LES NOMBRES COMPLEXES, LES POLYNOMES ET LES GRAPHES

I. But

Au terme de ce TP, vous serez en mesure de déclarer, de manipuler, les nombres complexes, les polynômes et de tracer des courbes planes.

II. Les complexes

II.1 Les nombre complexes

II.1.1 Les parties réelle et imaginaire d'un nombre complexe

Les parties réelle et imaginaire du nombre complexe **Z** sont obtenues respectivement avec les commandes **:** >> **real(Z), imag(Z).**

- **Exercice 1 :**

✓ Trouvez les parties réelle et imaginaire de chacun des nombres complexes suivants :

Z1 = 1 – 2j, Z2 = - 9j, Z3 = -1 – 2j, Z4 = – 2.5j, Z5 = -1
Z6 = 5 e^{-5j} , Z7 = $\notin 9e^{-3j\neq/4}$ Z8 = 2 + $\notin 7e^{-2j\neq/3}$

II.1.2 Le module et l'argument d'un nombre complexe

Le module et l'argument du nombre complexe : $Z = x + jy = re^{j\theta}$
sont obtenus respectivement avec les commandes: >> $r = abs(Z)$ *et* >> $\theta = angle(Z)$
Tandis que l'expression : $\mathbf{Z} = r\,\mathbf{e}^{j\theta}$ convertit Z à sa forme cartésienne, c'est à dire : Z = x + j*y

- **Exercice 2 :**

✓ Trouvez le module et l'argument des nombres complexes suivants :

Z1 = 1 – 2j, Z2 = - 9j, Z3 = - 1 – 2j, Z4 = – 2.5j, Z5 = -1

✓ Mettez les nombres complexes suivants sous la forme cartésienne :

Z1 = 5 e^{-5j} , Z2 = $9e^{-3j\neq/4}$ Z3 = 2 + $7e^{-2j\neq/3}$

II.1.3 Le conjugué d'un nombre complexe

le conjugué du nombre complexe Z est obtenu avec la commande : >> **conj(Z).**

- **Exercice 3 :**

✓ Trouvez le conjugué de chacun des nombres complexes suivants :

Z1 = 1 – 2j, Z2 = - 9j, Z3 = -1 – j, Z4 = – 2.5j, Z5 = -1
Z6 = 5 e^{-5j} , Z7 = $\notin 9e^{-3j\neq/4}$ Z8 = 2 + $\notin 7e^{-2j\neq/3}$

II.2 Les matrices complexes

II.2.1 Les parties réelle et imaginaire d'une matrice complexe

Les parties réelle et imaginaire d'une matrice complexe sont obtenues de la même manière que précédemment.

➢ Exercice 4 :

✓ Trouvez les parties réelle et imaginaire des deux matrices suivantes :

A1= [2, -3j ; 1+1j, -6], A2= [2+2j, 1+3j ; 4+5j, 6-j]

II.2.2 Le module et l'argument

abs(A) : Vous renvoie une matrice, dont les éléments sont les valeurs absolues de chaque élément de A. Mais si A est complexe **abs(A)** vous donne le module de A, c'est à dire :

abs(A)=sqrt [real(A). ^2 +imag(A).^2].

angle(A) : Vous retourne les arguments en radian des éléments de la matrice A. Les angles sont donnés entre –≠ et +≠

➢ Exercice 5 :

✓ Trouvez le module et l'argument des deux matrices précédentes :

II.2.3 La conjuguée d'une matrice complexe

La matrice conjuguée de la matrice complexe A est la matrice dont les éléments sont les conjugués de ceux de A. Cette dernière est obtenue aussi avec la commande >> **conj(A).**

II.2.4 La transposée d'une matrice complexe

La matrice transposée d'une matrice complexe est obtenue avec **:** >> **A.'**

II.2.5 L'associée d'une matrice complexe

La matrice associée d'une matrice complexe A est la matrice conjuguée de sa transposée.
On l'obtient avec la commande : >> **A'.**

➢ Exercice 6:

✓ Trouvez la conjuguée, la transposée et l'associée de chacune des matrices suivantes :

X1 = [j 1 ; -6j 1+2j], X2 = [-j -8j ; -1 1-3j], X3 = [-5j 0; -1+6j 1-3j].

III. Les polynômes

III.1 Déclarations des polynômes

Les polynômes sont représentés comme des vecteurs ligne qui contiennent les coefficients dans l'ordre décroissant de puissance. Par exemple pour introduire les deux polynômes suivants :

➢ Exercice 7:

Introduisez les deux polynômes suivants :

P1(p) = p^2+4p+1 **et P2(x) =** $3x^4 + 5x^2 + 4x - 7$**;**

III.2 Les racines d'un polynôme

Les racines d'un polynôme sont les éléments du vecteur colonne qui résulte de la commande :
>> roots ([Vecteur des coefficients du polynôme P]).
Les racines des deux derniers polynômes sont obtenues en exécutant :

> ☞ **N.B**: On peut reconstruire n'importe quel polynôme en partant de ses racines, en utilisant la commande : *>> poly ([Vecteur colonne des racines])*

➢ Exercice 8:

Trouvez les racines des polynômes P_1 et P_2 , puis reconstruisez ces polynômes à partir de leurs racines:

>> **R1 = roots (P 1) , R2 = roots (P 2)**

>> **P1 = poly (R1), P2 = poly (R2)**

III.3 Le produit de deux polynômes

Le produit de polynômes est la convolution de leurs coefficients. Donc, C(s) = A(s) * B(s), peut être obtenu en exécutant : **C = conv (A , B).**

➢ Exercice 9:

Calculez les produits : $\mathbf{P_1(s) = A_1(s)*A_2(s)}$, $\mathbf{P_2(p) = A_3(s)^3}$, des polynômes suivants :

$\mathbf{A_1(s) = 3s^2 - 1.5}$, $\mathbf{A_2(s) = s^2 + 4s + 5}$, $\mathbf{A_3(s) = s + 1}$. Pour cela introduisez **:**

>> **A_1= [3 0 –1.5] , A_2 = [1 4 5], A_3 = [1 1];**

>> **P_1 = conv (A_1 ,A_2)** vous retourne : >>**P_1 = [3.0000 12.0000 13.5000 -6.0000 -7.5000].**

Cela n'est autre que la représentation de Matlab de : $P(s) = 3s^4 + 12s^3 + 13.5s^2 - 6s - 7.5$.

P_2 = ?

III.4 La division de deux polynômes

Pour diviser le polynôme **$P_1(s)$** par **$A_2(s)$,** utilisez la commande : **[q ,r] = deconv [P_1 ,A_2].**

Cette dernière produira **:** **q = [1 4 5]** **et** **r = [0 0 0 0 0]**

Où **q** et **r** représentent respectivement le quotient et le reste de la division.

III.5 L'évaluation polynomiale

Si P est un vecteur dont les éléments sont les coefficients du polynôme P(s), suivant ses puissances décroissantes, alors : **polyval (P, s_0)** est la valeur du polynôme P(s) au point **s_0 ,** c'est à dire P(s_0).

➢ Exercice 10 :

Trouvez la valeur du polynôme $\mathbf{P(s) = 3s^2 + 2s + 1}$ au point $\mathbf{s_0 = 5}$.

La commande **polyvalm(P ,A)** évalue le polynôme P au sens matriciel. Considérons la matrice J suivante :

$$J = \begin{bmatrix} -2+j2\sqrt{3} & 0 & 0 \\ 0 & -2-j2\sqrt{3} & 0 \\ 0 & 0 & -10 \end{bmatrix}$$

la commande ..>> **P = poly (J),** donne le polynôme caractéristique de J, c'est à dire :

>> **P = [1 14 56 160]**. Ceci est l'expression de Matlab du polynôme caractéristique de J.

Poly (J) $= \Phi(J) = J^3 + 14J^2 + 56J + 160I$, où I est la matrice unité. Pour la matrice A :

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -6 & -11 & -6 \end{bmatrix}$$ la commande **polyvalm (poly (J) , A),** évalue l'expression

Φ (A) suivante : $\Phi(A) = A^3 + 14A^2 + 56A + 160I = \begin{bmatrix} 154 & 45 & 8 \\ -48 & 66 & -3 \\ 18 & -15 & 84 \end{bmatrix}$.

IV. Les graphiques

Matlab peut afficher des courbes planes, des courbes 3D, des surfaces maillées 3D ou des surfaces à facettes 3D. Les commandes respectives sont **plot**, **plot3**, **mesh** et **surf**.
Il dispose aussi d'un ensemble de routines, qui donnent des sorties graphiques. en coordonnées linéaires, logarithmiques et polaires. Ces deux derniers types sont obtenus en substituant la commande **plot** relative aux échelles linéaires par **loglog, semilogx, semilogy** ou **polar**. Toutes ces commandes sont utilisées de la même manière, sauf qu'elles affectent juste l'échelle et de là, la manière d'affichage du graphe.

IV.1 Courbes planes

La commande **plot**(*y*) trace la courbe correspondant au vecteur *y* en fonction des indices de ses éléments. **plot** (t , *y*) tracera le vecteur *y* en fonction du vecteur *t* si ils sont de même taille. La commande **plot** ouvre une fenêtre graphique pour tracer la courbe. Les fenêtres graphiques sont numérotées, la fenêtre active étant par défaut la fenêtre 1. Pour rendre active une fenêtre graphique on fera précéder le tracé par la commande **figure(n)**, où **n** est le numéro de la fenêtre.
>> **x = -pi : .01 : pi; y = sin(x);   plot(x,y)**
tracera la fonction **sinus(x)** dans l'intervalle **[-pi, +pi]** avec un pas de **0.01**.
Une deuxième commande **plot** effacera le premier graphique pour tracer le nouveau dans la même fenêtre si la commande **figure** n'a pas été utilisée entre temps.
Matlab permet également de tracer directement à partir d'une fonction avec la commande **fplot**. Soit la fonction sinus définie dans le fichier **sinus.m** par:

function y = sinus(x)
y = sin(x);
En exécutant: **fplot**('sinus', [-pi pi]), on obtiendra le graphe.

IV.2 Comment tracer plusieurs courbes sur une même figure

Matlab permet de tracer plusieurs courbes sur un même figure, en utilisant la commande **plot** avec plusieurs arguments : >> **plot (t1 , y1 , t2 , y2 ,…..,tn , yn).** Les variables **t1, y1, t2, y2,…..,tn, yn,** etc… sont des paires de vecteurs où chaque paire *(ti, yi)* sera représentée par une courbe.

Cependant il est possible également de tracer des graphiques en surimpression à l'aide de la commande **hold on**. Cette commande a pour effet de conserver le graphe présent avant de tracer le suivant. La commande **hold off** annule cette possibilité.

IV.3 Les attributs d'une figure

Le figure obtenue peut être agrémentée d'un titre, de légendes, de texte,ou quadrillée. Les commandes Matlab correspondantes sont:

- **title**(' Ceci est le titre de ma figure ');
- **xlabel**(' Légende de l'axe des abscisses ');
- **ylabel**(' Légende de l'axe des ordonnées ');
- **gtext** et **text :** positionnement et écriture dans la zone graphique.
- **grid** : Pour quadriller une figure,

IV.4 Comment écrire du texte sur un graphe

Pour écrire du texte à partir du point (x_0 ,y_0) sur un graphe utiliser soit la commande :
>>text (x0 , y0, ' Votre texte '), soit **gtext(' Votre texte ')**
Par exemple l'expression : **text (3, 0.45, ' sin (t) ')**, écrit **sin (t)** horizontalement à partir du point **(3 , 0.45)**. tandis que la commande **gtext(' Votre texte ') ,** vous permet de choisir avec votre souris le point (x_0 ,y_0) à partir du quel vous voulez écrire votre texte. Une fois votre choix est fait, cliquez sur le bouton gauche de votre souris, Matlab insère le texte se trouvant entre les deux apostrophes.

Les commandes : **plot (x1, y1, x2, y2) , text (x1,y1, ' 1 '), text (x2, y2 , ' 2 ')** marquent les deux courbes de sorte qu'elles soient facilement différentiables, voir les Exercices 2.1 et 2.2.

IV.5 Les attributs d'un graphe

Un graphe possède des attributs qui peuvent être modifiés à l'aide de commandes, on consultera l'aide de **axis** et **plot**. C'est à dire qu'il est possible de spécifier la couleur, le style de ligne et le type de marqueur avec la commande: >> *plot (t , y , ' Couleur_Style_Marqueur').*
Où couleur, style et type de marqueur sont respectivement un caractère des trois listes suivantes :

N°	Couleur		Le type de marqueur		Le style de ligne	
1	**y**	yellow	**.**	point	**-**	solid
2	**m**	magenta	**o**	circle	**:**	dotted
3	**c**	cyan	**x**	x-mark	**-.**	dashdot
4	**r**	red	**+**	plus	**--**	dashed
5	**g**	green	*****	star		none
6	**b**	blue	**s**	Square *		
7	**w**	white	**d**	Diamond *		
8	**k**	black	**v**	triangle (down) *		
9			**^**	triangle (up) *		
10			**<**	triangle (left) *		
11			**>**	triangle (right) *		
12			**p**	Pentagram *		
13			**h**	Hexagram *		

***N.B* : (*) Propre à la version 5.3**

IV.6 Les sous graphes

Il est également possible de tracer plusieurs graphes dans la même fenêtre à l'aide de la commande **subplot** *(voir aide)*. La commande **subplot** vous permet d'afficher plusieurs graphes sur une même fenêtre ou les imprimer sur une même feuille de papier.
L'exécution de subplot(m , n , p), partagera la figure active en m*n parties égales et sélectionne la p^{ième} partie pour le graphe en cours. Les figures sont numérotées de la gauche à la droite et de la première ligne à la m^{ième} ligne voir Exercice (   ).

IV.7 Les courbes polaires

Polar(theta,rho) : donnera la courbe en coordonnées polaire de l'angle ***theta*** en fonction du rayon ***rho***. L'utilisation de de la commande ***grid*** produit un quadrillage polaire de la figure.

IV.8 Les courbes logarithmiques

Les commandes :
- **loglog :** Dessine la courbe en coordonnées logarithmiques, c'est à dire que les axes sont gradués en $\log_{10}$

- **semilogx :** Trace la courbe en échelle semi-logarithmique. L'axe des x est gradué en log10 tandis que l'axe des y est gradué linéairement.
- **semilogy :** Trace la courbe en échelle semi-logarithmique. L'axe des y est gradué en

log10 tandis que l'axe des y est gradué linéairement.

N.B : si Z est un vecteur complexe, alors plot(Z) est équivalent à plot (real(Z), imag(Z)).

Exercices:

```
% Exemple 2.1
t=(0:0.001:1)';
y1=[1;zeros(99,1)];
y2=ones(100,1);
y3=t;
y4=1/2*t.^2;
y5=square(4*t);
plot(y1),pause,plot(y2),pause,
plot(t,y3),pause,plot(t,y4),pause
plot(t,y5)
```

```
% Exemple 2.2

t=-pi:0.05:+pi;
y=sin(t);z=t.*exp(-t);
figure(1)
plot(t,y)
grid
figure(2)
plot(t,z)
```

```
% Exemple 2.3

hold on
y=5*ones(1,100);
plot(y)
t=0:0.1:100;
z=t.^2.*cos(t);
plot(t,z)
hold off
```

```
% Exemple 2.4

t=linspace(-5,5);
y=sinc(t);
plot(t,y)
grid
xlabel('t[Sec]')
ylabel('Sinc')
title('Graphe de sinus cardinal ')
```

```
% Exemple 2.5

t=0 :0.25 :10 ;
y=sin(t) ;z=cos(t) ;
plot(t,y,t,z), text(t,y,'1'), text(t,z,'2')
grid
title (' LES COURBES DU SINUS ET DU COSINUS ')
xlabel ('[Sec]')
ylabel ('y=sin(t) , z=cos(t)')

% Exemple 2.6
t=0 :0.05 :10 ;
y=sin(t) ; z=cos(t) ;
plot(t,y,'0',t,z,'x')
grid
title (' LES COURBES DU SINUS ET DU COSINUS ')
xlabel ('[Sec]')
ylabel ('y=sin(t) , z=cos(t)')
text(3,0.45,'sin(t)')
text(0.8,-0.3,'cos(t)')
% Exemple 2.7
x=0 :0.1 :3 ;
y=x.ˆ2 ;
plot(x,y)
grid
title (' LE GRAPHE DE Y= x.ˆ2')
xlabel ('x')
ylabel ('y')
gtext('y=x.^2')

% Exemple 2.8
f=10000;
t=0:1/f:1.5;
y=1+sawtooth(2*pi*50*t);
plot(t,y),axis([0 0.2 -0.5 +2])
grid
```