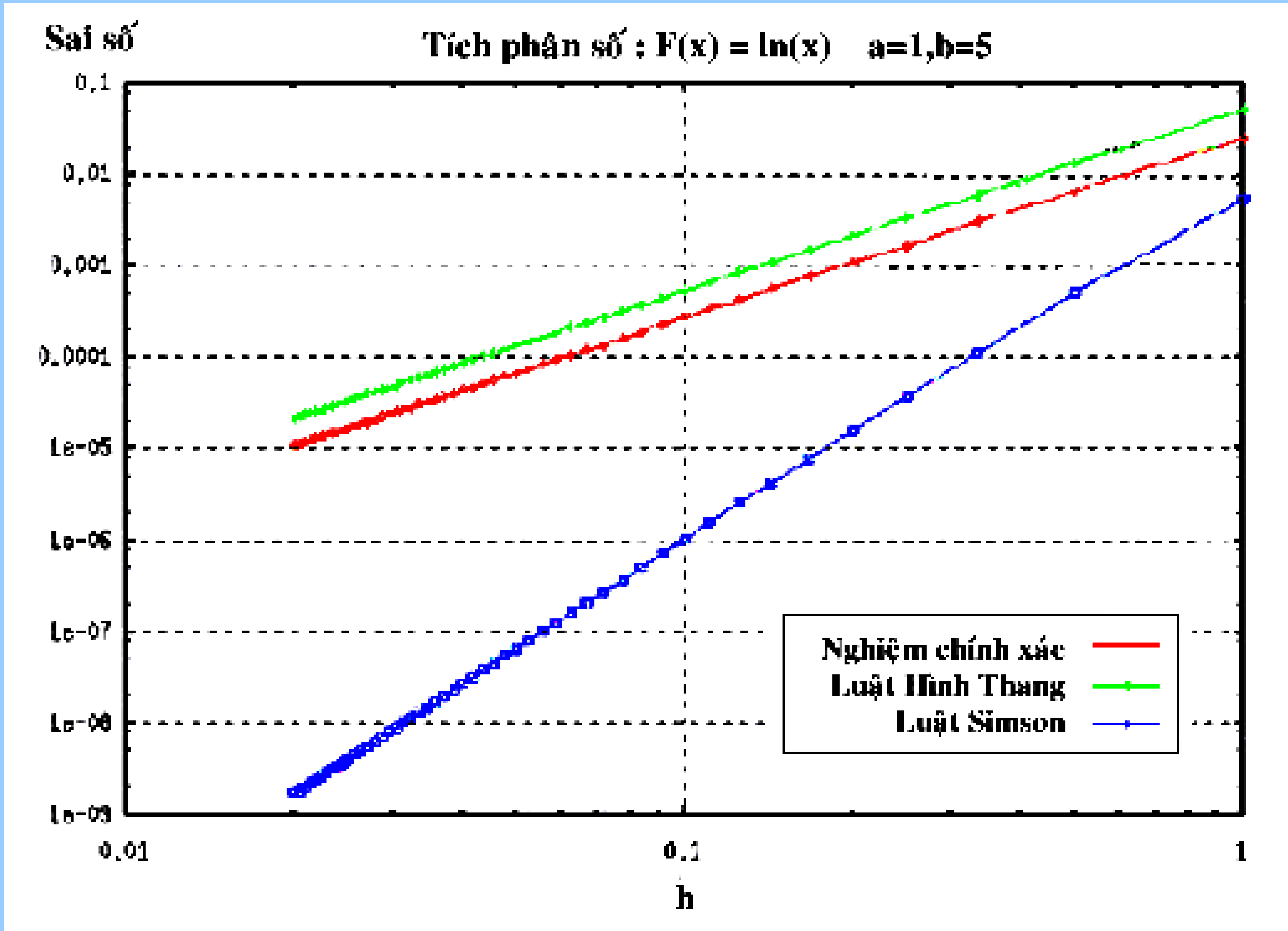
Kết qủa:

N	h	S_h	E_h
2	1.	11.7809	-0.0523
4	0.5	11.7318	-0.0032
8	0.25	11.7288	-0.0002
16	0.125	11.7286	-0.0000
32	0.0625	11.7286	-0.0000
64	0.03125	11.7286	-0.0000



TS Nguyễn Hoài Sơn

Sai số phương pháp:



TS Nguyễn Hoài Sơn

3. Tích phân Gauss (Gauss quadrature):

$$I = \int_{-1}^{1} f(x)\,dx \approx w_1 f(x_1) + w_2 f(x_2) + \cdots + w_n f(x_n)$$

Ví dụ:

$$I = \int_{-1}^{1} (0.2 + 25x - 200x^2 + 675x^3 - 900x^4 + 400x^5)\,dx$$

Matlab program

Tính với 4 điểm Gauss:

```
clear all
 clc
format long
x1=-0.861136;
x2=-0.339981;
x3=0.339981;
x4=0.861136;
% ------trọng số-----------------------------------
w1=0.347855;
w2=0.652145;
w3=0.652145;
w4=0.347855;
f1=w1*gauss1(x1);
f2=w2*gauss1(x2);
f3=w3*gauss1(x3);
f4=w4*gauss1(x4);
m=f1+f2+f3+f4
%------------------------------------------------------------------
function ff=gauss1(x)
        ff=400*x^5-900*x^4+675*x^3-200*x^2+25*x+0.2;
kết quả:
        I=-4.929329328775451e+002
```

TS Nguyễn Hoài Sơn

MATLAB - FEM

Bài tập 3.4

```
clear all; clc; close all
echo off
%-----------------------------------------------------------------
 Edof=[1  1  2  3  4  5  6;
          2  4  5  6  7  8  9];
%-----------------------------------------------------------------
 K=zeros(9);                f=zeros(9,1); f(8)=-88.9/2;
%-----------------------------------------------------------------
h=17.9; tw=0.315; bf=6.015;tf=0.525;
A=2*tf*bf+tw*(h-2*tf);
I=2.5e-2; E=2.1e8; L=6.1;
ep=[E A I];
 Ex=[0             L;
       L          3*L/2];
 Ey=zeros(2,2);
 Eq=zeros(2,2);
%-----------------------------------------------------------------
for i=1:2
  [Ke,fe]=beam2e(Ex(i,:),Ey(i,:),ep);
  [K,f]=assem(Edof(i,:),K,Ke,f,fe);
end
%-----------------------------------------------------------------
 bc=[1 0;2 0;4 0;5 0;7 0;9 0];
 a=solveq(K,f,bc);
 %-----------------------------------------------------------------
Ed=extract(Edof,a);
[es1,edi1,eci1]=beam2s(Ex(1,:),Ey(1,:),ep,Ed(1,:),Eq(1,:),20);
[es2,edi2,eci2]=beam2s(Ex(2,:),Ey(2,:),ep,Ed(2,:),Eq(2,:),10);
%-----------------------------------------------------------------
```

```
function [Ke,fe]=beam2e(ex,ey,ep,eq);
%-----------------------------------------------------------------
%    INPUT:
%          ex = [x1 x2]
%          ey = [y1 y2]      element node coordinates
%
%          ep = [E A I]   element properties
%                              E: Young's modulus
%                              A: Cross section area
%                              I: Moment of inertia
%
%          eq = [qx qy] distributed loads, local directions
%
%    OUTPUT: Ke : element stiffness matrix (6 x 6)
%                  fe : element load vector (6 x 1)
%-----------------------------------------------------------------
  b=[ ex(2)-ex(1); ey(2)-ey(1) ];
  L=sqrt(b'*b);  n=b/L;
```

TS Nguyễn Hoài Sơn

MATLAB - FEM

```
E=ep(1);  A=ep(2);  I=ep(3);
qx=0; qy=0;  if nargin>3; qx=eq(1); qy=eq(2); end

 Kle=[E*A/L   0          0      -E*A/L     0        0 ;
      0   12*E*I/L^3   6*E*I/L^2  0   -12*E*I/L^3
6*E*I/L^2;
      0   6*E*I/L^2    4*E*I/L    0   -6*E*I/L^2   2*E*I/L;
    -E*A/L  0          0      E*A/L     0        0 ;
      0   -12*E*I/L^3 -6*E*I/L^2  0   12*E*I/L^3  -
6*E*I/L^2;
      0   6*E*I/L^2    2*E*I/L    0   -6*E*I/L^2
4*E*I/L];

 fle=L*[qx/2 qy/2 qy*L/12 qx/2 qy/2 -qy*L/12]';

 G=[n(1) n(2)  0    0    0   0;
   -n(2) n(1)  0    0    0   0;
    0    0    1    0    0   0;
    0    0    0   n(1) n(2) 0;
    0    0    0  -n(2) n(1) 0;
    0    0    0    0    0   1];

 Ke=G'*Kle*G;   fe=G'*fle;
%-------------------------end----------------------------
```

```
function [K,f]=assem(edof,K,Ke,f,fe)
%------------------------------------------------------------
% INPUT: edof:      dof topology matrix
%        K :        the global stiffness matrix
%        Ke:        element stiffness matrix
%        f :        the global force vector
%        fe:        element force vector
%
% OUTPUT: K :       the new global stiffness matrix
%         f :       the new global force vector
%------------------------------------------------------------
   [nie,n]=size(edof);
   t=edof(:,2:n);
   for i = 1:nie
     K(t(i,:),t(i,:)) = K(t(i,:),t(i,:))+Ke;
     if nargin==5
        f(t(i,:))=f(t(i,:))+fe;
     end
   end
%-------------------------end------------------------------
```

TS Nguyễn Hoài Sơn

MATLAB - FEM

```
function [d,Q]=solveq(K,f,bc)
% a=solveq(K,f)
% [a,Q]=solveq(K,f,bc)
%-------------------------------------------------------------
% PURPOSE
%  Solve static FE-equations considering boundary conditions.
%
% INPUT: K : global stiffness matrix, dim(K)= nd x nd
%        f : global load vector, dim(f)= nd x 1
%
%        bc : boundary condition matrix
%            dim(bc)= nbc x 2, nbc : number of b.c.'s
%
% OUTPUT:  a : solution including boundary values
%          Q : reaction force vector
%              dim(a)=dim(Q)= nd x 1, nd : number of dof's
%-------------------------------------------------------------
 if nargin==2 ;
   d=K\f ;
 elseif nargin==3;
   [nd,nd]=size(K);
   fdof=[1:nd]';
%
   d=zeros(size(fdof));
   Q=zeros(size(fdof));
%
   pdof=bc(:,1);
   dp=bc(:,2);
   fdof(pdof)=[];
s=K(fdof,fdof)\(f(fdof)-K(fdof,pdof)*dp);
   %A=K(fdof,fdof);
   %B=(f(fdof)-K(fdof,pdof)*dp);
   %s=pcg(A,B);
%
   d(pdof)=dp;
   d(fdof)=s;
 end
   Q=K*d-f;

%-------------------------end------------------------------
```

```
function [ed]=extract(edof,a)
%-------------------------------------------------------------
% PURPOSE
%  Extract element displacements from the global
% displacement
%  vector according to the topology matrix edof.
% INPUT:   a:  the global displacement vector
%          edof:  topology matrix
% OUTPUT: ed:  element displacement matrix
%-------------------------------------------------------------
   [nie,n]=size(edof);
   t=edof(:,2:n);
%
   for i = 1:nie
      ed(i,1:(n-1))=a(t(i,:))';
   end
%
%-------------------------end------------------------------
```

TS Nguyễn Hoài Sơn

MATLAB - FEM

```
function [es,edi,eci]=beam2s(ex,ey,ep,ed,eq,n)
%   PURPOSE
%     Compute section forces in two dimensional beam element
%(beam2e).
%   INPUT:  ex = [x1 x2]
%           ey = [y1 y2]     element node coordinates
%           ep = [E A I]     element properties,
%                        E:  Young's modulus
%                        A:  cross section area
%                        I:  moment of inertia
%           ed = [u1 ... u6] element displacements
%           eq = [qx qy]     distributed loads, local directions
%           n : number of evaluation points ( default=2 )
%
%   OUTPUT:
%  es = [ N1 V1 M1 ;  section forces, local directions, in   N2 V2 M2
%;  n points along the beam, dim(es)= n x 3  .........]
%   edi = [ u1 v1 ;    element displacements, local directions,  u2 v2
%;    in n points along the beam, dim(es)= n x 2 ....]
%   eci = [ x1  ;      local x-coordinates of the evaluation
%    x2 ;      points, (x1=0 and xn=L)        ...]
%-------------------------------------------------------------
 EA=ep(1)*ep(2); EI=ep(1)*ep(3);
 b=[ ex(2)-ex(1); ey(2)-ey(1) ];
 L=sqrt(b'*b);

 if length(ed(:,1)) > 1
     disp('Only one row is allowed in the ed matrix !!!')
     return
   end
    qx=0; qy=0;  if nargin>4;  qx=eq(1); qy=eq(2); end
    ne=2;       if nargin>5;  ne=n; end;
```

```
  C=[0   0   0    1   0   0;
     0   0   0    0   0   1;
     0   0   0    0   1   0;
     L   0   0    1   0   0;
     0   L^3  L^2 0   L   1;
     0 3*L^2 2*L  0   1   0];
  n=b/L;
  G=[n(1) n(2)  0    0    0   0;
    -n(2) n(1)  0    0    0   0;
      0    0    1    0    0   0;
      0    0    0   n(1) n(2) 0;
      0    0    0  -n(2) n(1) 0;
      0    0    0    0    0   1];
   M=inv(C)*(G*ed'-[0 0 0 -qx*L^2/(2*EA)
qy*L^4/(24*EI) qy*L^3/(6*EI)]' );
  A=[M(1) M(4)]';  B=[M(2) M(3) M(5) M(6)]';
    x=[0:L/(ne-1):L]';   zero=zeros(size(x));
one=ones(size(x));

  u=[x one]*A-(x.^2)*qx/(2*EA);
  du=[one zero]*A-x*qx/EA;
  v=[x.^3 x.^2 x one]*B+(x.^4)*qy/(24*EI);
% dv=[3*x.^2 2*x one zero]*B+(x.^3)*qy/(6*EI);
  d2v=[6*x 2*one zero zero]*B+(x.^2)*qy/(2*EI);
  d3v=[6*one zero zero zero]*B+x*qy/EI;

  N=EA*du; M=EI*d2v; V=-EI*d3v;
  es=[N V M];
  edi=[u v];
  eci=x;
%-------------------------end------------------------------
```

TS Nguyễn Hoài Sơn