EECE 360

Lecture 28 Addendum

State-space reference tracking and other control problems

Dr. Oishi

Electrical and Computer Engineering

University of British Columbia, BC

http://courses.ece.ubc.ca/360

eece360.@gmail.com

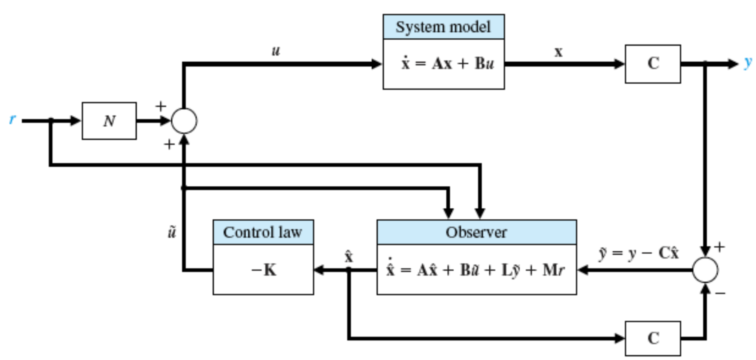
Chapters 11.6-11.8

Today's lecture

- LQR Control
  - Choosing closed-loop pole locations through minimization of a cost function
- Static reference tracking
  - Methods for tracking constant-value reference inputs
- Internal Model Control
  - State-space methods for robust tracking of dynamic reference inputs

Reference inputs

- Output-based command following



$(\tilde{u} = u - Nr)$

Reference inputs

- For command following, additional control components may be needed
- The control $u = -Kx+r$ can lead to steady-state errors when tracking a reference input r.
- **Solution**: Pre-multiply r by carefully chosen matrix N

Case 1: Full-state feedback

$$u = -Kx + Nr$$

Case 2: Full-state feedback with full-order observer

$$u = -K\hat{x} + Nr$$

$$\dot{\hat{x}} = A\hat{x} + B(u - Nr) + L(y - C\hat{x}) + Mr$$

Reference inputs: Case 1

Full-state feedback regulation of steady-state output y_{ss} to desired steady-state value r_{ss}

- Standard procedure:
  - To track a constant desired position x_{ss} (ss="steady-state") with control u_{ss}, note that

$$0 = Ax_{ss} + Bu_{ss}$$
$$y_{ss} = Cx_{ss} + Du_{ss}$$

  - and substituting $x_{ss}=N_x r_{ss}$, $u_{ss}=N_u r_{ss}$, we get

$$0 = AN_x r_{ss} + BN_u r_{ss}$$
$$r_{ss} = CN_x r_{ss} + DN_u r_{ss}$$

  - when $y_{ss}=r_{ss}$ in the steady-state

Reference inputs: Case 1

- In matrix form,

$$\begin{bmatrix}0\\1\end{bmatrix} r_{ss} = \begin{bmatrix}A & B\\ C & D\end{bmatrix}\begin{bmatrix}N_x\\ N_u\end{bmatrix} r_{ss}$$

$$\begin{bmatrix}N_x\\ N_u\end{bmatrix} = \begin{bmatrix}A & B\\ C & D\end{bmatrix}^{-1}\begin{bmatrix}0\\1\end{bmatrix}$$

- and therefore the full-state regulating input is

$$u = -Kx + Nr, \quad N = N_u + KN_x$$
$$= -Kx + (N_u + KN_x)r$$

to ensure **no** steady-state error in tracking *r*.

Reference inputs: Case 2

Command following with full-state feedback and full-order observer

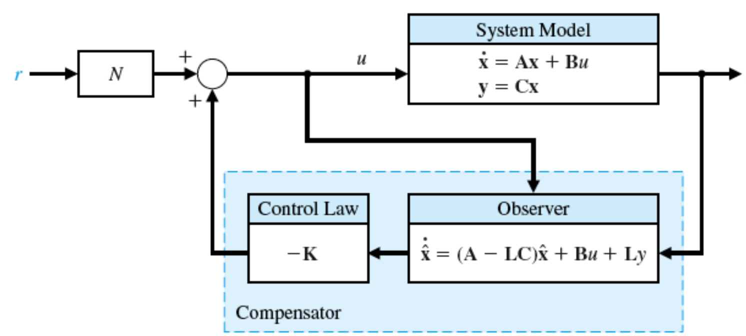
- General form:

$$u = -K\hat{x} + Nr$$
$$\dot{\hat{x}} = A\hat{x} + B(u - Nr) + L(y - C\hat{x}) + Mr$$

- There are a variety of ways to choose *M*, *N*
  - Autonomous estimator: Select *M* and *N* so that the state estimator error equation is independent of *r*
  - Tracking-error estimator: Select *M* and *N* so that only the **tracking** error ($e_r = r-y$) is used in the control
  - And others…

Reference inputs: Case 2

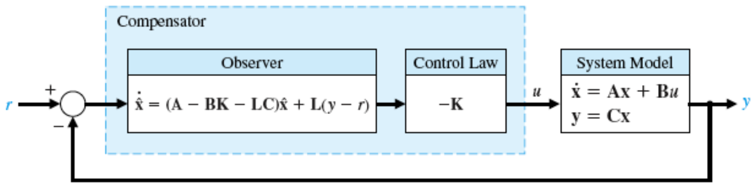
- Autonomous estimator: Select *M* and *N* so that the state estimator error equation is independent of *r*



Reference inputs: Case 2

- Tracking-error estimator: Select M and N so that only the **tracking** error (e_r = r-y) is used in the control

$$N = 0, \quad M = -L$$



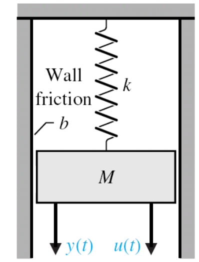
Example: Spring-Mass-Damper

- Case 1: No observer

$$\begin{bmatrix} N_x \\ N_u \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

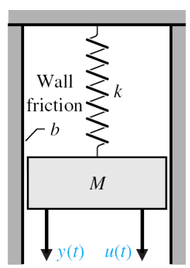
$$= \begin{bmatrix} 0 & 1 & 0 \\ -\frac{k}{M} & -\frac{b}{M} & 1 \\ 1 & 0 & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ \frac{b}{M} & 1 & \frac{k}{M} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ \frac{k}{M} \end{bmatrix}$$



Example: Spring-Mass-Damper

- Case 1: No observer



$$N_x = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, N_u = \frac{k}{M}$$

$$u = -Kx + Nr$$

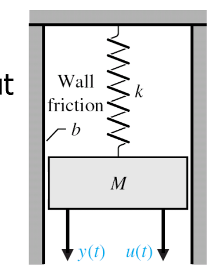
$$= -Kx + (N_u + KN_x)r$$

$$= -\left[\omega_n^2 - \frac{k}{M} \quad 2\zeta\omega_n - \frac{b}{M}\right]x + \left(\frac{k}{M} + \left[\omega_n^2 - \frac{k}{M} \quad 2\zeta\omega_n - \frac{b}{M}\right]\begin{bmatrix} 1 \\ 0 \end{bmatrix}\right)r$$

$$= -\left[\omega_n^2 - \frac{k}{M} \quad 2\zeta\omega_n - \frac{b}{M}\right]x + \boxed{\omega_n^2 r}$$

Example: Spring-Mass-Damper

- Case 2: With observer,
- Autonomous estimation (reference input appears in controller but not in estimator)



$$\boxed{N = \omega_n^2,} \quad u = -K\hat{x} + Nr$$

$$\boxed{M = BN = \begin{bmatrix} 0 \\ \omega_n^2 \end{bmatrix}} \text{ to make}$$

$$\dot{e} = (A - LC)e$$

- So the estimation dynamics do not depend on the reference command

Example: Spring-Mass-Damper

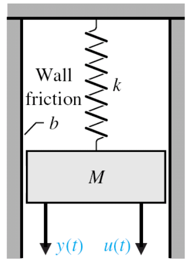
- Case 3: With observer,
- Tracking estimation (reference input *r* appears in estimator but not in controller)

$$N = 0 \quad u = -K\hat{x}$$

$$M = -L \text{ to make control depend on } \hat{x} \text{ only}$$

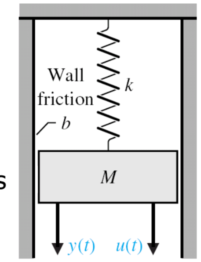
$$\dot{\hat{x}} = (A - BK - LC)\hat{x} + L(y - r)$$

- Comparable to tracking with standard classical control (based on deviations between actual and desired output)



Example: Spring-Mass-Damper

- Because the system is controllable and observable, the closed-loop poles of the error dynamics and the system dynamics can be placed arbitrarily.
- However, the further away the closed-loop poles are placed from the open-loop poles, the higher the control effort.
- Additionally, excessively high observer gains can lead to amplification of noise inherent to the output measurements.



Summary

- **Separation principle** allows independent design of the controller and observer
- Generally design controller first, then observer to be faster than the controller
- Reference inputs
  - To track a reference input, the regulating controller/observer must be modified
  - Construct *N* in $u = -Kx+Nr$ to remove steady-state error with full-state feedback
  - More complicated options in observer design depend on control problem at hand.

Internal Model Principle

- For a step input, zero steady-state tracking error can be achieved if the loop transfer function contains one integrator. For a ramp, two, etc…
- This idea can be formalized via the **internal model principle**

Purpose: design a compensator that provides asymptotic tracking of a reference input with *zero steady-state error*.

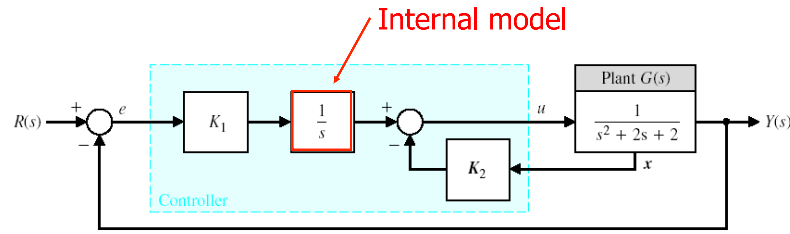
Internal Model Principle

- Consider a unit feedback system with loop transfer function $G(s)G_c(s)$
- The internal model principle states that

> If $G(s)G_c(s)$ contains $R(s)$, then $y(t)$ will track $r(t)$ asymptotically.

Internal Model Principle

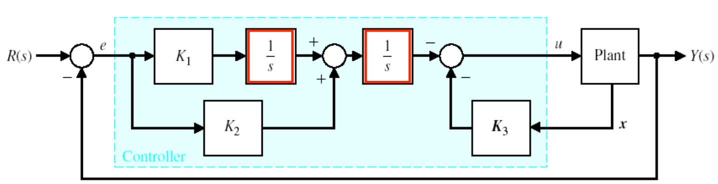
Example: Internal model design for a step



Design of a controller to enable the tracking of a step reference input with zero steady-state error.

Internal Model Principle

Example: Internal model design for a ramp



Can be extended to other types of reference inputs, e.g sine wave, etc…

Other possibilities

- Kalman filter
  - Observer gain which minimizes effect of noise from measurements
- Linear Quadratic Regulator (LQR)
  - Controller which optimizes a cost function (e.g. minimize fuel usage, error magnitude, control authority, etc.)
- Reduced state-control
  - When only part of x needs to be controlled
- Reduced-order observer
  - When only part of x needs to be estimated
- And **many** others…

Optimal Control

- An alternative to pole placement is optimal control.
- We want to find the controller that minimizes a performance index

Linear Quadratic Regulator (LQR) Control

- The performance index

$$J = \int_0^\infty (x^T Q x + r u^2) dt$$

- is minimized when

$$u(t) = -K^T x \text{ with } K = PBr^{-1}$$

where P satisfies the Riccati equation

$$A^T P + PA - PBB^T P r^{-1} + Q = 0$$

Control Sub-disciplines

- Discrete control
  - Control laws which account for sampling of a continuous signal from a continuous process
- Robust control
  - Design of shaping functions to make the system mathematically robust to uncertainty and noise
- Discrete event systems and control
  - Automata theory for supervisory control
- Optimal control
  - Design and computation of optimal controllers and observers to minimize a specific cost

Control Sub-disciplines

- Nonlinear control
  - Control and observation of general systems dx/dt=f(x,u)
- Hybrid control
  - Analysis and design of systems which have continuous processes as well as discrete mode-logic (e.g., hierarchical structure)
- Embedded systems and control
  - Computation and control of systems with integrated physical and computed components
- Biological/biomedical control
  - Analysis and design for biological modeling and biomedical control
- And many others…

Related Disciplines

- Information theory
- Robotics, haptics
- Verification (automata theory)
- Signal processing
- Chemical and industrial processes
- Artificial intelligence, fuzzy logic
- Complex systems, chaos theory
- Dynamics of mechanical systems
- Biological cell signaling and biological networks
- Scientific computation
- Mechatronics, sensors, and actuators