HOMEWORK # 7

Andres Rainiero Hernandez Coronado

November 22, 2017

Due date of the homework is: Sunday, November 26th @ 11:59pm.

---

1. The following LTI system is given:

$$\dot{x}(t) = \begin{bmatrix} 0 & \alpha \\ -\alpha & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t), \quad y(t) = \begin{bmatrix} 0 & 1 \end{bmatrix} x(t).$$

(a) Assume that $\alpha = 2$. Find a state feedback controller $u(t) = -Kx(t)$ such that the closed loop evalues are both at -1.

$$K = \begin{bmatrix} k_1 & k_2 \end{bmatrix} \quad \Rightarrow \quad BK = \begin{bmatrix} 0 & 0 \\ k_1 & k_2 \end{bmatrix} \quad \Rightarrow \quad A - BK = \begin{bmatrix} 0 & 2 \\ -2 - k_1 & -k_2 \end{bmatrix}$$

$$\pi(A_{CL}) = (\lambda + 1)(\lambda + 1) = \lambda^2 + 2\lambda + 1 = 0$$

$$\pi(A_{CL}) = (-\lambda)(-k_2 - \lambda) - (-2 - k_1)(2) = \lambda^2 + (k_2)\lambda + (4 + 2k_1) = 0$$

$$k_2 = 2 \qquad k_1 = (1 - 4)/2 = -3/2$$

$$u(t) = \begin{bmatrix} -3/2 & 2 \end{bmatrix} x(t)$$

(b) Assume now that α is not known. Find the range of values of α that would make the state-feedback controlled system stable. That is, set $u(t) = -Kx(t)$ and find the range of values of α that would produce a stable closed loop system performance.

The characteristic polynomial needs to follow the next rules:

$$\pi(A_{CL}) = (\lambda + x_1)(\lambda + x_2) = 0$$

$$x_1, x_2 > 0 \quad x_1 + x_2 > 0 \quad x_1 x_2 > 0$$

$$A - BK = \begin{bmatrix} 0 & \alpha \\ -\alpha + 3/2 & -2 \end{bmatrix}$$

$$\pi(A_{CL}) = (-\lambda)(-2 - \lambda) - (-\alpha + 3/2)(\alpha) = \lambda^2 + 2\lambda + (\alpha^2 - 3/2\alpha) = 0$$

From the characteristic polynomial it is known that $x_1 + x_2 = 2$ and that $x_1 x_2 \leq 1$, which gives the upper bound of $\alpha \leq 2$. The lower bound is given by:

$$x_1 x_2 = \alpha^2 - 3/2\alpha \quad \Rightarrow \quad \alpha^2 > 3/2\alpha$$

Since α^2 needs to be larger than $3/2\alpha$, then the lower bound is $\alpha > 3/2$ (Note that this range of α will make the closed-loop system asymptotically stable).

$$3/2 < \alpha \leq 2$$

(c) For the question in part (b), what is the percentage of change in α from $\alpha = 2$ that we considered in part (a), that the system can tolerate before becoming unstable?

Given that $\alpha = 2 = 100\%$, α can be changed in less than 25%.

(d) Assume again that $\alpha = 2$, and that $x(t)$ is not available in real-time. This requires building a state estimator or an observer. Design an observer gain L such that the estimation error evalues are both at -1.

$$\pi(A_D) = (\lambda + 1)(\lambda + 1) = \lambda^2 + 2\lambda + 1 = 0$$

$$\pi(A_D) = (-\lambda)(-l_2 - \lambda) - (-2)(2 - l_1) = \lambda^2 + (l_2)\lambda + (4 - 2l_1) = 0$$

$$l_1 = 3/2 \quad l_2 = 2$$

$$L = \begin{bmatrix} 3/2 \\ 2 \end{bmatrix}$$

(e) After solving (d), combine the controller from (a) with the observer in (d) to arrive at an observer-based state-feedback controller.

By taking the form:

$$\begin{bmatrix} \dot{x}(t) \\ \dot{\hat{x}}(t) \end{bmatrix} = \begin{bmatrix} A & -BK \\ LC & A - LC - BK \end{bmatrix} \begin{bmatrix} x(t) \\ \hat{x}(t) \end{bmatrix}$$

$$\begin{bmatrix} \dot{x}(t) \\ \dot{\hat{x}}(t) \end{bmatrix} = \begin{bmatrix} 0 & 2 & 0 & 0 \\ -2 & 0 & 3/2 & -2 \\ 0 & 3/2 & 0 & 1/2 \\ 0 & 2 & -1/2 & -4 \end{bmatrix} \begin{bmatrix} x(t) \\ \hat{x}(t) \end{bmatrix}$$

(f) Implement the whole setup on MATLAB via the `ode45` solver. Show that the observer is succeeding in estimating the system states. Consider that the estimator initial conditions are zero (i.e., $\hat{x}(0) = 0$) and that the system's initial conditions are $x(0) = \begin{bmatrix} -2 & 2 \end{bmatrix}^\top$.

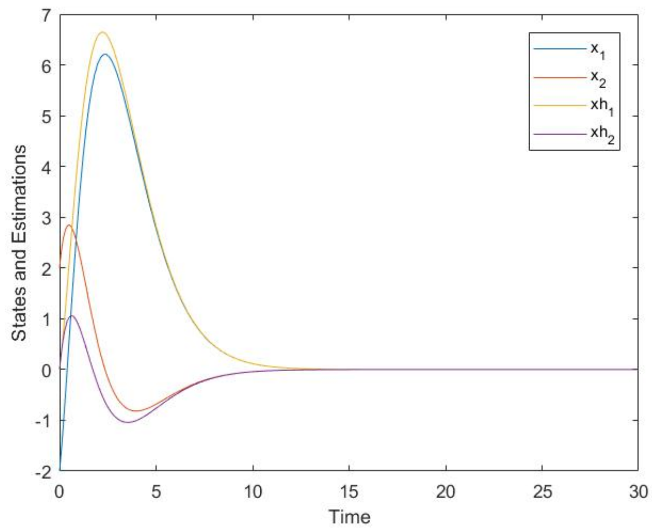
The following script will be used to describe the system from (e):

```
function dx = ocb(t, x)
A = [0 2 0 0; -2 0 3/2 -2; 0 3/2 0 1/2; 0 2 -1/2 -4];
dx = A*x;
end
```

Using the following commands, a plot can be obtained:

```
>> x0 = [-2; 2; 0; 0];
>> tspan = [0 30];
>> [t, y] = ode45(@ocb, tspan, x0);
>> plot(t,y)
```

The plot of the solution looks as follows:



From the image it is evident that the estimation for the first state is very accurate, and that the first state is correctly estimated at around 4 seconds. The second state takes longer to be estimated correctly, around 7 seconds.

(g) Assume now that we connect the observer designed in part (d) with the system with unknown α in part (b). What are the values of α that would drive the observer-based controller system to stay stable?

Hint: For any fourth order polynomial

$$s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0 = 0$$

to have roots with negative real parts, the necessary and sufficient conditions are that:

$$\{a_3, a_2, a_1, a_0\} > 0, \quad a_1 a_2 a_3 - b_1 a_3^2 - a_1^2 > 0.$$

It is not necessary to solve a fourth order polynomial, the next representation has the same evalues as the representation provided in (e):

$$\hat{A} = \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix}$$

Note that this form is block diagonal, meaning that the evalues of the blocks $\hat{A}_{11}, \hat{A}_{22}$ are the evalues of $\hat{A}$. The calculations of the evalues of each block is as follows:

$$det(A - BK) = det(\hat{A}_{11}) = (-\lambda)(-2 - \lambda) - (-\alpha + 3/2)(\alpha) = 0$$
$$det(A - LC) = det(\hat{A}_{22}) = (-\lambda)(-2 - \lambda) - (-\alpha)(\alpha - 3/2) = 0$$

The characteristic polynomial of both blocks is identical, meaning that the range of values of α to make the evalues of the blocks stable (and the OBC system) are given by the solution obtained in (b).

$$3/2 < \alpha \leq 2$$

2. The following autonomous LTI system is given:

$$\dot{x}(t) = Ax(t) = \begin{bmatrix} -1 & 1 \\ 0 & 1 \end{bmatrix} x(t), \quad y(t) = Cx(t) = \begin{bmatrix} 0 & 1 \end{bmatrix} x(t).$$

(a) Is the above system observable? Prove that via the four tests of observability.
Test 1:

$$\mathcal{O} = \begin{bmatrix} C \\ CA \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$$

$$rank(\mathcal{O}) = 1 < 2$$

The controllability matrix is not full rank.
Test 2:

$$eig(A) = -1, 1$$

$$rank(\begin{bmatrix} (1)I - A \\ C \end{bmatrix}) = \begin{bmatrix} 2 & -1 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} = 2$$

$$rank(\begin{bmatrix} (-1)I - A \\ C \end{bmatrix}) = \begin{bmatrix} 0 & -1 \\ 0 & -2 \\ 0 & 1 \end{bmatrix} = 1$$

The rank of the computed matrix for -1 is less than 2.
Test 3:

$$eig(A) = -1, 1$$

$$(A - (-1)I)v_1 = \begin{bmatrix} 0 & 1 \\ 0 & 2 \end{bmatrix} v_1 = 0 \quad \Rightarrow \quad v_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$Cv_1 = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 0$$

$$(A - (1)I)v_2 = \begin{bmatrix} -2 & 1 \\ 0 & 0 \end{bmatrix} v_2 = 0 \quad \Rightarrow \quad v_2 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

$$Cv_2 = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} = 2$$

The test fails for the evector v_1 that corresponds to the eigenvalue -1.
Test 4:

$$e^{A\tau} = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} e^{-\tau} & 0 \\ 0 & e^{\tau} \end{bmatrix} \begin{bmatrix} 1 & -1/2 \\ 0 & 1/2 \end{bmatrix} = \begin{bmatrix} e^{-\tau} & \dfrac{-e^{-\tau} + e^{\tau}}{2} \\ 0 & e^{\tau} \end{bmatrix}$$

$$(e^{A\tau})' = \begin{bmatrix} e^{-\tau} & 0 \\ \dfrac{-e^{-\tau} + e^{\tau}}{2} & e^{\tau} \end{bmatrix}$$

$$\int_0^t (e^{A\tau})' C' C e^{A\tau} d\tau = \int_0^t \begin{bmatrix} 0 & 0 \\ 0 & e^{2\tau} \end{bmatrix}$$

The integration is not necessary, it can be concluded that the resultant matrix will be singular.

(b) What is the unobservable subspace (if any)?

The unobservable subspace is the null-space of the Observability matrix.

$$N(\mathcal{O}) = span\{\begin{bmatrix}1\\0\end{bmatrix}\}$$

(c) Is the above system detectable?

Yes, the evalue 1 has an stable evector which was proven in Test 3 from (a), since $Cv_2 \neq 0$.

(d) Design a Luenberger observer such that the closed loop estimation error dynamics have both evalues placed at -1.

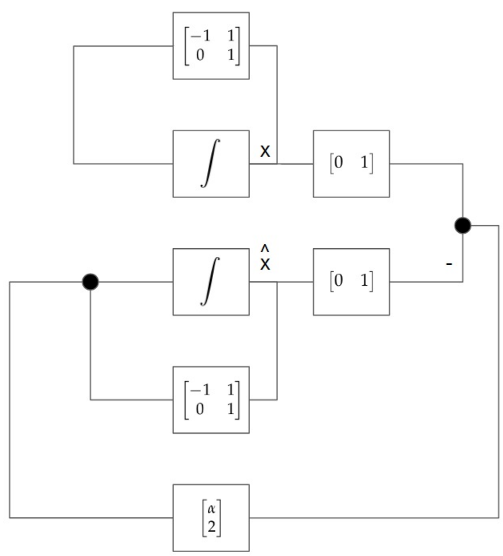
$$M = A - LC = \begin{bmatrix}-1 & 1\\0 & 1\end{bmatrix}\begin{bmatrix}L_1\\L_2\end{bmatrix}\begin{bmatrix}0 & 1\end{bmatrix} = \begin{bmatrix}-1 & 1-L_1\\0 & 1-L_2\end{bmatrix}$$

$$\pi(A_{CL}) = (\lambda+1)(\lambda+1) = \lambda^2 + 2\lambda + 1 = 0$$

$$\pi(M) = (-1-\lambda)(1-L_2-\lambda) = \lambda^2 + L_2\lambda + (L_2 - 1) = 0$$

From simple inspection $L_2 = 2$, the value of L_1 is redundant, since it wont affect the values of the evalues of the closed loop matrix.

(e) Show an overall diagram of the dynamic system and the observer.



(f) Find $\lim_{t\to\infty} e(t)$?

Assume that $L = \begin{bmatrix}0 & 2\end{bmatrix}^T$.

$$\dot{e}(t) = (A - LC)e(t) \quad \Rightarrow \quad e(t) = e^{(A-LC)t}$$

$$e(t) = exp(\begin{bmatrix}-1 & 1\\0 & 1\end{bmatrix} - \begin{bmatrix}0\\2\end{bmatrix}\begin{bmatrix}0 & 1\end{bmatrix}) = exp(\begin{bmatrix}-1 & 1\\0 & -1\end{bmatrix})$$

$$e(t) = \begin{bmatrix}e^{-t} & te^{-t}\\0 & e^{-t}\end{bmatrix}$$

Note that $e(t)$ will always have the form of the exponential of a Jordan block, since the values on the diagonal are -1, the limit of $e(t) \Rightarrow$ is a zero matrix.

$$\lim_{t\to\infty} e(t) = \begin{bmatrix}0 & 0\\0 & 0\end{bmatrix}$$

(g) Assume now we slightly change the dynamics of the system. Specifically, we change the A-matrix to:

$$\bar{A} = \begin{bmatrix} -0.9 & 1 \\ 0 & 1 \end{bmatrix}.$$

Given this slight change in matrix A, will the observer designed in the previous part yield converging and asymptotically stable estimation error dynamics or no?

Hint: Derive the closed loop dynamics of $\begin{bmatrix} \dot{x}(t) \\ \dot{e}(t) \end{bmatrix}$ and investigate the stability of the closed loop estimation error dynamics.

Yes, the observer will work and will yield asymptotically stable estimation error dynamics, simply because $A - LC$ corrects the positive evalue 1 and won't modify the evalue -0.9. For the new closed-loop system, the evalues will be $-0.9, -1$ which are in the LHP, making the solution of the estimation error asymptotically stable.

3. You are given a CT-LTI system defined by

$$A = \begin{bmatrix} 3 & -2 \\ 4 & -3 \end{bmatrix}, B = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$

(a) Design a dynamic observer such that the error dynamics have eigenvalues at -4 and -2. First, you should derive the dynamics of the observer and explain how it works. Then, obtain a gain L that achieves the desired objective.

$$y(t) = Cx(t) + Du(t)$$
$$\hat{y}(t) = C\hat{x}(t) + Du(t)$$

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

The dynamics of the observer are designed in a way that the output of the observer needs to correct itself depending on the difference of the estimated output and the actual output, described by $L(y - \hat{y})$.

$$\dot{e}(t) = \dot{x}(t) - \dot{\hat{x}}(t)$$
$$\dot{e}(t) = Ax(t) + Bu(t) - A\hat{x}(t) - Bu(t) - LCx(t) + LC\hat{x}(t) + Du(t) - Du(t)$$

Note that $Bu(t)$ and $Du(t)$ get cancelled, meaning that the design of the observer is independent of the input.

$$\dot{e}(t) = Ax(t) + A\hat{x}(t) - LCx(t) + LC\hat{x}(t)$$
$$\dot{e}(t) = (A - LC)(x(t) - \hat{x}(t))$$
$$\dot{e}(t) = (A - LC)e(t)$$

The solution of $e(t) = e^{A-LC}$, meaning that $A - LC$ needs to have stable evalues to provide an asymptotically stable solution for $e(t)$, that is, make the estimation error go to zero.

The design of L is as follows:

$$p(A_{CL}) = (\lambda + 4)(\lambda + 2) = 0$$

$$A_{CL} = \begin{bmatrix} 3 - L_1 & -2 \\ 4 - L_2 & -3 \end{bmatrix}$$

$$p(A_{CL}) = (3 - L_1 - \lambda)(-3 - \lambda) - (4 - \lambda)(-2) = 0$$

$$p(A_{CL}) = \lambda^2 + L_1\lambda + (-9 + 8 + 3L_1 - 2L_2) = 0$$

$$L_1 = 6 \qquad L_2 = 4.5 \qquad L = \begin{bmatrix} 6 \\ 4.5 \end{bmatrix}$$

(b) Finally, plot the norm of the estimation error dynamics, i.e., $||e(t)||_2$ as a function of time given that the observer's initial conditions are $\hat{x}(0) = \begin{bmatrix} 10 & -10 \end{bmatrix}^\top$ and $x(0) = \begin{bmatrix} 0 & 0 \end{bmatrix}^\top$. Consider that the input is $u(t) = 2\sin(10t)$. Your plot should be a single plot versus time, since the norm function returns a scalar. Implement the whole setup on MATLAB via the `ode45` solver.

The system will be implemented as follows:

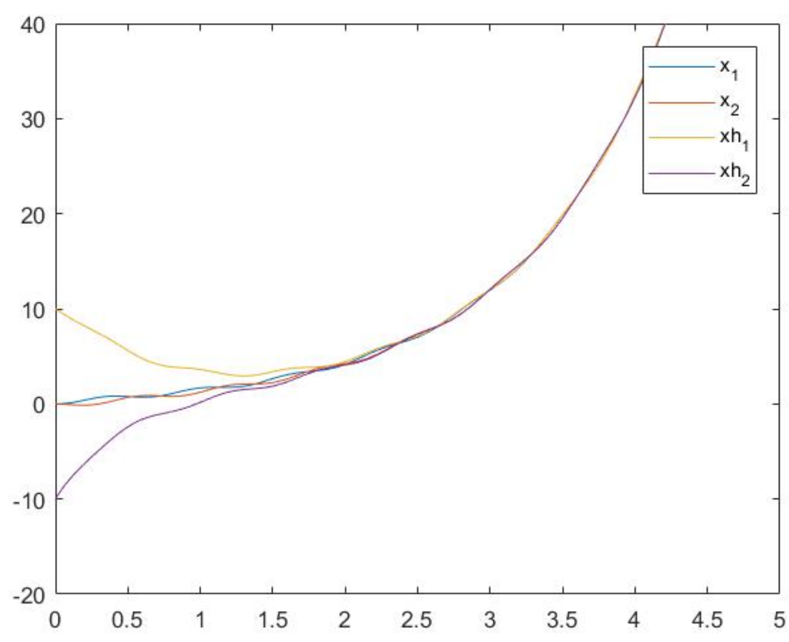
A function that represents the dynamics of the real states and the observer states is created:

$$\begin{bmatrix} x(t) \\ \hat{x}(t) \end{bmatrix} = \begin{bmatrix} A & 0 \\ LC & A - LC \end{bmatrix} \begin{bmatrix} x(t) \\ \hat{x}(t) \end{bmatrix} + \begin{bmatrix} B \\ B \end{bmatrix} u(t)$$

```
function dx = problem3(t, x)
L = [6; 4.5];
A = [3 -2; 4 -3];
B = [1; -1];
C = [1 0];
u = 2*sin(10*t);

dx = [A zeros(2); L*C (A-L*C)]*x + [B;B]*u;
end
```

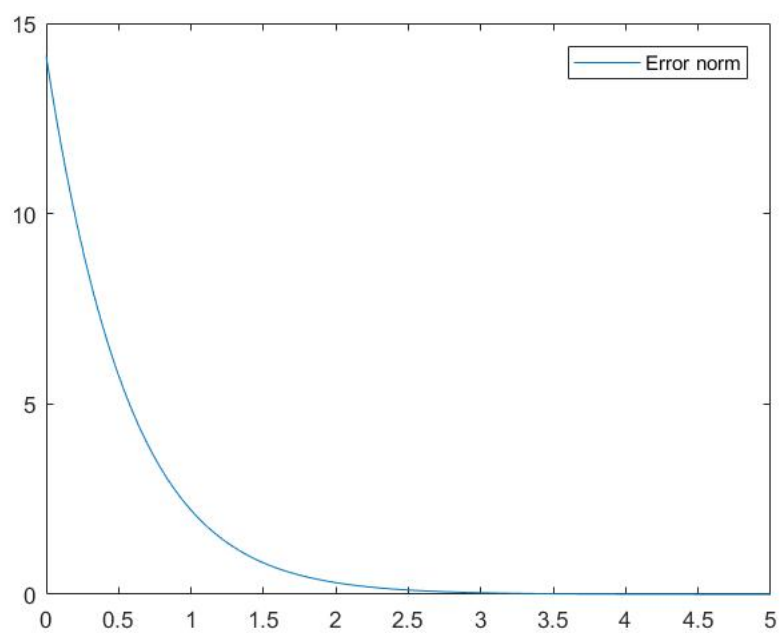
Using the ode45 solver, the values of the states are obtained:



With the information about the values of the states, an different function is created to calculate the estimation error norm:

```
function e = errorNorm(dx)
[m,n] = size(dx);
e = zeros(m,1);
for i=1:m
e(i) = norm(dx(i,1:2) - dx(i,3:4));
end
end
```

With this function, the following plot is obtained:



When comparing both plots, it is evident that the estimation is accurate at around 3 seconds.

(c) How can you improve the convergence of the estimation error? Show that both analytically and via implementation.

It can be done by providing the A_{CL} matrix with more aggresive evalues, i.e., making the evalues larger negatives.

Proof:

The solution of the error function is as follows:

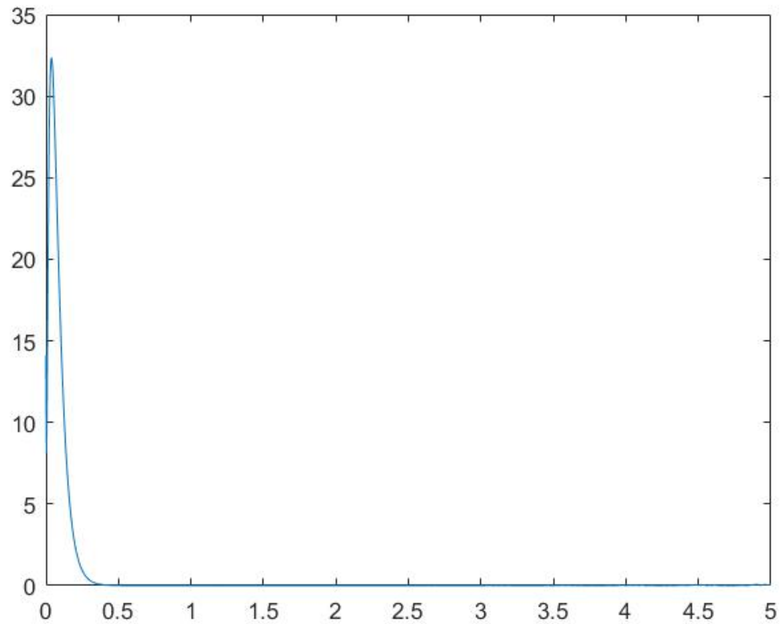
$$e(t) = e^{A-CL}$$

If the evalues of $A - CL$ are large negatives, this means that $e(t)$ will converge faster to zero as time goes on.

Using the previous code, L will be modified to provide $A - CL$ with evales -40 and -20:

$$L = \begin{bmatrix} 60 \\ -310.5 \end{bmatrix}$$

Using the exact same steps as in (b) the next plot is obtained:



Note that the estimation error becomes zero in less than a second. However, at the very beginning there is a high error value (almost equal to 35).

(d) Most systems are noisy, in the sense that the real dynamics are more like:

$$\dot{x}(t) = Ax(t) + Bu(t) + w(t)$$

where $w(t)$ is the noise. Often, we assume that $w(t)$ is a random variable with a well known distribution. Assume that $w(t)$ is a white Gaussian noise. You can implement this via the `randn` or `rand` commands in MATLAB. For example, you can write in the ODE solver file `w=0.1*randn(n,1)` which defines a vector of random quantities.

Investigate whether the observer you designed in the above parts is robust enough to this noise signal in the state evolution. Of course, you should not add noise in the observer dynamics, but only in the state dynamics. You can change the scaling factor 0.1 to something smaller or bigger—depending on the robustness of this observer.

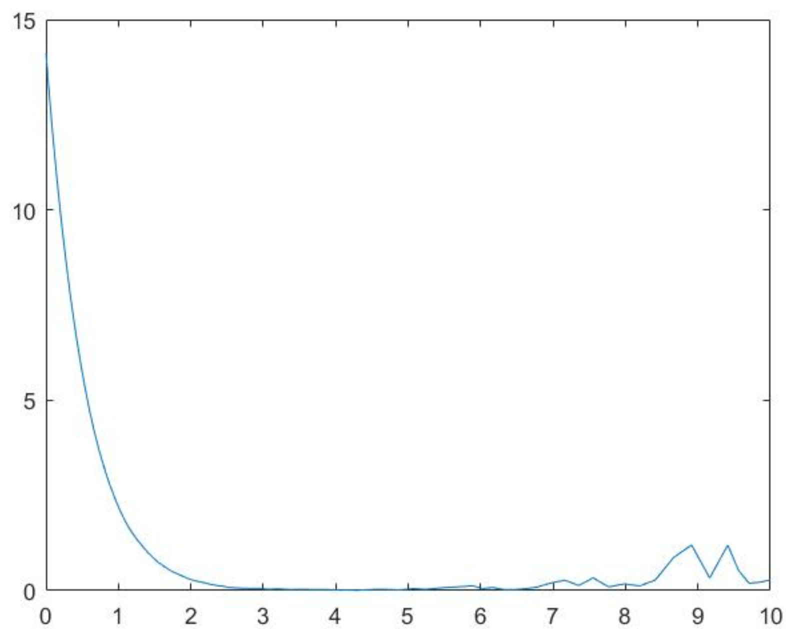
Can you draw any conclusions?

The noise will be applied as follows:

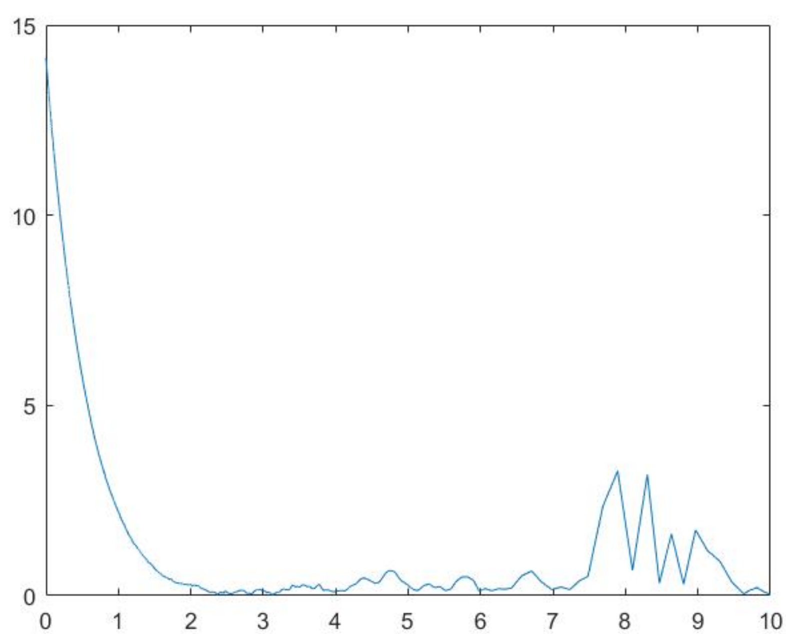
```
function dx = problem3(t, x)
L = [6; 4.5];
A = [3 -2; 4 -3];
B = [1; -1];
C = [1 0];
u = 2*sin(10*t);
w = zeros(4,1);
w(1:2) = 1*randn(2,1);

dx = [A zeros(2); L*C (A-L*C)]*x + [B;B]*u + w;
end
```

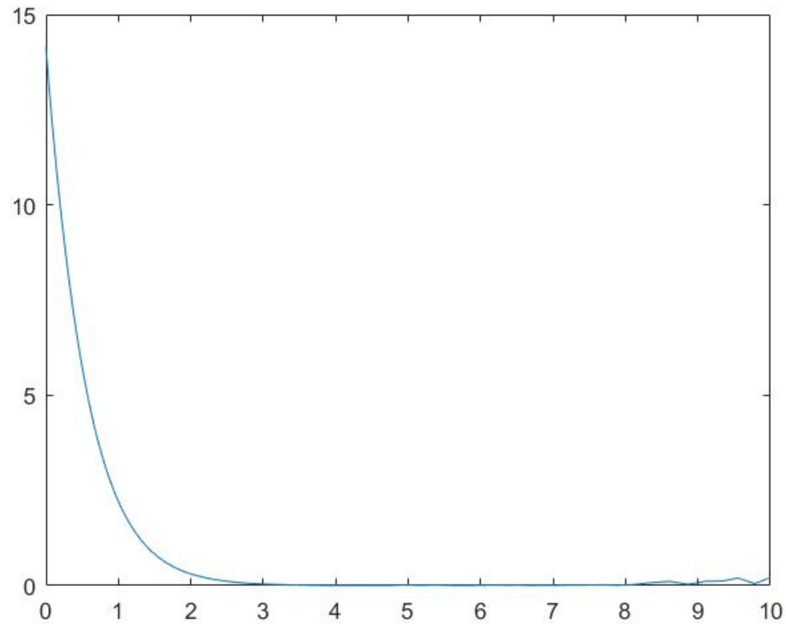
When solving the system of equations with the noise (0.1*randn(n,1)) using ode45 for the first L proposed in (b), the next error norm plot is obtained:



From the image it is evident that the estimator is able to keep up with the noise present on the system, however, there is a noticeable degree of error, but it might bearable.
When changing the noise to 1*randn(n,1) the following plot is obtained:



When increasing the scale of error, it can be concluded that the estimator fails to estimate the internal states, and a great degree of error is produced, especially after 7 seconds.
Finally, when the noise is .01*randn(n,1) the following plot is obtained:



As expected, the error norm for this level of noise is minimum, and it could be concluded that the estimator can handle the presence of the noise perfectly.