Plan of the Lecture

- Review: control, feedback, etc.
- Today’s topic: state-space models of systems; linearization

**Goal:** a general framework that encompasses all examples of interest. Once we have mastered this framework, we can proceed to *analysis* and then to *design*.

**Reading:** FPE, Sections 1.1, 1.2, 2.1–2.4, 7.2, 9.2.1. Chapter 2 has lots of cool examples of system models!!
Notation Reminder

We will be looking at *dynamic systems* whose evolution *in time* is described by *differential equations* with *external inputs*.

We will not write the time variable $t$ explicitly, so we use

- $x$ instead of $x(t)$
- $\dot{x}$ instead of $x'(t)$ or $\frac{dx}{dt}$
- $\ddot{x}$ instead of $x''(t)$ or $\frac{d^2x}{dt^2}$

etc.
Example 1: Mass-Spring System

Newton’s second law (translational motion):

\[ F_{\text{total}} = ma = \text{spring force} + \text{friction} + \text{external force} \]

- **spring force**: \( -kx \) (Hooke’s law)
- **friction force**: \( -\rho \dot{x} \) (Stokes’ law — linear drag, only an approximation!!)

\[ m\ddot{x} = -kx - \rho \dot{x} + u \]

Move \( x, \dot{x}, \ddot{x} \) to the LHS, \( u \) to the RHS:

\[ \ddot{x} + \frac{\rho}{m} \dot{x} + \frac{k}{m} x = \frac{u}{m} \]

2nd-order linear ODE
Example 1: Mass-Spring System

\[ \ddot{x} + \frac{\rho}{m} \dot{x} + \frac{k}{m} x = \frac{u}{m} \]

2nd-order linear ODE

Canonical form: convert to a system of 1st-order ODEs

\[ \dot{x} = v \quad \text{(definition of velocity)} \]
\[ \dot{v} = -\frac{\rho}{m} v - \frac{k}{m} x + \frac{1}{m} u \]
Example 1: Mass-Spring System

State-space model: express in matrix form

\[
\begin{pmatrix}
\dot{x} \\
\dot{\nu}
\end{pmatrix} = \begin{pmatrix}
0 & 1 \\
\frac{-k}{m} & \frac{-\rho}{m}
\end{pmatrix} \begin{pmatrix}
x \\
\nu
\end{pmatrix} + \begin{pmatrix}
0 \\
\frac{1}{m}
\end{pmatrix} u
\]

Important: start reviewing your linear algebra now!!

- matrix-vector multiplication; eigenvalues and eigenvectors; etc.
General $n$-Dimensional State-Space Model

State $x = \begin{pmatrix} x_1 \\ \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{R}^n$  
Input $u = \begin{pmatrix} u_1 \\ \\ \vdots \\ u_m \end{pmatrix} \in \mathbb{R}^m$

\[
\begin{pmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{pmatrix} = \begin{pmatrix} A \\ \vdots \\ A_{n \times n \text{ matrix}} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} + \begin{pmatrix} B \\ \vdots \\ B_{n \times m \text{ matrix}} \end{pmatrix} \begin{pmatrix} u_1 \\ \vdots \\ u_m \end{pmatrix}
\]

\[
\dot{x} = Ax + Bu
\]
Partial Measurements

state $x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{R}^n$  
input $u = \begin{pmatrix} u_1 \\ \vdots \\ u_m \end{pmatrix} \in \mathbb{R}^m$

output $y = \begin{pmatrix} y_1 \\ \vdots \\ y_p \end{pmatrix} \in \mathbb{R}^p$  
$y = Cx$  
$C$ is a $p \times n$ matrix

$\dot{x} = Ax + Bu$

$y = Cx$

Example: if we only care about (or can only measure) $x_1$, then

$y = x_1 = \begin{pmatrix} 1 & 0 & \ldots & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$
State-space models are useful and convenient for writing down system models for different types of systems, in a unified manner.

When working with state-space models, what are *states* and what are *inputs*?

— match against $\dot{x} = Ax + Bu$
Example 2: RL Circuit

\[-V_S + V_R + V_L = 0\]
\[V_R = RI\]
\[V_L = LI\]

\[-V_S + RI + LI = 0\]

\[\dot{I} = \frac{-R}{L}I + \frac{1}{L}V_S\]  \hspace{1cm} (1st-order system)

\(I\) – state, \(V_S\) – input

Q: How should we change the circuit in order to implement a 2nd-order system?  
A: Add a capacitor.
Example 3: Pendulum

Newton's 2nd law (rotational motion):

\[ T = J \cdot \alpha \]

= pendulum torque + external torque

pendulum torque = \(-mg \sin \theta \cdot \ell\)

moment of inertia \( J = ml^2 \)

\[-mg\ell \sin \theta + T_e = ml^2 \ddot{\theta} \]

\[ \ddot{\theta} = -\frac{g}{\ell} \sin \theta + \frac{1}{ml^2} T_e \]

(nonlinear equation)
Example 3: Pendulum

$$\ddot{\theta} = -\frac{g}{l} \sin \theta + \frac{1}{ml^2} T_e$$  \hspace{1cm} \text{(nonlinear equation)}$$

For small $\theta$, use the approximation $\sin \theta \approx \theta$

State-space form: $\theta_1 = \theta$, $\theta_2 = \dot{\theta}$

$$\dot{\theta}_2 = -\frac{g}{l} \theta + \frac{1}{ml^2} T_e = -\frac{g}{l} \theta_1 + \frac{1}{ml^2} T_e$$

$$\begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\frac{g}{l} & 0 \end{pmatrix} \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{ml^2} \end{pmatrix} T_e$$
**Linearization**

Taylor series expansion:

\[ f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{1}{2}f''(x_0)(x - x_0)^2 + \ldots \]

\[ \approx f(x_0) + f'(x_0)(x - x_0) \quad \text{linear approximation around } x = x_0 \]

Control systems are generally *nonlinear*:

\[ \dot{x} = f(x, u) \quad \text{nonlinear state-space model} \]

\[
\begin{pmatrix}
  x_1 \\
  \vdots \\
  x_n
\end{pmatrix} =
\begin{pmatrix}
  u_1 \\
  \vdots \\
  u_m
\end{pmatrix},
\quad
f =
\begin{pmatrix}
  f_1 \\
  \vdots \\
  f_n
\end{pmatrix}
\]

Assume \( x = 0, u = 0 \) is an *equilibrium point*: \( f(0, 0) = 0 \)

This means that, when the system is at rest and no control is applied, the system does not move.
**Linearization**

Linear approx. around \((x, u) = (0, 0)\) to all components of \(f\):

\[
\dot{x}_1 = f_1(x, u), \quad \ldots, \quad \dot{x}_n = f_n(x, u)
\]

For each \(i = 1, \ldots, n\),

\[
f_i(x, u) = f_i(0, 0) + \left. \frac{\partial f_i}{\partial x_j} \right|_{x=0, u=0} (0, 0)x_j + \ldots + \left. \frac{\partial f_i}{\partial x_n} \right|_{x=0, u=0} (0, 0)x_n \\
+ \left. \frac{\partial f_i}{\partial u_k} \right|_{x=0, u=0} (0, 0)u_k + \ldots + \left. \frac{\partial f_i}{\partial u_m} \right|_{x=0, u=0} (0, 0)u_m
\]

Linearized state-space model:

\[
\dot{x} = Ax + Bu,
\]

where \(A_{ij} = \left. \frac{\partial f_i}{\partial x_j} \right|_{x=0, u=0}\), \(B_{ik} = \left. \frac{\partial f_i}{\partial u_k} \right|_{x=0, u=0}\)

**Important:** since we have ignored the higher-order terms, this linear system is only an approximation that holds only for small deviations from equilibrium.
Example 3: Pendulum, Revisited

Original nonlinear state-space model:

\[
\begin{align*}
\dot{\theta}_1 &= f_1(\theta_1, \theta_2, T_e) = \theta_2 \quad \text{— already linear} \\
\dot{\theta}_2 &= f_2(\theta_1, \theta_2, T_e) = -\frac{g}{\ell} \sin \theta_1 + \frac{1}{m\ell^2} T_e
\end{align*}
\]

Linear approx. of \( f_2 \) around equilibrium \((\theta_1, \theta_2, T_e) = (0, 0, 0)\):

\[
\begin{align*}
\frac{\partial f_2}{\partial \theta_1} &= -\frac{g}{\ell} \cos \theta_1 \\
\frac{\partial f_2}{\partial \theta_2} &= 0 \\
\frac{\partial f_2}{\partial T_e} &= \frac{1}{m\ell^2}
\end{align*}
\]

\[
\begin{align*}
\left. \frac{\partial f_2}{\partial \theta_1} \right|_0 &= -\frac{g}{\ell} \\
\left. \frac{\partial f_2}{\partial \theta_2} \right|_0 &= 0 \\
\left. \frac{\partial f_2}{\partial T_e} \right|_0 &= \frac{1}{m\ell^2}
\end{align*}
\]

Linearized state-space model of the pendulum:

\[
\begin{align*}
\dot{\theta}_1 &= \theta_2 \\
\dot{\theta}_2 &= -\frac{g}{\ell} \theta_1 + \frac{1}{m\ell^2} T_e 
\end{align*}
\]

valid for small deviations from equ.
General Linearization Procedure

- Start from nonlinear state-space model

\[ \dot{x} = f(x, u) \]

- Find equilibrium point \((x_0, u_0)\) such that \(f(x_0, u_0) = 0\)

*Note:* different systems may have different equilibria, not necessarily \((0, 0)\), so we need to shift variables:

\[ x = x - x_0, \quad u = u - u_0 \]

\[ f(x, u) = f(x + x_0, u + u_0) = f(x, u) \]

Note that the transformation is *invertible*:

\[ x = \underline{x} + x_0, \quad u = \underline{u} + u_0 \]
General Linearization Procedure

- Pass to shifted variables \( x = x - x_0, \ u = u - u_0 \)

\[
\dot{x} = \dot{x} \quad (x_0 \text{ does not depend on } t)
\]

\[
= f(x, u)
\]

\[
= f(x, u)
\]

— equivalent to original system

- The transformed system is in equilibrium at \((0, 0)\):

\[
f(0, 0) = f(x_0, u_0) = 0
\]

- Now linearize:

\[
\dot{x} = Ax + Bu,
\]

where \( A_{ij} = \frac{\partial f_i}{\partial x_j} \bigg|_{x=x_0, \ u=u_0} \), \( B_{ik} = \frac{\partial f_i}{\partial u_k} \bigg|_{x=x_0, \ u=u_0} \)
General Linearization Procedure

- Why do we require that $f(x_0, u_0) = 0$ in equilibrium?
- This requires some thought. Indeed, we may talk about a linear approximation of any smooth function $f$ at any point $x_0$:

  $$ f(x) \approx f(x_0) + f'(x_0)(x - x_0) \quad \text{— } f(x_0) \text{ does not have to be 0} $$

- The key is that we want to approximate a given nonlinear system $\dot{x} = f(x, u)$ by a linear system $\dot{x} = Ax + Bu$ (may have to shift coordinates:
  
  $x \mapsto x - x_0, \ u \mapsto u - u_0$

Any linear system must have an equilibrium point at $(x, u) = (0, 0)$:

$$ f(x, u) = Ax + Bu \quad f(0, 0) = A0 + B0 = 0. $$