Last Time:

- Continuous-time Dynamics
- Manipulator Dynamics
- Linear Systems

Today:

- Equilibria
- Stability
- Discrete-Time Dynamics + Simulation

Equilibria:

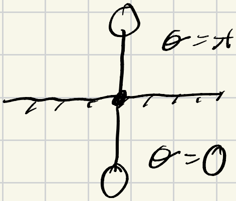
- A point where the system will "remain at rest"

$$\Rightarrow \quad \dot{x} = f(x,u) = 0$$

- Algebraically, roots of the dynamics

- Pendulum:

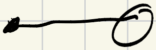
$$\dot{x} = \begin{bmatrix} \dot{\theta} \\ -\frac{g}{\ell}\sin(\theta) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Rightarrow \begin{matrix} \dot{\theta} = 0 \\ \theta = 0, \pi \end{matrix}$$



First Control Problem

- Can I move the equilibria?

$\theta = \pi/2$



$$\dot{x} = \begin{bmatrix} \dot{\theta} \\ -\frac{g}{\ell}\sin(\pi/2) + \frac{1}{m\ell^2}u \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\Rightarrow \frac{1}{m\ell^2}u = \frac{g}{\ell}\sin(\pi/2) \Rightarrow u = mg\ell$$

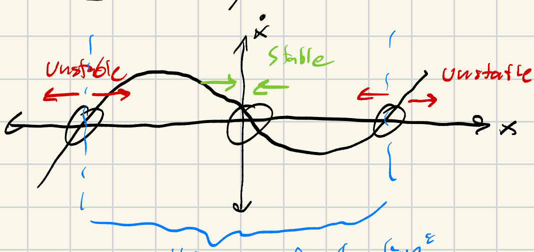
- In general, we get a root-finding problem in u:

$$f(x^*, u) = 0$$

Stability of Equilibria:

- When will we stay "near" an equilibrium point under perturbations?

- Look at a 1D system ($x \in \mathbb{R}$)



$\frac{\partial f}{\partial x} < 0 \Rightarrow$ stable $\qquad \frac{\partial f}{\partial x} > 0 \Rightarrow$ Unstable

- In higher dimensions:

$\frac{\partial f}{\partial x}$ is a Jacobian matrix

- Take an eigendecomposition $\Rightarrow$ decouple into n 1D systems

$$Re\left[\text{eigvals}\left(\frac{\partial f}{\partial x}\right)\right] < 0 \Rightarrow \text{stable}$$

otherwise $\Rightarrow$ unstable

- Pendulum:

$$f(x) = \begin{bmatrix} \dot{\theta} \\ -\frac{g}{\ell}\sin(\theta) \end{bmatrix} \Rightarrow \frac{\partial f}{\partial x} = \begin{bmatrix} 0 & 1 \\ -g/\ell \cos(\theta) & 0 \end{bmatrix}$$

$$\left.\frac{\partial f}{\partial x}\right|_{\theta=\pi} = \begin{bmatrix} 0 & 1 \\ g/\ell & 0 \end{bmatrix} \Rightarrow \text{eigvals}\left(\frac{\partial f}{\partial x}\right) = \pm\sqrt{\frac{g}{\ell}}$$

$\Rightarrow$ unstable

$$\left.\frac{\partial f}{\partial x}\right|_{\theta=0} = \begin{bmatrix} 0 & 1 \\ -g/\ell & 0 \end{bmatrix} \Rightarrow \text{eigvals}\left(\frac{\partial f}{\partial x}\right) = \pm i\sqrt{\frac{g}{\ell}}$$

$\Rightarrow$ undamped oscillation

- Add damping (e.g. $u = -K_d\dot{\theta}$) results in strictly negative real part.

Discrete-Time Dynamics:

- Motivation:

  * In general, we can't solve $\dot{x} = f(x)$ for $x(t)$

  * Computationally, need to represent $x(t)$ with discrete x_n

  * Discrete-time models can capture some effects that continuous ODEs can't

- "Explicit Form":

$$x_{n+1} = \underbrace{f_d}_{\text{"discrete"}}(x_n, u_n)$$

- Simplest discretization:

$$\underbrace{x_{n+1} = x_n + \underbrace{h}_{\text{"time step"}} f(x_n, u_n)}_{f_d(x_n, u_n)}$$

"Forward Euler Integration"

- Pendulum Sim:

$l = m = 1$, $h = 0.1, 0.01$

  * blows up!

Stability of Discrete-Time Systems:

- In discrete time, dynamics is an iterated map:

$$x_N = f_d(f_d(f_d(\cdots f_d(x_0))))$$

- Linearize + apply chain rule:

$$\frac{\partial x_N}{\partial x_0} = \left.\frac{\partial f_d}{\partial x}\right|_{x_0} \left.\frac{\partial f_d}{\partial x}\right|_{x_0} \cdots \left.\frac{\partial f_d}{\partial x}\right|_{x_0} = A_d^N$$

- Assume $x_0 = 0$ is an equilibrium

stable $\Rightarrow$ $\lim_{k\to\infty} A_d^k x_0 = 0 \quad \forall x_0$

$$\Rightarrow \lim_{n\to\infty} A_d^n = 0$$

$$\Rightarrow |\text{eigvals}(A_d)| < 1$$

(inside unit circle)

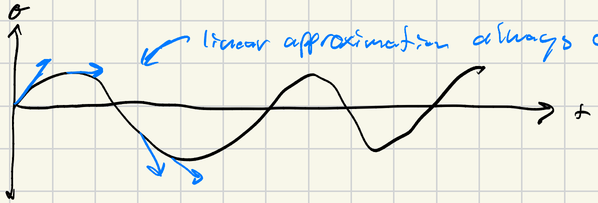
- Pendulum with Forward Euler:

$$x_{n+1} = \underbrace{x_n + h f(x_n)}_{f_d(x_n)}$$

$$A_d = \frac{\partial f_d}{\partial x} = I + hA = I + h\begin{bmatrix} 0 & 1 \\ -g/\ell \cos(\theta) & 0 \end{bmatrix}$$

$$\text{eigvals}(A_d|_{\theta=0}) \approx 1 \pm 0.313i$$

- Intuition:



- Take aways:

  - Be careful
  - Always sanity check e.g. energy, momentum behavior
  - Never use forward Euler!

* A better explicit integrator:

  - 4th-order Runge-Kutta Method
  - RK4 fits a cubic polynomial to $x(t)$ rather than a line

    $\Rightarrow$ much better accuracy!

  - Pseudo-code:

$$x_{n+1} = f_d(x_n)$$

$$K_1 = f(x_n)$$
$$K_2 = f(x_n + h/2\, K_1)$$
$$K_3 = f(x_n + h/2\, K_2)$$
$$K_4 = f(x_n + h\, K_3)$$

$$x_{n+1} = x_n + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4)$$

* Take Away:

- Accuracy >> additional compute cost
- Even "good" integrators have issues

$\Rightarrow$ always sanity check