Lecture 7: Luenberg Observer

In the previous lecture we discussed the concept of *observability* which gives us a binary answer to whether the state of a system can be determined from its outputs. Now, we will describe the design of an *observer* to actually achieve this goal, that is, to determine the state from a series of observations. Note that an *observer* and an *estimator* are closely related concepts but usually have different connotations. An observer is usually designed for a *deterministic system* (all the systems we've studied so far are deterministic) whereas a estimator is usually designed for a *stochastic system* (i.e., one in which noise or uncertainty is present). Generally observers are designed to stabilize an error variable that measures the discrepency between the actual state and the observer-determined state. Estimators on the other hand emphasize optimization to minimize an error while considering the noise properties of the system.

Observer design [Rugh, 1996, Ch. 15]. Consider a state space system

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}(t)\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t), \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0 \tag{1}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}(t)\boldsymbol{x}(t) \tag{2}$$

with the initial state $\boldsymbol{x}_0$ unknown. Our goal is to generate a $n \times 1$ function $\hat{\boldsymbol{x}}(t)$ that is an estimate of $\boldsymbol{x}(t)$ in the sense that as time increases the error in the estimate approaches zero

$$\lim_{t\to\infty}[\boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t)] = 0 \tag{3}$$

using only the available information $\boldsymbol{u}(t)$, $\boldsymbol{y}(t)$ for $t \in [t_0, t_1]$ where $t_1 > t_0$ is some time interval. Before proceeding we should first determine that the system is observable on this time interval. Assuming this property is confirmed, we can proceed to design the observer. The basic idea is to construct an analogous dynamic system that is driven by *both* the input and the output of the original system. In other words, the output $\boldsymbol{y}(t)$ becomes similar to a control input for the new observer system. Consider an observer system of the following form:

$$\dot{\hat{\boldsymbol{x}}}(t) = \boldsymbol{F}(t)\hat{\boldsymbol{x}}(t) + \boldsymbol{G}\boldsymbol{u}(t) + \boldsymbol{H}\boldsymbol{y}(t), \quad \hat{\boldsymbol{x}}(t_0) = \hat{\boldsymbol{x}}_0 \tag{4}$$

where the "$\hat{\boldsymbol{x}}$" symbol represents the observer's estimate of the state, with $\hat{\boldsymbol{x}}_0$ being the initial estimate assumed by the observer, and $\boldsymbol{F}(t) \in \mathbb{R}^{n\times n}$, $\boldsymbol{G}(t) \in \mathbb{R}^{n\times m}$, and $\boldsymbol{H}(t) \in \mathbb{R}^{n\times p}$.

Consistency of the observer. One obvious requirement for this new system is that if we initialize it with the correct guess, i.e., we choose $\hat{\boldsymbol{x}}_0 = \boldsymbol{x}_0$, then the our observer system should continue to produce the correct state, i.e., $\hat{\boldsymbol{x}}(t) = \boldsymbol{x}(t)$. We can choose our system matrices to satisfy this requirement by selecting

$$\boldsymbol{F}(t) = \boldsymbol{A}(t) - \boldsymbol{H}(t)\boldsymbol{C}(t) \tag{5}$$

$$\boldsymbol{G}(t) = \boldsymbol{B}(t) \tag{6}$$

With this choice our system becomes

$$\begin{aligned}\dot{\hat{\boldsymbol{x}}}(t) &= (\boldsymbol{A}(t) - \boldsymbol{H}(t)\boldsymbol{C}(t))\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{H}\boldsymbol{y}(t) &(7)\\ &= \boldsymbol{A}(t)\hat{\boldsymbol{x}}(t) - \boldsymbol{H}(t)\boldsymbol{C}(t)\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{H}\boldsymbol{y}(t) &(8)\\ &= \boldsymbol{A}(t)\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{H}[\boldsymbol{y}(t) - \boldsymbol{C}(t)\hat{\boldsymbol{x}}(t)] &(9)\end{aligned}$$

and it is clear that if $\hat{\boldsymbol{x}}(t) = \boldsymbol{x}(t)$ then $\boldsymbol{C}(t)\hat{\boldsymbol{x}}(t) = \boldsymbol{y}(t)$ and the last term vanishes leaving the estimate to evolve according to the true system dynamics (and thereby satisfying our consistency requirement).

Error convergence to zero. Having chosen $\boldsymbol{F}(t)$ and $\boldsymbol{G}(t)$ the matrix $\boldsymbol{H}(t)$ remains the only free parameter. This system is known as the *Luenberg* observer (named after David G. Luenberger, a professor at Stanford University). In the more general case that the initial estimate is chosen incorrectly $\hat{\boldsymbol{x}}(t) \neq \boldsymbol{x}(t)$ we want the property that the observer will asymptotically converge to the correct value, $\hat{\boldsymbol{x}}(t) \to \boldsymbol{x}(t)$ as $t \to \infty$. An equivalent condition is that the error

$$\boldsymbol{e}(t) = \boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t) \tag{10}$$

should go to zero, $\boldsymbol{e}(t) \to 0$ as $t \to \infty$. Our next task is to choose $\boldsymbol{H}(t)$ which is called the *observer gain* to satisfy this condition. We begin by writing the error dynamics by differentiating (10) with (1), (2), (4), (5), (6):

$$\begin{aligned}
\dot{\boldsymbol{e}}(t) &= \dot{\boldsymbol{x}}(t) - \dot{\hat{\boldsymbol{x}}}(t) && (11)\\
&= [\boldsymbol{A}(t)\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t)] - [\boldsymbol{F}(t)\hat{\boldsymbol{x}}(t) + \boldsymbol{G}\boldsymbol{u}(t) + \boldsymbol{H}(t)\boldsymbol{C}(t)\boldsymbol{x}(t)] && (12)\\
&= [\boldsymbol{A}(t)\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t)] - [(\boldsymbol{A}(t) - \boldsymbol{H}(t)\boldsymbol{C}(t))\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{H}(t)\boldsymbol{C}(t)\boldsymbol{x}(t)] && (13)\\
&= \boldsymbol{A}(t)\boldsymbol{x}(t) + \boldsymbol{B}(t)\boldsymbol{u}(t) - \boldsymbol{A}(t)\hat{\boldsymbol{x}}(t) + \boldsymbol{H}(t)\boldsymbol{C}(t)\hat{\boldsymbol{x}}(t) - \boldsymbol{B}(t)\boldsymbol{u}(t) - \boldsymbol{H}(t)\boldsymbol{C}(t)\boldsymbol{x}(t) && (14)\\
&= \boldsymbol{A}(t)(\boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t)) + \boldsymbol{H}(t)\boldsymbol{C}(t)(\hat{\boldsymbol{x}}(t) - \boldsymbol{x}(t)) && (15)\\
&= \boldsymbol{A}(t)\boldsymbol{e}(t) - \boldsymbol{H}(t)\boldsymbol{C}(t)\boldsymbol{e}(t) && (16)\\
&= (\boldsymbol{A}(t) - \boldsymbol{H}(t)\boldsymbol{C}(t))\boldsymbol{e}(t) && (17)
\end{aligned}$$

If we define $\boldsymbol{K} = \boldsymbol{A}(t) - \boldsymbol{H}(t)\boldsymbol{C}(t)$ then the above error dynamics are

$$\dot{\boldsymbol{e}}(t) = \boldsymbol{K}(t)\boldsymbol{e}(t)\ . \tag{18}$$

Observer Stability via Pole Placement. The error dynamics are stable (i.e., converge to zero) for an appropriate choice of $\boldsymbol{H}(t)$ (which determines $\boldsymbol{K}(t)$). The stabilizing gain matrix $\boldsymbol{K}(t)$ can be chosen using various techniques for linear control systems, for example, using pole placement. In pole placement we first compute the characteristic polynomial of the observer

$$\det(s\boldsymbol{I} - \boldsymbol{K}) = \det(s\boldsymbol{I} - [\boldsymbol{A} - \boldsymbol{H}\boldsymbol{C}]) = s^n + b_{n-1}s^{n-1} + \cdots + b_1 s + b_0 = 0 \tag{19}$$

with the observer feedback gains (i.e,. elements of $\boldsymbol{H}$) as unknown variables to be solved for. (Note that in the above equations the values $b_{n-1}, \cdots, b_1, b_0$ depend on the elements of $\boldsymbol{H}$.) Then we choose a set of desired eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_n$ for the closed loop system. Recall that placing the poles/eigenvalues in the left-half of the complex plane gives a stable mode corresponding to each eigenvalue. This is the desired condition for the error dynamics to converge to zero. The characteristic polynomial for our desired closed loop system is

$$(s - \lambda_1)(s - \lambda_2)\cdots(s - \lambda_n) = s^n + c_{n-1}s^{n-1} + \cdots + c_1 s + c_0 = 0 \tag{20}$$

By equating the coefficients of (19) and (20) we obtain a system of n equations:

$$c_{n-1} = b_{n-1}(\boldsymbol{H}) \tag{21}$$
$$c_{n-2} = b_{n-2}(\boldsymbol{H}) \tag{22}$$
$$\vdots \tag{23}$$
$$c_1 = b_1(\boldsymbol{H}) \tag{24}$$
$$c_0 = b_0 \tag{25}$$

where we've emphasized above that the b_i coefficients depend on $\boldsymbol{H}$. The above system can be solved algebraically to determine the observer gain matrix.

Aside: The above pole placement approach is conveniently implemented in MATLAB using the `H = place(A',C',P)'` command where `P` is a column vector of n desired eigenvalues. Notice the transpose `'` appearing in the matrices involved with the `place` command. This is due to the fact that `place` is primarily intended to find the control (not observer) gains of a closed loop system. However, we are exploiting the controllability-observability duality for pole placement. The system $\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x}$ with $\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x}$ is observable if and only if the system $\dot{\boldsymbol{x}} = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{C}^{\mathrm{T}}\boldsymbol{u}$ is controllable.

Example. Consider the system

$$\dot{\boldsymbol{x}}(t) = \begin{bmatrix} 0 & -1 \\ 1 & -2 \end{bmatrix} \boldsymbol{x}(t) + \begin{bmatrix} 2 \\ 1 \end{bmatrix} u(t) \tag{26}$$
$$\boldsymbol{y}(t) = \begin{bmatrix} 1 & 1 \end{bmatrix} \boldsymbol{x}(t) \tag{27}$$

with initial condition $\boldsymbol{x}(t_0) = [10, -3]^{\mathrm{T}}$ and a constant control input $u(t) = 1$. Suppose we have a (very inaccurate!) guess that the initial state is $\hat{\boldsymbol{x}}(t_0) = [-10, 5]^{\mathrm{T}}$. We wish to design a Luenberg observer to converge to the true state using only the scalar output $y(t) = x_1 + x_2$. The observer gain matrix is $\boldsymbol{H} \in \mathbb{R}^2$ in this case, since $n = 2$ and $p = 1$:

$$\boldsymbol{H} = \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} \tag{28}$$

Suppose that the desired poles are $\lambda_1 = -1$ and $\lambda_2 = -3$. The characteristic polynomial of the desired system is

$$(\lambda + 1)(\lambda + 3) = \lambda^2 + 4\lambda + 3 = 0 \tag{29}$$

Now, compute the poles of observer:

$$\det(s\boldsymbol{I} - [\boldsymbol{A} - \boldsymbol{H}\boldsymbol{C}]) = 0$$
$$\det\left(\begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \left\{\begin{bmatrix} 0 & -1 \\ 1 & -2 \end{bmatrix} - \begin{bmatrix} h_1 \\ h_2 \end{bmatrix}\begin{bmatrix} 1 & 1 \end{bmatrix}\right\}\right) = 0$$
$$\det\left(\begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \left\{\begin{bmatrix} 0 & -1 \\ 1 & -2 \end{bmatrix} - \begin{bmatrix} h_1 & h_1 \\ h_2 & h_2 \end{bmatrix}\right\}\right) = 0$$
$$\det\left(\begin{bmatrix} s + h_1 & 1 + h_1 \\ -1 + h_2 & s + 2 + h_2 \end{bmatrix}\right) = 0$$
$$(s + h_1)(s + 2 + h_2) - (1 + h_1)(-1 + h_2) = 0$$
$$s^2 + 2s + h_2 s + h_1 s + 2h_1 + h_1 h_2 + 1 - h_2 + h_1 - h_1 h_2 = 0$$
$$s^2 + \underbrace{(2 + h_2 + h_1)}_{b_1} s + \underbrace{(3h_1 - h_2 + 1)}_{b_0} = 0$$

which leads to two equations

$$4 = 2 + h_2 + h_1 \tag{30}$$
$$3 = 3h_1 - h_2 + 1 \tag{31}$$

From the first equation, $h_1 = 2 - h_2$ and substituting into the second:

$$3 = 3(2 - h_2) - h_2 + 1 \implies h_2 = 1 \implies h_1 = 1 \tag{32}$$

We can confirm that with $H = [1, 1]^{\mathrm{T}}$ the eigenvalues of the error dynamics are $\lambda_1 = -1$ and $\lambda_2 = -3$ and the system is stable

```
eig(A-H*C)
ans =
    -1
    -3
```

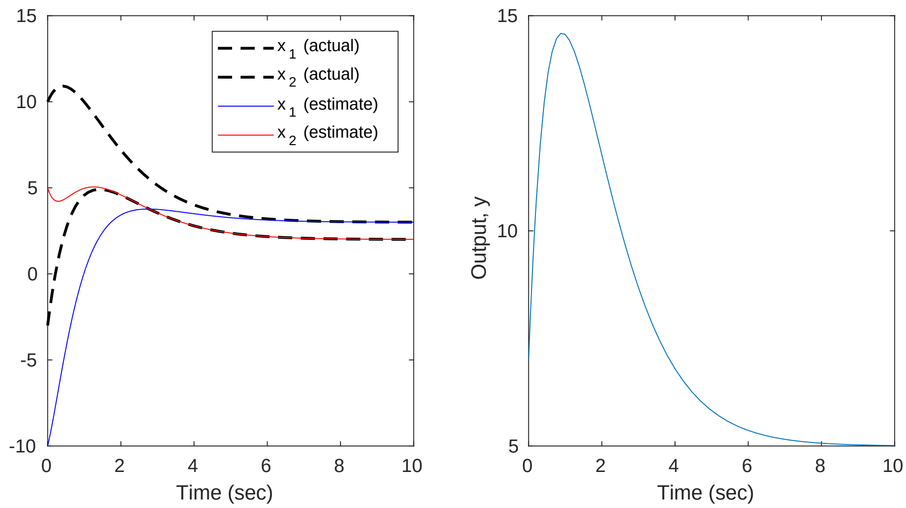
The observer gain could also be obtained using `place`:

```
>> A = [0 -1;1 -2];
>> C = [1 1];
>> H = place(A',C',[-1, -3])'
H =
    1.0000
    1.0000
```

The Luenberg observer is implemented in the simulation below using `ode45`. Note that in the simulation a system is defined with a state vector $\boldsymbol{z}(t) = [x_1(t), x_2(t), \hat{x}_1(t), \hat{x}_(t)]^{\mathrm{T}}$ for convenience. That is, we augment the original state with the observed states for ease of simulation.

```
params.A = [0 -1;1 -2];
params.B = [2; 1];
params.u = 1;
params.C = [1 1];
params.H = [1; 1];
x0 = [10 -3]';
xhat0 = [-10 5]';
z0 = [x0; xhat0];
tspan = [0, 10];
options = [];

% simulate observer
[t,Z] = ode45(@(t,z) observer_system(t,z,params), tspan, z0, options);

for i = 1:1:length(t) % recover output
    x = Z(i,1:2)';
    y(i) = params.C*x;
end

function dzdt = observer_system(t,z,params)
    A = params.A;
    B = params.B;
    C = params.C;
    H = params.H;
    u = params.u;

    x = z(1:2);
    dzdt(1:2,1) = A*x + B*u;  % actual system
    y = C*x;

    F = A - H*C;
    G = B;
    xhat = z(3:4,1);
    dzdt(3:4) = F*xhat + G*u + H*y; % observer
end
```

The solution can be plotted to give the response of the actual system and the estimated observer states as shown below.



References

[Rugh, 1996] Rugh, W. J. (1996). *Linear System Theory*. Prentice-Hall, Inc.