# 6.3100 April 29, 2026 Lecture:
# LQR Optimization of Luenberger Observers

April 30, 2026

## Last Lecture

- Luenberger (state) observers

## Outline for Today

- Example: Luenberger observer for a part of maglev model
- Using LQR to optimize Luenberger observer gain

## Last lecture: Luenberger (State) Observer

For a state space model of the form

$$\dot{x}(t) = Ax(t) + B_1 w(t) + B_2 u(t), \qquad y(t) = C_2 x(t) + D_{21} w(t) + D_{22} u(t),$$

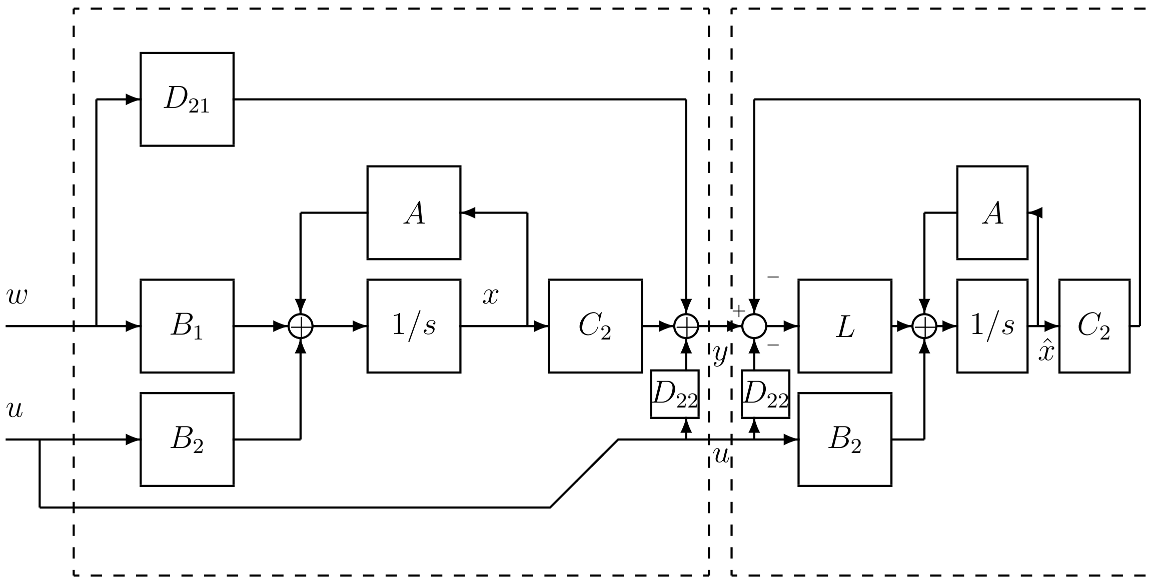
where

- $x(t) \in \mathbb{R}^n$ is the state vector to estimate;
- $u(t) \in \mathbb{R}^m$ is the control command, assumed to be known "in real time" (i.e., $u(t)$ is known at time $t$);
- $y(t) \in \mathbb{R}^k$ is the sensor measurement (multi-dimensional when there are multiple sensors), assumed to be known in real time time;
- $w(t) \in \mathbb{R}^d$ is the noise vector, not known;
- $A$, $B_1$, $B_2$, $C_2$, $D_{21}$, $D_{22}$ are known constant real matrices of appropriate dimensions;

the *Luenberger observer* has been introduced in the form

$$\frac{d\hat{x}(t)}{dt} = A\hat{x}(t) + B_2u(t) + L(y(t) - C_2\hat{x}(t) - D_{22}u(t)),$$

depicted on the block diagram below:




where $L$ is a constant matrix to be chosen, depending on the original coefficients $A$, $B_1$, $C_2$, $D_{21}$. The observer formula is derived by starting with the original equation for $dx/dt$, adding the term $L(y - C_2x - D_{21}w - D_{22}u)$ (which equals zero) to the right side, and then replacing $x(t)$ with $\hat{x}(t)$ and removing all terms with $w(t)$ from the result. Since the estimation error $e(t) = \hat{x}(t) - x(t)$ will satisfy the differential equation

$$\dot{e}(t) = (A - LC_2)e(t) + (LD_{21} - B_1)w(t),$$

it is required that all eigenvalues of $A - LC_2$ have negative real part.

The *observer gain* $L$ can be designed by choosing $L$ to place the eigenvalues of matrix $A - LC_2$ into desired positions in the left half plane, which is the same as placing the eigenvalues of its transpose

$$(A - LC_2)^T = A^T - C_2^T L^T = A^T - C_2^T K, \qquad \text{where} \quad K = L^T,$$

and hence can be handled by the `place` algorithm. However, the same objections voiced to designing full state feedback by pole placement (basically, "how do we know the pole placement locations which offer a god compromise between improving closed loop performance and satisfying control effort limitations?") prompt us to look for an LQR-like strategy of optimizing $L$. This will be the main focus of this lecture.

First, however, we re-visit the observer design for the position/velocity part of our MAGLEV model, this time using a Luenberger observer.

## State Observer for MAGLEV Position and Velocity

While it is possible (and perhaps preferable) to design a Luenberger observer for the complete MAGLEV model (3 states), we will use a smaller example of its position/velocity dynamics here, capitalizing on the fact that it is driven by the coil current, which is observed.

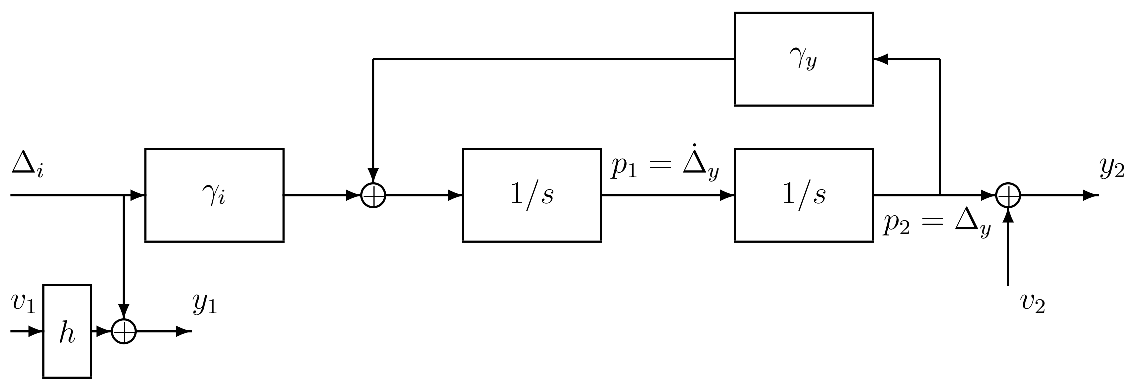
We will work with the differential equation model

$$\begin{aligned} \dot{p}_1(t) &= \gamma_i \Delta_i(t) + \gamma_y p_2(t), \\ \dot{p}_2(t) &= p_1(t), \end{aligned}$$

where $p_1(t) = \dot{\Delta}_y(t)$, $p_2(t) = \Delta_y(t)$, and $\gamma_i, \gamma_y$ are used as shortcut notation for $\gamma_{da/di}$ and $\gamma_{da/dy}$, respectively. We assume that the available measurements $y_1, y_2$ are given by

$$\begin{aligned} \dot{y}_1(t) &= \Delta_i(t) + h v_1(t), \\ \dot{y}_2(t) &= \Delta_y(t) + v_2(t), \end{aligned}$$

where $v_1, v_2$ are normalized measurement noises, and $h$ is the coefficient representing the known ratio of intensities between the magnetic field and position sensors noises.




How to translate this into the standard state observer setup? In other words, what is the meaning of $u(t)$, $x(t)$, $y(t)$, $w(t)$, and what are the coefficient matrices $A$, $B_1$, $B_2$, $C_2$, $D_{21}$, $D_{22}$?

- What is $x(t)$? The two obvious inertial signals in this model are $p_1(t) = \dot{\Delta}_y(t)$ and $p_2(t) = \Delta_y(t)$, so it is natural to define $x(t)$ as the two-dimensional vector with components $p_1(t)$ and $p_2(t)$. Note that, since we are considering only a part of the overall MAGLEV model, $\Delta_i(t)$, as an external input, cannot be used as a state for this part, even if it is a good choice of a state component for the complete MAGLEV model.

- What is $w(t)$? There are two noises in this model, so defining $w(t)$ as the two-dimensional vector with components $w_1(t) = v_1(t)$ and $w_2(t) = v_2(t)$ will work well.
- What is $u(t)$? While it is tempting to use $u(t) = \Delta_i(t)$, as $\Delta_i(t)$ is the external input into the model, this will not work out, as $\Delta_i(t)$ is not measured. Instead, we should use $u(t) = y_1(t)$, which means $\Delta_i(t) = u(t) - hw_1(t)$.
- What is $y(t)$? While, practically, both $y_1(t)$ and $y_2(t)$ are sensor measurements, we've just declared $y_1(t)$ as the control input $u(t)$, which leaves $y(t) = y_2(t)$.

Overall, this leaves us with the model

$$\begin{aligned} \dot{x}_1(t) &= \gamma_i(u(t) - hw_1(t)) + \gamma_y x_2(t), \\ \dot{x}_2(t) &= x_1(t), \\ y(t) &= x_2(t) + w_2(t), \end{aligned}$$

which means

$$A = \begin{bmatrix} 0 & \gamma_y \\ 1 & 0 \end{bmatrix}, \ B_1 = \begin{bmatrix} -h\gamma_i & 0 \\ 0 & 0 \end{bmatrix}, \ B_2 = \begin{bmatrix} \gamma_i \\ 0 \end{bmatrix}, \ C_2 = \begin{bmatrix} 0 & 1 \end{bmatrix}, \ D_{21} = \begin{bmatrix} 0 & 1 \end{bmatrix}, \ D_{22} = 0.$$

Actually, while the coefficient matrices $A$ and $C_2$ are needed to use the `place` algorithm to figure out a good $L$ matrix, and using LQR for designing $L$ requires knowing $A$, $B_1$, $C_2$, and $D_{21}$, one does not need the coefficient matrices $A$, $B_1$, $B_2$, $C_2$, $D_{21}$, $D_{22}$ in order to write down the Luenberger observer equations. Instead, it is sufficient to realize that, for

$$\hat{x}(t) = \begin{bmatrix} \hat{x}_1(t) \\ \hat{x}_2(t) \end{bmatrix}, \qquad L = \begin{bmatrix} L_1 \\ L_2 \end{bmatrix},$$

we have

$$y(t) - C_2\hat{x}(t) = y_2(t) - \hat{x}_2(t),$$

and hence the observer differential equation for $\hat{x}_k(t)$ $(k = 1, 2)$ is obtained from the original differential equation for $x_k$ by replacing $x_1(t)$, $x_2(t)$ with $\hat{x}_1(t)$, $\hat{x}_2(t)$, setting $w_1(t) = w_2(t) = 0$, and adding the "correction" term $L_k(y_2(t) - \hat{x}_2(t))$:

$$\begin{aligned} \frac{d\hat{x}_1(t)}{dt} &= \gamma_i y_1(t) + \gamma_y \hat{x}_2(t) + L_1(y_2(t) - \hat{x}_2(t)), \\ \frac{d\hat{x}_2(t)}{dt} &= \hat{x}_1(t) + L_2(y_2(t) - \hat{x}_2(t)). \end{aligned}$$

The observer dynamics is stable when its "A" matrix (which is the same as $A - LC_2$)

$$A_{obs} = \begin{bmatrix} 0 & \gamma_y - L_1 \\ 1 & -L_2 \end{bmatrix}$$

has all eigenvalues with negative real part. Since the characteristic polynomial of $A_{obs}$ is $s^2 + L_2 s + (L_1 - \gamma_y)$, we can make it anything we want by selecting $L_1$ and $L_2$.

It is important to notice that the Luenberger observer produces estimates for *all* states, even those that are measured. In particular, in this example it generates the estimate $\hat{x}_2(t)$ of $\Delta_y(t)$. Is it better than $y_2(t)$, the direct measurement of $\Delta_y(t)$? The answer is "it depends", mostly on the relative noise levels, but also on the accuracy of the original model (in particular, how accurate our values of $\gamma_i$ and $\gamma_y$ are). When the model is accurate, but the noise level in measuring $\Delta_y(t)$ directly is large compared to the noise level in measuring $\Delta_i(t)$, it makes sense to use $\hat{x}_2(t)$ instead of $y_2(t)$ as the estimate of $\Delta_y(t)$.

## Using LQR to Optimize the State Observer Gain

Assuming a specific performance measure used to assess the quality of Luenberger observer, the standard LQR algorithm can be used to optimize the observer gain $L$.

We return back to the general setup with state equations

$$\dot{x}(t) = Ax(t) + B_1 w(t) + B_2 u(t), \qquad y(t) = C_2 x(t) + D_{21} w(t) + D_{22} u(t).$$

We allow an arbitrary linear system with constant coefficients

$$\text{E:} \qquad \dot{x}_e(t) = A_e x_e(t) + B_{ey} y(t) + B_{eu} u(t), \qquad \hat{x}(t) = C_e x_e(t) \tag{1}$$

which is *stable* (i.e., such that all eigenvalues of matrix $A_e$ have negative real part) to serve as an estimator of $x(t)$. In particular, the Luenberger observer produces an estimate $\hat{x}(t)$ of $x(t)$ given by

$$\frac{d\hat{x}(t)}{dt} = A\hat{x}(t) + B_2 u(t) + L(y(t) - C_2 \hat{x}(t) - D_{22} u(t)),$$

which fits the generic allowed estimator format, with

$$A_e = A - LC_2, \quad B_{ey} = L, \quad B_{eu} = B_2 - LD_{22}, \quad C_e = I.$$

We define the performance of an estimator (1) in terms of the transfer functions $H_{w_k \to e_i}(s)$ from the $k$-th componet $w_k(t)$ of $w(t)$ to the $i$-th component $e_i(t)$ of the estimation error $e(t) = x(t) - \hat{x}(t)$. Specifically, the performance measure equals

$$J(E) = \frac{1}{2\pi} \sum_{k,i} \int_{-\infty}^{\infty} |H_{w_k \to e_i}(j\omega)|^2 d\omega$$

when all $H_{w_k \to e_i}(s)$ are *stable* (i.e., no poles with non-negative real part), and $+\infty$ otherwise. It can be shown that, assuming $w$ is the so-called *normalized white noise*, $J(u)$

equals the mean square value $\mathbb{E}[|e(t)|^2]$ of the estimation error $e(t)$ in steady-state (if any of $H_{w_k \to e_i}(s)$ is unstable then there is no steady state, and the mean square value converges to $+\infty$ as $t \to +\infty$). The "normalized white noise" assumption means that the samples of different components of $w$ at different times (say, $w_k(t)$ and $w_i(\tau)$, where either $k \neq i$ or $t \neq \tau$) are independent and have equal intensity. Unfortunately, mathematical meaning of "white noise" also implies that, for every fixed $t$, the standard deviation of every component $w_k(t)$ of $w(t)$ is infinite, which seems a bit counter intuitive, if not bonkers crazy. We will have better luck interpreting "white noise" in discrete time setting, in the next lecture.

The main statement on optimizing Luenberger observer via LQR is as follows:

LET $q(t) = -Kp(t)$ BE THE OPTIMAL FEEDBACK IN THE TASK OF MINIMIZING THE INTEGRAL

$$J_{lqr} = \int_0^\infty |B_1^T p(t) + D_{21}^T q(t)|^2 dt$$

SUBJECT TO

$$\dot{p}(t) = A^T p(t) + C_2^T q(t), \quad p(0) = p_0, \qquad \lim_{t \to +\infty} p(t) = 0.$$

(WE KNOW THERE IS JUST *one* $K$ MINIMIZING $J_{lqr}$ SIMULTANEOUSLY FOR *all* $p_0$.) THEN THE LUENBERGER OBSERVER WITH $L = K^T$ MINIMIZES $J(E)$.

For example, in Python the optimal observer gain $L$ can be designed using

```
import control as ct
K = ct.lqr(A.T, C2.T, B1@B1.T, D21@D21.T, B1@D21.T)
L = K.T
```

It is instructive (at least, for *memorizing* the statement above) to note that the coefficients $A, B_1, C_2, D_{21}$ of the original setup (they are the only ones relevant to the dependence of the estimation error $e(t)$ of the Luenberger observer on the noise $w(t)$) form "naturally" the matrix

$$M = \begin{bmatrix} A & B_1 \\ C_2 & D_{21} \end{bmatrix}, \qquad \text{so that} \qquad \begin{bmatrix} \dot{x}(t) \\ y(t) \end{bmatrix} = M \begin{bmatrix} x(t) \\ w(t) \end{bmatrix}$$

when $u(t) = 0$. On the other hand, the coefficients of the differential equation and the cost output $z(t) = B_1^T p(t) + D_{21}^T q(t)$ for the resulting LQR setup are given by

$$M^T = \begin{bmatrix} A^T & C_2^T \\ B_1^T & D_{21}^T \end{bmatrix}, \qquad \text{so that} \qquad \begin{bmatrix} \dot{p}(t) \\ z(t) \end{bmatrix} = M^T \begin{bmatrix} p(t) \\ q(t) \end{bmatrix}.$$

In other words, apparently, the relation between full state feedback and state estimation is that of *transposition* ("duality" in fancier math terminology).