Préparation au TP1 du cours CAVAN

C. Afri, B. Marinescu

Important Cette préparation est à rendre **avant** le début du TP, faute de quoi la note finale sera impactée.

1 Objectif

Cette préparation a pour but de cibler l'ensemble des notions autour desquelles se déroulera le TP sur les commandes optimales linéaires LQ/LQG. Elle comporte un ensemble de questions auxquelles on doit répondre en se référant principalement au cours et à des ressources sur internet.

2 Questions

1. Pour un système donné par la forme d'état ci-dessous
$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases}, \tag{1}$$
comment s'écrit la commande $u(t)$ permettant de minimiser le critère
$$J = \int_0^\infty y^T(t)\, Q_y y(t) + u^T(t)\, R u(t)\,.$$
? Réécrire J sous la forme
$$J = \int_0^\infty x^T(t)\, Q_x x(t) + 2x^T(t)\, N u(t) + u^T(t)\, R_u u(t)$$
en donnant les expressions des matrices Q_x, N et R_u. Avec la commande $u(t)$ qui minimise J, à quoi correspondent donc les pôles du système (1) en boucle fermée.
2. C'est quoi un filtre de Kalman, et que doivent vérifier les bruits $w(t)$ et $v(t)$ qui affectent l'état et la mesure respectivement.
3. Rappeler l'équation dynamique du filtre de Kalman et dire à quoi correspondent ces pôles.
4. C'est quoi la commande LQG et en quoi elle est différente de la commande LQ.
5. Donnez l'expression des matrices $A_{lqg}, B_{lqg}, C_{lqg}, D_{lqg}$ correspondantes à la boucle fermée de la commande LQG, i.e., qui vérifient
$$\begin{cases} \begin{pmatrix} \dot{x} \\ \dot{\hat{x}} \end{pmatrix} = A_{lqg} \begin{pmatrix} x \\ \hat{x} \end{pmatrix} + B_{lqg} \begin{pmatrix} w(t) \\ v(t) \end{pmatrix} \\ y(t) = C_{lqg} \begin{pmatrix} x \\ \hat{x} \end{pmatrix} + D_{lqg} \begin{pmatrix} w(t) \\ v(t) \end{pmatrix} \end{cases}. \tag{2}$$
Quels sont les pôles de (2) ?
6. À quoi elle sert la fonction lsim de Matlab (le help en annexe A)

A Help de la fonction Lsim

lsim
Simulate time response of dynamic system to arbitrary inputs
Syntax
lsim
lsim(sys,u,t)
lsim(sys,u,t,x0)
lsim(sys,u,t,x0,'zoh')
lsim(sys,u,t,x0,'foh')
lsim(sys)
Description

lsim simulates the (time) response of continuous or discrete linear systems to arbitrary inputs. When invoked without left-hand arguments, lsim plots the response on the screen.

lsim(sys,u,t) produces a plot of the time response of the dynamic system model sys to the input time history t,u. The vector t specifies the time samples for the simulation (in system time units, specified in the TimeUnit property of sys), and consists of regularly spaced time samples.

t = 0 :dt :Tfinal The matrix u must have as many rows as time samples (length(t)) and as many columns as system inputs. Each row u(i, :) specifies the input value(s) at the time sample t(i).

The LTI model sys can be continuous or discrete, SISO or MIMO. In discrete time, u must be sampled at the same rate as the system (t is then redundant and can be omitted or set to the empty matrix). In continuous time, the time sampling dt=t(2)-t(1) is used to discretize the continuous model. If dt is too large (undersampling), lsim issues a warning suggesting that you use a more appropriate sample time, but will use the specified sample time. See Algorithms for a discussion of sample times.

lsim(sys,u,t,x0) further specifies an initial condition x0 for the system states. This syntax applies only to state-space models.

lsim(sys,u,t,x0,'zoh') or lsim(sys,u,t,x0,'foh') explicitly specifies how the input values should be interpolated between samples (zero-order hold or linear interpolation). By default, lsim selects the interpolation method automatically based on the smoothness of the signal U.

Finally,

lsim(sys1,sys2,...,sysN,u,t) simulates the responses of several LTI models to the same input history t,u and plots these responses on a single figure. As with bode or plot, you can specify a particular color, linestyle, and/or marker for each system, for example,

lsim(sys1,'y :',sys2,'g–',u,t,x0) The multisystem behavior is similar to that of bode or step.

When invoked with left-hand arguments,

[y,t] = lsim(sys,u,t) [y,t,x] = lsim(sys,u,t) [y,t,x] = lsim(sys,u,t,x0) return the output response y, the time vector t used for simulation, and the state trajectories x (for state-space models only). No plot is drawn on the screen. The matrix y has as many rows as time samples (length(t)) and as many columns as system outputs. The same holds for x with "outputs" replaced by states.

lsim(sys) opens the Linear Simulation Tool GUI. For more information about working with this GUI, see Working with the Linear Simulation Tool.

Indication Pour plus de précisions et exemples voir le help de Matlab en ligne en tapant le mot clé 'lsim matlab' dans google.