EE 661 Project #4: **LQR Observer-Based Controller Design**

Due Date: Monday, December 15, 2014

This project involves the LQR controller-design process, starting with a reduced-order model of the system to be controlled, and ending up with a stable standalone observer-based controller. You will also analyze the stability margins of the resulting controller from various perspectives.

You will need three zip-files from MyCourses:

"**`cbeam_3x2_frfdata_Ts20kHz_hlfadv_20kPts__struc.zip`**" (also used in Proj. #2)

"**`cbeam_3x2_sysmodel_Ts20kHz_194pole__struc.zip`**" (also used in Proj. #2)

"**`cbeam_3x2_reduced_model_40pole_Ts20kHz__struc.zip`**"

Here are the things you need to do for this assignment:

(1) Based on the 40-pole model of the cantilevered-beam, use LQR to design a state-feedback matrix K, so that significant damping is achieved on all modes below 300 Hz, but not much pole movement above 300 Hz, and not much movement of real-valued poles. This requires computing (modal-canonic) state-weights based on the pole frequencies, with a "lowpass" frequency-weighting scheme. All the real-valued poles should be assigned very small weights, such as 0.001.

(2) Again based on the 40-pole model, use LQR to design an observer-error feedback matrix L, so that all the poles move at least slightly, but pole-pairs below 300 Hz move significantly to heavier-damped locations. Use the same Q and R as in step 1 (except here size of R is 3x3 instead of 2x2).

(3) Ascertain that the stand-alone controller poles **`eig(`** $A\text{-}BK\text{-}LC\text{+}LDK$ **`)`** are all stable (*i.e.*, inside the unit-circle). If not, change the R matrices you use in the LQR computations. Typically, increasing the values in the input-weighting matrix R makes the controller more conservative, because input-effort becomes more costly, and the poles won't move as far. Repeat steps 1 and 2 to **make the diagonal elements of the two "R" matrices as small as possible** while still achieving a stable standalone observer-based controller. This results in the most "aggressive" stable LQR controller.

(4) After completing steps 1 through 3, plot the characteristic loci of the measured FRF compensated by the FRF of your 40-pole observer-based controller. Estimate the stability margins of your compensated closed-loop system based on the Nichols chart of the characteristic loci.

(5) Use the higher-order 194-pole system model, combined with your 40-pole controller, to estimate the closed-loop pole locations. Note that the closed-loop-system model will have 194 poles from the system model, plus 40 poles from the observer, for a total of 234 poles. Verify the stability of this larger closed-loop model, by making sure that all 234 poles are strictly inside the unit circle. All these poles will move to different locations from where they started in open-loop, and all of them affect the closed-loop dynamics. In the same axes, plot the MIMO root locus, the compensated open-loop poles, compensated closed-loop poles, and intended (design) closed-loop pole locations.

(6) In this step, you investigate using Matlab's "place" function to achieve the same closed-loop pole locations. With the eigenvalues of $A\text{-}BK$ (the gain $\boldsymbol{K}$ designed above by LQR), use Matlab's "**`place`**" function to get a different feedback gain matrix, $\boldsymbol{K_m}$, that will put the closed loop poles in the same locations. Likewise, from the eigenvalues of $A\text{-}LC$ (the gain $\boldsymbol{L}$ designed above by LQR), use "**`place`**" to compute a gain matrix $\boldsymbol{L_m}$ that will put the closed-loop poles in the same locations. Compare the norm of $\boldsymbol{K_m}$ to the norm of $\boldsymbol{K}$, and the norm

of $\boldsymbol{L_m}$ to the norm of $\boldsymbol{L}$, and determine if these gain matrices $\boldsymbol{K_m}$ and $\boldsymbol{L_m}$ will give a stable standalone controller, **`eig(`** $A - BK_m - L_mC + L_mDK_m$ **).**

Submit an annotated Matlab “script” or “function” m-file that carries out all of the design steps. The m-file should be well-commented to explain what the various steps in the m-file are accomplishing.

The m-file should also create

(a) Nichols chart of the **characteristic loci** based on the compensated <u>measured</u>-FRF

(b) the **uncompensated open-loop** <u>Bode FRF magnitude</u> plots,

(c) the **closed-loop** <u>Bode FRF magnitude</u> plots,

(d) the **z-plane MIMO root-locus**, with open-loop poles, transmission zeros, and closed-loop poles (the 80 *design* locations and the 234 *actual* locations) indicated. Plot open-loop poles by **x**’s, transmission zeros by **o**’s, C.L. <u>design</u> poles by **squares**, <u>actual</u> C.L. pole locations by **triangles**. Make these markers big enough to be seen easily.

Plots must have informative titles and axis labels.

Submit a (hardcopy) report that

(*i*) compares K and L with K_m and L_m (step 6),

(*ii*) analyzes the differences in the open-loop and closed-loop FRF plots,

(*iii*) analyzes the MIMO root-locus, especially at frequencies below 300 Hz,

(*iv*) compares the actual closed-loop poles to where they were designed to be, including the results of the 234-pole closed-loop stability check.

(*v*) analyzes the stability margins using the characteristic loci.