

THỰC HÀNH ĐIỀU KHIỂN THIẾT BỊ ĐIỆN

BÀI 1: THAO TÁC TRONG CỬA SỔ LỆNH CỦA MATLAB

I. MỤC TIÊU

Bài thí nghiệm này giúp sinh viên làm quen nhanh với Matlab bằng các thao tác đơn giản trên ma trận, vectơ, biểu thức toán học, các lệnh đồ họa, ..., thực hiện ngay trên cửa sổ lệnh (command window) của Matlab.

II. THỰC HÀNH

Từ cửa sổ lệnh của Matlab, sinh viên lần lượt thực hiện các thao tác sau:

1. Ma trận

Để tạo ma trận trong Matlab ta chỉ cần liệt các phần tử của ma trận trong cặp dấu ngoặc vuông ([...]). Các phần tử trên cùng hàng được phân biệt bởi dấu phẩy (,) hoặc khoảng trắng (space). Các hàng của ma trận, phân cách nhau bởi dấu chấm phẩy (;).

Ví dụ: Nhập ma trận A có 4 hàng, 4 cột như sau:

```
>> A=[16 3 2 13; 5 10 11 8; 9 6 7 12; 4 15 14 1]
```

```
>> size(A)
```

Để truy xuất đến từng phần tử của ma trận ta dùng chỉ số phần tử tương ứng.

Ví dụ, phần tử ở hàng thứ 2, cột thứ 3 của A là A(2,3).

```
>> A(2,3)
```

Bài thực hành:

Bài 1: Cho ma trận $A=[2 \ 4 \ 1; 6 \ 7 \ 2; 3 \ 5 \ 9]$, sinh viên dùng các lệnh cần thiết để:

a. Lấy dòng đầu tiên của ma trận A.

b. Tạo ma trận B bằng 2 dòng cuối cùng của A.

c. Tính tổng các phần tử trên các cột của A. (gợi ý: tính tổng các phần tử trên cột 1: $\text{sum}(A(:,1))$).

d. Tính tổng các phần tử trên các dòng của A.

Bài 2: Cho ma trận $A=[2 \ 7 \ 9 \ 7; 3 \ 1 \ 5 \ 6; 8 \ 1 \ 2 \ 5]$, sinh viên giải thích kết quả của các lệnh sau:

a. A' ; b. $A(:, [1 \ 4])$; c. $A([2 \ 3], [3 \ 1])$; d. $\text{reshape}(A, 2, 6)$

e. $A(:)$; f. $[A \ A(\text{end}, :)]$; g. $A(1:3, :)$; h. $[A ; A(1:2, :)]$

i. $\text{sum}(A)$; j. $\text{sum}(A')$; k. $[[A ; \text{sum}(A)] [\text{sum}(A, 2) ; \text{sum}(A(:))]]$

Bài 3: Giải hệ phương $Ax=b$, với:

THỰC HÀNH ĐIỀU KHIỂN THIẾT BỊ ĐIỆN

$$A = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 5 & 3 \\ 3 & -1 & 0 \end{bmatrix} \text{ và } b = \begin{bmatrix} 1 \\ 1 \\ -2 \end{bmatrix}$$

Gợi ý: $x=A \setminus b$

2. Vector

Vector thực chất cũng là ma trận có kích thước $(n \times 1)$ hay $(1 \times n)$, nên ta có thể tạo ra vector như cách tạo ra ma trận. Ngoài ra, có thể dùng một số cách sau:

```
>> x=0:0.1:1
>> y=linspace(1, 10, 20)    % vecto 20 phần tử cách đều nhau từ 1 đến 10
>> z=rand(10,1)
```

Bài thực hành:

Bài 1: Cho vector $x = [3 \ 1 \ 5 \ 7 \ 9 \ 2 \ 6]$, giải thích kết quả của các lệnh sau:

- $x(3)$;
- $x(1:7)$;
- $x(1:end)$;
- $x(1:end-1)$
- $x(6:-2:1)$;
- $x([1 \ 6 \ 2 \ 1 \ 1])$;
- $\text{sum}(x)$

Bài 2: Sinh viên hãy tạo một vector x có 100 phần tử, sao cho: $x_n = \frac{(-1)^{n+1}}{2n-1}$

Gợi ý: Tạo vector n có 100 phần tử từ 1 đến 100, dùng toán tử dấu chấm (.) để xác định x .

3. Các đa thức

Các đa thức trong Matlab được mô tả bằng các vector hàng với các phần tử của vector chính là các hệ số của đa thức, xếp theo thứ tự số mũ giảm dần. Ví dụ, đa thức $m = s^4 - s^3 + 4s^2 - 5s - 1$ được biểu diễn là:

```
>> m=[1 -1 4 5 -1]
```

Để xác định giá trị của đa thức, ta dùng lệnh `polyval`. Ví dụ, xác định giá trị của đa thức tại điểm $s=2$:

```
>> polyval(m,2)
```

Để xác định nghiệm của đa thức, ta dùng lệnh `roots`.

```
>> roots(m)
```

Bài thực hành:

Bài 1: Cho phương trình $x^2 - 4x + 5 = 0$, giải phương trình theo 2 cách,

Cách 1: tính delta theo phương pháp cổ điển,

Cách 2: dùng hàm `roots`, hãy so sánh kết quả.

THỰC HÀNH ĐIỀU KHIỂN THIẾT BỊ ĐIỆN

Cách 1:

```
>> a=1;
>> b=-4;
>> c=5
>> x1=(-b+sqrt(b^2-4*a*c))/(2*a)
>> x2=(-b-sqrt(b^2-4*a*c))/(2*a)
```

Cách 2:

```
>> m=[a b c];
>> x=roots(m)
```

Hãy thay đổi các giá trị khác nhau của a, b và c tương ứng trong 2 cách giải trên. So sánh kết quả và nhận xét.

Bài 2: Giải phương trình $x^3 - 2x^2 + 4x + 5 = 0$. Kiểm chứng kết quả thu được bằng hàm polyval. Nhận xét gì về kết quả kiểm chứng.

Bài 3: Lập lại câu trên cho phương trình $x^7 - 2 = 0$.

Bài 4: Thử dùng hàm poly để tạo đa thức từ các nghiệm cho trước.

```
>> help poly
```

4. Đồ họa

Matlab hỗ trợ chế độ đồ họa rất mạnh, bao gồm đồ họa 2D và 3D, với các trục tọa độ tuyến tính và phi tuyến bất kỳ.

4.1. Đồ họa 2D

Đồ họa 2D chủ yếu dựa trên lệnh plot. Để được giúp đỡ, ta gõ:

```
>> help plot
```

Ví dụ: Vẽ đồ thị hàm số $y_1 = \sin x \cdot \cos 2x$ và hàm số $y_2 = \sin^2 x$ trong $[0-2\pi]$, trên cùng hệ trục tọa độ, ta lần lượt thực hiện như sau:

```
>> x=0:0.01:2*pi;
>> y1=sin(x).*cos(2*x); %nhan tuong tung tung phan tu
>> plot(x,y1)
>> grid on %hien thi luoi
```

Sau khi thu được đồ thị hàm y_1 , để vẽ y_2 trên cùng đồ thị, ta thực hiện:

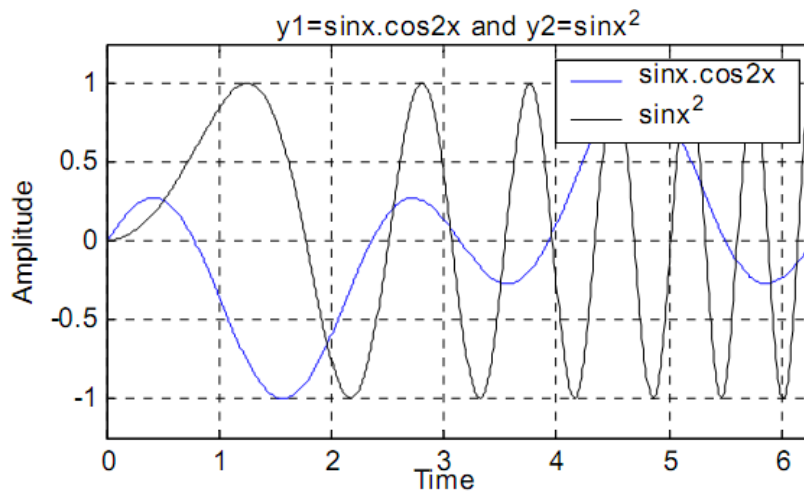
```
>> hold on % giu hinh, mac nhien la hold off
>> y2=sin(x.^2); %luy thua tung tung phan tu
```

THỰC HÀNH ĐIỀU KHIỂN THIẾT BỊ ĐIỆN

```
>> plot(x,y2,'r') %duong ve co mau do  
>> axis([0 4*pi -1.25 1.25]) %dinh lai toa do hien thi
```

Ta có thể đặt nhãn cho các trục cũng như tiêu đề cho đồ thị:

```
>> xlabel('Time')  
>> ylabel('Amplitude')  
>> title('y1=sinx.cos2x and y2=sin(x^2)')  
>> legend('sinx.cos2x','sinx^2')
```



Matlab hỗ trợ rất nhiều thuộc tính đồ họa, để có thể kiểm soát các thuộc tính này ta cần dùng đến thẻ đồ họa.

Ví dụ:

```
>>close all  
>>x=[0 1 2 3];  
>>y=[0 4 1 5];  
>>h=plot(x,y)
```

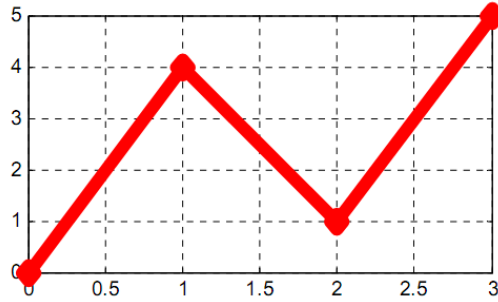
h chính là thẻ đồ họa của hàm plot, để thấy các thuộc tính đồ họa, ta dùng lệnh:

```
>> set(h)
```

Bây giờ ta thử đặt một số thuộc tính đồ họa cho h.

```
>>set(h,'Color','r') % dat lai mau do  
>>set(h,'LineWidth',6) % dat do rong duong  
>>set(h,'Marker','v','MarkerSize',6)
```

THỰC HÀNH ĐIỀU KHIỂN THIẾT BỊ ĐIỆN

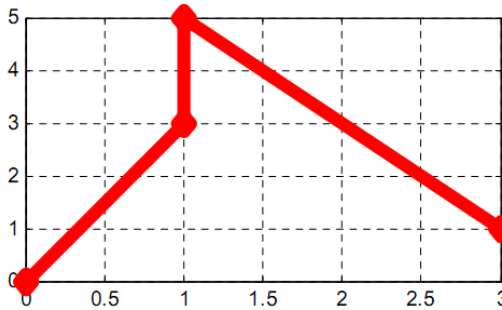


Dữ liệu dùng để vẽ được đặt trong 'XData' và 'YData'.

```
>> set(h,'XData',[0 1 1 3])
```

```
>>set(h,'YData',[0 3 5 1])
```

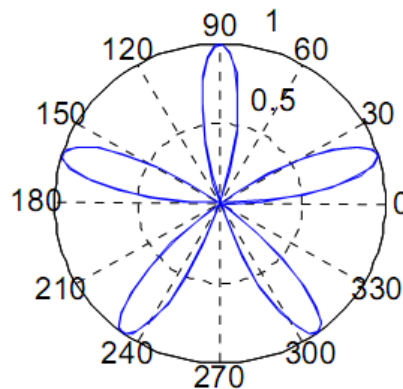
Từ kết quả này, ta nhận thấy nếu dữ liệu trong 'XData' và 'YData' biến thiên theo thời gian, thì trên cửa sổ đồ họa (figure) ta sẽ thấy hình ảnh sinh động kiểu 'animation'.



Tương tự như hàm plot, sinh viên thử dùng các hàm semilogx, semilogy và loglog cho trường hợp trục tọa độ phi tuyến.

Ngoài các lệnh biểu diễn đường cong trong tọa độ Descartes, Matlab cũng hỗ trợ việc vẽ đồ thị hàm số trong hệ tọa độ cực bằng hàm polar.

```
>> theta=0:0.05:2*pi;  
>> r=sin(5*theta);  
>> polar(theta,r)
```



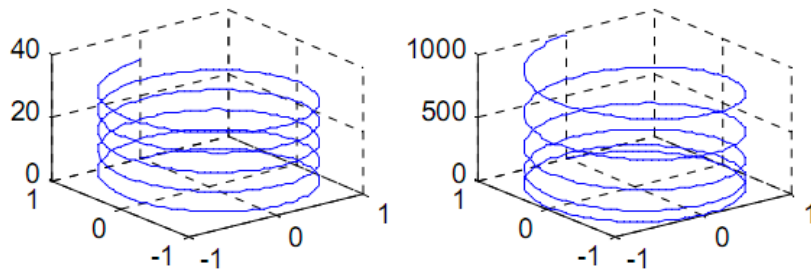
4.2. Đồ họa 3 D

THỰC HÀNH ĐIỀU KHIỂN THIẾT BỊ ĐIỆN

Matlab cung cấp nhiều hàm vẽ đồ thị 3D, chẳng hạn: plot3 - dùng để vẽ các đường trong không gian 3 chiều; mesh và surf - dùng để vẽ vật thể 3D (gõ help mesh và help surf để biết thêm các hàm 3D có liên quan).

Vẽ đồ thị 3D bằng hàm plot3:

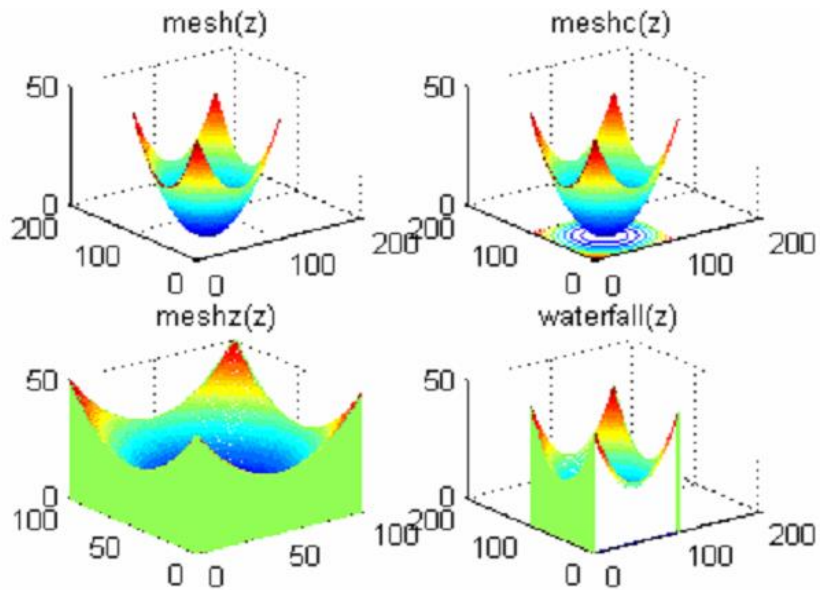
```
t=0:pi/50:10*pi;
x=sin(t);
y=cos(t);
z=t;
subplot(121), plot3(x,y,z) %ve tren o thu nhat
grid on
subplot(122), plot3(x,y,t.^2) %ve tren o thu hai
grid on
```



Vẽ mặt paraboloid $z = x^2 + y^2$ trong không gian 3 chiều:

```
close all
t=-5:0.1:5;
[x,y]=meshgrid(t); %dinh luoi ve
z=x.^2+y.^2;
subplot(2,2,1), mesh(z) %ve mat luoi 3D
title('mesh(z)')
subplot(2,2,2), meshc(z) %giiong mesh nhung co them duong vien
title('meshc(z)')
subplot(2,2,3), meshz(z) %co them luoi tren mat x,y
title('meshz(z)')
subplot(2,2,4), waterfall(z) %chi ve luoi theo 1 huong
title('waterfall(z)')
```

THỰC HÀNH ĐIỀU KHIỂN THIẾT BỊ ĐIỆN



Vẽ mặt paraboloid $z = \frac{\sin(\sqrt{x^2 + y^2})}{\sqrt{x^2 + y^2}}$ trong không gian 3 chiều:

```
x=-8:0.5:8;  
y=x;  
[x,y]=meshgrid(x,y);  
r=sqrt(x.^2+y.^2);  
z=sin(r)./r;  
surf(x,y,z)
```

Sinh viên thử vẽ mặt trụ $z = \sqrt{x^4 + y^2}$ bằng hàm mesh và hàm surf.

Bài tập:

1. Giải hệ phương trình sau:

$$2x_1 + 4x_2 + 6x_3 - 2x_4 = 0$$

$$x_1 + 2x_2 + x_3 + 2x_4 = 1$$

$$2x_2 + 4x_3 + 2x_4 = 2$$

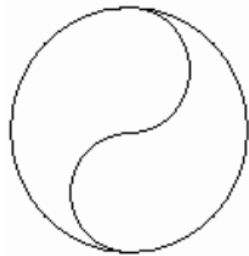
$$3x_1 - x_2 + 10x_4 = 10$$

2. Chứng tỏ rằng $(A+B)C=AC+BC$, với:

THỨC HÀNH ĐIỀU KHIỂN THIẾT BỊ ĐIỆN

$$A = \begin{bmatrix} 10 & -2 \\ 20 & 4 \\ 3 & 6 \end{bmatrix}; B = \begin{bmatrix} 3 & 1 \\ -10 & 2 \\ 0 & 5 \end{bmatrix}; C = \begin{bmatrix} -3 & 4 \\ 6 & 1 \end{bmatrix}$$

3. Hãy vẽ hình sau:



4. Từ cửa sổ lệnh của Matlab, nhập: demos. Chọn MATLAB → Graphics → 3D-plots. Chạy chương trình demo này.