

Pense bête : Matlab

2015

1 Introduction

Matlab est un environnement de calcul numérique matriciel. Il est largement utilisé pour l'étude et l'application de l'Automatique. Il est distribué par l'entreprise américaine MathWorks® sous des licences payantes.

2 L'environnement Matlab

Nous allons faire abstraction de l'installation de Matlab car cette phase est dépendante du système d'exploitation utilisée. On attaquera directement sur l'utilisation pratique du logiciel. Une fois lancé, Matlab se présente sous forme d'une fenêtre composée de sous fenêtre répertoire de travail, une historique de commande, un espace de travail, une fenêtre de commande. Cette dernière servira comme interface entre les entrées de l'utilisateur et le logiciel.

2.1 Les variables

Matlab manipule essentiellement des variables qui sont stockées en mémoire. Par défaut, chaque variable est une matrice de par même la nature du logiciel. La dimension de chaque matrice définira si la variable est un scalaire, un vecteur ou une matrice de dimension $n \times m$.

Exemple :

Un nombre réel $S = \pi * r^2$

```
>>S=pi*r^2
```

Un vecteur $x = (1 - 1)^T$

```
>>x=[1;-1]
```

Une matrice $A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$

```
>>A=[1 2 3;4 5 6]
```

Pour définir une variable sous forme vectorielle avec un pas fixe :

```
>>t=0 :0.1 :10;
```

La variable t (qui peut être le temps) prend des valeurs de 0 à 10 par pas de 0.1.

3 Utilisation de la *Control Toolbox*

3.1 Généralités

Matlab dispose de boîte à outils spécialisés pour différents thèmes. Control Toolbox est dédié à la commande et dispose de nombreux outils d'analyse pour l'automatique.

Un système est défini par sa fonction de transfert notée :

$$G(p) = \frac{N(p)}{D(p)} = \frac{p + 2}{p^2 + 4p + 4} \quad (1)$$

où p désigne la variable de Laplace. Dans la fenêtre de commande, on définit la fonction de transfert :

```
>> G=tf([1 2], [1 4 4])
```

>> $G = zp([-2], [-2 \ -2], 1)$ à partir des zéros, des pôles et du gain
avec le zéro, les pôles et le gain de la fonction de transfert.

3.2 Les opérations avec les fonctions de transfert

Les opérations entre les fonctions de transfert de différents sous-systèmes :

>>G1*G2	ou >>series(G1*G2)	G1 en série avec G2
>>G1+G2	ou >>parallel(G1*G2)	G1 en parallèle avec G2
>>feedback(G1,1)		G1 bouclé avec un retour unitaire négative

3.3 Les fonctions utiles pour analyser un système

Soit un système défini par sa fonction de transfert $G(p)$, il est possible d'obtenir ces propriétés par les commandes suivantes :

>>pole(G)	fournit les pôles
>>zero(G)	fournit les zéros
>>step(G)	trace la réponse indicielle
>>impulse(G)	trace la réponse impulsionnelle
>>lsim(G,u,t)	simule la sortie du système G soumis à une entrée quelconque u pendant la durée t
>>bode(G)	trace le diagramme de Bode
>>nyquist(G)	trace le diagramme de Nyquist
>>nichols(G)	trace le diagramme de Black-Nichols, suivi de la commande <i>grid on</i> pour obtenir les courbes d'iso-gain et d'iso-phase
>>rlocus(G)	trace le lieu d'Evans
>>rlocfind(G)	fournit les valeurs des pôles et du gain en se basant sur le lieu d'Evans
>>damp(G)	fournit les pôles, ainsi que la pulsation propre et l'amortissement correspondant
>>pzmap(G)	place les pôles et les zéros dans le plan complexe
>>margin(G)	fournit les marges de gain et de phase, ainsi que les fréquences de coupure

Autres commandes utiles : plot, grid on, xlabel, ylabel,

Il est possible de regrouper plusieurs commandes dans un fichier **.m* ou un fichier script. Les commandes seront par la suite exécuter ensemble.

Faire un commentaire dans un fichier **.m* :

% Le commentaire sera ici.