
TP 04: Les graphes

Exercice 1.

La commande de base permettant de dessiner des graphes est `plot`. Sous sa forme la plus simple, `plot` permet de représenter les valeurs d'un tableau :

```
>> plot([1, 2, 4, 8]);
>> plot([1; 2; 1; 2]);
```

Si l'argument de `plot` est une matrice à plusieurs lignes, Matlab trace chacune des colonnes séparément :

```
>> m = [1, 2; 1, 3; 1, 2]
>> plot(m)
```

On peut également modifier l'apparence du tracé en ajoutant un argument final composé de symboles cryptiques entre guillemets simples :

```
>> plot([1, 1, 1, 2, 4, 8, 4], 'r+--')
>> plot([2, 3, 4, 5, 3, 1], 'md:')
```

La commande `help plot` donne plus de détails, en particulier la liste des codes permettant de choisir la couleur du tracé, la forme des marqueurs des points (croix, cercle, losange, ...) et le type de tracé (pointillé, plein, tirets, etc.).

On peut également (et c'est ce que l'on va utiliser le plus souvent par la suite) représenter les valeurs d'un tableau `y` en fonction d'un tableau `x` :

```
>> x = 0:1:1
>> y = x.^2
>> plot(x, y)
```

1. Dans l'exemple précédent, que fait la ligne `y = x.^2` ? Quelle différence y a-t-il entre `x^2` et `x.^2` ?

Réponse :

Exercice 2.

Soit un vecteur y contenant des valeurs comprises entre -6π et 6π avec un pas de 0.001. Soit deux fonctions h et g définies par :

$$h(x) = \sin(\pi/4x) \quad \text{et} \quad g(x) = \cos(\pi/4x)$$

Ecrire un script Matlab représentant h et g en fonction de y sur le même graphe.

Réponse:

Gestion des axes

Les différentes fonctions suivantes permettent de gérer les labels des axes et commentaires sur les figures, ainsi que diverses fonctions pour manipuler les graphiques.

- title : title('Texte du titre') Ajoute un titre à la figure.
- xlabel : xlabel('Unité des x')
- ylabel : ylabel('Unité des y')
- legend : legend('Nom de la courbe 1' , 'nom courbe 2' , ...)
- grid : grid on, grid off, quadrille ou non le graphique.
- clf : efface la figure en cours d'utilisation.
- ginput : ginput(n) récupère les coordonnées de n points cliqués à la souris dans la figure en cours.

Exercice 3.

Graphes 3D

Il est également possible de demander à Matlab de tracer des courbes ou des surfaces en 3 dimensions à l'aide des commandes plot3, ezmesh et ezsurf.

Recopiez la fonction suivante dans un script intitulé: troisdim.m

```
function res = troisdim(x, y)
r = sqrt(x.^2 + y.^2);
res = sin(r) ./ r;
```

1. En regardant un peu la définition de la fonction troisdim(x, y), essayez de vous représenter la forme de la surface qu'elle définit.

Représentez la surface sous la forme d'un maillage à l'aide de l'instruction:

```
>> ezmesh(@troisdim)
```

Si vous voulez changer les axes, la fonction axis fonctionne toujours...

Si vous voulez représenter la surface de manière plus lisse, utilisez les commandes

```
>> ezsurf(@troisdim)
>> shading interp
```