Plan of the Lecture

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

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*.

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

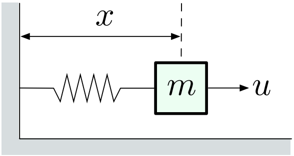
We will be looking at *dynamic systems* whose evolution *in time* is described by *differential equations* with *external inputs*.

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

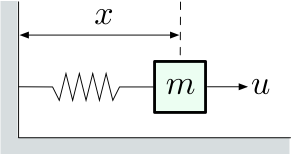
$$
\begin{array}{lll}
x & \text{instead of} & x(t) \\
\dot{x} & \text{instead of} & x'(t) \text{ or } \dfrac{\mathrm{d}x}{\mathrm{d}t} \\
\ddot{x} & \text{instead of} & x''(t) \text{ or } \dfrac{\mathrm{d}^2x}{\mathrm{d}t^2}
\end{array}
$$

etc.

Example 1: Mass-Spring System



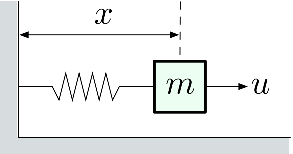
Example 1: Mass-Spring System



Newton's second law (translational motion):

$$\underbrace{F}_{\text{total force}} = ma$$

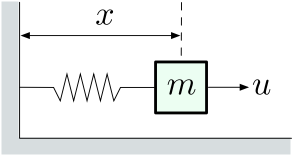
Example 1: Mass-Spring System



Newton's second law (translational motion):

$$\underbrace{F}_{\text{total force}} = ma = \text{spring force} + \text{friction} + \text{external force}$$

Example 1: Mass-Spring System



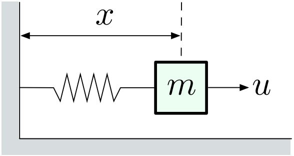
Newton's second law (translational motion):

$$\underbrace{F}_{\text{total force}} = ma = \text{spring force} + \text{friction} + \text{external force}$$

$$\text{spring force} = -kx \quad (\text{Hooke's law})$$

$$\text{friction force} = -\rho\dot{x} \quad (\text{Stokes' law — linear drag, only an approximation!!})$$

Example 1: Mass-Spring System



Newton's second law (translational motion):

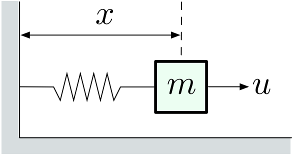
$$\underbrace{F}_{\text{total force}} = ma \;=\; \text{spring force} + \text{friction} + \text{external force}$$

$$\text{spring force} \;= -kx \quad (\text{Hooke's law})$$

$$\text{friction force} \;= -\rho\dot{x} \quad (\text{Stokes' law — linear drag, only an approximation!!})$$

$$F = -kx - \rho\dot{x} + u$$

Example 1: Mass-Spring System



Newton's second law (translational motion):

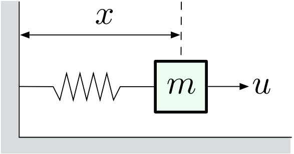
$$\underbrace{F}_{\text{total force}} = ma \; = \; \text{spring force} + \text{friction} + \text{external force}$$

$$\text{spring force} \; = -kx \quad \text{(Hooke's law)}$$

$$\text{friction force} \; = -\rho\dot{x} \quad \text{(Stokes' law — linear drag, only an approximation!!)}$$

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

Example 1: Mass-Spring System



Newton's second law (translational motion):

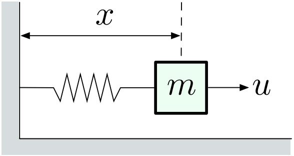
$$\underbrace{F}_{\text{total force}} = ma = \text{spring force} + \text{friction} + \text{external force}$$

$$\begin{aligned} \text{spring force} &= -kx \quad (\text{Hooke's law}) \\ \text{friction force} &= -\rho\dot{x} \quad (\text{Stokes' law — linear drag, only an approximation!!}) \\ m\ddot{x} &= -kx - \rho\dot{x} + u \end{aligned}$$

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

$$m\ddot{x} + \rho\dot{x} + kx = u$$

Example 1: Mass-Spring System



Newton's second law (translational motion):

$$\underbrace{F}_{\text{total force}} = ma = \text{spring force} + \text{friction} + \text{external force}$$

$$\text{spring force} = -kx \quad (\text{Hooke's law})$$

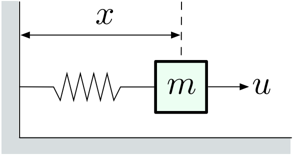
$$\text{friction force} = -\rho\dot{x} \quad (\text{Stokes' law — linear drag, only an approximation!!})$$

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

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

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

Example 1: Mass-Spring System



Newton's second law (translational motion):

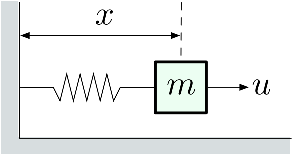
$$\underbrace{F}_{\text{total force}} = ma = \text{spring force} + \text{friction} + \text{external force}$$

$$\begin{aligned} \text{spring force} &= -kx \quad \text{(Hooke's law)} \\ \text{friction force} &= -\rho\dot{x} \quad \text{(Stokes' law — linear drag, only an approximation!!)} \\ m\ddot{x} &= -kx - \rho\dot{x} + u \end{aligned}$$

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

$$\ddot{x} + \frac{\rho}{m}\dot{x} + \frac{k}{m}x = \frac{u}{m} \qquad \text{2nd-order linear ODE}$$

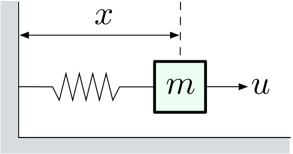
Example 1: Mass-Spring System



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

2nd-order linear ODE

Example 1: Mass-Spring System

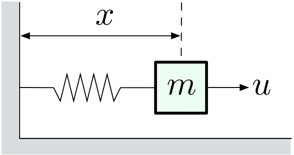


$$\ddot{x} + \frac{\rho}{m}\dot{x} + \frac{k}{m}x = \frac{u}{m}$$

2nd-order linear ODE

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

Example 1: Mass-Spring System



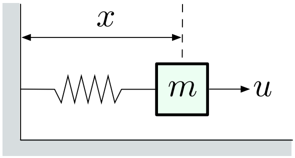
$$\ddot{x} + \frac{\rho}{m}\dot{x} + \frac{k}{m}x = \frac{u}{m}$$ 2nd-order linear ODE

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

$$\dot{x} = v \qquad \text{(definition of velocity)}$$

$$\dot{v} = -\frac{\rho}{m}v - \frac{k}{m}x + \frac{1}{m}u$$

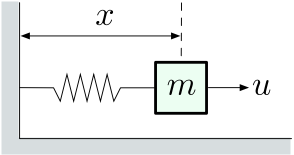
Example 1: Mass-Spring System



State-space model: express in *matrix form*

$$\begin{pmatrix} \dot{x} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{\rho}{m} \end{pmatrix} \begin{pmatrix} x \\ v \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{m} \end{pmatrix} u$$

Example 1: Mass-Spring System



State-space model: express in *matrix form*

$$\begin{pmatrix} \dot{x} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{\rho}{m} \end{pmatrix} \begin{pmatrix} x \\ v \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{m} \end{pmatrix} u$$

Important: start reviewing your linear algebra ***now***!!

- matrix-vector multiplication; eigenvalues and eigenvectors; etc.

General n-Dimensional State-Space Model

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

General n-Dimensional State-Space Model

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

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

General n-Dimensional State-Space Model

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

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

$$\dot{x} = Ax + Bu$$

Partial Measurements

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

Partial Measurements

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

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

Partial Measurements

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

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

$$\dot{x} = Ax + Bu$$
$$y = Cx$$

Partial Measurements

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

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

$$\dot{x} = Ax + Bu$$
$$y = Cx$$

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

$$y = x_1 = \begin{pmatrix} 1 & 0 & \dots & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

State-Space Models: Bottom Line

$$\dot{x} = Ax + Bu$$

$$y = Cx$$

State-Space Models: Bottom Line

$$\dot{x} = Ax + Bu$$
$$y = Cx$$

State-space models are useful and convenient for writing down system models for different types of systems, in a unified manner.

State-Space Models: Bottom Line

$$\dot{x} = Ax + Bu$$
$$y = Cx$$

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*?

State-Space Models: Bottom Line

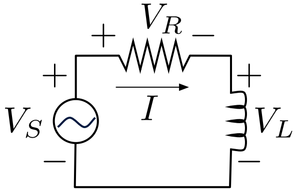
$$\dot{x} = Ax + Bu$$
$$y = Cx$$

State-space models are useful and convenient for writing down system models for different types of systems, in a unified manner.

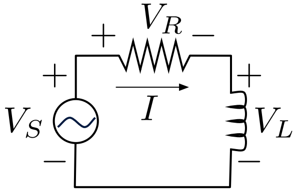
When working with state-space models, what are *states* and what are *inputs*?

— match against $\dot{x} = Ax + Bu$

Example 2: RL Circuit



Example 2: RL Circuit



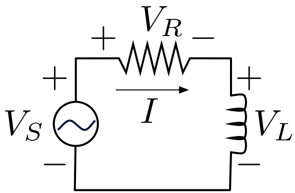
$$-V_S + V_R + V_L = 0 \qquad \text{Kirchhoff's voltage law}$$

$$V_R = RI \qquad \text{Ohm's law}$$

$$V_L = L\dot{I} \qquad \text{Faraday's law}$$

$$-V_S + RI + L\dot{I} = 0$$

Example 2: RL Circuit



$$-V_S + V_R + V_L = 0 \qquad \text{Kirchhoff's voltage law}$$

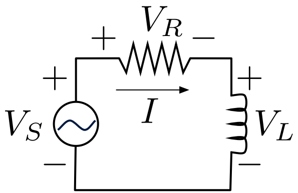
$$V_R = RI \qquad \text{Ohm's law}$$

$$V_L = L\dot{I} \qquad \text{Faraday's law}$$

$$-V_S + RI + L\dot{I} = 0$$

$$\dot{I} = -\frac{R}{L}I + \frac{1}{L}V_S \qquad \text{(1st-order system)}$$

Example 2: RL Circuit



$$-V_S + V_R + V_L = 0 \qquad \text{Kirchhoff's voltage law}$$

$$V_R = RI \qquad \text{Ohm's law}$$

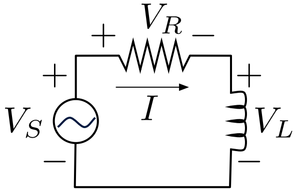
$$V_L = L\dot{I} \qquad \text{Faraday's law}$$

$$-V_S + RI + L\dot{I} = 0$$

$$\dot{I} = -\frac{R}{L}I + \frac{1}{L}V_S \qquad \text{(1st-order system)}$$

I – state, V_S – input

Example 2: RL Circuit



$$-V_S + V_R + V_L = 0 \qquad \text{Kirchhoff's voltage law}$$

$$V_R = RI \qquad \text{Ohm's law}$$

$$V_L = L\dot{I} \qquad \text{Faraday's law}$$

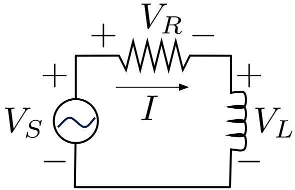
$$-V_S + RI + L\dot{I} = 0$$

$$\dot{I} = -\frac{R}{L}I + \frac{1}{L}V_S \qquad \text{(1st-order system)}$$

I – state, V_S – input

Q: How should we change the circuit in order to implement a *2nd-order system*?

Example 2: RL Circuit



$$-V_S + V_R + V_L = 0 \qquad \text{Kirchhoff's voltage law}$$

$$V_R = RI \qquad \text{Ohm's law}$$

$$V_L = L\dot{I} \qquad \text{Faraday's law}$$

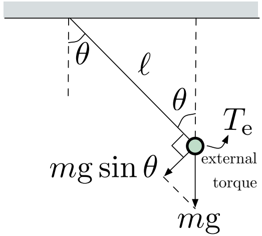
$$-V_S + RI + L\dot{I} = 0$$

$$\dot{I} = -\frac{R}{L}I + \frac{1}{L}V_S \qquad \text{(1st-order system)}$$

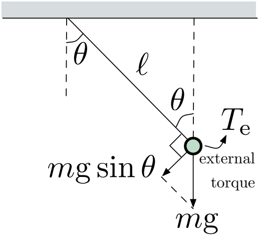
I – state, V_S – input

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

Example 3: Pendulum

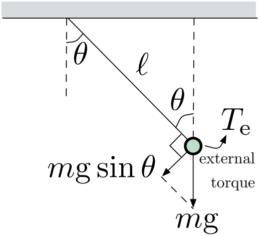


Example 3: Pendulum



Newton's 2nd law (rotational motion):

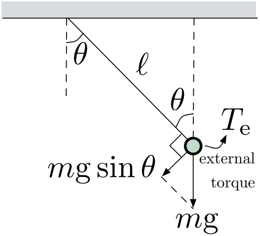
Example 3: Pendulum



Newton's 2nd law (rotational motion):

$$\underbrace{T}_{\text{total torque}} = \underbrace{J}_{\text{moment of inertia}} \underbrace{\alpha}_{\text{angular acceleration}}$$

Example 3: Pendulum

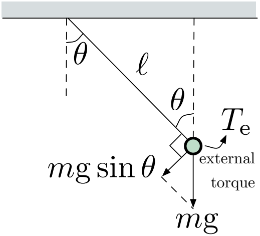


Newton's 2nd law (rotational motion):

$$\underbrace{T}_{\text{total torque}} = \underbrace{J}_{\text{moment of inertia}} \underbrace{\alpha}_{\text{angular acceleration}}$$

$$= \text{pendulum torque} + \text{external torque}$$

Example 3: Pendulum



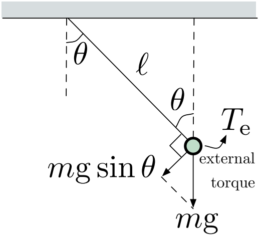
Newton's 2nd law (rotational motion):

$$\underbrace{T}_{\text{total torque}} = \underbrace{J}_{\text{moment of inertia}} \underbrace{\alpha}_{\text{angular acceleration}}$$

$$= \text{pendulum torque} + \text{external torque}$$

$$\text{pendulum torque} = \underbrace{-mg\sin\theta}_{\text{force}} \cdot \underbrace{\ell}_{\text{lever arm}}$$

Example 3: Pendulum



Newton's 2nd law (rotational motion):

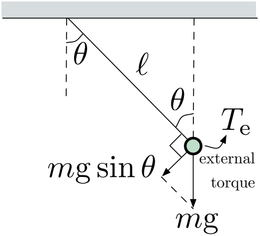
$$\underbrace{T}_{\text{total torque}} = \underbrace{J}_{\text{moment of inertia}} \underbrace{\alpha}_{\text{angular acceleration}}$$

$$= \text{pendulum torque} + \text{external torque}$$

$$\text{pendulum torque} = \underbrace{-mg\sin\theta}_{\text{force}} \cdot \underbrace{\ell}_{\text{lever arm}}$$

moment of inertia $J = m\ell^2$

Example 3: Pendulum



Newton's 2nd law (rotational motion):

$$\underbrace{T}_{\text{total torque}} = \underbrace{J}_{\text{moment of inertia}} \underbrace{\alpha}_{\text{angular acceleration}}$$

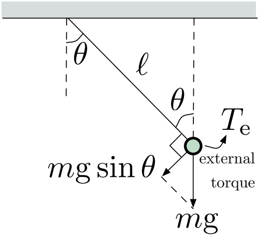
$$= \text{pendulum torque} + \text{external torque}$$

$$\text{pendulum torque} = \underbrace{-mg\sin\theta}_{\text{force}} \cdot \underbrace{\ell}_{\text{lever arm}}$$

$$\text{moment of inertia } J = m\ell^2$$

$$-mg\ell\sin\theta + T_e = m\ell^2\ddot{\theta}$$

Example 3: Pendulum



Newton's 2nd law (rotational motion):

$$\underbrace{T}_{\text{total torque}} = \underbrace{J}_{\text{moment of inertia}} \; \underbrace{\alpha}_{\text{angular acceleration}}$$

$$= \text{pendulum torque} + \text{external torque}$$

$$\text{pendulum torque} = \underbrace{-mg\sin\theta}_{\text{force}} \cdot \underbrace{\ell}_{\text{lever arm}}$$

$$\text{moment of inertia } J = m\ell^2$$

$$-mg\ell\sin\theta + T_\mathrm{e} = m\ell^2\ddot{\theta}$$

$$\ddot{\theta} = -\frac{g}{\ell}\sin\theta + \frac{1}{m\ell^2}T_\mathrm{e} \qquad \text{(nonlinear equation)}$$

Example 3: Pendulum

$$\ddot{\theta} = -\frac{\mathrm{g}}{\ell}\sin\theta + \frac{1}{m\ell^2}T_{\mathrm{e}} \qquad \text{(nonlinear equation)}$$

Example 3: Pendulum

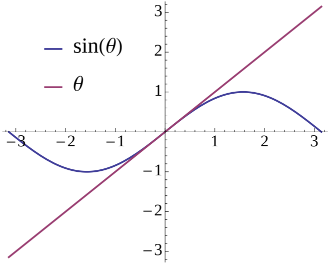
$$\ddot{\theta} = -\frac{\mathrm{g}}{\ell}\sin\theta + \frac{1}{m\ell^2}T_{\mathrm{e}} \qquad \text{(nonlinear equation)}$$

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

Example 3: Pendulum

$$\ddot{\theta} = -\frac{\mathrm{g}}{\ell}\sin\theta + \frac{1}{m\ell^2}T_{\mathrm{e}} \qquad \text{(nonlinear equation)}$$

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

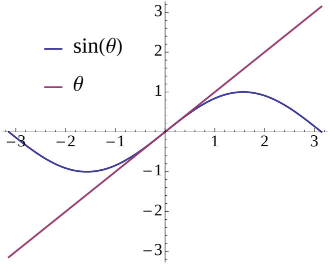


$$\ddot{\theta} = -\frac{\mathrm{g}}{\ell}\theta + \frac{1}{m\ell^2}T_{\mathrm{e}}$$

Example 3: Pendulum

$$\ddot{\theta} = -\frac{g}{\ell}\sin\theta + \frac{1}{m\ell^2}T_e \qquad \text{(nonlinear equation)}$$

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



$$\ddot{\theta} = -\frac{g}{\ell}\theta + \frac{1}{m\ell^2}T_e$$

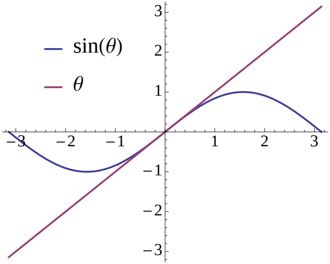
State-space form: $\theta_1 = \theta,\ \theta_2 = \dot{\theta}$

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

Example 3: Pendulum

$$\ddot{\theta} = -\frac{g}{\ell}\sin\theta + \frac{1}{m\ell^2}T_e \qquad \text{(nonlinear equation)}$$

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



$$\ddot{\theta} = -\frac{g}{\ell}\theta + \frac{1}{m\ell^2}T_e$$

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

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

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

Linearization

Taylor series expansion:

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{1}{2} f''(x_0)(x - x_0)^2 + \dots$$

$$\approx f(x_0) + f'(x_0)(x - x_0) \qquad \text{linear approximation around } x = x_0$$

Linearization

Taylor series expansion:

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{1}{2}f''(x_0)(x - x_0)^2 + \dots$$

$$\approx f(x_0) + f'(x_0)(x - x_0) \qquad \text{linear approximation around } x = x_0$$

Control systems are generally *nonlinear*:

$$\dot{x} = f(x, u) \qquad \text{nonlinear state-space model}$$

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \quad u = \begin{pmatrix} u_1 \\ \vdots \\ u_m \end{pmatrix} \quad f = \begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix}$$

Linearization

Taylor series expansion:

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{1}{2} f''(x_0)(x - x_0)^2 + \dots$$

$$\approx f(x_0) + f'(x_0)(x - x_0) \qquad \text{linear approximation around } x = x_0$$

Control systems are generally *nonlinear*:

$$\dot{x} = f(x, u) \qquad \text{nonlinear state-space model}$$

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \quad u = \begin{pmatrix} u_1 \\ \vdots \\ u_m \end{pmatrix} \quad f = \begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix}$$

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:

$$\dot{x}_1 = f_1(x, u), \qquad \ldots, \qquad \dot{x}_n = f_n(x, u)$$

Linearization

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

$$\dot{x}_1 = f_1(x,u), \qquad \ldots, \qquad \dot{x}_n = f_n(x,u)$$

For each $i = 1, \ldots, n$,

$$\begin{aligned} f_i(x,u) = \underbrace{f_i(0,0)}_{=0} &+ \frac{\partial f_i}{\partial x_1}(0,0)x_1 + \ldots + \frac{\partial f_i}{\partial x_n}(0,0)x_n \\ &+ \frac{\partial f_i}{\partial u_1}(0,0)u_1 + \ldots + \frac{\partial f_i}{\partial u_m}(0,0)u_m \end{aligned}$$

Linearization

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

$$\dot{x}_1 = f_1(x, u), \qquad \ldots, \qquad \dot{x}_n = f_n(x, u)$$

For each $i = 1, \ldots, n$,

$$\begin{aligned} f_i(x, u) = \underbrace{f_i(0, 0)}_{=0} &+ \frac{\partial f_i}{\partial x_1}(0, 0)x_1 + \ldots + \frac{\partial f_i}{\partial x_n}(0, 0)x_n \\ &+ \frac{\partial f_i}{\partial u_1}(0, 0)u_1 + \ldots + \frac{\partial f_i}{\partial u_m}(0, 0)u_m \end{aligned}$$

Linearized state-space model:

$$\dot{x} = Ax + Bu, \qquad \text{where } A_{ij} = \left. \frac{\partial f_i}{\partial x_j} \right|_{\substack{x=0 \\ u=0}}, \; B_{ik} = \left. \frac{\partial f_i}{\partial u_k} \right|_{\substack{x=0 \\ u=0}}$$

Linearization

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

$$\dot{x}_1 = f_1(x, u), \qquad \ldots, \qquad \dot{x}_n = f_n(x, u)$$

For each $i = 1, \ldots, n$,

$$\begin{aligned} f_i(x, u) = \underbrace{f_i(0,0)}_{=0} &+ \frac{\partial f_i}{\partial x_1}(0,0)x_1 + \ldots + \frac{\partial f_i}{\partial x_n}(0,0)x_n \\ &+ \frac{\partial f_i}{\partial u_1}(0,0)u_1 + \ldots + \frac{\partial f_i}{\partial u_m}(0,0)u_m \end{aligned}$$

Linearized state-space model:

$$\dot{x} = Ax + Bu, \qquad \text{where } A_{ij} = \left.\frac{\partial f_i}{\partial x_j}\right|_{\substack{x=0 \\ u=0}}, \; B_{ik} = \left.\frac{\partial f_i}{\partial u_k}\right|_{\substack{x=0 \\ u=0}}$$

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:

$$\dot{\theta}_1 = f_1(\theta_1, \theta_2, T_e) = \theta_2 \qquad \text{— already linear}$$

$$\dot{\theta}_2 = f_2(\theta_1, \theta_2, T_e) = -\frac{g}{\ell}\sin\theta_1 + \frac{1}{m\ell^2}T_e$$

Example 3: Pendulum, Revisited

Original nonlinear state-space model:

$$\dot{\theta}_1 = f_1(\theta_1, \theta_2, T_\mathrm{e}) = \theta_2 \qquad \text{— already linear}$$

$$\dot{\theta}_2 = f_2(\theta_1, \theta_2, T_\mathrm{e}) = -\frac{\mathrm{g}}{\ell}\sin\theta_1 + \frac{1}{m\ell^2}T_\mathrm{e}$$

Linear approx. of f_2 around equilibrium $(\theta_1, \theta_2, T_\mathrm{e}) = (0, 0, 0)$:

$$\frac{\partial f_2}{\partial \theta_1} = -\frac{\mathrm{g}}{\ell}\cos\theta_1 \qquad \frac{\partial f_2}{\partial \theta_2} = 0 \qquad \frac{\partial f_2}{\partial T_\mathrm{e}} = \frac{1}{m\ell^2}$$

$$\left.\frac{\partial f_2}{\partial \theta_1}\right|_0 = -\frac{\mathrm{g}}{\ell} \qquad \left.\frac{\partial f_2}{\partial \theta_2}\right|_0 = 0 \qquad \left.\frac{\partial f_2}{\partial T_\mathrm{e}}\right|_0 = \frac{1}{m\ell^2}$$

Example 3: Pendulum, Revisited

Original nonlinear state-space model:

$$\dot{\theta}_1 = f_1(\theta_1, \theta_2, T_e) = \theta_2 \qquad \text{— already linear}$$

$$\dot{\theta}_2 = f_2(\theta_1, \theta_2, T_e) = -\frac{g}{\ell}\sin\theta_1 + \frac{1}{m\ell^2}T_e$$

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

$$\frac{\partial f_2}{\partial \theta_1} = -\frac{g}{\ell}\cos\theta_1 \qquad \frac{\partial f_2}{\partial \theta_2} = 0 \qquad \frac{\partial f_2}{\partial T_e} = \frac{1}{m\ell^2}$$

$$\left.\frac{\partial f_2}{\partial \theta_1}\right|_0 = -\frac{g}{\ell} \qquad \left.\frac{\partial f_2}{\partial \theta_2}\right|_0 = 0 \qquad \left.\frac{\partial f_2}{\partial T_e}\right|_0 = \frac{1}{m\ell^2}$$

Linearized state-space model of the pendulum:

$$\dot{\theta}_1 = \theta_2$$

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

valid for *small* deviations from equ.

General Linearization Procedure

- Start from nonlinear state-space model

$$\dot{x} = f(x, u)$$

General Linearization Procedure

- Start from nonlinear state-space model

$$\dot{x} = f(x, u)$$

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

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:

$$\underline{x} = x - x_0 \qquad \underline{u} = u - u_0$$
$$\underline{f}(\underline{x}, \underline{u}) = f(\underline{x} + x_0, \underline{u} + u_0) = f(x, u)$$

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:

$$\underline{x} = x - x_0 \qquad \underline{u} = u - u_0$$
$$\underline{f}(\underline{x}, \underline{u}) = f(\underline{x} + x_0, \underline{u} + u_0) = f(x, u)$$

  Note that the transformation is *invertible*:

$$x = \underline{x} + x_0, \qquad u = \underline{u} + u_0$$

General Linearization Procedure

General Linearization Procedure

- Pass to shifted variables $\underline{x} = x - x_0$, $\underline{u} = u - u_0$

$$
\begin{aligned}
\dot{\underline{x}} &= \dot{x} && (x_0 \text{ does not depend on } t) \\
&= f(x, u) \\
&= \underline{f}(\underline{x}, \underline{u})
\end{aligned}
$$

— equivalent to original system

General Linearization Procedure

- Pass to shifted variables $\underline{x} = x - x_0$, $\underline{u} = u - u_0$

$$\begin{aligned} \dot{\underline{x}} &= \dot{x} && (x_0 \text{ does not depend on } t) \\ &= f(x, u) \\ &= \underline{f}(\underline{x}, \underline{u}) \end{aligned}$$

— equivalent to original system

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

$$\underline{f}(0, 0) = f(x_0, u_0) = 0$$

General Linearization Procedure

- Pass to shifted variables $\underline{x} = x - x_0$, $\underline{u} = u - u_0$

$$\begin{aligned} \dot{\underline{x}} &= \dot{x} \qquad\qquad (x_0 \text{ does not depend on } t) \\ &= f(x, u) \\ &= \underline{f}(\underline{x}, \underline{u}) \end{aligned}$$

  — equivalent to original system

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

$$\underline{f}(0, 0) = f(x_0, u_0) = 0$$

- Now linearize:

$$\dot{\underline{x}} = A\underline{x} + B\underline{u}, \qquad \text{where } A_{ij} = \left.\frac{\partial f_i}{\partial x_j}\right|_{\substack{x=x_0 \\ u=u_0}}, \ B_{ik} = \left.\frac{\partial f_i}{\partial u_k}\right|_{\substack{x=x_0 \\ u=u_0}}$$

General Linearization Procedure

- Why do we require that $f(x_0, u_0) = 0$ in equilibrium?

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:

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

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:

  $f(x) \approx f(x_0) + f'(x_0)(x - x_0)$ — $f(x_0)$ 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: $x \mapsto x - x_0$, $u \mapsto u - u_0$)

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:

  $f(x) \approx f(x_0) + f'(x_0)(x - x_0)$ — $f(x_0)$ 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 $(x, u) = (0, 0)$:

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