

## TP 2

# INFORMATIQUE 3

### 1- Objectif

Dans ce TP nous allons manipuler des matrices, résoudre un système linéaire, manipuler des polynômes, créer /compiler des programmes (\*.m) et tracer des courbes.

### 2- Manipuler une matrice

2.1. Créer une matrice nommée A, comme suit :

$$\begin{bmatrix} 1 & 2 & 3 \\ 6 & 5 & 4 \\ 7 & 8 & 9 \end{bmatrix}$$

2.2. Déterminer la valeur de la 2<sup>ème</sup> ligne du 2<sup>ème</sup> colonne de la matrice A.

2.3. Extraire la 2<sup>ème</sup> ligne de la matrice A. Utiliser les parenthèses et les deux points ( :).

2.4. Déterminer la matrice transposée de A.

2.5. Exécuter les instructions suivantes :

```
A=eye(3)
diag(A)
```

Quel est le rôle des instructions « eye » et « diag » ?

### 3- Résoudre un système linéaire

Nous allons maintenant utiliser le formalisme matriciel pour résoudre un système d'équations. Le système que nous cherchons à résoudre est le suivant :

$$\begin{cases} 3x + 5y + z = 1 \\ 7x - 2y + 4z = -3 \\ -6x + 3y + 2z = 3 \end{cases}$$

Afin de résoudre le système ci-dessus, on définit alors la matrice A à partir des coefficients de ce système, comme :

$$A = \begin{bmatrix} 3 & 5 & 1 \\ 7 & -2 & 4 \\ -6 & 3 & 2 \end{bmatrix}$$

Si on note  $X = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$ , on peut voir que les trois lignes du produit  $A.X$  correspondent aux trois

lignes du système. On peut alors définir la matrice  $b = \begin{pmatrix} 1 \\ -3 \\ 3 \end{pmatrix}$  tel que  $A.X=B$ .

Afin de résoudre le système linéaire, il ne reste plus qu'à inverser la matrice  $A$ . On a donc  $X=A^{-1}B$ .

Il existe deux méthodes pour implémenter de ce calcul sous Matlab. La première méthode correspond à ce que nous avons appris précédemment. C.-à-d., on doit définir la matrice  $A$  et la matrice  $b$ , calculer l'inverse de  $A$  puis faire le produit avec  $b$ .

Compiler le code ci-dessous afin de résoudre le système linéaire.

```
A = [3 5 1 ; 7 -2 4 ; -6 3 2];  
b = [1 -3 3]';  
X=inv(A)*b
```

La seconde méthode est plus simple. Elle introduit la division à gauche. Lorsque l'on note sous Matlab  $A \backslash b$  cela signifie  $inv(A) * b$ . C'est la division à gauche. Résoudre le système considéré avec cette deuxième méthode.

#### 4- Manipuler un Polynôme

Au niveau du MATLAB, un polynôme est une liste de coefficients ordonnés par ordre décroissant :

**Exemple :**

Soit le polynôme  $p(x) = 1 - 2x + x^2 + 4x^3$ .

3.1. Exécuter les instructions suivantes :

```
>> p = [ 4 1 -2 1]  
p =  
    4    1   -2    1  
>> polyval(p,0) = [ 4 1 -2 1]    % polyval permet d'évaluer le polynôme en un point ou des points donnés.  
ans =  
     1  
>> polyval(p,[1 2])  
ans =  
     4    33
```

3.2. Calculer les racines du polynôme avec la commande `roots` :

```
>> p = [ 4 1 -2 1];  
>> roots(p)    % Calcul des racines du polynôme p.  
ans =  
-1.0000 + 0.0000i  
 0.3750 + 0.3307i  
 0.3750 - 0.3307i
```

3.4. Effectuer la multiplication de deux polynômes via la commande `conv(x1,x2)`, c'est à dire les coefficients du produit des deux polynômes, soit l'exemple :

```
>> x1 = [2 3];  
>> x2 = [1 4];  
>> conv(x1,  
x2)ans =  
     2    11    12
```

## 5- Création de fichiers (\*.m) et graphisme 2D

MATLAB en plus de ses grandes possibilités de calcul numériques produit des graphiques en 2 ou 3 dimensions.

La fonction plot (x,y) permet de tracer une courbe liant un ensemble de valeurs (vecteur) y en fonction d'un autre vecteur x (bien entendu de même dimension).

Dans l'exemple suivant on se propose de tracer la fonction suivante :

$$y = \cos(2x) + 2 \sin(0.1x).$$

La variable x est un vecteur dont les valeurs vont de  $-\pi$  à  $+\pi$  avec un pas de  $\pi/100$ .

5.1. Aller dans la barre du menu MATLAB, sélectionner *New Script* (ou *Ctrl+N*), donner un nom au fichier d'extension \*.m puis sauvegarder.

5.2 Appuyer sur *Run* (ou F5) pour compiler.

```
x = -pi : pi/200 : pi;  
y = cos ( 2 * x ) + 2 * sin ( 0.1 * x );  
plot ( x , y )  
grid  
xlabel ( ' variable x ' )  
ylabel ( ' variable y ' )  
title ( ' y =cos ( 2 x)+2 sin ( 0.1 x ) ' )
```

5.3 Afficher la figure.

5.4 On suppose qu'on a une autre fonction  $z=2*y$ . Tracer sur la même figure  $y(x)$  et  $z(x)$ .

5.4 Quel est le rôle des instructions « plot » et « grid » ?

## 6- Graphique 3D

Les graphes en des fonctions de deux variables sont légèrement plus compliqués à manier que ceux à une variable. En effet, de la même façon que l'on avait créé un vecteur qui discrétisait l'axe des abscisses, on doit créer un maillage qui discrétise le plan : c'est le but de la fonction **meshgrid**.

```
x = -10:0.1:10;  
y = -5:0.1:5;  
[X,Y] = meshgrid(x,y);  
g = exp(-X.^2).*exp(-Y.^2);  
figure(1)  
surf(x,y,g)
```

6.1 Créer un nouveau programme puis réaliser l'exemple ci-dessus.

6.2 Que est le rôle de 1<sup>er</sup>, 2<sup>ème</sup> et 3<sup>ème</sup> ligne.

6.3 Rajouter la ligne ci-dessous puis exécuter.

```
mesh(x,y,g)
```

Que fait cette dernière ?

6.4 Rajouter la ligne ci-dessous puis exécuter.

```
contour(x,y,g)
```

Que fait cette dernière ?