

## Lecture 12

*Lecturer: Asst. Prof. M. Mert Ankarali*

### 12.1 Discrete-time Luenberger Observer

In general the state,  $x[k]$ , of a system is not accessible and *observers (estimators, filters)* have to be used to extract this information. The output,  $y[k]$ , represents the measurements which is a function of  $x[k]$  and  $u[k]$ .

$$\begin{aligned}x[k+1] &= Gx[k] + Hu[k] \\ y[k] &= Cx[k]\end{aligned}$$

A Luenberger observers is built using a “simulated” model of the system and the errors caused by the mismatched initial conditions  $x_0 \neq \hat{x}_0$  (or other types of perturbations) are reduced by introducing output error feedback.

Let’s assume that the state vector of the simulated system is  $\hat{x}[k]$ , then the state space equation of this synthetic system takes the form

$$\begin{aligned}\hat{x}[k+1] &= G\hat{x}[k] + Hu[k] \\ \hat{y}[k] &= C\hat{x}[k]\end{aligned}$$

Note that since  $u[k]$  is the input that is supplied by the controller, we assume that it is known apriori. If  $x[0] = \hat{x}[0]$  and when there is no model mismatch or uncertainty in the system then we expect that  $x[k] = \hat{x}[k]$  and  $y[k] = \hat{y}[k]$  for all  $k$ . When  $x[0] \neq \hat{x}[0]$ , then we should observe a difference between the measured and predicted output  $y[k] \neq \hat{y}[k]$ . The core idea in Luenberger observer is feeding the error in the output prediction  $y[k] - \hat{y}[k]$  to the simulated system via a linear feedback gain.

$$\begin{aligned}\hat{x}[k+1] &= G\hat{x}[k] + Hu[k] + L(y[k] - \hat{y}[k]) \\ \hat{y}[k] &= C\hat{x}[k]\end{aligned}$$

In order to understand how a Luenberger observer works and to choose a proper observer gain  $L$ , we define an error signal  $e[k] = x[k] - \hat{x}[k]$ . The dynamics w.r.t  $e[k]$  can be derived as

$$\begin{aligned}e[k+1] &= x[k+1] - \hat{x}[k+1] \\ &= (Gx[k] + Hu[k]) - (G\hat{x}[k] + Hu[k] + L(y[k] - \hat{y}[k])) \\ e[k+1] &= (G - LC)e[k]\end{aligned}$$

where  $e[0] = x[0] - \hat{x}[0]$  denotes the error in the initial condition.

If the matrix  $(G - LC)$  is stable then the errors in initial condition will diminish eventually. Moreover, in order to have a good observer/estimator performance the observer convergence should be sufficiently fast.

### 12.1.1 Observer Gain & Pole Placement

Similar to the state-feedback gain design, the fundamental principle of “pole-placement” Observer design is that we first define a desired closed-loop eigenvalue set and compute the associated desired characteristic polynomial.

$$\begin{aligned}\mathcal{E}^* &= \{\lambda_1^*, \dots, \lambda_n^*\} \\ p^*(z) &= (z - \lambda_1^*) \cdots (z - \lambda_n^*) \\ &= z^n + a_1^* z^{n-1} + \cdots + a_{n-1}^* z + a_n^*\end{aligned}$$

The necessary and sufficient condition on arbitrary observer pole-placement is that the system should be fully Observable. Then, we tune  $L$  such that

$$\det(zI - (G - LC)) = p^*(z)$$

### Direct Design of Observer Gain

If  $n$  is small, the most efficient method could be the direct design.

**Example:** Consider the following DT system

$$\begin{aligned}x[k+1] &= \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} x[k] + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u[k] \\ y[k] &= \begin{bmatrix} 1 & -1 \end{bmatrix} x[k]\end{aligned}$$

Design an observer such that estimator poles are located at  $\lambda_{1,2} = 0$  (Dead-beat Observer)

**Solution:** Desired characteristic equation can be computed as

$$p^*(z) = z^2$$

Let  $L = \begin{bmatrix} l_2 \\ l_1 \end{bmatrix}$ , then the characteristic equation of  $(G - LC)$  can be computed as

$$\begin{aligned}\det(zI - (G - LC)) &= \det\left(\begin{bmatrix} z - 1 + l_2 & -l_2 \\ l_1 & z - 2 - l_1 \end{bmatrix}\right) \\ &= z^2 + z(l_2 - l_1 - 3) + (l_1 - 2l_2 + 2)\end{aligned}$$

If we match the equations

$$\begin{aligned}l_2 - l_1 &= 3 \\ -l_1 + 2l_2 &= 2 \\ l_2 &= -1 \\ l_1 &= -4\end{aligned}$$

Thus  $L = \begin{bmatrix} -1 \\ -4 \end{bmatrix}$

## 12.2 Closed-Loop Observer & State-Feedback

In the state-feedback control policy the input is ideally defined by the following law

$$u[k] = -Kx[k]$$

However, as mentioned in Observer lecture, in general we don't have direct access to the all states of the system. In this case, we learnt how to design an Observer/Estimator of the states. In this respect, it is natural to assume that in a closed-loop system, the control policy that define the input should depend on the estimated states

$$u[k] = -K\hat{x}[k]$$

However the important question how this coupling affect the closed-loop behavior, and even deeper question can be even use such a policy. The advantage of LTI systems is that state-feedback gain, and observer gain can be seperately designed and we guarantee a stable closed-loop performance. In this section, we will analyze the coupled system Equations of motion for the closed-loop observer & state-feedback based control system is given below

$$\begin{aligned} x[k+1] &= Gx[k] + Hu[k] \\ y[k] &= Cx[k] \\ \hat{x}[k+1] &= G\hat{x}[k] + Hu[k] + L(y[k] - \hat{y}[k]) \\ \hat{y}[k] &= C\hat{x}[k] \\ u[k] &= -K\hat{x}[k] \end{aligned}$$

If we eliminate  $u[k]$  and  $\hat{y}[k]$  we obtain following dynamical representation

$$\begin{aligned} x[k+1] &= Gx[k] - HK\hat{x}[k] \\ \hat{x}[k+1] &= G\hat{x}[k] - HK\hat{x}[k] + LC(x[k] - \hat{x}[k]) \\ y[k] &= Cx[k] - DK\hat{x}[k] \end{aligned}$$

Now let's replace  $\hat{x}[k]$  with  $e[k] = x[k] - \hat{x}[k]$

$$\begin{aligned} x[k+1] &= (G - HK)x[k] + HK e[k] \\ e[k+1] &= (G - LC)e[k] \\ y[k] &= (C - DK)x[k] + DK e[k] \end{aligned}$$

Now let's define a state for the whole system,  $z[k] = \begin{bmatrix} x[k] \\ e[k] \end{bmatrix}$  then the state-space representation is given by

$$\begin{aligned} z[k+1] &= \begin{bmatrix} (G - HK) & HK \\ 0_{n \times n} & (G - LC) \end{bmatrix} z[k] \\ y[k] &= \begin{bmatrix} (C - DK) & DK \end{bmatrix} z[k] \end{aligned}$$

The system matrix is in block diagonal form and the eigenvalues of this new system matrix is find by taking the union of eigenvalues of  $(G - HK)$  and eigenvalues of  $(G - LC)$ . Thus a separate pole-placement can be performed for the state-feedback controller and the observer.