

TP 5 : Les fonctions et le graphique dans le MATLAB

1. Les fonctions

Les expressions arithmétiques nécessitent le plus souvent d'autres opérations que l'addition, la soustraction, la division et la multiplication. On trouve ainsi souvent des logarithmes, des exponentielles, ou des fonctions trigonométriques.

Quelques fonctions de base sont données dans le tableau suivant :

min valeur minimale	max valeur maximale	mean valeur moyenne	std écart type	cov covariance
sum somme	abs valeur absolue; module (nb. complexe)	angle argument (nb. complexe)	real partie réelle	imag partie imaginaire
conj conjuguée (nb. complexe)	round arrondir	fix arrondir (vers zéro)	floor arrondir (vers $-\infty$)	ceil arrondir (vers ∞)
sqrt racine carrée	rem reste	exp exponentielle	log logarithme base e	log10 logarithme base 10

Les fonctions trigonométriques sont données dans le tableau suivant:

sin	cos	tan
asin	acos	atan
sinh	cosh	tanh
asinh	acosh	atanh

Exemple 1 : soit un nombre complexe $x = -2 + 5i$

- donner la partie réelle, imaginaire, le module et l'angle pour x

Exemple 2 : soit $w = 50$ et $t = 0.5e-3$, calculer $y = 25\exp(-4t)\cos(wt)$

Exemple 3 : soit deux nombres complexes suivants. $u = 11 - 7i$ et $v = -1 + 3i$

- calculer les modules de u et v .
- calculer uv et $v\bar{u}$.
- calculer la partie réelle et imaginaire de $u^3 + v^2$.

1.1 Création d'une fonction

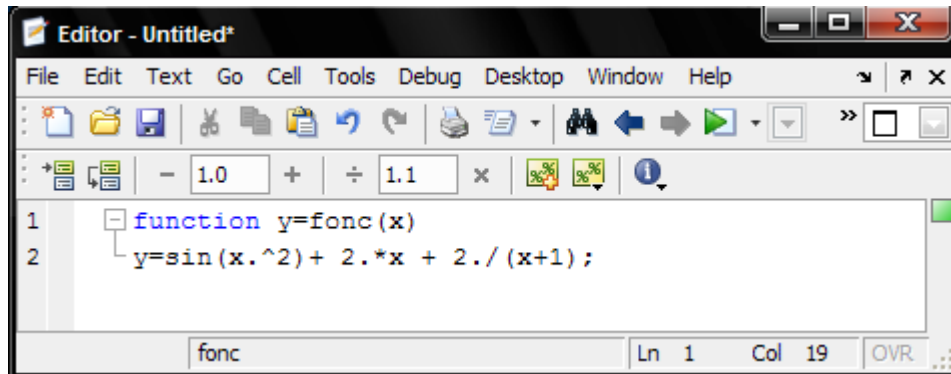
L'utilisateur peut créer les fonctions particulières pour ses applications.

Exemple : soit la fonction suivante

$$y = \sin(x^2) + 2x + \frac{2}{x+1}$$

- Commencer par l'ouverture de l'éditeur de texte : « M-file Editor ».
- Donner un nom à cette fonction (dans cet exemple **fonc**) et saisir son expression mathématique.

Attention : il faut mettre un point devant les opérateurs de multiplication, division et puissance : `.*` `./` `.^`



- Sauvegarder le fichier sous le nom **fonc**
- Faire un calcul pour **x = 5**

Solution

>> fonc(5)

2. Graphiques

Le graphique dans le matlab est utilisé pour le traçage des courbes.

2.1 Graphiques 2D

On utilise l'instruction **plot** pour tracer un graphique 2D, cette instruction est saisie comme suit :

`plot(x)` : Tracer le vecteur x

`plot(x, 'b*')` : Tracer le vecteur x en trait étoile bleu

`plot(x,y)` : Tracer le vecteur y en fonction du vecteur x

`plot(t,x,t,y,t,z)` : Tracer x(t), y(t) et z(t) sur le même graphique

`plot(t,z,'r--')` : Tracer z(t) en trait pointillé rouge

2.2 Format de graphique

On peut aussi choisir le format du graphique:

`plot(x,y)` : Tracer y(x) avec échelles linéaires

`semilogx(f,A)` : Tracer A(f) avec échelle log(f)

`semilogy(w,B)` : Tracer B(w) avec échelle log(B)

`polar(theta,r)` : Tracer r(theta) en coordonnées polaires

`bar(x,y)` : Tracer y(x) sous forme des barres

`grid` : Ajouter une grille

Exemple 1:

Tester les instructions suivantes :

```
>> t=0:0.01e-3:0.06;  
>> y=10*exp(-60*t).*cos(120*pi*t);  
>> z=10*exp(-60*t).*sin(120*pi*t);  
>> a=10*exp(-60*t);  
>> plot(t,y,'r',t,z,'g'),grid  
>> hold on
```

Remarque : "**hold on**" permet de tracer plusieurs graphes sur la même figure; sinon une deuxième commande plot (ou autre) effacera le graphe précédent pour effectuer la dernière commande.

```
>> plot(t,a,'b--')  
>> plot(t,-a,'b--')  
>> title('Fonctions sinusoidales amorties')  
>> xlabel('Temps , s'),ylabel('Tension , V')
```

Si vous voulez tracer un autre graphe, vous pouvez faire:

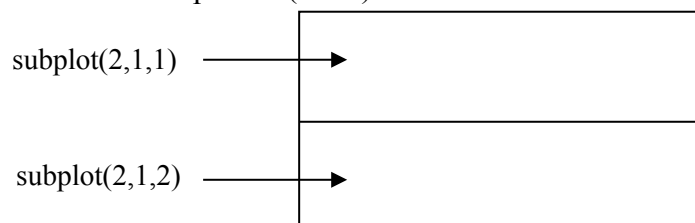
```
>> hold off  
>> plot(y,z),grid  
>> axis equal  
>> xlabel('y'),ylabel('z')
```

"**hold off**" : recommencera un nouveau graphe à la prochaine commande graphique (en restant dans la même fenêtre graphique),

2.3 Graphique multiple

On peut tracer plusieurs graphiques dans la même fenêtre en utilisant l'instruction **subplot** pour diviser la fenêtre en plusieurs parties.

- Diviser la fenêtre en deux parties (2 x 1)



Que signifient les instructions suivantes :

subplot(1,2,1), subplot(1,2,2), subplot(2,2,1), subplot(2,2,1) et subplot(4,1,1)

Exemple 2:

Tracer les deux fonctions suscitées $y(t)$ et $z(t)$ de l'exemple précédent en deux parties dans la même fenêtre.

Exemple 3:

Tester les instructions suivantes :

```
>> w=logspace(0,3,1000);
>> s=j*w;
>> H=225./(s.*s+3*s+225);
>> AdB=20*log10(abs(H));
>> phase=angle(H)*(180/pi);
>> subplot(2,1,1),semilogx(w,AdB),grid
>> xlabel('w , rad/s'),ylabel('Amplitude , dB')
>> subplot(2,1,2),semilogx(w,phase),grid
>> xlabel('w , rad/s'),ylabel('Phase , degre')
```

2.4 Ajout du texte au graphique

title('Titre du graphique') : Donner un titre au graphique

xlabel('Temps') : Étiquette de l'axe x

ylabel('Tension') : Étiquette de l'axe y

text(x,y,'Valeur absolue') : Ajouter du texte au graphique à la position x,y

gtext('Valeur absolue') : Ajouter du texte au graphique avec la souris

2.5 Manipulation de graphique

axis([-1 5 -10 10]) : Choix des échelles x = (-1,5) et y = (-10,10)

hold on : Garder le graphique sur l'écran (pour tracer plusieurs courbes sur le même graphique)

2.6 Graphique 3D

Le traçage des graphiques 3D est illustré par les deux exemples suivants :

Exemple 1 :

```
>> t = 0 : 0.05 : 25 ;
>> x = exp(-0.05*t).*cos(t) ;
>> y = exp(-0.05*t).*sin(t) ;
>> z = t ;
>> plot3(x,y,z), grid
```

Exemple 2 :

```
>> b = 1200*pi ;
>> dt = 50e-6 ;
>> for j = 1 : 15
    for i = 1 : 15
        k(j) = j ;
        a = (16 - j)*50;
        t(i) = (i - 1)* dt;
        y(j,i) = exp(-a*t(i)).*sin(b*t(i));
    end
end
>> [K,T] = meshgrid(k,t);
>> mesh(T,K,y)
```