

# Chapitre 4

## Les Graphiques sous Matlab

### 4.1 Introduction

Matlab offre un puissant système de visualisation pour la présentation et l'affichage graphique des données d'une manière à la fois efficace et facile.

Dans cette partie, nous allons présenter les principes de base indispensables pour dessiner des courbes en Matlab.

### 4.2 Graphique 2 dimensions (2D)

#### 4.2.1 La fonction plot

La fonction la plus simple pour la création des graphiques du type 2D est la fonction « plot », elle peut prendre en arguments deux vecteurs ou bien deux matrices de même taille, permettant de tracer une courbe affichée sur un graphique à deux axes (axe des abscisses et axe des ordonnées), en reliant les points de coordonnées définis dans ses arguments ayant plusieurs formes.

Par exemple, `plot(x, y)` marquera un point pour chaque couple  $[x(i), y(i)]$  avec  $i$  allant de 1 à `length(x)`. On représente ainsi les valeurs de  $y$  en fonction de  $x$ , sachant que  $x$  et  $y$  ayant la même taille.

#### a) Formes des arguments

1. Deux vecteurs de la même taille comme arguments :

#### Exemple 1 :

```
>> x= [1 2 3]
x =
     1     2     3
>> y= [-2 1 3]
y =
    -2     1     3
>> plot(x,y)
```

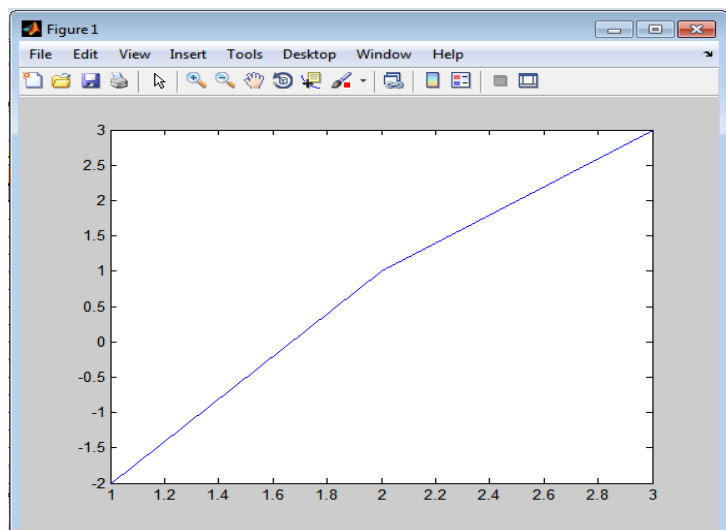


Figure 4.1 Fenêtre graphique (Exemple 1).

**Remarque :**

La variable x représente les valeurs du premier vecteur (axe des abscisses) et la variable y représente les valeurs du deuxième vecteur (axe des ordonnées).

**2. Un seul vecteur comme argument :**

**Exemple 2:**

```
>> plot([1 3 5 2]);
```

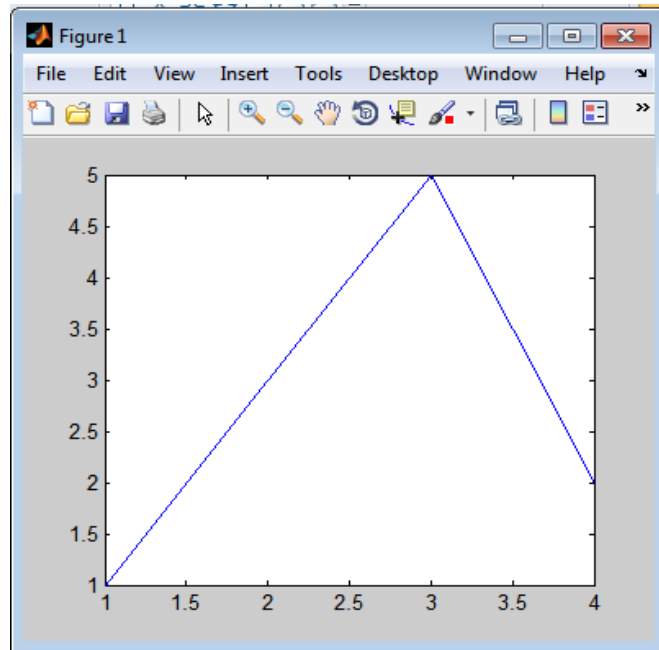


Figure 4.2 Fenêtre graphique (Exemple 2).

**Remarque :**

Les valeurs du vecteur [1 3 5 2] représentent les éléments de l'axe y (axe des ordonnées), et leurs positions représentent les éléments de l'axe x (axe des abscisses).

**3. Une seule matrice comme argument :**

**Exemple 3:**

```
>> M=[1 0 2;2 3 1;0 1 2]
```

M =

```
1 0 2
2 3 1
0 1 2
```

```
>> plot(M)
```

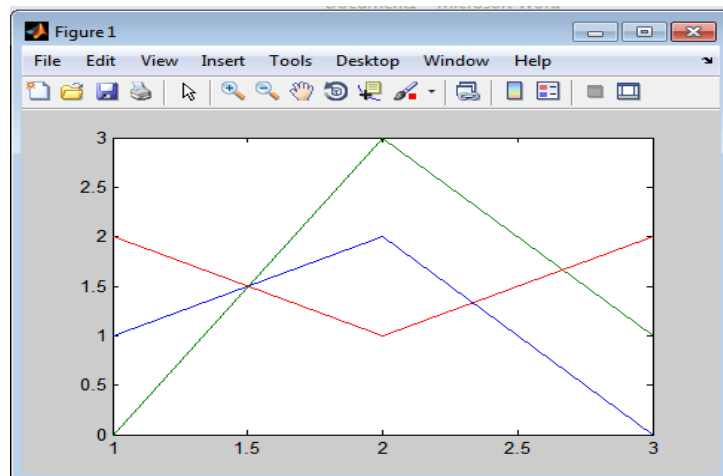


Figure 4.3 Fenêtre graphique (Exemple 3).

### Remarque :

- Chaque colonne de la matrice M sera représentée par une courbe.
- La fonction plot(M) donnera plusieurs courbes avec des couleurs différentes, sachant que la première colonne est représentée par la courbe bleue, la deuxième colonne est représentée par la courbe rouge et la troisième colonne est représentée par la courbe verte.
- Les valeurs de chaque colonne de la matrice M représentent les éléments de l'axe y et leurs positions (le numéro de ligne) comme les valeurs de l'axe x.

### 4. Deux matrices comme arguments :

#### Exemple 4 :

```
>> K=[1 1 1;2 2 2;3 3 3]
```

```
K =
```

```
    1    1    1
    2    2    2
    3    3    3
```

```
>> L=[0 -2 1;2 0 3;-3 3 -2]
```

```
L =
```

```
    0   -2    1
    2    0    3
   -3    3   -2
```

```
>> plot(K,L)
```

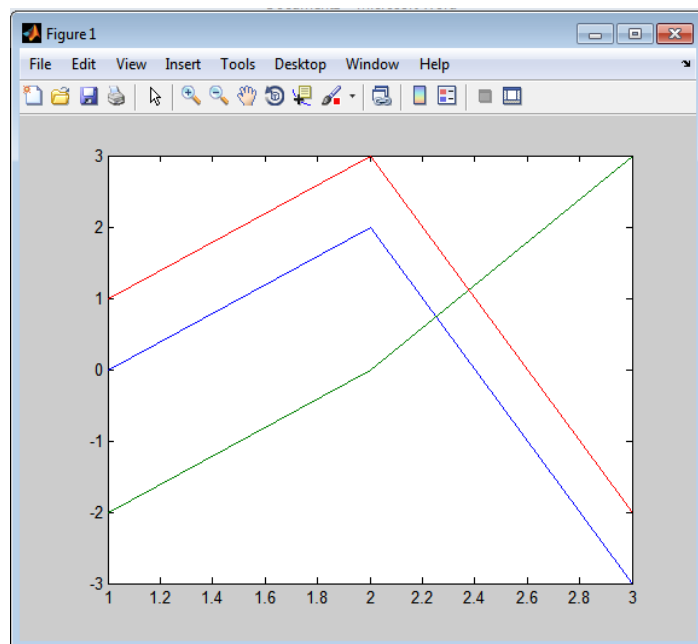


Figure 4.4 Fenêtre graphique (Exemple 4).

### Remarque :

- Les valeurs de chaque colonne de la première matrice représentent les éléments de l'axe x.
- Les valeurs de chaque colonne de la deuxième matrice représentent les éléments de l'axe y.

#### 4.2.2. Modification de l'apparence d'une courbe

Le traçage des courbes sous Matlab s'affiche par défaut en bleu, mais il est possible de changer l'apparence d'une courbe en modifiant sa couleur, la forme des points de coordonnées et le type de ligne reliant les points.





Pour cela, on ajoute un nouveau argument de type chaîne de caractère appelé « marquer » à la fonction plot en utilisant la syntaxe suivante :

**plot(x, y, 'marquer')**

## Chapitre 4 : Les Graphiques sous Matlab (partie 1)

Le mot marquer peut contenir un ou plusieurs caractères parmi les caractères rassemblés dans les tableaux suivants :

Couleur de la courbe	
Le caractère	Son effet
b ou blue	Courbe en bleu
g ou green	Courbe en vert
r ou red	Courbe en rouge
c ou cyan	Entre le vert et le bleu
m ou magenta	En rouge violacé vif
y ou yellow	Courbe en jaune
k ou black	Courbe en noir

Style de la courbe	
Le caractère	Son effet
-	En ligne plein 
:	En pointillé 
-.	En point tiret 
--	En tiret 

Représentation des points	
Le caractère	Son effet
.	Un point
o	Un cercle
x	Le symbole x
+	Le symbole +
*	Le symbole *
s	Un carré
d	Un losange
v	Triangle inférieur
^	Triangle supérieur
<	Triangle gauche
>	Triangle droit

### Exemple :

- Pour  $x = [0 \dots 6]$  tracer le graphe de la fonction  $y = \cos(x \cdot \pi)$  avec un pas = 0,1.
- Changer l'apparence de la courbe en utilisant les deux marqueurs suivants : ('rd-', 'c :^').

### Solution :

```
>> x=[0:0.1:6];  
>> y=cos(x*pi);  
>> plot(x,y,'rd-')
```

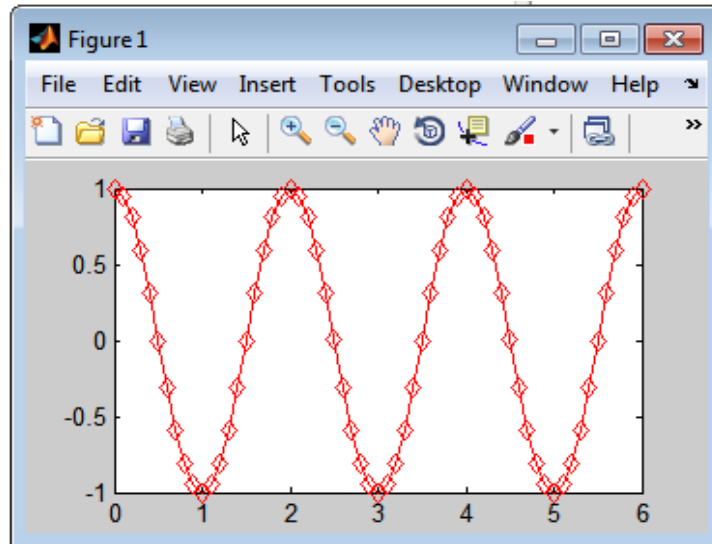


Figure 4.5 Courbe avec couleur rouge, en ligne plein avec des losanges.

```
>> x=[0:0.1:6];  
>> y=cos(x*pi);  
>> plot(x,y,'c :^')
```

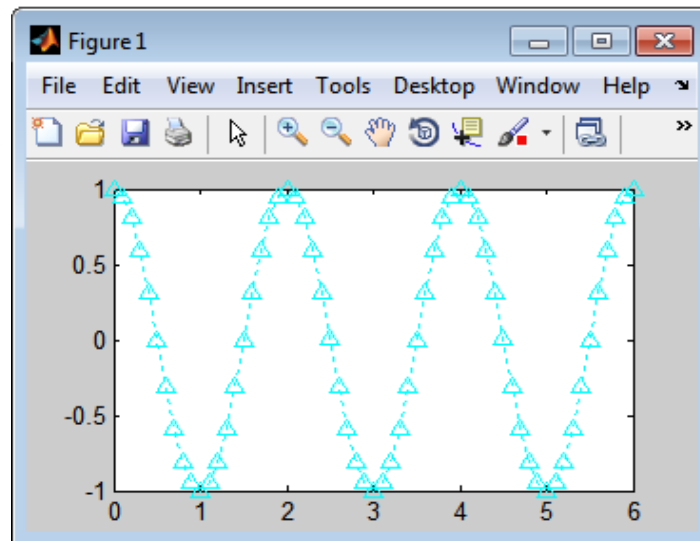


Figure 4.6 Courbe avec couleur cyan, en pointillé avec des triangles supérieur.

#### 4.2.3. Annotation d'une figure

Pour une meilleure compréhension du graphe, il est préférable d'ajouter des informations dans le graphique.

##### 1. Titre de la figure :

On utilise la fonction « **title** » pour donner un titre à la figure utilisé comme ceci :

```
>> title('titre de la figure')
```

##### 2. La commande « xlabel » et « ylabel » :

On utilise la commande « **xlabel** » pour donner un nom à l'axe des abscisses et « **ylabel** » pour donner un nom à l'axe des ordonnées de la figure utilisé comme ceci :

```
>> xlabel ('Ceci est l'axe des abscisses X')
```

```
>> ylabel ('Ceci est l'axe des ordonnées Y')
```

### 3. La commande « text » :

Permet d'introduire un texte dans la fenêtre du graphique à une position indiquée par les coordonnées X et Y utilisée comme ceci :

```
>> text (x, y, 'veuillez respecter le message suivant')
```

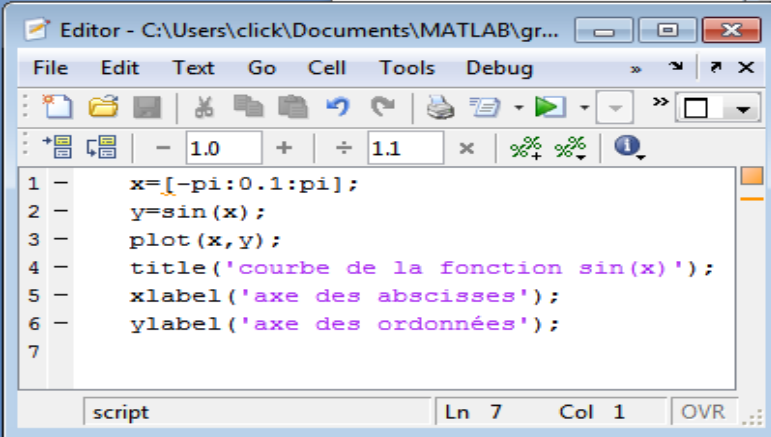
### 4. La commande « grid » :

Pour représenter une figure sous forme de grille on utilise la commande **grid** (ou **grid on**) et pour l'enlever on utilise la commande **grid** (ou **grid off**).

### Exemple :

Dans un nouveau script, écrire un programme qui permet de tracer le graphe de la fonction  $y=\sin(x)$  avec  $x=[-\pi \dots \pi]$  et un pas=0,1. Donnez un titre à la figure, un nom à l'axe des abscisses et l'axe des ordonnées.

### Solution :



```
1 - x=[-pi:0.1:pi];
2 - y=sin(x);
3 - plot(x,y);
4 - title('courbe de la fonction sin(x)');
5 - xlabel('axe des abscisses');
6 - ylabel('axe des ordonnées');
7
```

