Topic #16

16.31 Feedback Control

State-Space Systems

- Open-loop Estimators
- Closed-loop Estimators

- **Observer Theory (no noise) – Luenberger**
  IEEE TAC Vol 16, No. 6, pp. 596–602, December 1971.
- **Estimation Theory (with noise) – Kalman**

Estimators/Observers

- **Problem:** So far we have assumed that we have full access to the state $x(t)$ when we designed our controllers.
  - Most often all of this information is not available.

- Usually can only feedback information that is developed from the sensors measurements.
  - Could try "output feedback"

$$u = Kx \quad \Rightarrow \quad u = \hat{K}y$$

  - Same as the proportional feedback we looked at at the beginning of the root locus work.
  - This type of control is very difficult to design in general.

- **Alternative approach:** Develop a replica of the dynamic system that provides an "estimate" of the system states based on the measured output of the system.

- **New plan:**
  1. Develop estimate of $x(t)$ that will be called $\hat{x}(t)$.
  2. Then switch from $u = -Kx(t)$ to $u = -K\hat{x}(t)$.

- Two key questions:
  - How do we find $\hat{x}(t)$?
  - Will this new plan work?

Estimation Schemes

- Assume that the system model is of the form:

$$\begin{aligned} \dot{x} &= Ax + Bu \ , \quad x(0) \text{ unknown} \\ y &= Cx \end{aligned}$$

where

1. A, B, and C are known.
2. $u(t)$ is known
3. Measurable outputs are $y(t)$ from $C \neq I$

- **Goal:** Develop a dynamic system whose state

$$\hat{x}(t) = x(t)$$

for all time $t \geq 0$. Two primary approaches:

  – Open-loop.

  – Closed-loop.

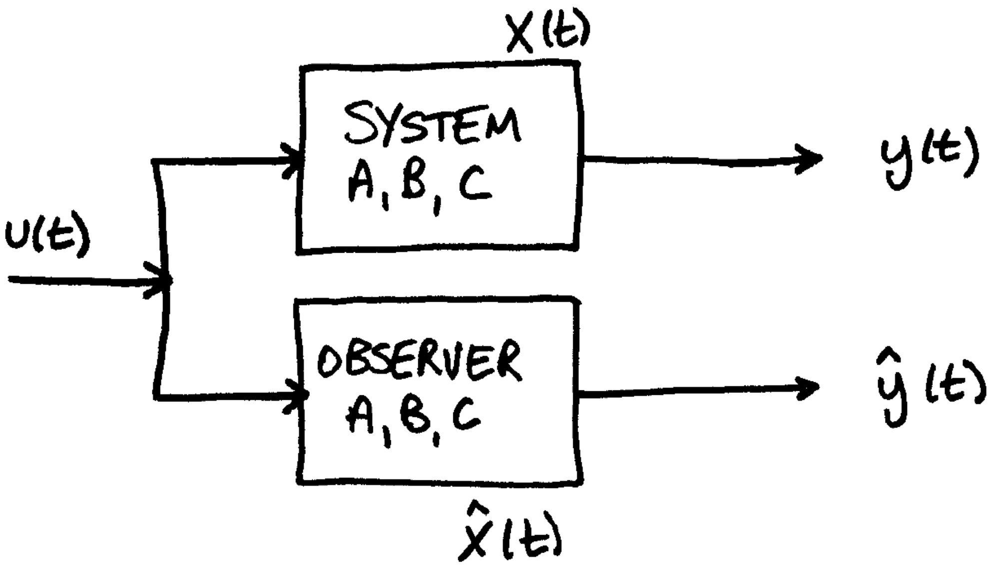
Open-loop Estimator

- Given that we know the plant matrices and the inputs, we can just perform a simulation that runs in parallel with the system

$$\dot{\hat{x}}(t) = A\hat{x} + Bu(t)$$

- Then $\hat{x}(t) \equiv x(t) \ \forall \ t$ provided that $\hat{x}(0) = x(0)$

- **Major Problem:** We do not know $x(0)$



- Analysis of this case:

$$\begin{aligned} \dot{x}(t) &= Ax + Bu(t) \\ \dot{\hat{x}}(t) &= A\hat{x} + Bu(t) \end{aligned}$$

- Define the **estimation error** $\tilde{x}(t) = x(t) - \hat{x}(t)$.
  Now want $\tilde{x}(t) = 0 \ \forall \ t$. (But is this realistic?)

- Subtract to get:

$$\frac{d}{dt}(x - \hat{x}) = A(x - \hat{x}) \quad \Rightarrow \quad \dot{\tilde{x}}(t) = A\tilde{x}$$

  which has the solution

$$\tilde{x}(t) = e^{At}\tilde{x}(0)$$

  – Gives the estimation error in terms of the initial error.

- Does this guarantee that $\tilde{x} = 0 \; \forall \; t$?
  Or even that $\tilde{x} \to 0$ as $t \to \infty$? (which is a more realistic goal).

  – Response is fine if $\tilde{x}(0) = 0$. But what if $\tilde{x}(0) \neq 0$?

- If A stable, then $\tilde{x} \to 0$ as $t \to \infty$, but the dynamics of the estimation error are completely determined by the open-loop dynamics of the system (eigenvalues of A).

  – Could be very slow.

  – No obvious way to modify the estimation error dynamics.

- Open-loop estimation does not seem to be a very good idea.

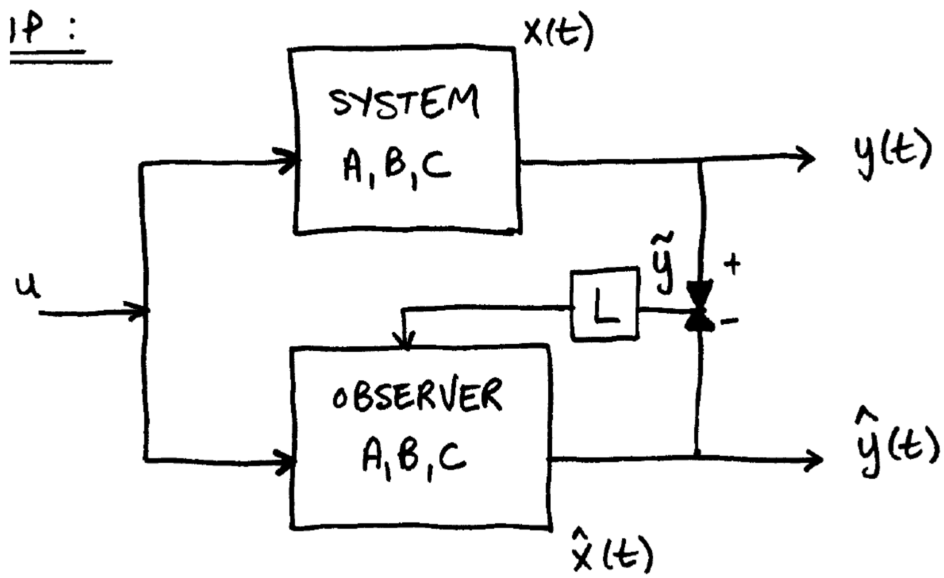
Closed-loop Estimator

- An obvious way to fix this problem is to use the additional information available:

  – How well does the estimated output match the measured output?

  $$\text{Compare: } \; y = Cx \; \text{ with } \; \hat{y} = C\hat{x}$$

  – Then form $\tilde{y} = y - \hat{y} \equiv C\tilde{x}$



- **Approach:** Feedback $\tilde{y}$ to improve our estimate of the state. Basic form of the estimator is:

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x}(t) + Bu(t) + \boxed{L\tilde{y}(t)} \\ \hat{y}(t) &= C\hat{x}(t) \end{aligned}$$

where L is the *user selectable gain matrix.*

- **Analysis:**

$$\begin{aligned} \dot{\tilde{x}} &= \dot{x} - \dot{\hat{x}} = [Ax + Bu] - [A\hat{x} + Bu + L(y - \hat{y})] \\ &= A(x - \hat{x}) - L(Cx - C\hat{x}) = A\tilde{x} - LC\tilde{x} = (A - LC)\tilde{x} \end{aligned}$$

- So the closed-loop estimation error dynamics are now

$$\dot{\tilde{x}} = (A - LC)\tilde{x} \quad \text{with solution} \quad \tilde{x}(t) = e^{(A-LC)t}\,\tilde{x}(0)$$

- **Bottom line:** Can select the gain L to attempt to improve the convergence of the estimation error (and/or speed it up).
  - But now must worry about observability of the system model.

- Note the similarity:

  – **Regulator Problem:** pick K for $A - BK$

  ◇ Choose $K \in \mathcal{R}^{1\times n}$ (SISO) such that the closed-loop poles

  $$\det(sI - A + BK) = \Phi_c(s)$$

  are in the desired locations.

  – **Estimator Problem:** pick L for $A - LC$

  ◇ Choose $L \in \mathcal{R}^{n\times 1}$ (SISO) such that the closed-loop poles

  $$\det(sI - A + LC) = \Phi_o(s)$$

  are in the desired locations.

- These problems are obviously very similar – in fact they are called **dual problems**.

Estimation Gain Selection

- For regulation, were concerned with controllability of (A, B)

  For a controllable system we can place the eigenvalues of $A - BK$ arbitrarily.

- For estimation, were concerned with observability of pair (A, C).

  For a observable system we can place the eigenvalues of $A - LC$ arbitrarily.

- Test using the observability matrix:

$$\texttt{rank}\ \mathcal{M}_o \triangleq \texttt{rank} \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix} = n$$

- The procedure for selecting L is very similar to that used for the regulator design process.

- Write the system model in **observer canonical** form

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -a_1 & 1 & 0 \\ -a_2 & 0 & 1 \\ -a_3 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

- Now very simple to form

$$
\begin{aligned}
A - LC &= \begin{bmatrix} -a_1 & 1 & 0 \\ -a_2 & 0 & 1 \\ -a_3 & 0 & 0 \end{bmatrix} - \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} -a_1 - l_1 & 1 & 0 \\ -a_2 - l_2 & 0 & 1 \\ -a_3 - l_3 & 0 & 0 \end{bmatrix}
\end{aligned}
$$

  – The closed-loop poles of the estimator are at the roots of

$$
\det(sI - A + LC) = s^3 + (a_1 + l_1)s^2 + (a_2 + l_2)s + (a_3 + l_3) = 0
$$

- So we have the freedom to place the closed-loop poles as desired.
  – Task greatly simplified by the selection of the state-space model used for the design/analysis.

- Another approach:
  - Note that the poles of $(A - LC)$ and $(A - LC)^T$ are identical.
  - Also we have that $(A - LC)^T = A^T - C^T L^T$
  - So designing L^T for this transposed system looks like a standard regulator problem $(A - BK)$ where

$$\begin{array}{ccc} A & \Rightarrow & A^T \\ B & \Rightarrow & C^T \\ K & \Rightarrow & L^T \end{array}$$

  So we can use

$$K_e = \texttt{acker}(A^T, C^T, P) \ , \quad L \equiv K_e^T$$

- Note that the estimator equivalent of Ackermann's formula is that

$$L = \Phi_e(s)\mathcal{M}_o^{-1} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

Estimators Example

- Simple system

$$A = \begin{bmatrix} -1 & 1.5 \\ 1 & -2 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad x(0) = \begin{bmatrix} -0.5 \\ -1 \end{bmatrix}$$
$$C = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad D = 0$$

  – Assume that the initial conditions are not well known.

  – System stable, but $\lambda_{\max}(A) = -0.18$

  – Test observability:

$$\texttt{rank} \begin{bmatrix} C \\ CA \end{bmatrix} = \texttt{rank} \begin{bmatrix} 1 & 0 \\ -1 & 1.5 \end{bmatrix}$$

- Use open and closed-loop estimators

  – Since the initial conditions are not well known, use

$$\hat{x}(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

- Open-loop estimator:

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu \\ \hat{y} &= C\hat{x} \end{aligned}$$

- Closed-loop estimator:

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + L\tilde{y} = A\hat{x} + Bu + L(y - \hat{y}) \\ &= (A - LC)\hat{x} + Bu + Ly \\ \hat{y} &= C\hat{x} \end{aligned}$$

  – Which is a dynamic system with poles given by $\lambda_i(A - LC)$ and which takes the measured plant outputs as an input and generates an estimate of x.

- Typically simulate both systems together for simplicity

- Open-loop case:

$$
\begin{aligned}
\dot{x} &= Ax + Bu \\
y &= Cx \\
\dot{\hat{x}} &= A\hat{x} + Bu \\
\hat{y} &= C\hat{x}
\end{aligned}
$$

$$
\Rightarrow \begin{bmatrix} \dot{x} \\ \dot{\hat{x}} \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & A \end{bmatrix} \begin{bmatrix} x \\ \hat{x} \end{bmatrix} + \begin{bmatrix} B \\ B \end{bmatrix} u \ , \quad \begin{bmatrix} x(0) \\ \hat{x}(0) \end{bmatrix} = \begin{bmatrix} -0.5 \\ -1 \\ 0 \\ 0 \end{bmatrix}
$$

$$
\begin{bmatrix} y \\ \hat{y} \end{bmatrix} = \begin{bmatrix} C & 0 \\ 0 & C \end{bmatrix} \begin{bmatrix} x \\ \hat{x} \end{bmatrix}
$$

- Closed-loop case:

$$
\begin{aligned}
\dot{x} &= Ax + Bu \\
\dot{\hat{x}} &= (A - LC)\hat{x} + Bu + LCx
\end{aligned}
$$

$$
\Rightarrow \begin{bmatrix} \dot{x} \\ \dot{\hat{x}} \end{bmatrix} = \begin{bmatrix} A & 0 \\ LC & A - LC \end{bmatrix} \begin{bmatrix} x \\ \hat{x} \end{bmatrix} + \begin{bmatrix} B \\ B \end{bmatrix} u
$$

- Example uses a strong $u(t)$ to shake things up

Figure 1: Open-loop estimator. Estimation error converges to zero, but very slowly.

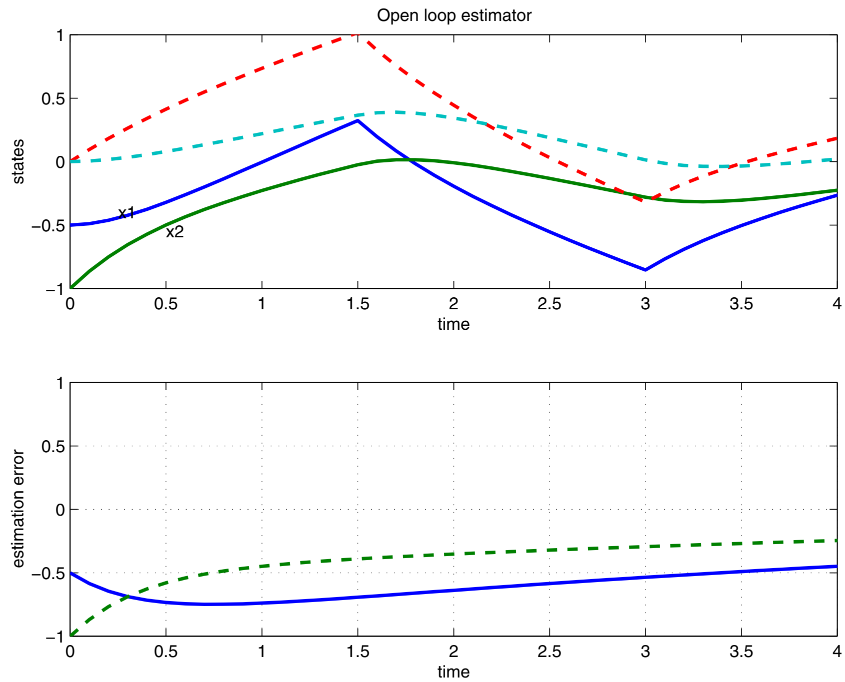
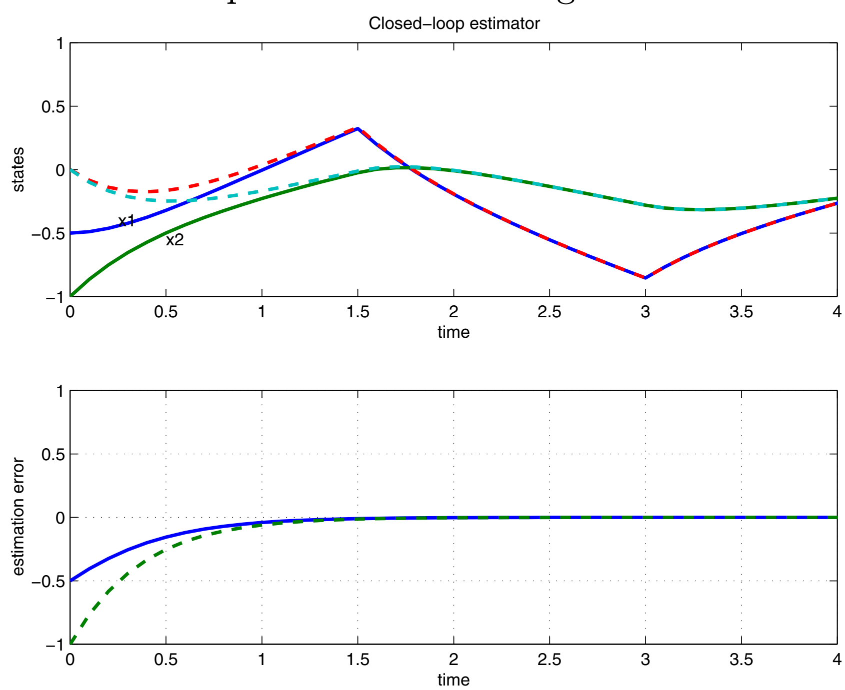


Figure 2: Closed-loop estimator. Convergence looks much better.



Where to put the Estimator Poles?

- Location heuristics for poles still apply – use Bessel, ITAE, ...
  - Main difference: probably want to make the estimator faster than you intend to make the regulator – should enhance the control, which is based on $\hat{x}(t)$.
  - ROT: Factor of 2–3 in the time constant $\zeta\omega_n$ associated with the regulator poles.

- **Note:** When designing a regulator, were concerned with "bandwidth" of the control getting too high ⇒ often results in control commands that *saturate* the actuators and/or change rapidly.

- Different concerns for the estimator:
  - Loop closed inside computer, so saturation not a problem.
  - However, the measurements y are often "noisy", and we need to be careful how we use them to develop our state estimates.

⇒ **High bandwidth estimators** tend to accentuate the effect of sensing noise in the estimate.

- State estimates tend to "track" the measurements, which are fluctuating randomly due to the noise.

⇒ **Low bandwidth estimators** have lower gains and tend to rely more heavily on the plant model

- Essentially an open-loop estimator – tends to ignore the measurements and just uses the plant model.

- Can also develop an **optimal estimator** for this type of system.
  - Which is apparently what Kalman did one evening in 1958 while taking the train from Princeton to Baltimore...
  - **Balances effect** of the various types of random noise in the system on the estimator:

$$\begin{aligned} \dot{x} &= Ax + Bu + B_w w \\ y &= Cx + v \end{aligned}$$

    where:

    ◇ w is called "process noise" – models the uncertainty in the system model.

    ◇ v is called "sensor noise" – models the uncertainty in the measurements.

- A *symmetric root locus* exists for the optimal estimator.
  - Define $G_{yw}(s) = C(sI - A)^{-1}B_w \equiv N(s)/D(s)$
  - SRL for the closed-loop poles $\lambda_i(A - LC)$ of the estimator which are the LHP roots of:

$$D(s)D(-s) \pm \frac{R_w}{R_v} N(s)N(-s) = 0$$

  where R_w and R_v are, in some sense, associated with the **sizes** of the process/sensor noise (spectral density).

  - Pick sign to ensure that there are no poles on the jω-axis.

- Relative size of the noises determine where the poles will be located.
  - Similar to role of control cost in LQR problem.

- As $R_w/R_v \to 0$, the n poles go to the
  1. LHP poles of the system
  2. Reflection of the RHP poles of the system about the jω-axis.

  - **The "relatively noisy" sensor case**

    ⇒ Closed-loop estimator essentially reverts back to the open-loop case (but must be stable).

  - Low bandwidth estimator.

- As $R_w/R_v \to \infty$, the n poles go to
  1. LHP zeros (and reflections of the RHP zeros) of $G_{yw}(s)$.
  2. ∞ along the Butterworth patterns – same as regulator case

  - **The "relatively clean" sensor case**

    ⇒ Closed-loop estimator poles go to very high bandwidth to take full advantage o the information in y.

  - High bandwidth estimator.

- If you know R_w and R_v, then use them in the SRL, but more often than not we **just use them as "tuning" parameters** to develop low $\to$ high bandwidth estimators.
  - Typically fix R_w and tune estimator bandwidth using R_v

Final Thoughts

- Note that the feedback gain L in the estimator only stabilizes the estimation error.
  - If the system is unstable, then the state estimates will also go to ∞, with zero error from the actual states.

- Estimation is an important concept of its own.
  - Not always just "part of the control system"
  - Critical issue for guidance and navigation system

- More complete discussion requires that we study stochastic processes and optimization theory.

- **Estimation is all about which do you trust more: your measurements or your model.**