

OBJECTIFS GENERAUX

- Se familiariser avec les logiciels de CAO d'automatique
- Savoir simuler des systèmes asservis.
- Savoir utiliser des logiciels d'automatique CAO pour le calcul des structures de commande.

OBJECTIFS SPECIFIQUES

- A l'issue de cette manipulation, l'étudiant doit être capable de :
- Réaliser la simulation temporelle d'un système asservi linéaire en utilisant l'outil SIMULINK.
 - Réaliser la simulation fréquentielle d'un système asservi linéaire en utilisant l'outil SIMULINK
 - Réaliser la simulation fréquentielle d'un système asservi linéaire en utilisant la programmation sous MATLAB.

PRE - REQUIS

- Notions sur l'informatique
- Notions sur l'étude temporelle des systèmes asservis.
- Notions sur l'étude fréquentielle des systèmes asservis

EQUIPEMENT ET ACCESSOIRES

- Micro ordinateur.
- Logiciel MATLAB.
- Imprimante.

Conditions de fonctionnement

- Les manipulations s'effectuent dans le laboratoire "d'automatique".
- La durée de la manipulation est de 3 heures.
- Les étudiants travaillent par groupes de 4 au maximum.
- Chaque groupe doit présenter au début de la séance une recherche bibliographique sur le thème de la manipulation.

PRESENTATION DU LOGICIEL :

MATLAB (MATrix LABoratory) est un système interactif et convivial de calcul numérique et de visualisation graphique destiné aux ingénieurs et scientifiques. Il possède un langage de programmation à la fois puissant et simple d'utilisation.

Dans MATLAB, l'élément de base est la matrice. L'utilisateur ne s'occupe pas des allocations mémoire ou de redimensionnement comme dans les langages classiques. Les problèmes numériques peuvent être résolus en un temps record, qui ne représente qu'une fraction infime du temps à passer avec d'autres langages comme le Basic, C, C++ ou le Fortran.

En complément de MATLAB, l'outil additionnel SIMULINK est proposé pour la modélisation et la simulation de systèmes dynamiques en utilisant une représentation de type graphique (schéma bloc).

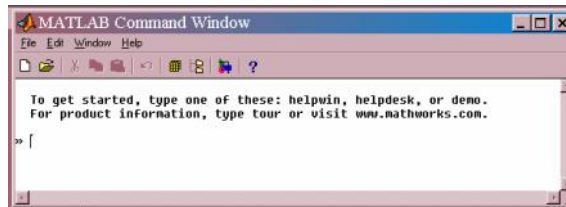
1. Initiation :

Programmation sous MATLAB

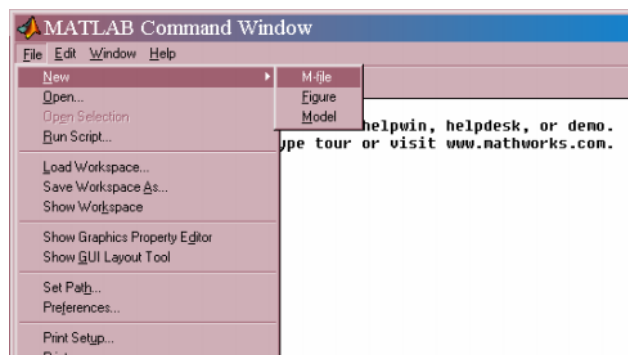
Si on veut développer un programme, on doit suivre les étapes suivantes :

- Lancer MATLAB en cliquant deux fois sur son icône sous Windows

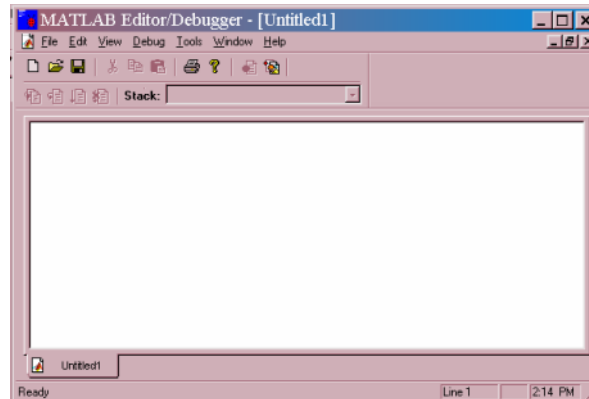
Une fenêtre « MATLAB command Windows » est alors affichée.



- Cliquer sur « File » et choisir « New » « Mfile »



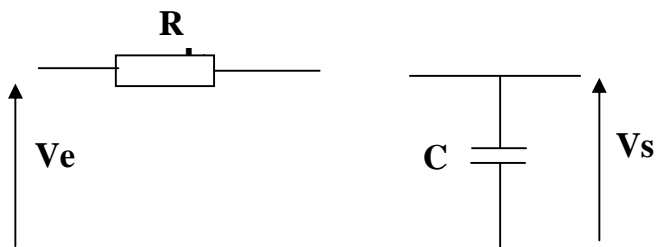
une nouvelle fenêtre est ouverte dans la quelle on peut écrire notre programme



- Donner un nom au programme qu'on enregistrera
- Pour l'exécution du programme taper son nom dans la fenêtre « MATLAB command Windows »

Exemple de programmation : Etude d'un circuit électrique RLC

Le circuit RLC suivant est attaqué par un signal sinusoïdal $e(t) = e^{j\omega t}$



Avec : $R = 100 \Omega$; $L = 0.1 \text{ H}$; $C = 1 \text{ mF}$
et en domaine de pulsation allant de 0 à 5000 rad/s.

Dans le domaine fréquentiel, ce circuit est défini par la fonction de transfert suivant :

$$H(j\omega) = \frac{V_s(j\omega)}{V_e(j\omega)} = \frac{1}{1 - LC\omega^2 + jRC\omega}$$

Nous désirons calculer et tracer la réponse en fréquence de cette fonction de transfert.

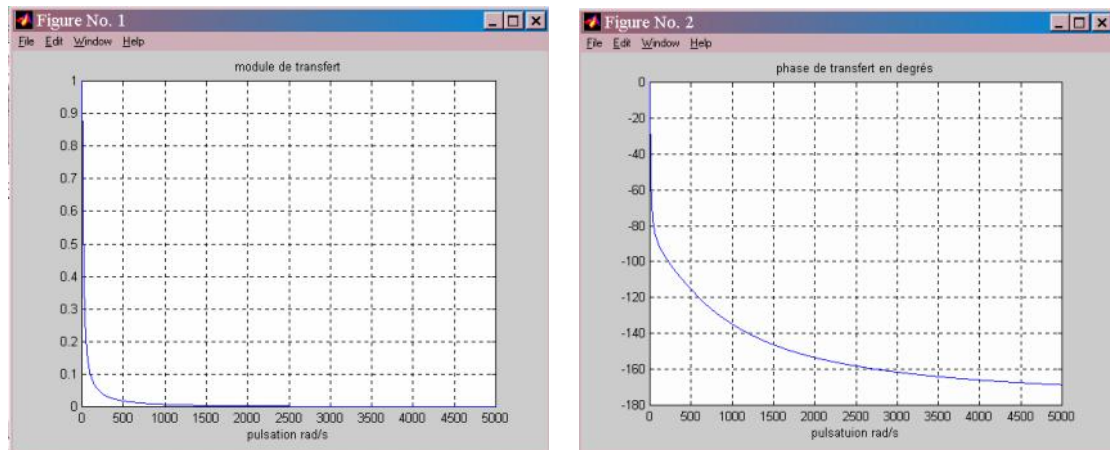
Pour ce la on doit développer le programme suivant :

```

MATLAB Editor/Debugger - [RLC.m - D:\Matlab\bin\RLC.m]
File Edit View Debug Tools Window Help
Stack:

%valeurs des composants du circuits
R = 100 ; C = 1e-3 ; L = 0.1 ;
%domaine des pulsations
w = 0:5000 ;
%fonction de transfert complexe
H = 1./ (1-L*C*w.^2+R*C*w*i);
figure(1)
%tracé du module de la fonction de transfert
plot (w,abs(H)),grid
xlabel ('pulsation rad/s')
title ('module de transfert')
figure(2)
%tracé de phase de la fonction de transfert
plot (w,180*angle(H) /pi)
grid, xlabel('pulsatuion rad/s')
title ('phase de transfert en degrés')
    
```

Résultat de l'exécution du programme



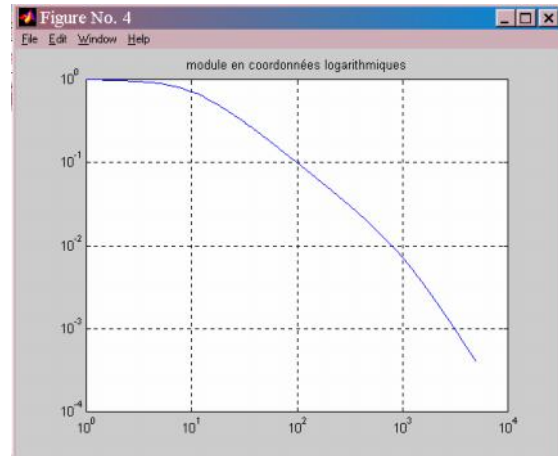
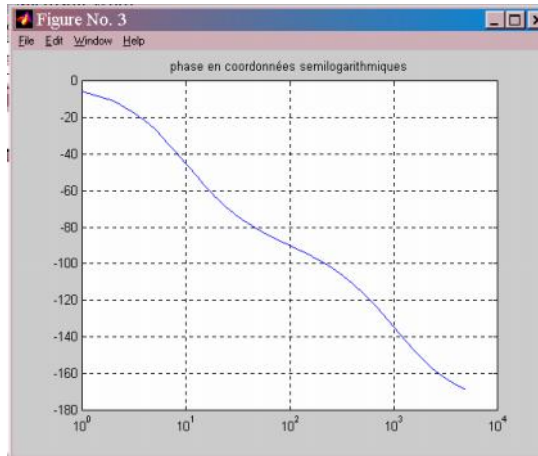
Lorsque le module décroît très rapidement, il est nécessaire d'utiliser les échelles logarithmiques pour la pulsation et le module.

Pour ce faire, il suffit d'ajouter les lignes suivantes au programme précédent :

```

% coordonnées logarithmiques
figure(3)
semilogx (w ; 180*angle (H) /pi)
grid , title ('phase en coordonnées semi logarithmiques')
figure(4)
loglog (w, abs(H))
grid, title ('module en coordonnées logarithmique')
    
```

Résultat de l'exécution du nouveau programme :

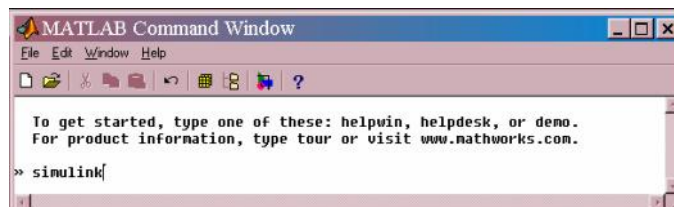


Utilisation de SIMULINK

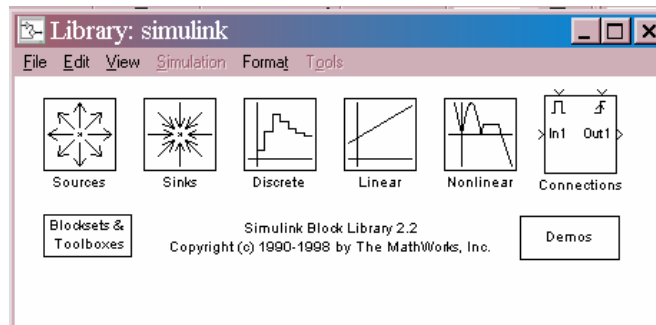
1 - Initiation

Si on veut utiliser le SIMULINK, on doit suivre les étapes suivantes :

- Lancer MATLAB en cliquant deux fois sur son icône sous Windows
- Taper la commande SIMULINK à partir de la fenêtre « MATLAB command Windows »



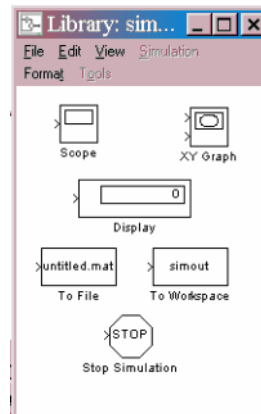
Une fenêtre est alors affichée sur la partie supérieure de l'écran. Elle contient les différentes familles de blocs disponibles dans la bibliothèque SIMULINK



Pour examiner le contenu d'une famille de blocs : pointer le rectangle correspondant et cliquer deux fois la souris, tous les blocs appartenant à cette famille seront affichés.

Pour fermer une fenêtre cliquer la souris sur le signe **x** en haut et à gauche de la fenêtre en question.

Par exemple en cliquant deux fois sur le bloc Sinks en aboutit au menu suivant :



Le contenu des autres blocs est :

Sources : générateurs de signaux (sinus, carré, etc...)

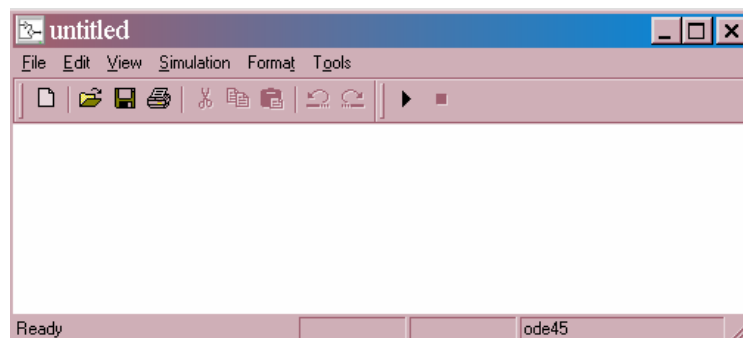
Discrete : modèles discrets

Linear : modèles analogiques linéaires

Nonlinear : non linéarité (seuil, relais, etc...)

Connections : blocs de connexion (multiplexeur , démultiplexeur ,etc...)

- Cliquer sur « File » et choisir « New », une nouvelle fenêtre est ouverte



La fenêtre ouverte est nommée « Untitled » ; vous pouvez la renommer quand vous la sauvegardez.

Pour construire le modèle à simuler on procède comme dans un laboratoire :

- Etape 1 : Placer les éléments de votre système (discrète, linear, N.linear, connections, ...)
- Etape 2 : Placer le(s) générateur(s) de signaux (sources).
- Etape 3 : Placer les appareils de mesure (Sinks)
- Etape 4 : Faire les connections (à l'aide de la souris)
- Etape 5 : Faire le réglage des différents éléments (gain, base de temps, fréquence, etc ...)
- Etape 6 : Choisir les caractéristiques des blocs et lancer la simulation.
- Etape 7 : Analysez les résultats (Scope, Enregistreur, Fichier,...)

2 – Sélectionner un objet :

Pour sélectionner un objet on doit cliquer une seule fois ; à ses coins apparaissent des carrés noir. On peut sélectionner tous les objet existant dans la fenêtre active en utilisant « **SELECT ALL** » du menu « **EDIT** ».

3 – Manipuler un bloc :

Les bloc possèdent deux propriétés : Générale et particulière

- **Générale** : commune pour tous les blocs (dimension et position).
- **Particulière** : chaque bloc a ses caractéristiques (la valeur du gain, la valeur initiale d'un intégrateur,)

Lorsqu'un bloc est copié, il hérite tous les propriétés de celui de l'original.

a – Comment changer ou copier un bloc ?

Les blocs peuvent être changés ou copiés dans une fenêtre par utilisation de la souris ou par l'usage de « **CUT** », « **COPY** » et « **PASTE** » du menu « **EDIT** »

Utilisation de la souris :

Pour copier un bloc de la bibliothèque,, il faut pointer, maintenir appuyé le bouton gauche de la souris tout en déplaçant vers l'endroit où on désire reproduire le schéma du bloc. Lâcher le bouton et le bloc sera inséré dans sa nouvelle position. Si la nouvelle position est dans la fenêtre active le bloc est déplacé et si elle dans une autre fenêtre, il est copié.

Pour dupliquer un bloc se trouvant dans la fenêtre de travail il faut procéder de la même manière mais en utilisant le bouton droit de la souris.

b – Comment supprimer un bloc ?

Pour supprimer un ou plusieurs bloc, il faut le(s) sélectionner puis actionner sur « **DELETE KEY** », ou choisir « **CLEAR** » ou « **CUT** » du menu **EDIT**

c – Comment redimensionner un bloc ?

Mettre le curseur dans l'un des carrés noirs qui se trouvent aux coins du bloc sélectionné quand on veut changer ses dimensions.

4 – Manipulation des lignes :**a – Comment connecter les blocs ?**

Pour relier la sortie (**OUTPUT**) d'un bloc à l'entrée (**INPUT**) de l'autre, sélectionner la sortie, maintenir appuyé sur le bouton gauche et traîner la pointe à l'entrée désirée . Lorsqu'on lâche le bouton, on obtient une ligne de connexion avec une extrémité qui désigne la direction du parcours du signal. La ligne de connexion est tracée dans deux sens possibles de l'entrée à la sortie ou inversement mais son orientation ne change pas.

b – Comment connecter deux lignes ?

Pointer la ligne, maintenir appuyé sur le bouton droit de la souris et l'on déplace vers l'endroit désiré. De cette façon on obtient une nouvelle dérivation et il nous apparaît un point noir qui indique la connexion entre deux lignes.

5 – Simulation des modèles :

Généralement on utilise un oscilloscope « **SCOPE** » ou un « **XYgraph** » ou « **To workspace** » pour voir les courbes de simulation de notre modèle. On démarre ou on arrête la simulation en choisissant, respectivement, « **Start** » ou « **Stop** » du menu « **Simulation** ».

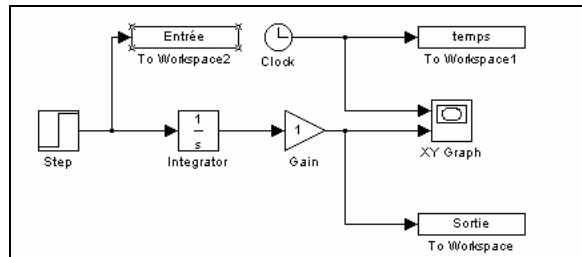
- Après le démarrage de la simulation, on clique deux fois sur l'oscilloscope pour voir la courbe.
- Si on utilise le « **To workspace** », on doit le nommer, une fois la simulation est terminée, on tape la commande **plot(name)** à partir de la fenêtre « **MATLAB command Windows** »
name : le nom du « **To workspace** »

MANIPULATION :

I- Première partie : Utilisation de SIMULINK :

1- Génération d'une rampe :

Réaliser le schéma de simulation suivant :



Donner la réponse à un échelon d'amplitude 1 pour les valeurs suivantes du gain : ($G = 1$; $G = 2$; $G = 5$).

2- Réponse temporelle des systèmes de premier et second ordre :

a – Tracer la réponse à un échelon ($E_0 = 1 \text{ V}$) d'un système de premier ordre pour différentes valeurs de T (5 s ; 7 s ; 10 s ; 15 s) ; $K = 1$. Déterminer le temps de stabilisation à 5% et à 10%.

b – Tracer la réponse à un échelon ($E_0 = 1 \text{ V}$) d'un système de premier ordre généralisé ($T = 4 \text{ s}$, $K = 1$) pour $\lambda = 0$; 0.5 ; 1 ; 1.5 ; 2).

c – Tracer la réponse à un échelon ($E_0 = 1 \text{ V}$) d'un système de second ordre ($f = 0.3 \text{ Hz}$, $K = 1$) pour $m = 0$; 0.1 ; 0.3 ; 0.5 ; 0.7). Déterminer le dépassement et le temps de pic, en déduire le temps de stabilisation à 5%.

d – Tracer la réponse à un échelon ($E_0 = 1.5 \text{ V}$) d'un système de second ordre ($f = 0.25 \text{ Hz}$, $K = 1$) pour $m = 1$; 1.3 ; 1.5). Déterminer le temps de stabilisation à 5%.

3- Réponse temporelle des systèmes linéaires :

Soit la fonction de transfert suivante :

$$H(p) = \frac{1}{p^2} - \frac{1}{p^2} \bullet e^{-2p} - \frac{1}{p^2} \bullet e^{-4p} + \frac{1}{p^2} \bullet e^{-6p}$$

Tracer la réponse à un échelon d'amplitude 1 V.

II- Deuxième partie : Programmation sous MATLAB :

1- Réponse en fréquence des système de 1^{er} et 2^{ième} ordre :

a– Tracer la réponse en fréquence (diagramme de Bode) d'un système de premier ordre ($K = 1$) pour $T = 5s$, en développant le programme suivant :

```
w=0.01:.1:100;
for i=1:length(w);t(i)=1/(1+5*j*w(i));end;
i=1:length(w);
subplot(2,1,1);semilogx(w,20*log(abs(t(i))));
title('amplitude en db');
subplot(2,1,2);semilogx(1/pi*180*angle(t(i)));
xlabel('système de 1er ordre T = 5 s');
title('phase');
```

b– Tracer la réponse en fréquence (diagramme de Bode) d'un système de premier ordre généralisé ($K = 1$; $T = 4 s$) pour $\lambda = 0.5 ; 1.5$

c– Tracer la réponse en fréquence (diagramme de bode) d'un système de second ordre ($f = 0.3 \text{ Hz}$; $K = 1$) pour $m = 0.1 ; 0.3$

Remarque :

Pour changer l'ordre de système il suffit de changer l'expression de la fonction de transfert « t »