CHEM-E7190/2021: Exercise IV - Luenberger + LQR for stirred tank

The dynamic equations of the stirred tank system in state-space form (with some steady-state values, you can use your own values) are the following:

$$\dot{x}(t) = \begin{bmatrix} -0.5 & 0 \\ 0 & -1 \end{bmatrix} x(t) + \begin{bmatrix} 1 & 0 & 0 \\ -0.1 & 1 & 1 \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} u(t)$$

1. The first step in designing a full-state feedback controller is to determine the open-loop poles of the system. Implement the system in m-file and calculate the eigenvalues of the system.

2. Before we design our controller, we will first verify that the system is controllable. Satisfaction of this property means that we can drive the state of the system anywhere we like in finite time (under the physical constraints of the system). For the system to be completely state controllable, the controllability matrix must have rank n where the rank of a matrix is the number of linearly independent rows (or columns). The number n corresponds to the number of state variables of the system. Calculate rank of the system.

3. We will use the linear quadratic regulation method for determining our state-feedback control gain matrix K. The MATLAB function lqr allows you to choose two parameters, R and Q, which will balance the relative importance of the control effort (u) and error (deviation from 0), respectively, in the cost function that you are trying to optimize. The simplest case is to assume $R = 1$, and $Q = C'C$. Design LQR control and simulate your system.

4. Modify your m-file so that the (1,1) element of Q is 100 and the (2,2) element is 5000, put more emphasize for temperature.

5. To address the situation where not all state variables are measured, a state estimator must be designed. Before we design our estimator, we will first verify that our system is observable. The property of observability determines whether or not based on the measured outputs of the system we can estimate the state of the system. Similar to the process for verifying controllability, a system is observable if its observability matrix is full rank. Calculate the observability of the system.

6. The dynamics of the state estimate are described by the following equation.

$$\dot{\hat{\mathbf{x}}} = A\hat{\mathbf{x}} + Bu + L(\mathbf{y} - \hat{\mathbf{y}})$$

The spirit of this equation is similar to that of closed-loop control in that last term is a correction based on feedback. Specifically, the last term corrects the state estimate based on the difference between the actual output $\mathbf{y}$ and the estimated output $\hat{\mathbf{y}}$. Now let's look at the dynamics of the error in the state estimate.

$$\dot{\mathbf{e}} = \dot{\mathbf{x}} - \dot{\hat{\mathbf{x}}} = (A\mathbf{x} + Bu) - (A\hat{\mathbf{x}} + Bu + L(C\mathbf{x} - C\hat{\mathbf{x}}))$$

Therefore, the state estimate error dynamics are described by

$$\dot{\mathbf{e}} = (A - LC)\mathbf{e}$$

and the error will approach zero ($\hat{\mathbf{x}}$ will approach $\mathbf{x}$) if the matrix $A - LC$ is stable (has negative eigenvalues). As is with the case for control, the speed of convergence depends on the poles of the estimator (eigenvalues of $A - LC$). Since we plan to use the state estimate as the input to our controller, we would like the state estimate to converge faster than is desired from our overall closed-loop system. That is, we would like the observer poles to be faster than the controller poles. A common guideline is to make the estimator poles 4-10 times faster than the slowest controller pole. Calculate the controllers poles and calculate the L matrix with Matlab `place` command.

7. We will combine our state-feedback controller from before with our state estimator to get the full compensator. The resulting closed-loop system is described by the following matrix equations.

(8)

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{e}} \end{bmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{e} \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} r$$

(9)

$$\mathbf{y} = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{e} \end{bmatrix} + \begin{bmatrix} 0 \end{bmatrix} r$$

Implement the above system and simulate it in Matlab.