

# Chapitre 2

## Graphisme

### Graphisme 2D

#### La commande plot

La commande plot permet de tracer l'ensemble des points de coordonnées  $(x_i, y_i)$ ,  $i = 1, \dots, N$ . La syntaxe est `plot(x,y)` où  $x_i$  sont les points de l'abscisse et  $y_i$  sont les points de l'ordonnée. Il est à noter que les vecteurs x et y doivent être de même dimension mais. Par défaut, les points  $(x_i, y_i)$  sont reliés entre eux par des segments de droites.

#### Exemple 1

```
1 x=[-2*pi:0.01:2*pi]; y = x.*exp(x);  
2 plot(x,y)
```

On peut spécifier à MATLAB la couleur de la courbe, le style de trait et/ou le symbole de chaque point  $(x_i, y_i)$ . Pour cela on donne un troisième paramètre d'entrée à la commande plot qui est une chaîne de caractère de la forme 'cst' avec c désignant la couleur du trait, s le symbole des points  $(x_i, y_i)$  et t le type de trait. Les possibilités sont résumés dans le tableau suivant.

Couleur du trait	style de trait	le symbole du point $(x_i, y_i)$
y jaune	. point	- trait plein
m magenta	o cercle	: pointillé court
c cyan	x marque x	- pointillé long
r rouge	+ plus	-. pointillé mixte
g vert	* étoile	
b bleu	s carré	
w blanc	d losange	
k noir	v triangle (bas)	
	^ triangle (haut)	
	< triangle (gauche)	
	> triangle (droit)	
	p pentagone	
	h hexagone	

La commande `grid` permet d'obtenir un quadrillage de la figure. Il est possible de tracer plusieurs courbes sur la même figure en spécifiant plusieurs vecteurs  $x_1, y_1, x_2, y_2, \dots$ , comme paramètres de la commande plot. Pour avoir chaque courbe ai une apparence différente, on utilisera les options de couleurs et/ou de styles de traits distincts après chaque couple de vecteurs  $x_i, y_i$ .

#### Exemple 2

```

1 x = [-10:0.01:10]; y = x.^2.*sin(x); z = x.*sin(x);
2 plot(x,y,'b-',x,z,'r:'); grid

```

## Amélioration et Annotation d'une figure

Il est préférable d'améliorer la lisibilité d'une figure et de mettre une description textuelle aidant à comprendre la signification des axes.

Commande	description
xlabel('Axe des abscisses X')	Pour donner un titre pour l'axe horizontal des abscisses x
ylabel('Axe des abscisses Y')	Pour donner un titre pour l'axe vertical des ordonnées y
title('titre de la figure')	Pour donner un titre à une figure
legend('chaîne de caractères',pos)	Pour ajouter une légende à un graphique, où pose = -1,0,1,2,3,ou 4
text(x,y,' du texte ')	Pour écrire un texte sur la fenêtre graphique à une position de coordonnées x et y
grid (ou grid on)	Pour mettre un quadrillage (une grille)

## Afficher plusieurs courbes dans une même fenêtre

```

1 figure
2 hold on
3 fplot('sin',[-pi pi])
4 fplot('x',[-1 1])
5 grid

```

## Décomposition d'une fenêtre en sous-fenêtres

Il est possible de décomposer une fenêtre en sous-fenêtres et d'afficher une figure différente sur chacune de ces sous-fenêtres grâce à la commande `subplot`. La syntaxe est la suivante :

`subplot(m,n,i)`

- m est le nombre de sous-fenêtres verticalement
- n est le nombre de sous-fenêtres horizontalement
- p sert à spécifier dans quelle sous-fenêtre doit s'effectuer l'affichage. Les fenêtres sont numérotées de gauche à droite et de haut en bas.

### Exemple 3

```

1 figure
2 subplot(2,2,1), fplot('cos',[0 2*pi]), title('cosinus'), grid
3 subplot(2,2,2), fplot('sin',[0 4*pi]), title('sinus'), grid
4 subplot(2,2,3), fplot('x^2',[-1 1]), title('x^2'), grid
5 subplot(2,2,4), fplot('x',[-1 1]), title('x'), grid

```

## Graphisme 3D

### Traçage des lignes de niveau d'une fonction de 2 variables

La commande `contour` permet de tracer les lignes de niveau d'une fonction de 2 variables réelles. Cette fonction peut être définie par une expression MATLAB, ou comme une fonction utilisateur. Pour tracer les lignes de niveau de la fonction  $f(x, y)$  on procède comme suit :

- Création d'un maillage, par à la commande `meshgrid`  
 $[X,Y] = \text{meshgrid}(xmin :h :xmax, ymin :h :ymax).$
- Evaluation de la fonction aux noeuds de ce maillage  $Z = f(X, Y)$
- Affichage des lignes de niveau grâce à la commande `contour`, `contour(X,Y,Z)`.

### Exemple 4

```
1 [X,Y] = meshgrid(-pi:pi/10:pi);  
2 Z = sin(X.*Y);  
3 contour(X,Y,Z)
```

## Représentation d'une surface d'équation $z = f(x, y)$

La commande `mesh` permet de tracer une surface d'équation  $z = f(x, y)$ . Pour tracer la surface d'équation  $z = f(x, y)$  on procède comme suit :

- Création d'un maillage, par à la commande `meshgrid`  
`[X,Y] = meshgrid(xmin :h :xmax, ymin :h :ymax)`.
- Evaluation de la fonction aux noeuds de ce maillage  $Z = f(X, Y)$
- Affichage de la surface grâce à la commande `mesh`, `mesh(X,Y,Z)`.

```
1 [X,Y] = meshgrid(-pi:pi/10:pi);  
2 Z = sin(X.*Y);  
3 mesh(X,Y,Z)
```