Group7-Assignment3

Akshay Seethanadi Aravind   Mukundh Balabhadra   Yogith Madha

December 16, 2022

Continuing to use the DC motor with flywheel modeled analyzed in the previous assignments M2, we will use the discrete time state space representation obtained.

Question a

From Imperical Rule, we know that data will fall within ($\mu \pm 3\sigma$) for 99.7% of a normal distribution. Thus, if V_a varies between ±0.3V. Then, V_a -

$$\mu \pm 3\sigma = \pm 0.3 \tag{1}$$

where we know it has zero mean ($\mu = 0$).

$$\sigma_{Va} = \frac{0.3}{3} = 0.1 \tag{2}$$

For T_e, we know that it is 10% of 1Nm,

$$\mu \pm 3\sigma = \pm 0.1 \tag{3}$$

$$\sigma_{Te} = \frac{0.1}{3} \tag{4}$$

$$Covariancematrix = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} = \begin{bmatrix} Cov_{11} & Cov_{12} \\ Cov_{21} & Cov_{22} \end{bmatrix}$$

As the parameters are un-correlated, $Cov_{12} = Cov_{21} = 0$.

$$Q_w = \begin{bmatrix} \sigma_{Va}^2 & 0 \\ 0 & \sigma_{Te}^2 \end{bmatrix} = \begin{bmatrix} 0.01 & 0 \\ 0 & 1.11 * 10^{-3} \end{bmatrix}$$

As the disturbance is added to the input,

$$u = \begin{bmatrix} Va + W_r(k) \\ Te + W_T(k) \end{bmatrix}$$

Thus, we can re-write this as,

$$\dot{x} = A_d x + B_d \begin{bmatrix} Va \\ Te \end{bmatrix} + B_d \begin{bmatrix} W_r(k) \\ W_T(k) \end{bmatrix} \tag{5}$$

Thus, **N =** B_d

Question b

We know that v_1, v_2 are measurement noise. Following the same procedure as in Question a, ($\mu \pm 3\sigma$ = Variation).

For v_1,

$$\mu + 3\sigma = 0.02 \tag{6}$$

$$\sigma_{v1} = \frac{0.02}{3} \tag{7}$$

For v_2,

$$\mu + 3\sigma = 0.01 \tag{8}$$

$$\sigma_{v2} = \frac{0.01}{3} \tag{9}$$

As they are un-correlated,

$$Q_v = \begin{bmatrix} \sigma_{v1}^2 & 0 \\ 0 & \sigma_{v2}^2 \end{bmatrix} = \begin{bmatrix} 4.44 * 10^{-5} & 0 \\ 0 & 1.11 * 10^{-5} \end{bmatrix}$$

Question c

The state-space equation is given by,

$$x(k+1) = Ax(k) + Bu(k) + Nw(k) \tag{10}$$

$$y(k) = Cx(k) + v(k) \tag{11}$$

We will use the discretised matrices(A_d, B_d, C_1 and D_1 in place of A, B, C and D) from the previous assignment M2. Thus, the code for the implementation is as shown in figure 1,

```
sys = ss(Ad,[Bd Bd],C1,[D_1 D_1]); % State space model discrete  system.
QN = [0.01 0; 0 1.11e-3]; %covariance matrix of Process Noise
RN = [4.44e-5 0; 0 1.11e-5];  %covariance matrix of Measurment Noise
NN = zeros(2,2);
[KEST,L,P1] = kalman(sys,QN,RN,NN); % Kalman filter KEST - Estimates states L - Kalman gain
eigen = eig(Ad - L*C1); % observer stability
```

Figure 1: matlab code for Kalman

We obtain the results as,

$$\textit{Observer gain matrix, } L = \begin{bmatrix} 0.0005 & 0.0009 \\ 0.0005 & 0.0018 \\ 0.0003 & 0.0113 \\ -0.0001 & 0.6970 \\ -0.0292 & 1.6755 \end{bmatrix}$$

$$\textit{State estimation error, } P = \begin{bmatrix} 0.0000 & 0.0000 & -0.0000 & 0.0001 & 0.0001 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ -0.0000 & 0.0000 & 0.0001 & 0.0011 & 0.0031 \\ 0.0001 & 0.0000 & 0.0011 & 0.3429 & 0.0248 \\ 0.0001 & 0.0000 & 0.0031 & 0.0248 & 0.1096 \end{bmatrix}$$

The eigen values are,

$$\textit{eigen values} = \begin{bmatrix} -0.6883 + 0.0000i \\ 0.9995 + 0.0000i \\ 0.1438 + 0.5209i \\ 0.1438 - 0.5209i \\ -0.0001 + 0.0000i \end{bmatrix} \tag{12}$$

As we can see from above, the eigen values are inside the unit circle, but one value is nearly on the unit circle, thus the system is marginally stable.

Question d

A LQG controller was designed to simulate a step response with reference $r_{\omega 2}$ from 10rad/s to 100rad/s on a discrete time noise corrupted system.

A Reference Gain

We created a controller that commands the process through its input u by using the system state x and the reference input r. The input u is calculated using the equation (12).

$$u(t) = -Kx(t) + K_r r(t) \tag{13}$$

$$Kr = (C_d(I - A_d + B_d K)^{-1} B_d)^{-1} \tag{14}$$

$$eigenvalues = (A_d - B_d K) \tag{15}$$

- The LQR gain K was computed using a Matlab command **"dlqr"**.
- K_r was calculated using equation 13.
- The dimensional weighing matrices Q_x and Q_u used to achieve an expensive control were:

$$Qx = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 150 & 0 \\ 0 & 0 & 0 & 0 & 1/700 \end{bmatrix}$$

$$Qu = \begin{bmatrix} 100 & 0 \\ 0 & 50 \end{bmatrix}$$

- The reference tracking for ω_2 is shown in figure 2. An input disturbance of 1V is added at 5sec.

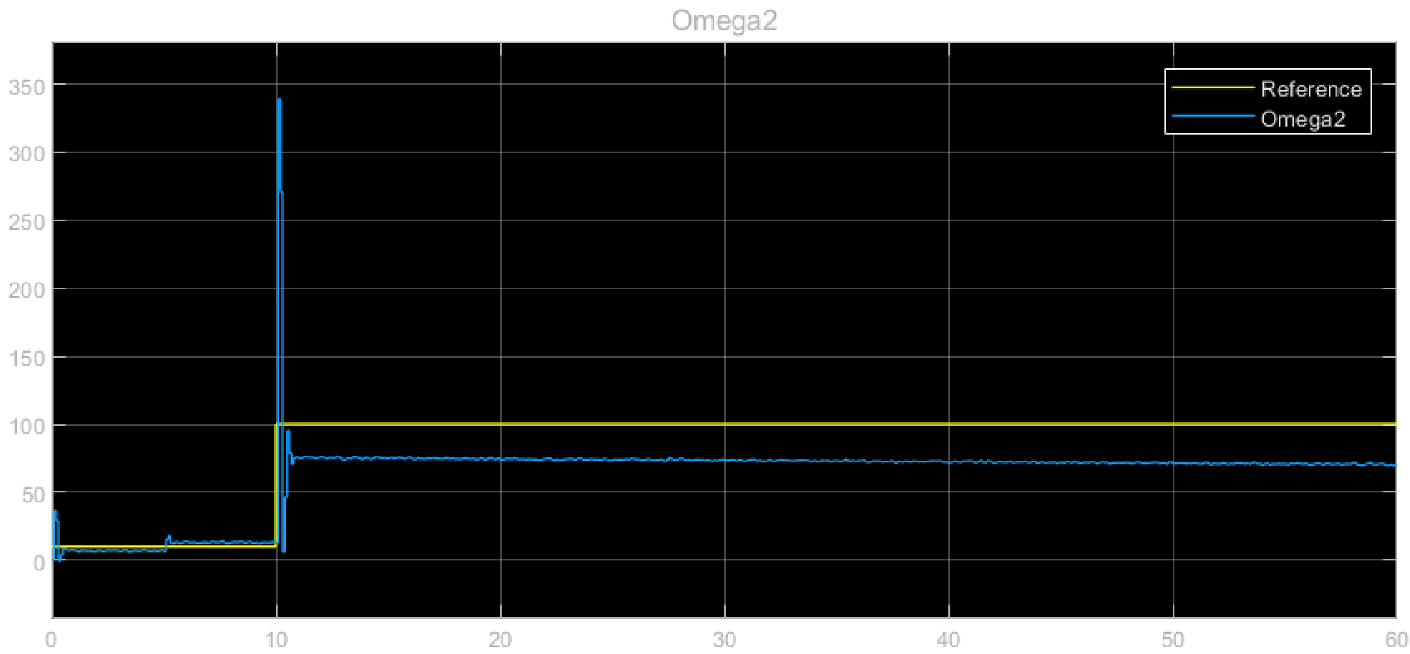

Figure 2: Reference Gain Simulink Model

- The tuning of the LQG controller using reference gain is quite difficult as the condition number is very high (**9.5012e+03**).
- From the eigen values (equation 12), we see that one of the values is approximately equal to unity, which signifies that the corresponding state is not reliable.
- The Simulink model used for reference gain controller is shown in figure 3.

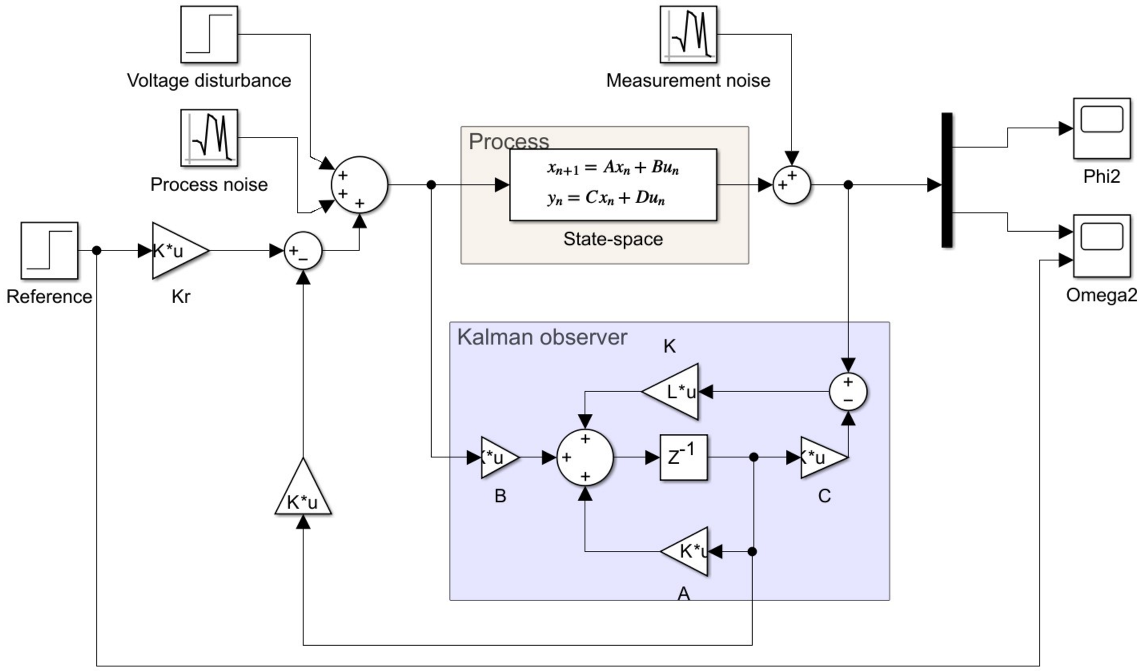

Figure 3: Reference Gain Simulink Model

B Integral Action

- The dimensional weighing matrices Q_x and Q_u used to achieve an expensive control were:

$$Qx = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.0014 \end{bmatrix}$$

$$Qu = \begin{bmatrix} 1000 & 0 \\ 0 & 500 \end{bmatrix}$$

- The values for the matrix were selected to achieve an expensive control. The values for the proportional (K_p) and integral (K_I) gains are as follows.

$$K_p = \begin{bmatrix} -4.3530 & 4.3640 & -0.0110 & 0.0019 & -0.0017 \\ 33.2917 & -36.1140 & 2.8230 & 0.0158 & 0.0727 \end{bmatrix}$$

$$K_I = \begin{bmatrix} -0.0027 \\ -0.0914 \end{bmatrix}$$

- The reference tracking for ω_2 is shown in figure 4. An input disturbance of 1V is added at 5sec.

- The Simulink model used for integral action controller is shown in figure 5.

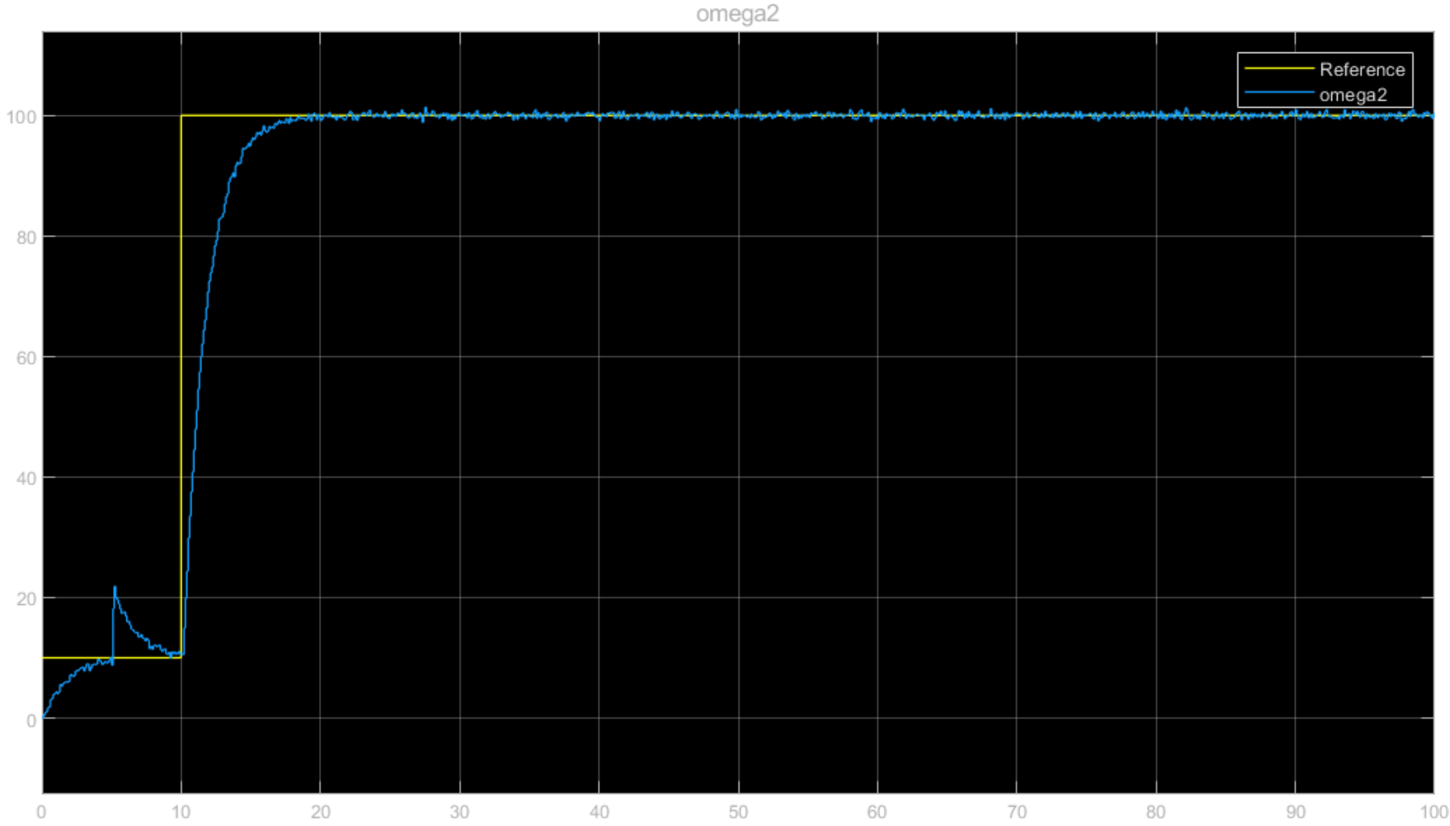

Figure 4: LQG with Integral action

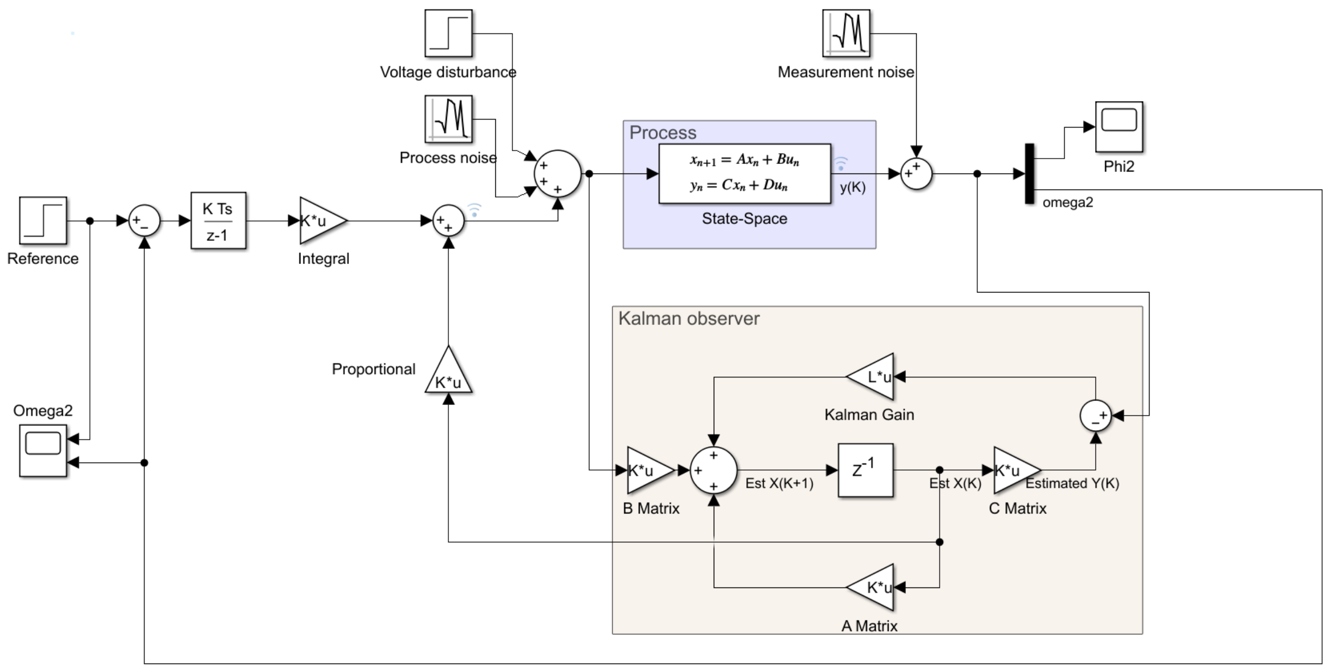

Figure 5: Simulink model for LQG with integral action