EECE 5550
Mobile Robotics

Lecture 19: Introduction to Feedback Control – Part 2

Derya Aksaray

Assistant Professor

Department of Electrical and Computer Engineering

Northeastern University

Last time: Intro to feedback control

Intro to control theory

Open- and closed-loop control

Intro to PID control

Parameter tuning

- Last time: largely empirical
- **Brief discussion on ODE solutions**

Today’s agenda

Step response

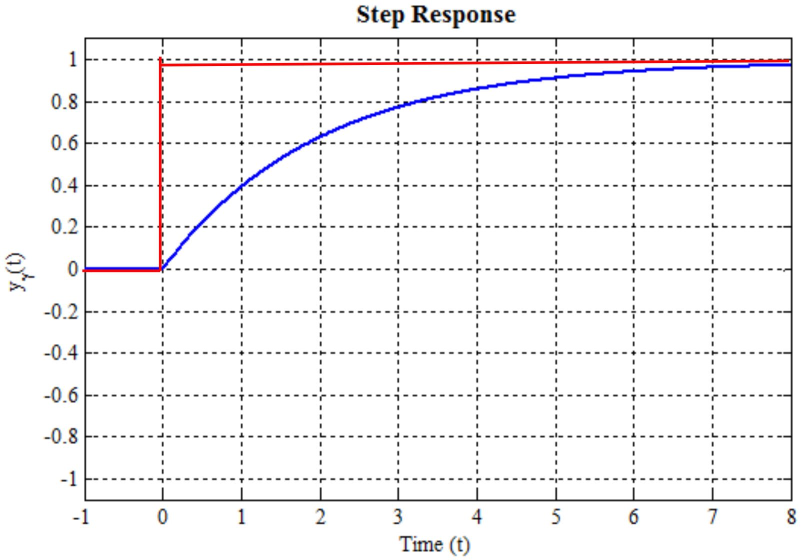
First order step response

Second order step response

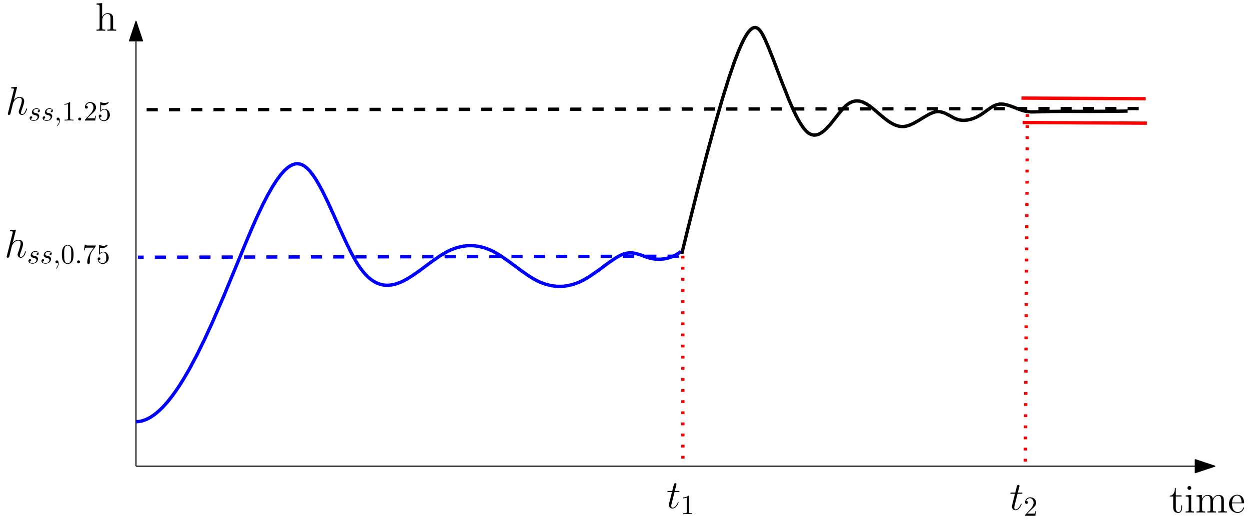
State-space representation

Step response

- Step response is how a system responds to a step input (a sudden input change)
- It gives an idea of the system's time response under an abrupt change of input.



Red: step input
Blue: system response



May not necessarily go to 1, can be perceived as going from one steady-state value to another
(e.g., a drone climbing from 0.75 m to 1.25 m)

First order step response

- Consider the following first order system:

$$\dot{y}(t) + a_0 y(t) = b_0 u(t)$$

$$\text{IC: } y(0) = 0$$

The coefficient a_1 in front of $\dot{y}(t)$ is normalized to simplify the derivation.

- Unit step input: $u(t) = 1$
- Solution to this ODE: $y(t) = y_P(t) + c_1 e^{st}$
- Characteristics equation: $s + a_0 = 0 \;\; \rightarrow \;\; s = -a_0$

If $s = -a_0 < 0$, **STABLE**

Particular solution for unit step response:

$$y_P(t) = \frac{b_0}{a_0}$$

General solution: $y(t) = \frac{b_0}{a_0} - \frac{b_0}{a_0} e^{-a_0 t}$

If $s = -a_0 \geq 0$, **UNSTABLE**

If $s = -a_0 > 0$

Particular solution for unit step response:

$$y_P(t) = \frac{b_0}{a_0}$$

General solution: $y(t) = \frac{b_0}{a_0} - \frac{b_0}{a_0} e^{-a_0 t}$

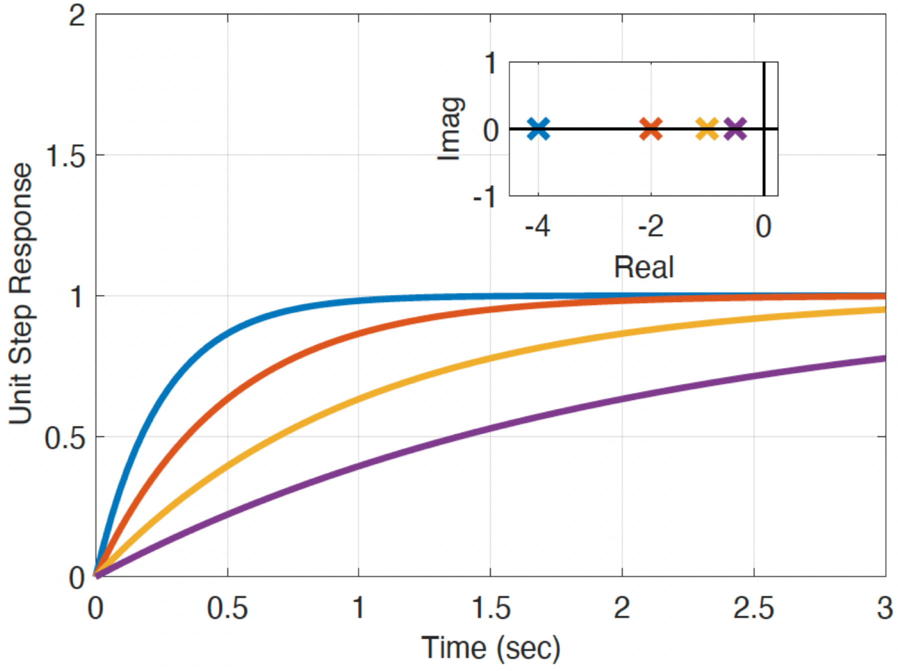
If $s = -a_0 = 0$,

ODE becomes: $\dot{y}(t) = b_0$

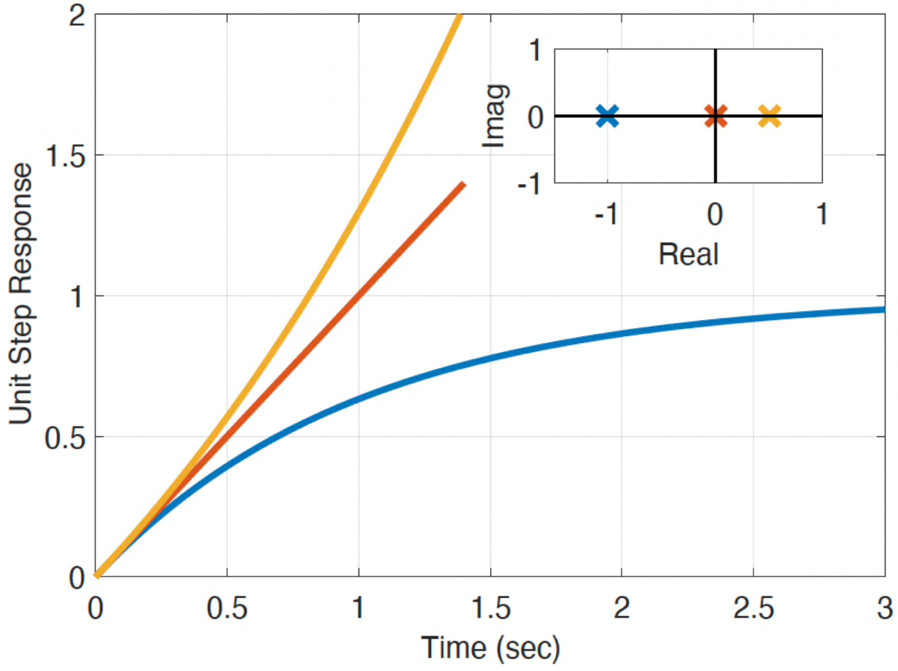
Particular solution: $y_P(t) = b_0 t$

General solution: $b_0 t$

First order step response



Stable roots



Stable/Unstable roots

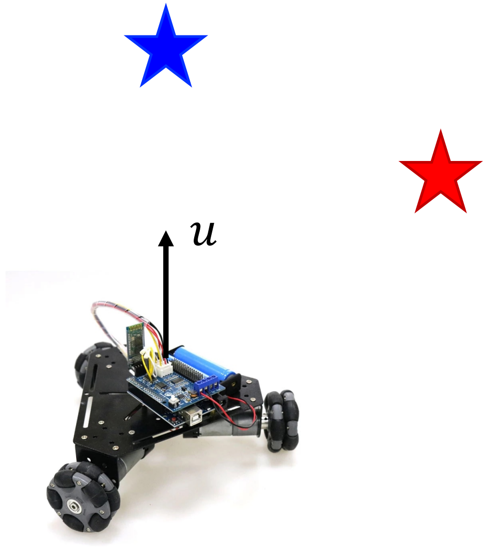
The left plot shows stable step responses with $a_1 = 1$ and $(a_0; b_0) = \{(4; 4), (2; 2), (1; 1), (0:5; 0:5)\}$. Each of these responses corresponds to a steady state value of $y = 1$. The right plot compares three responses with $a_1 = 1$, $b_0 = 1$, and a0 = {1, 0,-1}.

Example

In general, for an arbitrary goal state:
- Design a controller for motion in x direction
- Design a controller for motion in y direction
- Use kinematic model to find wheel velocities

- Motion in y direction: $\dot{y} = u, y_0 = 0$. Proportional control: $u(t) = K_p e(t)$.

(x_0, y^*)



(x_0, y_0)

$$\dot{y}(t) + a_0 y(t) = b_0 u(t)$$

$$a_0 = 0,\ b_0 = 1$$

Let's design $u(t) = K_p(y^* - y)$

$$\dot{y}(t) = K_p(y^* - y) \quad \leftrightarrow \quad \dot{y}(t) + K_p y = K_p y^*$$

What selection of K_p ensures stability?

In the closed loop model, our input is r.

The general solution will become: $y(t) = \frac{K_p y^*}{K_p} - \frac{K_p}{K_p} e^{-K_p t}$

What is the influence of increasing K_p?

Second order step response

- Consider the following second order system:

$$\ddot{y}(t) + a_1\dot{y}(t) + a_0 y(t) = b_0 u(t)$$

IC: $y(0) = 0, \dot{y}(0) = 0$

For stable systems, another representation

$$\ddot{y}(t) + 2\zeta\omega_n\dot{y}(t) + \omega_n^2 y(t) = b_0 u(t)$$

The coefficient a_2 in front of $\ddot{y}(t)$ is normalized to simplify the derivation.

ω_n: natural frequency (rad/sec) ⟷ Tendency to oscillate under no damping force

ζ: damping ratio (unitless) ⟷ How oscillations decay after a disturbance

Characteristic equation: $s^2 + 2\zeta\omega_n s + \omega_n^2 = 0$ ➡ $s_{1,2} = -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1}$

The system is stable when…?

Second order step response

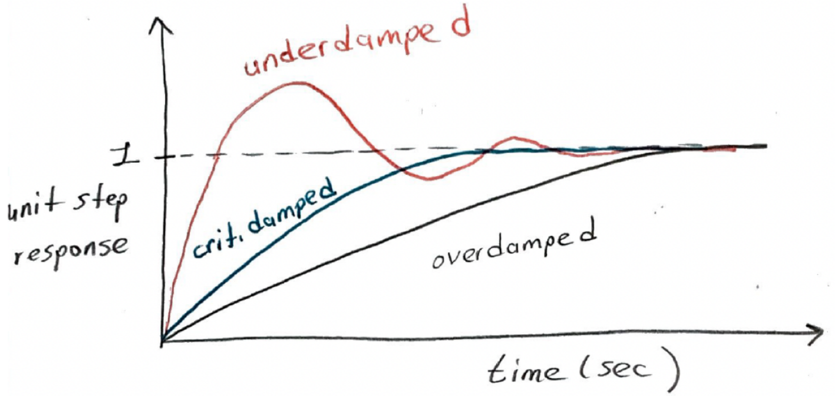
- Recall $s_{1,2} = -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1}$
- $\mathrm{sgn}(\zeta^2 - 1)$ is very important to characterize system behavior.
- The following behaviors are for stable systems:

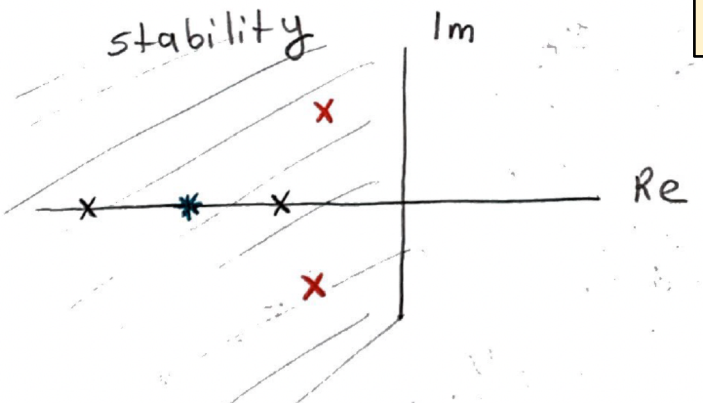
1. $\zeta^2 - 1 > 0 \;\rightarrow\; \zeta > 1 \;\rightarrow$ 2 distinct real roots $\rightarrow$ OVERDAMPED

2. $\zeta^2 - 1 = 0 \;\rightarrow\; \zeta = 1 \;\rightarrow$ 2 repeated real roots $\rightarrow$ CRITICALLY DAMPED

3. $\zeta^2 - 1 < 0 \;\rightarrow\; 0 < \zeta < 1 \;\rightarrow$ 2 complex roots $\rightarrow$ UNDERDAMPED

Settling time ~ $\frac{3}{\zeta\omega_n}$ [sec]





Example – P control

- $\ddot{y}(t) + a_1\dot{y}(t) + a_0 y(t) = b_0 u(t)$ and $u(t) = K_p e(t)$.

$$e(t) = r(t) - y(t)$$

$$\ddot{y}(t) + a_1\dot{y}(t) + (a_0 - b_0 K_p) y(t) = b_0 K_p r(t)$$

$$\boxed{\ddot{y}(t) + 2\zeta\omega_n\dot{y}(t) + \omega_n^2 y(t) = b_0 u(t)}$$

- The system is stable: $a_1 > 0$ and $(a_0 - b_0 K_p) > 0$

- If $a_1 \leq 0$, it is not possible to stabilize the system!

$$\ddot{h} = \frac{4k_T u}{m} - g$$

IC: $h(0) = h_0, \dot{h}(0) = \dot{h}_0$

Example – PD control

- Altitude dynamics of a quadrotor

$$\ddot{h} = \frac{4k_T u}{m} - g$$

IC: $h(0) = h_0, \dot{h}(0) = \dot{h}_0$

$e = r - h$ and $\dot{e} = -\dot{h}$

Define a PD control: $u = K_p(r - h) - K_d\dot{h} + \frac{mg}{4k_T}$ Perfect gravity cancellation

If you plug this control in the system model, can you find relationships between ζ, ω_n, K_p, and K_d?

$$\ddot{y}(t) + 2\zeta\omega_n\dot{y}(t) + \omega_n^2 y(t) = b_0 u(t)$$

Solution

$$\ddot{h} = \frac{4k_T}{m}\left[K_p(r-h) - K_d\dot{h} + \frac{mg}{4k_T}\right] - g$$

$$\ddot{h} + \underbrace{\frac{4k_T}{m}K_d}_{2\zeta\omega_n}\dot{h} + \underbrace{\frac{4k_T}{m}K_p}_{\omega_n^2}h = \frac{4k_T}{m}r$$

For $\omega_n > 0$ and $\zeta > 0$,

Given K_p, $\omega_n = \sqrt{\frac{4k_T K_p}{m}}$	Given ω_n, $K_p = \frac{\omega_n^2 m}{4k_T}$
Given K_p, K_d $\zeta = \frac{4k_T}{m}K_d\frac{1}{2\omega_n}$	Given ω_n, ζ $K_d = \frac{\zeta\omega_n m}{2k_T}$

Linear State Space

n^{th} order linear state-space model with input u and output y takes the form:

$$\dot{x}(t) = \mathrm{Ax(t) + Bu(t)}$$
$$y(t) = Cx(t) + Du(t)$$
$$IC: x(0) = x_0 \in \mathbb{R}^n$$

$A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times m}, C \in \mathbb{R}^{k\times n}, D \in \mathbb{R}^{k\times m}$

For now, let's consider single input single output (SISO) case, and the derivatives of input u do not appear in the ODE:

$$y^{[n]}(t) + a_{n-1}y^{[n-1]}(t) + \cdots + a_1\dot{y}(t) + a_0 y(t) = b_0 u(t)$$

$$\text{IC: } y(0) = y_0,\ \dot{y}(0) = \dot{y}_0\ , \dots ,\ y^{[n-1]}(0) = y_0^{[n-1]}$$

Note: The coefficient a_n has been normalized to simplify the notation.

Let's define state variables: $x_1 \coloneqq y, x_2 := \dot{y}, \dots, x_n = y^{[n-1]}$

Then, the first $n-1$ state variables satisfy $\dot{x}_1(t) = x_2(t), \dot{x}_2(t) = x_3(t)$, etc.

Moreover, $\dot{x}_n(t) = y^{[n]}(t)$.

Linear State Space

Recall

$$y^{[n]}(t) + a_{n-1}y^{[n-1]}(t) + \cdots + a_1\dot{y}(t) + a_0 y(t) = b_0 u(t)$$

IC: $y(0) = y_0,\ \dot{y}(0) = \dot{y}_0\ , \dots ,\ y^{[n-1]}(0) = y_0^{[n-1]}$

$$\dot{x}(t) = \mathrm{Ax(t)} + \mathrm{Bu(t)}$$

$$y(t) = Cx(t) + Du(t)$$

$$IC: x(0) = x_0 \in \mathbb{R}^n$$

$$\Longleftrightarrow$$

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ b_0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \end{bmatrix}, \quad D = 0$$

IC: $x(0) = [y_0, \dot{y}_0, \dots, y_0^{[n-1]}]^T$

Eigenvalues of A are the roots of the characteristic equation.
System is stable = all eigenvalues of A have negative real parts!

Example – Joint model

Consider the dynamical joint model of a humanoid, which can be modeled as a 2nd order system:

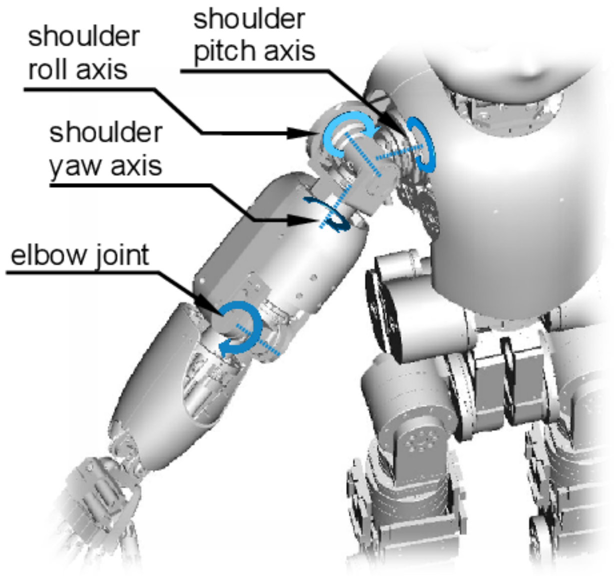
Torque coefficient

$$\ddot{\theta} = \frac{1}{J}(Ki - b\dot{\theta})$$

Moment of inertia

current

Friction coefficient



States: $\theta, \dot{\theta}$ **Control:** i **Output:** θ

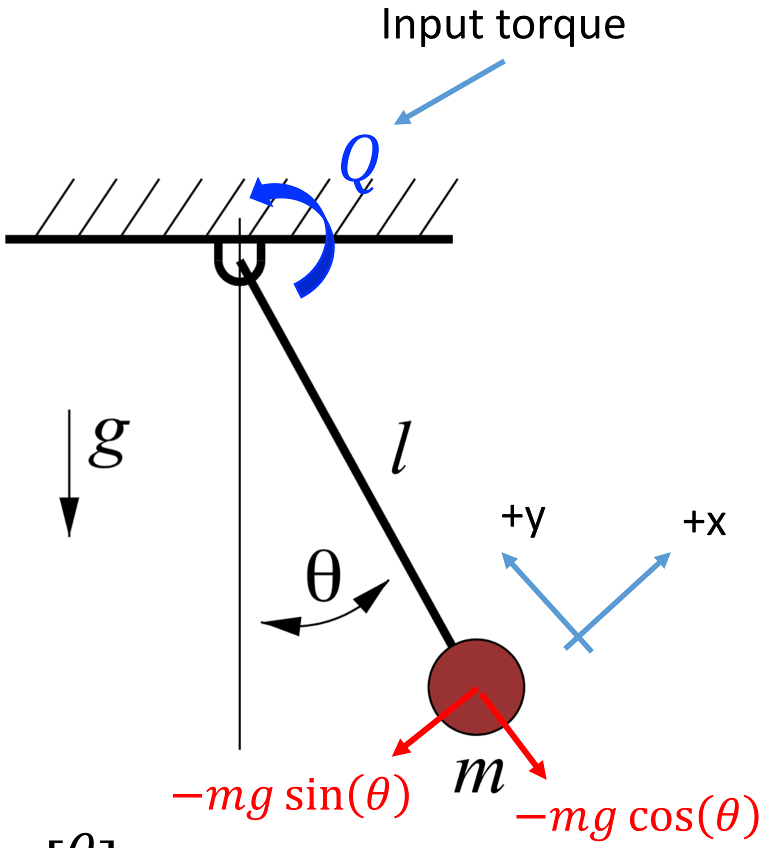
$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{b}{J} \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{K}{J} \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} x$$

Example - Pendulum

Net torque = Moment of inertia × angular acceleration

$$ml^2\ddot{\theta} = -mgl\sin(\theta) + Q = -mgl\sin(\theta) + cu$$



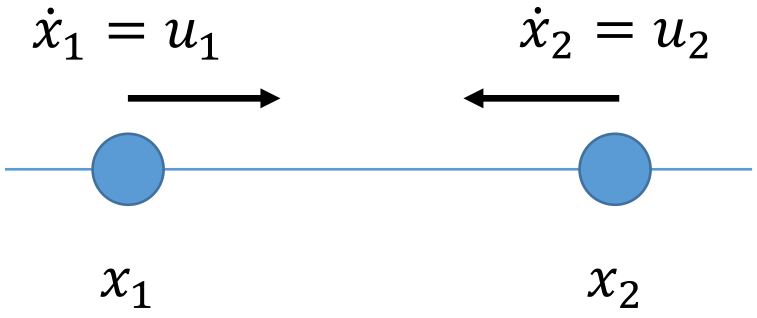
$$\ddot{\theta} = -\frac{g}{l}\sin(\theta) + c'u$$

Not linear!

Small angle approximation: $\theta \approx 0 \Rightarrow \quad \sin(\theta) \approx \theta$

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -g/l & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ c' \end{bmatrix} u \qquad y = \begin{bmatrix} 1 & 0 \end{bmatrix} x \qquad x = \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix}$$

Example – Two robots



$$\dot{x} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} u$$

Note that $A = 0$.

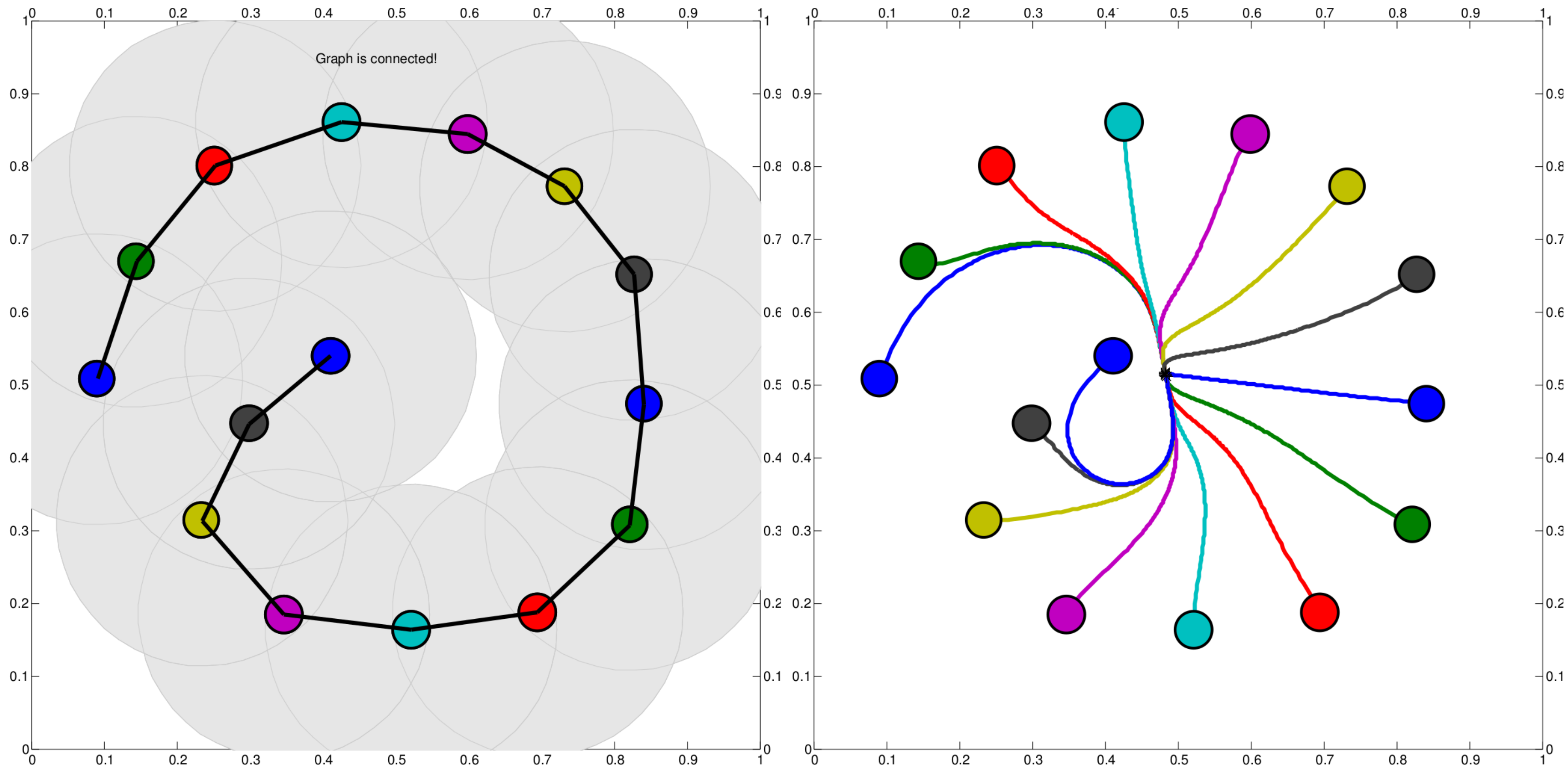
Rendezvous problem: Making them meet at the same location

Main idea: Having them aim towards each other

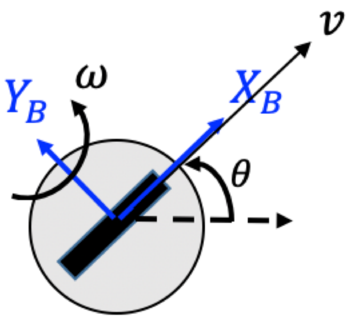
Control design $\begin{cases} u_1 = x_2 - x_1 \\ u_2 = x_1 - x_2 \end{cases}$

$$\dot{x} = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} x$$

Rendezvous of multi robots



Example – Unicycle robot



$$\dot{x} = v\cos(\theta)$$

$$\dot{y} = v\sin(\theta)$$

Not linear

$$\dot{\theta} = \omega$$

Small angle approximation: $\theta \approx 0 \Rightarrow \quad \sin(\theta) \approx \theta \quad$ and $\quad \cos(\theta) \approx 1$

$$\dot{x} = v$$

$$\dot{y} = v\theta$$

Still not linear

$$\dot{\theta} = \omega$$

A more systematic approach is needed.

Linearization

- The process of approximating a non-linear function with a linear one.
- For dynamical systems, the functions to linearize are $\boldsymbol{f}$ and $\boldsymbol{g}$.

$$\dot{x}(t) = \boldsymbol{f}\big(x(t), u(t)\big) \qquad\qquad y(t) = \boldsymbol{g}\big(x(t), u(t)\big)$$

$x(t) \in \mathbb{R}^n$: state vector $\quad$ $u(t) \in \mathbb{R}^m$: input vector $\quad$ $y(t) \in \mathbb{R}^p$: output vector

- **Goal:** Find a local, linear model around an operating point

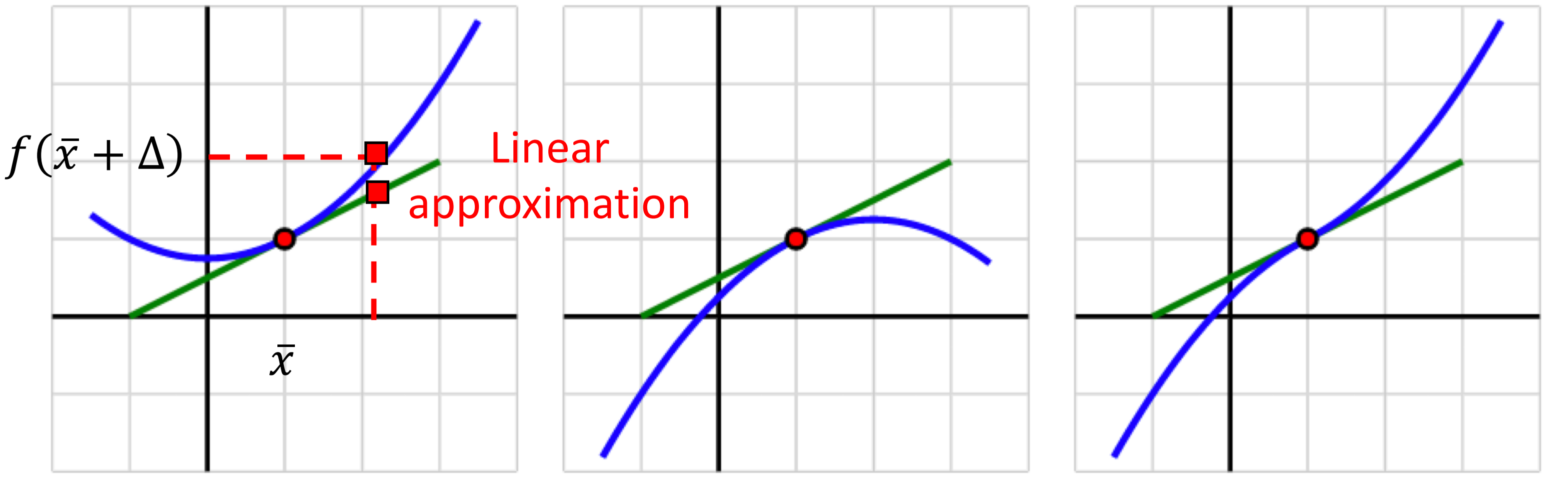
$$(x_0, u_0) \rightarrow (x_0 + \Delta_x, u = u_0 + \Delta_u)$$

Linearization in 1D

Consider a differentiable nonlinear function $f: \mathbb{R} \to \mathbb{R}$.

- e.g., $f(x) = x^2$, $f(x) = \ln(x)$, $f(x) = e^x$, $f(x) = \sin(x)$, etc.

Linearization of $f(x)$ around a point $\bar{x} \in \mathbb{R}$: the tangent line at $\bar{x}$.



For small $\Delta \in \mathbb{R}$

$$f(\bar{x} + \Delta) \approx f(\bar{x}) + \frac{df}{dx}(\bar{x})\Delta$$

Example

- Linearize $f(x) = \sin(x)$ around $\bar{x} = \frac{\pi}{3}$ and $\bar{x} = 0$.

$$\sin\left(\frac{\pi}{3} + \Delta\right) \approx \sin\left(\frac{\pi}{3}\right) + \frac{d\sin(x)}{dx}\left(\frac{\pi}{3}\right)\Delta = \frac{\sqrt{3}}{2} + \frac{1}{2}\Delta$$

$$\sin(\Delta) \approx \sin(0) + \cos(0)\Delta = \Delta$$

Connection to the Taylor Series

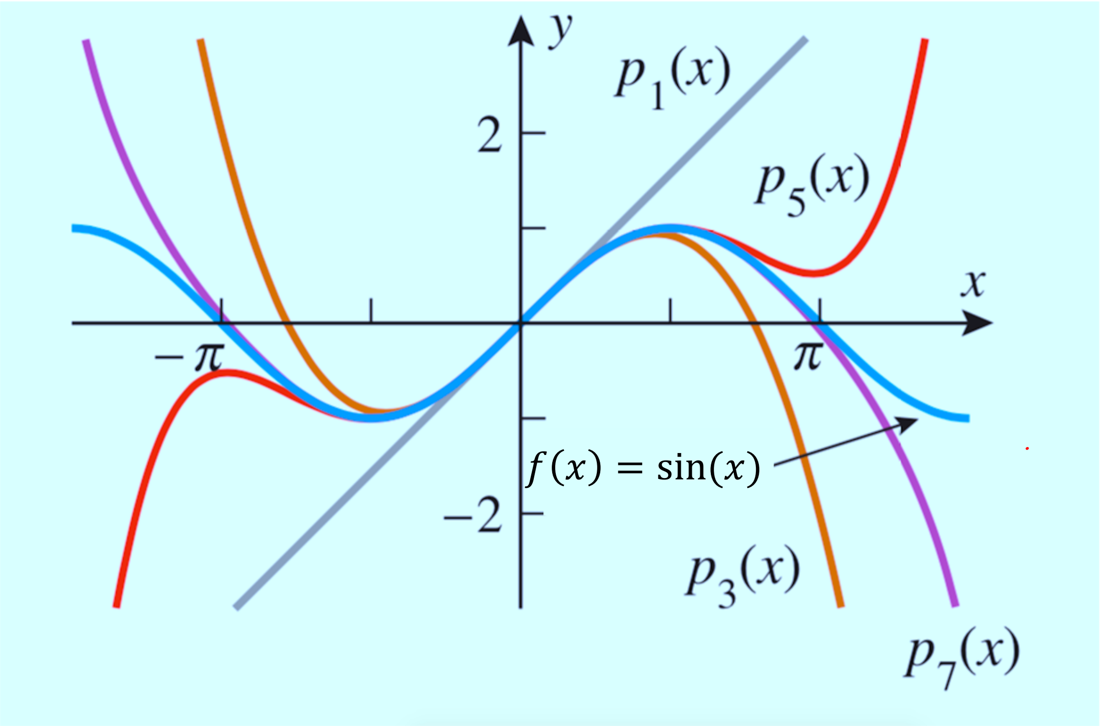
- For any $f(x)$ that is infinitely differentiable, the Taylor series at some $\bar{x}$ is

$$f(\bar{x}) + \sum_{k=1}^{\infty} \frac{d^k f}{dx^k}(\bar{x}) \frac{(x - \bar{x})^k}{k!}$$

- Taylor series converges to $f(x)$ around $\bar{x}$ if $f(x)$ is analytic (locally represented by power series)
  - e.g., $f(x) = x^2$, $f(x) = \ln(x)$, $f(x) = e^x$, $f(x) = \sin(x)$, …

Connection to the Taylor Series

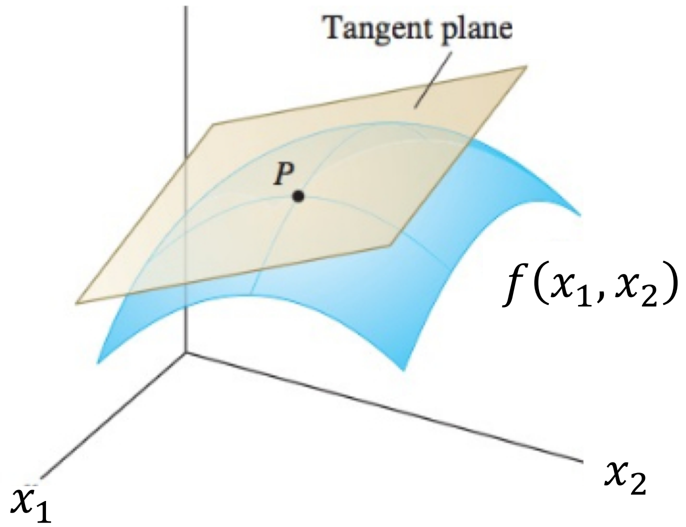
- Linearization: 1^{st} order Taylor polynomial (neglecting higher-order terms).



$$f(\bar{x}) + \sum_{k=1}^{\infty} \frac{d^k f}{dx^k}(\bar{x}) \frac{(x-\bar{x})^k}{k!}$$

Linearization in 2D

- Consider a differentiable nonlinear function $f: \mathbb{R}^2 \to \mathbb{R}$.
- Linearization of $f(x_1, x_2)$ around $(\bar{x}_1, \bar{x}_2)$: the tangent plane at $(\bar{x}_1, \bar{x}_2)$.



For small $\Delta_1, \Delta_2 \in \mathbb{R}$

$$f(\bar{x}_1 + \Delta_1, \bar{x}_2 + \Delta_2) \approx f(\bar{x}_1, \bar{x}_2) + \frac{\partial f}{\partial x_1}(\bar{x}_1, \bar{x}_2)\,\Delta_1 + \frac{\partial f}{\partial x_2}(\bar{x}_1, \bar{x}_2)\,\Delta_2$$

Linearization in Higher Dimensions

- Use vectors and matrices for a compact expression.
- Consider a differentiable nonlinear function $\boldsymbol{f}: \mathbb{R}^n \to \mathbb{R}^p$.
- Linearization around a point $\bar{x} \in \mathbb{R}^n$:

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} f_1(x) \\ \vdots \\ f_p(x) \end{bmatrix}$$

For small $\Delta \in \mathbb{R}^n$

$$\boldsymbol{f}(\bar{x} + \Delta) \approx \boldsymbol{f}(\bar{x}) + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\bar{x})\Delta$$

$$\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\bar{x}) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(\bar{x}) & \dots & \frac{\partial f_1}{\partial x_n}(\bar{x}) \\ \vdots & \dots & \vdots \\ \frac{\partial f_p}{\partial x_1}(\bar{x}) & \dots & \frac{\partial f_p}{\partial x_n}(\bar{x}) \end{bmatrix}$$

System Linearization

Given a dynamical system:

$$\dot{x}(t) = \boldsymbol{f}\big(x(t), u(t)\big) \qquad\qquad y(t) = \boldsymbol{g}\big(x(t), u(t)\big)$$

$x(t) \in \mathbb{R}^n$: state vector $\qquad$ $u(t) \in \mathbb{R}^m$: input vector $\qquad$ $y(t) \in \mathbb{R}^p$: output vector

- We linearize the system around a pair $\bar{x}(t), \bar{u}(t)$ satisfying the ODE:

$$\dot{\bar{x}}(t) = \boldsymbol{f}\big(\bar{x}(t), \bar{u}(t)\big)$$

- The pair $\bar{x}(t), \bar{u}(t)$ can correspond to an equilibrium point.
  - $\bar{x} \in \mathbb{R}^n$ is called an equilibrium point if there exists $\bar{u} \in \mathbb{R}^m$ such that $\boldsymbol{f}(\bar{x}, \bar{u}) = \mathbf{0}$
  - If $x(0) = \bar{x}$, $x(t) = \bar{x}$ for all $t \geq 0$ under the equilibrium input $u(t) = \bar{u}$.

System Linearization (Equilibrium Point)

- Let the pair $\bar{x}(t), \bar{u}(t)$ be an equilibrium point and an equilibrium input.

- What if we start slightly away from $\bar{x}$ and apply a slightly different input?

- Let the deviation variables be $\Delta_x(t) = x(t) - \bar{x}$ and $\Delta_u(t) = u(t) - \bar{u}$.

$$\dot{\Delta}_x(t) = \dot{x}(t) = \boldsymbol{f}\big(\bar{x} + \Delta_x(t), \bar{u} + \Delta_u(t)\big)$$

- Using the linearization of $\boldsymbol{f}$ around $(\bar{x}, \bar{u})$ we obtain

$$\dot{\Delta}_x(t) \approx \boldsymbol{f}(\bar{x}, \bar{u}) + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\bar{x}, \bar{u})\Delta_x(t) + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}(\bar{x}, \bar{u})\Delta_u(t)$$

System Linearization (Equilibrium Point)

- Using the linearization of $\boldsymbol{f}$ around $(\bar{x}, \bar{u})$ we obtain

$$\dot{\Delta}_x(t) \approx \boldsymbol{f}(\bar{x}, \bar{u}) + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\bar{x}, \bar{u})\Delta_x(t) + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}(\bar{x}, \bar{u})\Delta_u(t)$$

$$\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\bar{x}, \bar{u}) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(\bar{x}, \bar{u}) & \dots & \frac{\partial f_1}{\partial x_n}(\bar{x}, \bar{u}) \\ \vdots & \dots & \vdots \\ \frac{\partial f_n}{\partial x_1}(\bar{x}, \bar{u}) & \dots & \frac{\partial f_n}{\partial x_n}(\bar{x}, \bar{u}) \end{bmatrix} \qquad \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}(\bar{x}, \bar{u}) = \begin{bmatrix} \frac{\partial f_1}{\partial u_1}(\bar{x}, \bar{u}) & \dots & \frac{\partial f_1}{\partial u_m}(\bar{x}, \bar{u}) \\ \vdots & \dots & \vdots \\ \frac{\partial f_n}{\partial u_1}(\bar{x}, \bar{u}) & \dots & \frac{\partial f_n}{\partial u_m}(\bar{x}, \bar{u}) \end{bmatrix}$$

- Since $\boldsymbol{f}(\bar{x}, \bar{u}) = \boldsymbol{0}$, we can write this as $\dot{\Delta}_x(t) \approx \boldsymbol{A}\Delta_x(t) + \boldsymbol{B}\Delta_u(t)$.

$$\boldsymbol{A} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\bar{x}, \bar{u}) \qquad \boldsymbol{B} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}(\bar{x}, \bar{u})$$

System Linearization (Equilibrium Point)

- Linearization of the output equation $y(t) = \boldsymbol{g}\big(x(t), u(t)\big)$:

$$y(t) \approx \boldsymbol{g}(\bar{x}, \bar{u}) + \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}(\bar{x}, \bar{u})\Delta_x(t) + \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}}(\bar{x}, \bar{u})\Delta_u(t)$$

- Let $\Delta_y(t) = y(t) - \boldsymbol{g}(\bar{x}, \bar{u})$, then $\Delta_y(t) \approx \boldsymbol{C}\Delta_x(t) + \boldsymbol{D}\Delta_u(t)$.

$$\boldsymbol{C} = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}(\bar{x}, \bar{u}) \qquad\qquad \boldsymbol{D} = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}}(\bar{x}, \bar{u})$$

System Linearization (Equilibrium Point)

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}\big(x(t), u(t)\big) \qquad\qquad \boldsymbol{y}(t) = \boldsymbol{g}\big(x(t), u(t)\big)$$

- Given an equilibrium pair $\bar{x} \in \mathbb{R}^n$ and $\bar{u} \in \mathbb{R}^m$, define the deviation variables

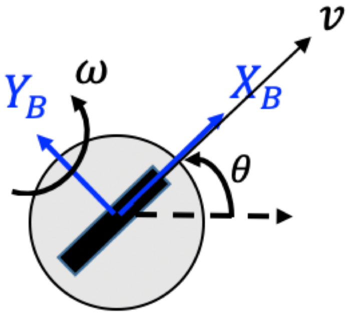
$$\Delta_x(t) = x(t) - \bar{x} \qquad \Delta_u(t) = u(t) - \bar{u} \qquad \Delta_y(t) = y(t) - \boldsymbol{g}(\bar{x}, \bar{u})$$

- For small $\Delta_x(t)$ and $\Delta_u(t)$, the deviation variables behave like a linear time invariant (LTI) sys.

$$\dot{\Delta}_x(t) \approx \boldsymbol{A}\Delta_x(t) + \boldsymbol{B}\Delta_u(t)$$

$$\Delta_y(t) \approx \boldsymbol{C}\Delta_x(t) + \boldsymbol{D}\Delta_u(t)$$

Example – Unicycle robot



$$\dot{x} = v\cos(\theta)$$

$$\dot{y} = v\sin(\theta)$$

$$\dot{\theta} = \omega$$

Not linear

Punchline

Linearization gives reasonable models
But not all the time!

If they work, they are very useful as they enable local analysis

Small angle approximation: $\theta \approx 0 \Rightarrow$ $\sin(\theta) \approx \theta$ and $\cos(\theta) \approx 1$

Linearization

$$\dot{x} = v \qquad \dot{y} = v\theta \qquad \dot{\theta} = \omega$$

$$x_1 = x \qquad x_2 = y \qquad x_3 = \theta$$

$$y_1 = x_1 \qquad y_2 = x_2 \qquad y_3 = x_3$$

$$u_1 = v \qquad u_2 = \omega$$

An equilibrium point

$$(\bar{x}, \bar{u}) = (\mathbf{0}, \mathbf{0})$$

$$\underbrace{A = \mathbf{0} \qquad B = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}}_{\dot{x}_2 = 0 \quad ?} \qquad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$