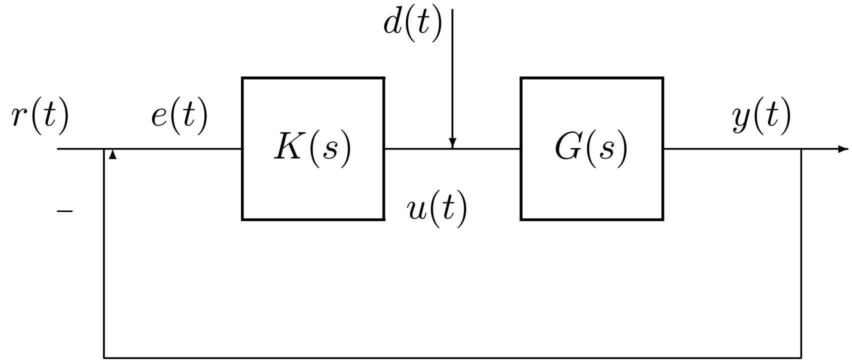
Lecture #1

16.31 Feedback Control

Introduction



- **Goal:** Design a controller $K(s)$ so that the *system* has some desired characteristics. Typical objectives:
  - **Stabilize** the system (Stabilization)
  - **Regulate** the system about some design point (Regulation)
  - **Follow** a given class of command signals (Tracking)
  - Reduce the response to **disturbances.** (Disturbance Rejection)
- Typically think of closed-loop control $\rightarrow$ so we would analyze the closed-loop dynamics.
  - Open-loop control also possible (called "feedforward") – more prone to modeling errors since inputs not changed as a result of measured error.
- Note that a typical *control system* includes the sensors, actuators, and the control law.
  - The sensors and actuators need not always be physical devices (*e.g.,* economic systems).
  - A good selection of the sensor and actuator can greatly simplify the control design process.
  - Course concentrates on the design of the control law given the rest of the system (although we will need to model the system).

Why Control?

OPERATION IRAQI FREEDOM -- An F-117 from the 8th Expeditionary Fighter Squadron out of Holloman Air Force Base, N.M., flies over the Persian Gulf on April 14, 2003. The 8th EFS has begun returning to Hollomann A.F.B. after having been deployed to the Middle East in support of Operation Iraqi Freedom. (U.S. Air Force photo by Staff Sgt. Derrick C. Goode). http://www.af.mil/photos.html.

- Easy question to answer for aerospace because many vehicles (spacecraft, aircraft, rockets) and aerospace processes (propulsion) need to be controlled just to function
  - Example: the F-117 does not even fly without computer control, and the X-29 is unstable

Feedback Control Approach

- Establish control objectives
  - Qualitative – don't use too much fuel
  - Quantitative – settling time of step response <3 sec
  - Typically requires that you understand the process (expected commands and disturbances) and the overall goals (bandwidths).
  - Often requires that you have a strong understanding of the physical dynamics of the system so that you do not "fight" them in appropriate (*i.e.,* inefficient) ways.
- Select sensors & actuators
  - What aspects of the system are to be sensed and controlled?
  - Consider sensor noise and linearity as key discriminators.
  - Cost, reliability, size, . . .
- Obtain model
  - Analytic (FEM) or from measured data (system ID)
  - Evaluation model $\rightarrow$ reduce size/complexity $\rightarrow$ Design model
  - Accuracy? Error model?
- Design controller
  - Select technique (SISO, MIMO), (classical, state-space)
  - Choose parameters (ROT, optimization)
- Analyze closed-loop performance. Meet objectives?
  - Analysis, simulation, experimentation, . . .
  - Yes $\Rightarrow$ done, No $\Rightarrow$ iterate . . .

Example: Blimp Control

- Control objective
  - Stabilization
  - Red blimp tracks the motion of the green blimp
- Sensors
  - GPS for positioning
  - Compass for heading
  - Gyros/GPS for roll attitude
- Actuators – electric motors (propellers) are very nonlinear.
- Dynamics
  - "rigid body" with strong apparent mass effect.
  - Roll modes.
- Modeling
  - Analytic models with parameter identification to determine "mass".
- Disturbances – wind

State-Space Approach

- Basic questions that we will address about the state-space approach:
  - What are state-space models?
  - Why should we use them?
  - How are they related to the transfer functions used in classical control design?
  - How do we develop a state-space model?
  - How do we design a controller using a state-space model?

- **Bottom line:**

  1. **What:** representation of the dynamics of an n^{th}-order system using n first-order differential equations:

     $$m\ddot{q} + c\dot{q} + kq = u \;\Rightarrow$$

     $$\begin{aligned} \begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -k/m & -c/m \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} + \begin{bmatrix} 0 \\ 1/m \end{bmatrix} u \\ \Rightarrow \dot{x} &= Ax + Bu \end{aligned}$$

  2. **Why:**
     - State variable form convenient way to work with complex dynamics. Matrix format easy to use on computers.
     - Transfer functions only deal with input/output behavior, but state-space form provides easy access to the "internal" features/response of the system.
     - Allows us to explore new analysis and synthesis tools.
     - Great for multiple-input multiple-output systems (MIMO), which are very hard to work with using transfer functions.

3. **How:** There are a variety of ways to develop these state-space models. We will explore this process in detail.
   – "Linear systems theory"

4. **Control design:** Split into 3 main parts
   – *Full-state feedback* –ficticious since requires more information than typically (ever?) available
   – *Observer/estimator design* –process of "estimating" the system state from the measurements that are available.
   – *Dynamic output feedback* –combines these two parts with provable guarantees on stability (and performance).
   – Fortunately there are **very** simple numerical tools available to perform each of these steps

   – Removes much of the "art" and/or "magic" required in classical control design $\rightarrow$ **design process more systematic.**

- **Word of caution:** – Linear systems theory involves extensive use of linear algebra.
  – Will not focus on the theorems/proofs in class – details will be handed out as necessary, but these are in the textbooks.
  – Will focus on using the linear algebra to understand the behavior of the system dynamics so that we can modify them using control. **"Linear algebra in action"**
  – Even so, this will require more algebra that most math courses that you have taken ....

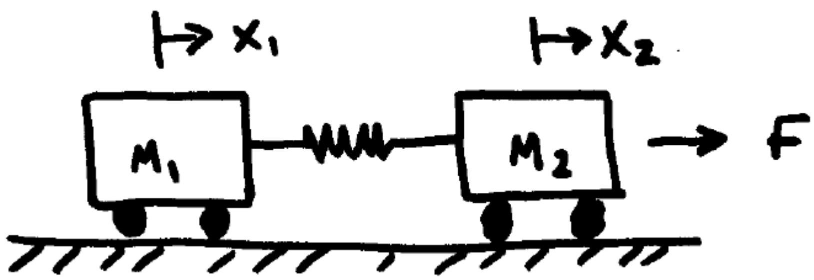
- My reasons for the review of classical design:
  - State-space techniques are just another to design a controller
  - But it is essential that you understand the basics of the control design process
  - Otherwise these are just a "bunch of numerical tools"
  - To truly understand the output of the state-space control design process, I think it is important that you be able to analyze it from a classical perspective.
    * Try to answer "why did it do that"?
    * Not always possible, but always a good goal.

- Feedback: muddy cards and office hours.
  - Help me to know whether my assumptions about your backgrounds is correct and whether there are any questions about the material.

- Matlab will be required extensively. If you have not used it before, then start practicing.

System Modeling

- Investigate the model of a simple system to explore the basics of system dynamics.
  - Provide insight on the connection between the system response and the pole locations.



- Consider the simple mechanical system (2MSS) – derive the system model
  1. Start with a free body diagram
  2. Develop the 2 equations of motion

$$\begin{aligned} m_1\ddot{x}_1 &= k(x_2 - x_1) \\ m_2\ddot{x}_2 &= k(x_1 - x_2) + F \end{aligned}$$

  3. How determine the relationships between x_1, x_2 and F?
     - Numerical integration - good for simulation, but not analysis
     - Use Laplace transform to get transfer functions
       * Fast/easy/lots of tables
       * Provides lots of information (poles and zeros)

- Laplace transform

$$\mathcal{L}\{f(t)\} \equiv \int_{0^-}^{\infty} f(t) e^{-st} dt$$

  - Key point: If $\mathcal{L}\{x(t)\} = X(s)$, then $\mathcal{L}\{\dot{x}(t)\} = sX(s)$ assuming that the initial conditions are zero.

- Apply to the model

$$\begin{aligned} \mathcal{L}\{m_1\ddot{x}_1 - k(x_2 - x_1)\} &= (m_1 s^2 + k)X_1(s) - kX_2(s) = 0 \\ \mathcal{L}\{m_2\ddot{x}_2 - k(x_1 - x_2) - F\} &= (m_2 s^2 + k)X_2(s) - kX_1(s) - F(s) = 0 \end{aligned}$$

$$\begin{bmatrix} m_1 s^2 + k & -k \\ -k & m_2 s^2 + k \end{bmatrix} \begin{bmatrix} X_1(s) \\ X_2(s) \end{bmatrix} = \begin{bmatrix} 0 \\ F(s) \end{bmatrix}$$

- Perform some algebra to get

$$\frac{X_2(s)}{F(s)} = \frac{m_1 s^2 + k}{m_1 m_2 s^2 (s^2 + k(1/m_1 + 1/m_2))} \equiv G_2(s)$$

- $G_2(s)$ is the **transfer function** between the input F and the system response x_2

- Given that $F \rightarrow G_2(s) \rightarrow x_2$. If $F(t)$ known, how find $x_2(t)$?
  1. Find $G_2(s)$
  2. Let $F(s) = \mathcal{L}\{F(t)\}$
  3. Set $X_s(s) = G_2(s) \cdot F(s)$
  4. Compute $x_2(t) = \mathcal{L}^{-1}\{X_2(s)\}$

- Step 4 involves an inverse Laplace transform, which requires an ugly contour integral that is hardly ever used.

$$x_2(t) = \frac{1}{2\pi i}\int_{\sigma_c - i\infty}^{\sigma_c + i\infty} X_2(s)e^{st}ds$$

  where σ_c is a value selected to be to the right of all singularities of $F(s)$ in the s-plane.
  - Partial fraction expansion and inversion using tables is much easier for problems that we will be dealing with.

- Example with $F(t) = 1(t) \Rightarrow F(s) = 1/s$

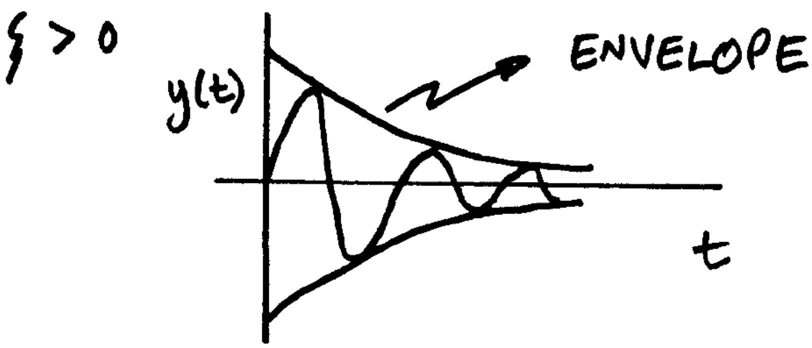
$$\begin{aligned} X_2(s) &= \frac{m_1 s^2 + k}{m_1 m_2 s^3(s^2 + k(1/m_1 + 1/m_2))} \\ &= \frac{c_1}{s} + \frac{c_2}{s^2} + \frac{c_3}{s^3} + \frac{c_4 s + c_5}{s^2 + k(1/m_1 + 1/m_2)} \end{aligned}$$

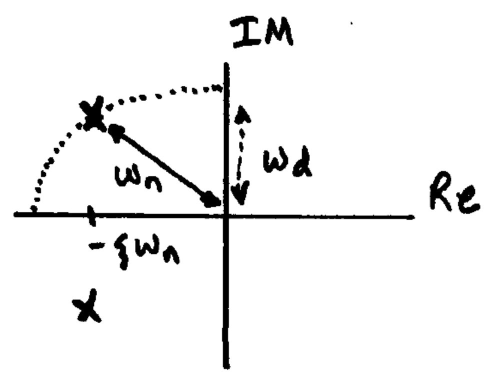
  - Solve for the coefficients c_i
  - Then inverse transform each term to get $x_2(t)$.

- Note that there are 2 special entries in the tables

  1. $\frac{1}{(s+a)} \Leftrightarrow e^{-at}$ which corresponds to a pole at $s+a=0$, or $s=-a$

  2. $\frac{\omega_n^2}{(s^2+2\zeta\omega_n s+\omega_n^2)} \Leftrightarrow e^{-\zeta\omega_n t}\sin(\omega_n\sqrt{1-\zeta^2}\;t)$





  – Corresponds to a damped sinusoidal response
  – ζ is the damping ratio
  – ω_n is the natural frequency
  – $\omega_d = \omega_n\sqrt{1-\zeta^2}$ is the damped frequency.

- These results point out that there is a very strong connection between the pole locations and the time response of the system
  – But there are other factors that come into play, as we shall see.

- For a second order system, we can be more explicit and relate the main features of the step response (time) and the pole locations (frequency domain).

$$G(s) = \frac{\omega_n^2}{(s^2 + 2\zeta\omega_n s + \omega_n^2)}$$

with $u(t)$ a step, so that $u(s) = 1/s$

- Then $y(s) = G(s)u(s) = \frac{\omega_n^2}{s(s^2+2\zeta\omega_n s+\omega_n^2)}$ which gives ($\sigma = \zeta\omega_n$)

$$y(t) = 1 - e^{-\sigma t}\left(\cos(\omega_d t) + \frac{\sigma}{\omega_d}\sin(\omega_d t)\right)$$

- Several key time domain features:
  - Rise time t_r (how long to get close to the final value?)
  - Settling time t_s (how long for the transients to decay?)
  - Peak overhsoot M_p, t_p (how far beyond the final value does the system respond, and when?)

- Can analyze the system response to determine that:
  1. $t_r \approx 2.2/w_h$

$$w_h = w_n\left(1 - 2\zeta^2 + \sqrt{2 - 4\zeta^2 + 4\zeta^4}\right)^{1/2}$$

     or can use $t_r \approx 1.8/w_n$
  2. $t_s(1\%) = 4.6/(\zeta\omega_n)$
  3. $M_p = e^{\frac{-\pi\zeta}{\sqrt{1-\zeta^2}}}$ and $t_p = \pi/\omega_d$

- Formulas relate time response to pole locations. Can easily evalute if the closed-loop system will respond as desired.
  - Use to determine acceptable locations for closed-loop poles.

- Examples:
  - Max rise time - min ω_n
  - Max settling time – min $\sigma = \zeta\omega_n$
  - Max overshoot – min ζ
- Usually assume that the response of more complex systems (i.e. ones that have more than 2 poles) is **dominated** by the lowest frequency pole pair.
  - Then the response is **approximately second order**, but we must check this
- These give us a good idea of where we would like the closed-loop poles to be so that we can meet the design goals.
  - Feedback control is all about changing the location of the system poles from the open-loop locations to the closed-loop ones.
  - This course is about a new way to do these control designs

Please refer to the "Design Aids" section of:

Franklin, Gene F., Powell, J. David and Abbas Emami-Naeini. 1994. *Feedback Control of Dynamic Systems – 3rd Ed.* Addison-Wesley