**TP 1 Présentation et analyse des systèmes linéaire continus dans l’espace d’état**

1. **Objectifs du TP :**

* Programmation et résolution des équations différentielles à l'aide du logiciel Matlab ;
* Représentation d’état;
* Commandebilité et observabilité d’un système,

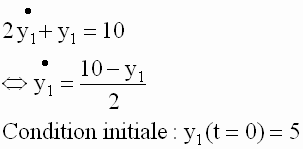
1. **Résolution d’équation différentielle ordinaire**
2. **1. Fonction ODE**

Une équation différentielle ordinaire ODE (ordinary differential equation) est une équation reliant une fonction d’une variable réelle et ses dérivées, c’est à dire de la forme



* **Exemple 1 : équation différentielle du premier ordre**

Soit la fonction y1(t) soumise à l’équation différentielle :



Pour programmer cette équation, nous créons le fichier f10.m :



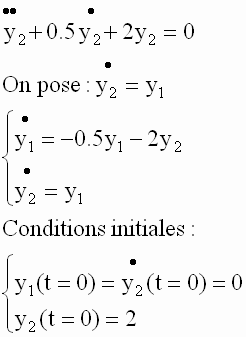
Sur l’espace de travaille du Matlab, taper le code suivant : **[ t , y ] = ode23 ( 'f10' , [0 20] , 0 ) ;**

**Plot(t,y)**

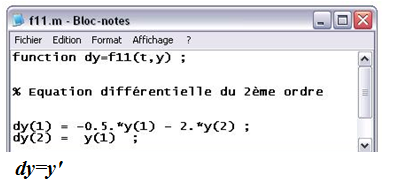
- Qu'est-ce qui fait la fonction ode23 dans MATLAB ?

- Faire changer l’ordre d’ode23 au ode45 puis tracez y(t) dans le même intervalle **[0 20].**

* **Exemple 2 : équation différentielle du deuxième ordre**



 Créer le fichier f11.m :



**[ t , y ]=ode23 ( 'f11' , [0 10] , [ 0 2 ] )**

**Plot (t,y)**

**2.2 Ffonction dsolve**

La syntaxe pour résoudre une équation différentielle simple est **dsolve(‘eqn’).** La fonction renvoie une solution formelle de l’équation différentielle définie par l’expression symbolique eqn.

D : représente la dérivée première, D2 la dérivée second..etc, les constantes arbitraires de la solution sont notées C1 C2..etc.

Ezplot(y,[tmin,tmax] : permet de ploter la solution dans l’intervalle de sa variable [tmin,tmax].

**Exemple 3**

**Fonction du 1er ordre**

soit l’équation : 2\*Dy+y=10

pour déterminer la solution formelle de cette équation on écrire sous Matlab :

**y=dsolve('2\*Dy+y=10', 'y(0)=5',t)**

**ezplot(y,[0,10])**

**Fonction du 2ème ordre** : D2y+0.5\*Dy+2\*y=0

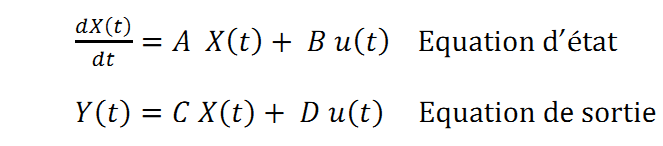
y=dsolve('D2y+0.5\*Dy+2\*y=0','y(0)=2, Dy(0)=0',t)

ezplot(y,[0,60])

**3 . Représentation d’état**

**3.1 Définition**

D’une manière générale, à tout système linéaire, causal et continu peuvent être associées les équations matricielles suivantes :



A : matrice d’état (ou dynamique) du système de dimension [n x n].

B : matrice de commande (d’entrée) de dimension [n x m].

C : matrice de mesure (de sortie) du système de dimension [p x n].

D : matrice de transmission directe de dimension [m x p].

X : vecteur d’état du système.

u : vecteur d’entrée du système.

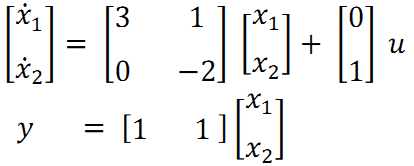
y : vecteur de sortie du système.

La fonction de transfert du système étant :

****

**Exemple**

Soit un système donné par sa représentation d’état suivante :

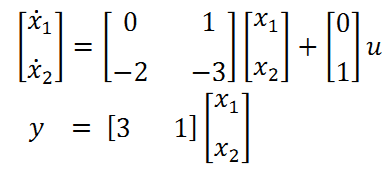


* Ecrie sous Matlab (Programmation puis sur Simulink) cette représentation d’état.
* Déterminer fonction de transfert du système, en utilisant l’équation de G(s) puis la fonctio, **ss2tf** en Matlab.
* En utilisant la fonction préétablie de Matlab (tf2ss), on obtient la représentation d’état. Matlab par défaut ne donne qu’une seule représentation

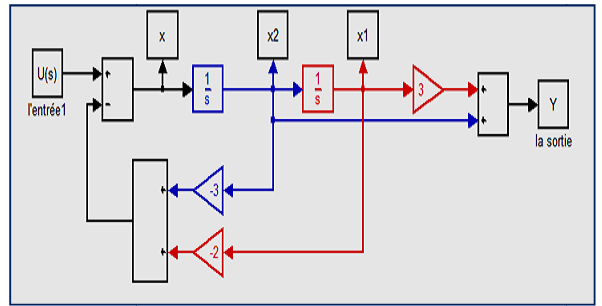
1. **Représentation d’état des systèmes dynamiques sous Matlab/Simulink**

**Exemple**

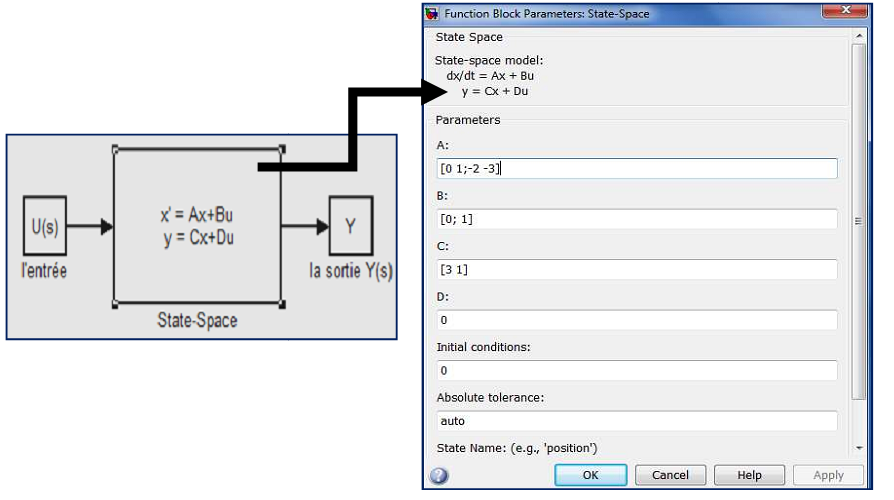
Soit le système suivant :



On peut schématiser cette représentation d’état sous Matlab/Simulink comme montré ci-desous



ou soit par le bloc state space comme suit :



1. **Analyse du système dans l’espace d’état**
   1. **Valeurs propres du système**

Matlab propose une fonction préétablit (eig) qui calcul les valeurs propres de A et qui servira à la résolution de l’équation:   
 

* 1. **Stabilité du système dans l’espace d’état**

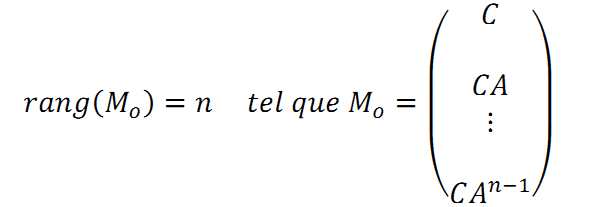
On peut démontrer que la stabilité d’un système est obtenue à partir

*det(A*-.

On dit que le système est *asymptotiquement* stabletoutes *les valeurs* propres *complexes de A ont une* partie réelle. Vérifier la stabilité d’exemple précédent.

* 1. **Obsavabilité et Commandabilité**

La paire (A, C) est observable si et seulement si :

******

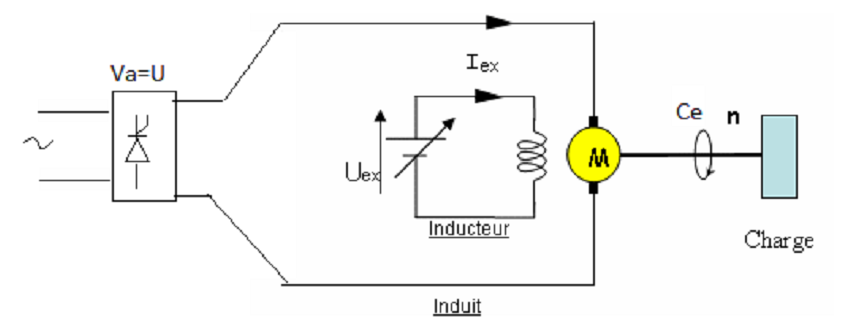
La paire (A, B) est commandable si et seulement si :



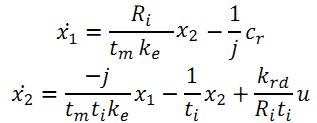
en Matlab on utilise ctrb(A,B)

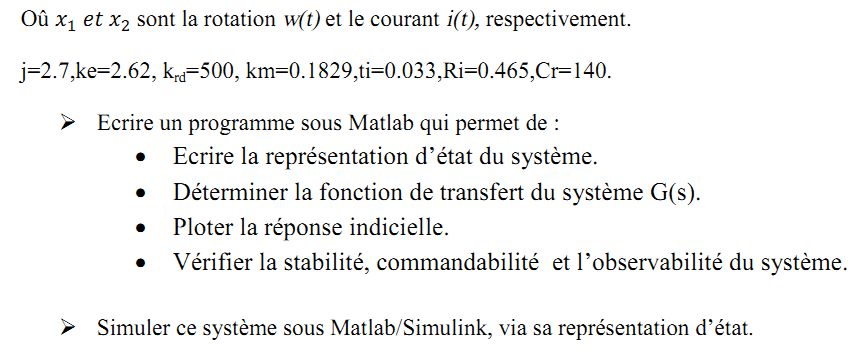
1. Travail demandé

Soit le moteur à CC illustré sur la figure ci-dessous :



Les équations d’état de ce système s’écrivent :





**Date :** …………………………..**Compte rendu du TP N°1**

**Nom et Prénom :** ……………………………………………………… **Groupe :**…………………

**Nom et Prénom :** ………………………………………………………………………………………

**Nom et Prénom :** ………………………………………………………………………………………

1. Représentions d’état du MCC à vide (cr=0) :

A= ,B= , C= ,D=

1. *F*onction de transfert du système en Boucle ouverte *G(s)*= …………………………………..
2. Réponse indicielle du système

Simulation sous Matlab Simulink

Réponse indicielle du système

1. Pôles de système p=[ ], Système est stable ? commandable ? observable ?
2. Simulation sous Matlab Simulink
3. Commentaires

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

1. Conclusion

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………