TP Matlab-Simulink sur le pendule inverse
M2 SAR - Commande non-linéaire

L'objectif de ce TD Matlab-Simulink est de mettre en application les éléments vus en cours pour la stabilisation **à partir du système linéarisé tangent**. Le système mécanique représenté sur la Figure 1 est appelé "pendule inversé". Il se compose d'un chariot qui peut se déplacer en translation suivant l'axe $(0, \vec{\imath})$, et d'une barre qui peut pivoter librement autour du point P. Les roues du chariot sont actionnées par le biais de moteurs qui délivrent une force $\vec{F} = f\vec{i}$, avec f assimilable à une variable de commande. Par contre, la liason pivot entre le chariot et la barre n'est pas actionnée.

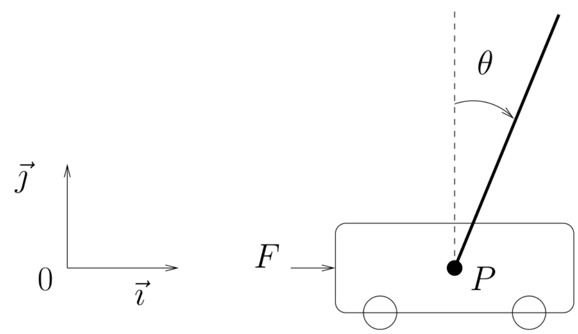

Figure 1

En notant M la masse de l'ensemble chariot/barre, m la masse de la barre, J son inertie, 2ℓ sa longueur, et x l'abscisse du point P (i.e. $\overrightarrow{OP} = x\vec{\imath}$), et en faisant le changement de variable de commande $f \longmapsto u = f + m\ell\dot{\theta}^2 \sin\theta$, on peut montrer que les équations dynamiques du système sont données par les équations suivantes:

$$\begin{pmatrix} \ddot{x} \\ \ddot{\theta} \end{pmatrix} = \frac{1}{\Delta(\theta)} \begin{pmatrix} J + m\ell^2 & -ml\cos\theta \\ -m\ell\cos\theta & M \end{pmatrix} \begin{pmatrix} T + m\ell\dot{\theta}^2 \sin\theta - \tau\dot{x} \\ mg\ell\sin\theta - \mu\dot{\theta} \end{pmatrix} \tag{1}$$

avec $\Delta(\theta) = M(J + m\ell^2) - m^2\ell^2\cos^2\theta > 0$ et $\tau, \mu > 0$.

Partie 0: **(question préparatoire à traiter avant le TP)**:

1. Mettre cette équation sous la forme d'un système $\dot{X} = f(X, T)$, où T désigne la variable de commande.
2. Déterminez les points d'équilibre de ce système.
3. Calculer le linéarisé tangent à l'état d'équilibre $X_e^+ = (x_e, \theta_e, \dot{x}_e, \dot{\theta}_e) = (0, 0, 0, 0)$.
4. Vérifier que ce linéarisé tangent est observable avec la mesure $y = (x_e, \theta_e)$.

Partie 1: Contrôle de l'état complet du système avec état connu.
On souhaite contrôler à la fois l'orientation de la barre et la position du chariot.

a) Déterminer numériquement la stabilité du linéarisé tangent en X_e^+ en boucle ouverte (i.e., avec une commande nulle). On pourra utiliser la fonction matlab "eig".

b) Déterminer un retour d'état linéaire qui rend X_e^+ cet équilibre asymptotiquement stable pour le système linéarisé tangent. On pourra utiliser la méthode LQR et tester différentes matrices de pondération du critère à minimiser.

c) Simuler les lois de commande synthétisées et discuter le comportement transitoire du système en fonction des matrices de pondération.

d) Introduire dans le modèle un échantillonage de la commande et des bruits sur la mesure (voir Figure 2 ci-dessous) et tester en simulation l'impact de la période d'échantillonage sur la stabilité du système et l'impact des bruits sur la précision de la stabilisation. Concernant les bruits, on prendra un bruit uniforme d'amplitude maximale ± 0.02 sur chaque variable d'état. Concernant l'échantillonage, on testera différentes valeurs entre 5 et 100 Hz, et on comparera les performances avec et sans bruit de mesure, en fonction de cette fréquence d'échantillonage.

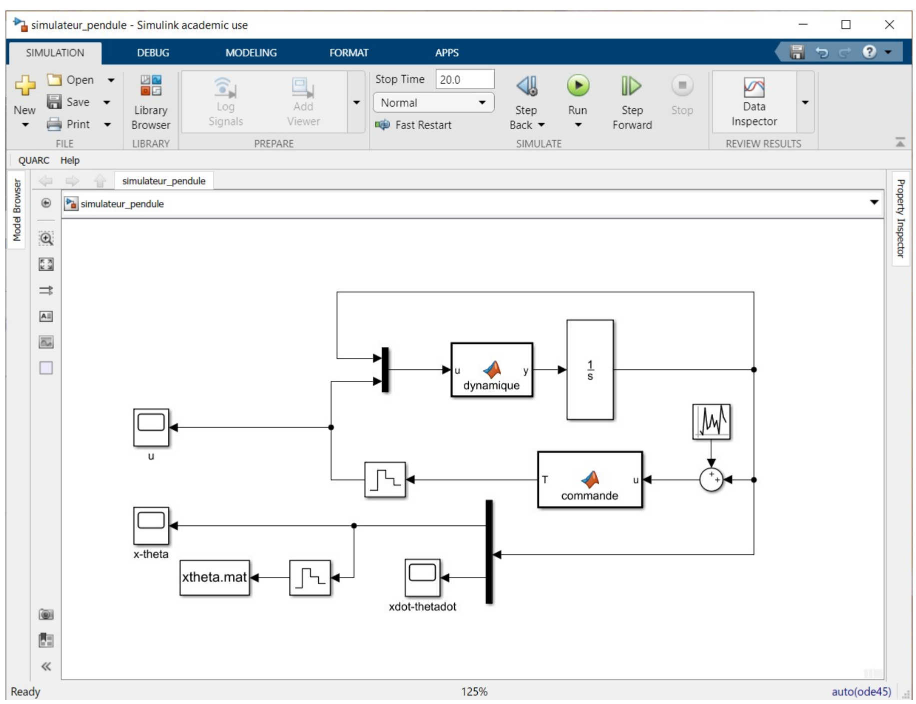

Figure 2: Echantillonage de la commande et ajout de bruit sur les mesures

<u>Question 2: Synthèse d'observateur par placement de pôles</u>

Les questions précédentes supposaient implicitement que tout l'état du système était mesuré. On considère dans cette question la situation plus réaliste dans laquelle on mesure seulement y, i.e., la position du chariot et l'orientation de la barre, mais pas les dérivées de ces variables.

1. Synthétiser un observateur d'état de Luenberger avec une technique de placement de pôles et implémenter cet observateur. On montrera que l'implémentation peut se faire comme indiqué

sur la Figure 3 ci-dessous, où l'on introduit également du bruit sur chacune des mesures (on utilisera les mêmes amplitudes de bruit que celles de la question 2). On pourra utiliser la commande Matlab "place" pour effectuer le placement de pôles.

2. Comparer les performances du système en fonction du choix des gains de l'observateur, et de la fréquence d'échantillonage. Concernant les valeurs des gains de l'observateur, on pourra tester différents ordres de grandeur entre -5 et -100.

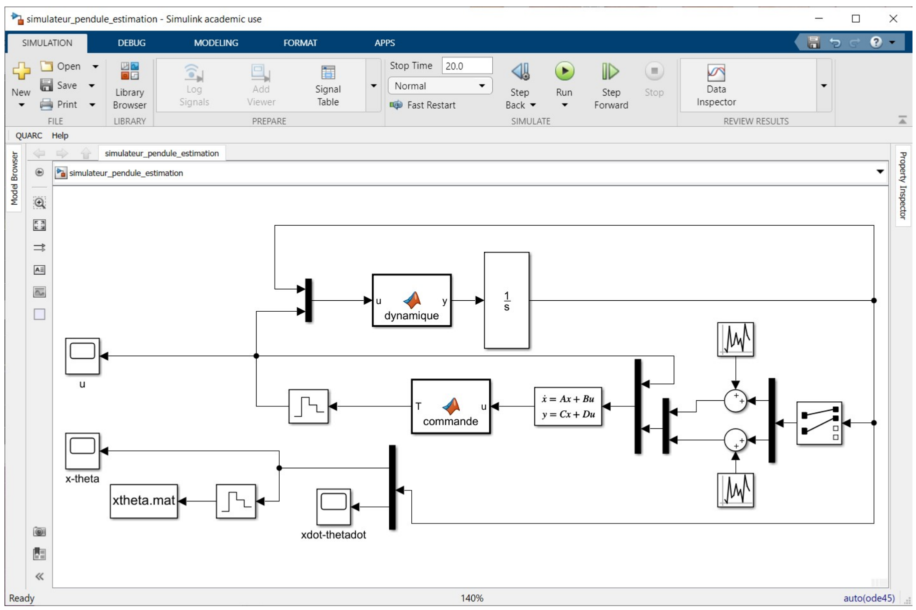

Figure 3: Simulateur avec observateur de Luenberger

Remarque Annexe: Chargement de paramètres dans Matlab

Afin de faciliter la mise en oeuvre sous Simulink, il est conseillé de faire les calculs dans un fichier .m de script, afin de générer les différentes matrices et gains dans le "Workspace" matlab, ce qui permettra ensuite de faire appel à ces valeurs sans devoir les retapper dans les blocs simulink, à l'image de la Figure 4 ci-dessous.

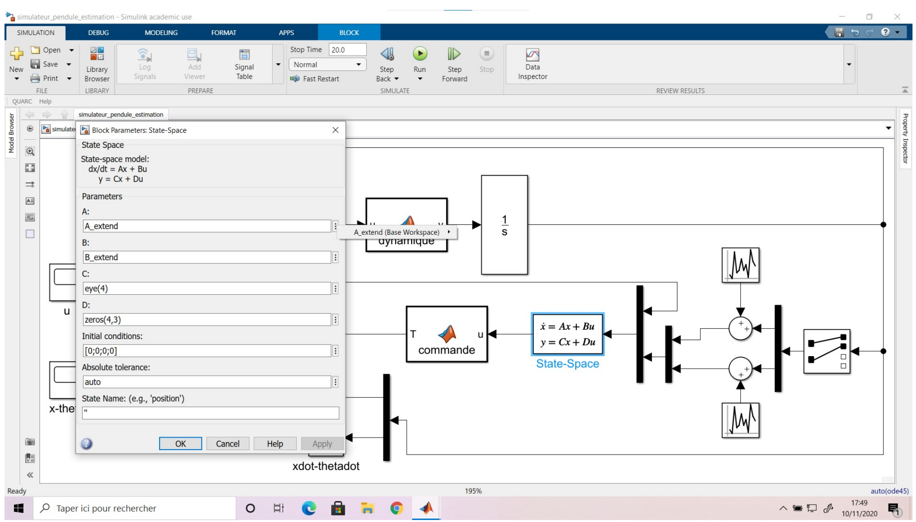

Figure 4: Chargement de variables dans un bloc simulink depuis le "Workspace" Matlab.