# Module 1-1: Linear Dynamics Modeling

## Linear Control Systems (2020)

Ding Zhao

Assistant Professor

College of Engineering

School of Computer Science

Carnegie Mellon University

# Table of Contents

# Course Administration

- Slides/assignments: Canvas
- Lectures/recitations/office hours will be on Zoom.Tips to join Zoom meetings.
- Submission: Gradescope (access on Canvas)
- Textbook: there is no textbook. All information needed is on the slides
- Forum: Campuswire (https://campuswire.com/)

# Table of Contents

# Linear Control SYSTEMS

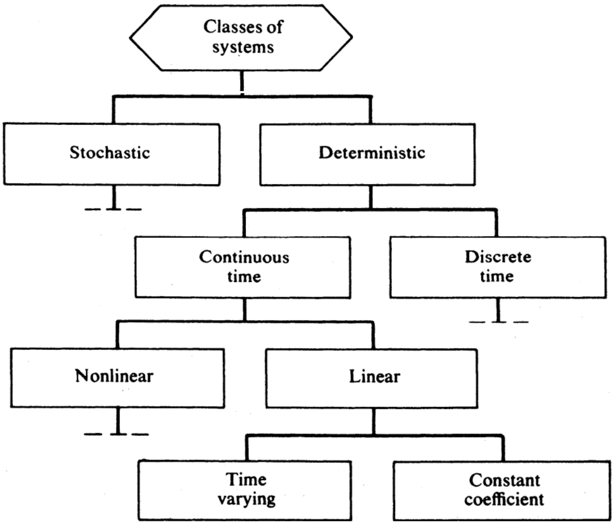
- We will take an abstract view of a system as a mapping from inputs $\mathbf{u} \in \mathbb{R}^m$ to output $\mathbf{y} \in \mathbb{R}^n$
- Both $\mathbf{u}$ and $\mathbf{y}$ are vectors
- Both $\mathbf{u}$ and $\mathbf{y}$ are functions of time
- Write $H$ as a mapping $H : \mathbb{R}^m \to \mathbb{R}^n$ where $\mathbb{R}^i$ is a set of $i$-dimensional signals

$$u \longrightarrow \boxed{H} \longrightarrow y$$

**Continuous Time System**:
$H$ is CT if both $\mathbf{u}$ and $\mathbf{y}$ are continuous time signals.
$\rightarrow$ Described by ODEs $\dot{\mathbf{y}}(t) = f(\mathbf{y}, \mathbf{u}(t))$

**Discrete Time System**:
$H$ is DT if both $\mathbf{u}$ and $\mathbf{y}$ are discrete time signals.
$\rightarrow$ Described by difference equations $\mathbf{y}[(k+1)T] = f(\mathbf{u}(kT)) \Rightarrow \mathbf{y}(k+1) = f(\mathbf{y}(k), \mathbf{u}(k))$

# Linear Control SYSTEMS -Types



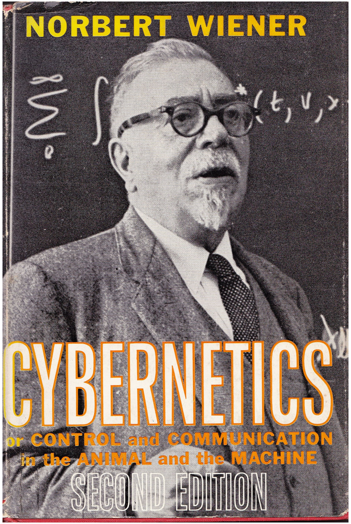

**Types of systems**

[Brogan, Modern Control Theory, 1990]

# Linear CONTROL Systems - A Brief History

**Control**:continuously operating dynamical systems

- **1920s**: Minorsky worked on automatic controllers (PID) for steering ships
- **1930s**: Nyquist/Bode (Bell Lab) developed methods for analyzing the stability of controlled systems
- **1940s**: Frequency response methods made it possible to design linear closed-loop Norbert Wiener (Cybernetics)
- **1950s**: Root-locus method due to Evans was fully developed
- **1960s**: State space methods linear model-based (MIMO) optimal/stochastic/ adaptive control Rudolf Kalman (Apollo)
- **1980s**: Machine learning controls nolinear model-free genetic algorithm, neural network reinforcement learning control
- **Present**: modern control theory in engr,bio,med,econ,social etc.




# PART I

## *ORIGINAL EDITION*

1948

# Knowledge in this will be Useful in the Following Courses

- 24-740 Combustion and Air Pollution Control
- 24-662 Robotic Systems and Internet of Things
- 24-671 Electromechanical Systems Design
- 24-673 Soft Robots
- 24-757 Vibrations
- 24-773 Multivariable Linear Control
- 24-774 Advanced Control Systems Integration
- 24-775 Robot Design & Experimentation
- 24-776 Nonlinear Controls
- 24-778 Mechatronic Design
- 24-785 Engineering Optimization
- 24-xxx Trustworthy AI Autonomy (Spring 2021)

- 16-642 Manipulation, Estimation, and Control
- 16-711 Kinematics, Dynamic Systems and Control
- 16-722 Sensing and Sensors
- 16-741 Mechanics of Manipulation
- 16-745 Dynamic Optimization
- 16-748 Underactuated Robots
- 16-861 Mobile Robot Design
- 16-865 Advanced Mobile Robot Design
- 16-868 Biomechanics and Motor Control

# Table of Contents

# Syllabus of This Course - Canvas

### Dynamic Modeling and Analysis

- Module 1-1: Linear Dynamics Modeling
- Module 1-2: Solving Linear Dynamics
- Module 1-3: Controllability and Observability
- Module 1-4: Realization (State Space vs Transfer Fun.)
- Module 1-5: Stability
- Each topic has a homework

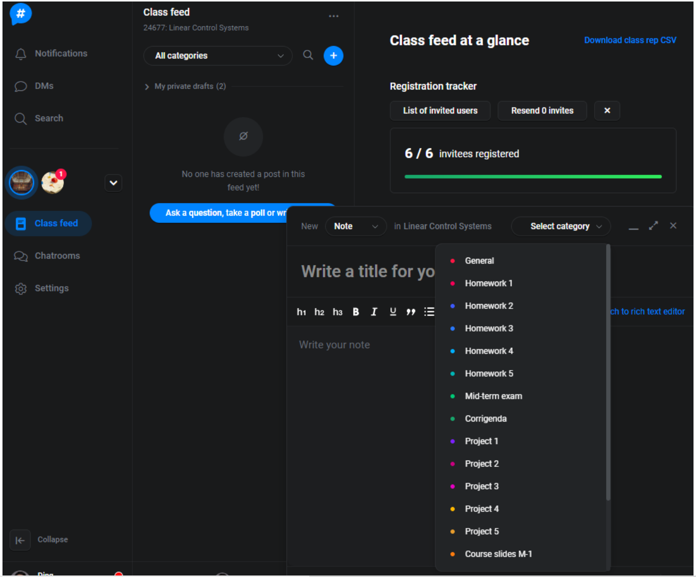
$\Rightarrow$ **Midterm**

### Design Methods for Control

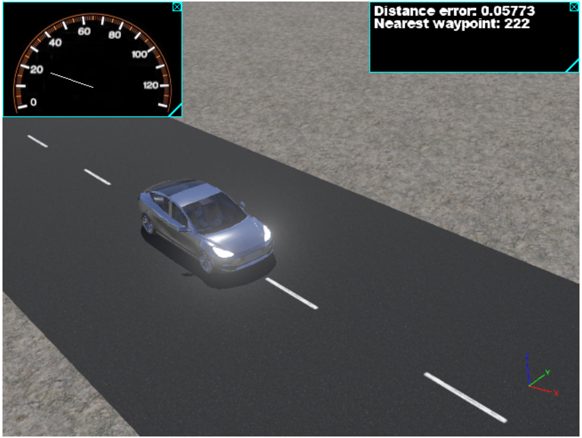
- Module 2-1: Feedback Control (Pole Placement)
- Module 2-2: Introduction to Optimal Control (LQR, MPC)
- Module 2-3: Introduction to Stochastic Control (Kalman Filter)
- Module 2-4: Introduction to Adaptive Control (MRAC)
- Module 2-5: Introduction to Control-Based Methods in Learning (if time permits)
- Sub-module 1-4 have 5 projects

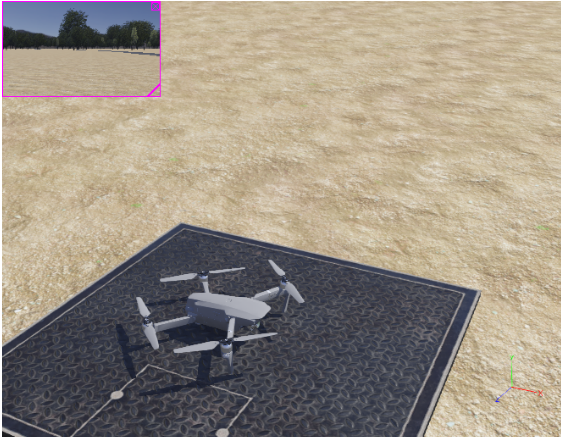
# Assessment

- Homework: 20%
- Midterm : 25%
- Projects: 50%
- Participation: 5%



# Projects


Distance error: 0.05773
Nearest waypoint: 222
0
20
40
60
80
100
120



# Specific Goals of This Course

I will help you to be

- Confident in the interview for positions of control engineers
- Familiar with the most important control nomenclature, theories, algorithms
- Capable to implement real-world control algorithms independently

I have talked to my students and summarized the following jargons that have been often mentioned in their interviews. By the end of the class, you will be able to explain them.

- **Core math**: QR, LU, SVD, Cholesky, Jordan, diagonal decomposition, Riccati Equation, Cayley Hamilton Theorem, and why they are useful in control?!!
- **Control**: Lyapunov stability, controllability, observability, LQR, LQG, MPC, separation principle, realization, Kalman Decomposition, Kalman Filter, EKF, UKF
- **Programming**: Python related questions (need extra practice)

The most common question being asked is: explain LQR - is it always stable? why?
I will spend a lecture to derive it with you.

# WARNINGS

- This is a math course. NOT a project-based course.
  Modern controller design is based on a solid mathematical foundation.
  Different from e.g. deep learning.
- Pure empirical hand-tuning is difficult and risky. -"don't tell me, prove it to me".
  Safety critical applications include space aircraft, automobile, economy, electric grids,etc.
- I will try to balance between the math rigorousness and application broadness in teaching and assignment. But I also need your devotion to spend enough time and efforts learning the math.
- **Drop the course if you have already selected 2+ "hardcore" courses, or 2 courses plus heavy research duty (research masters). The workload of this course may make you feel uncomfortable.**
- Remain in this course, only if you want to build an expertise on control/automation. This is graduate school, five courses per semester even with a high GPA may not bring you good job offers. Focus on building your expertise. Companies hire you to work on specific tasks not to cover a broad area!

# Prerequisites

If the following contents sound unfamiliar to you, you may consider take this course next year.

**These topics are ones that I assume you know, I may barely review them:**

- **Calculus** (e.g. derivative/integral)
  21-120 Calculus I, 21-122 Calculus II
- **Linear algebra** (e.g. matrix inversion, eigenvectors/eigenvalues)
  24-282 Special Topics: Linear Algebra and Vector Calculus for Engineers
  Gilbert Strang, "Introduction to Linear Algebra (5th Edition)," 2016
- **Classical control** (e.g. Laplace transformation, transfer function, poles/zeros)
  24-451 Feedback Control Systems
  Katsuhiko Ogata, "Modern control engineering (5th Edition)", Pearson, 2009
- **Basic statistics** (e.g. mean, variance, Gaussian distribution)
  Robert S. Witte, John S. Witte, "Statistics (10th Edition)," Wiley, 2009

**These topics are ones that I assume you have learned. I will review but will not teach them:**

- Stability (positive poles) for classical control theories
- PID controllers

# Influence of Covid 19

- We will work together to rise to this challenge.
- I will try my best to accommodate you to your difficulties.
- Raise hands or type in questions in the chat to encourage communication.
- Use breakout rooms during the lecture and Campuswire to find study mates.
- Arrange meetings with me and TAs.

# Table of Contents

# LINEAR Control Systems

**Linearity**: $H$ is linear if it is both **additive** and **homogeneous**

- Additivity:

$$H\left(\mathbf{u}_1+\mathbf{u}_2\right)=H\left(\mathbf{u}_1\right)+H\left(\mathbf{u}_2\right), \forall u_1, u_2 \in S^m$$

- Homogeneity:

$$H(\alpha \mathbf{u})=\alpha H(\mathbf{u}), \forall \mathbf{u}, \forall \text{ scalar } \alpha$$

These can be combined into

$$H\left(\alpha \mathbf{u}_1+\beta \mathbf{u}_2\right)=\alpha H\left(\mathbf{u}_1\right)+\beta H\left(\mathbf{u}_2\right)$$

# Time Invariant Systems

- A time-invariant system has a time-dependent system function that is not a direct function of time.
- Mathematically speaking, a "time-invariance" system has the following property: Given a system with a time-dependent output function $y(t)$ and a time-dependent input function $x(t)$ the system will be considered time-invariant if a time-delay on the input $x(t+\tau)$ directly equates to a time-delay of the output $y(t+\tau)$ function.

# Example

For the following, determine if the system is 1) linear 2) time-invariant

a) $\ddot{\mathbf{y}} = \mathbf{u} - k\mathbf{y}^2$, for $k \in \mathbb{R}$
Let 2 I/O pairs be given by $\mathbf{y}_1 = H(\mathbf{u}_1)$, $\mathbf{y}_2 = H(\mathbf{u}_2)$

$$\ddot{\mathbf{y}}_1 = \mathbf{u}_1 - k\mathbf{y}_1^2$$

$$\ddot{\mathbf{y}}_2 = \mathbf{u}_2 - k\mathbf{y}_2^2$$

Add together

$$\ddot{\mathbf{y}}_1 + \ddot{\mathbf{y}}_2 = \mathbf{u}_1 + \mathbf{u}_2 - k\left(\mathbf{y}_1^2 + \mathbf{y}_2^2\right)$$

If additive, $\left(\mathbf{y}_1 \ddot{+} \mathbf{y}_2\right) = \mathbf{u}_1 + \mathbf{u}_2 - k\left(\mathbf{y}_1 + \mathbf{y}_2\right)^2 \Rightarrow$ <u>Not linear</u>
For time invariance, substitute $t = t - \tau$

$$\ddot{\mathbf{y}}(t-\tau) = \mathbf{u}(t-\tau) - k\mathbf{y}(t-\tau)^2$$

Dynamics are unchanged by delay $\Rightarrow$ <u>time-invariant</u>

# Example (OTB)

b) $\ddot{\mathbf{y}} = 5t^2\dot{\mathbf{y}} + 3\mathbf{y} + \mathbf{u}$

Let $\mathbf{y}_1 = H(\mathbf{u}_1)$, $\mathbf{y}_2 = H(\mathbf{u}_2)$

$$\ddot{\mathbf{y}}_1 = 5t^2\dot{\mathbf{y}}_1 + 3\mathbf{y}_1 + \mathbf{u}_1$$

$$\ddot{\mathbf{y}}_2 = 5t^2\dot{\mathbf{y}}_2 + 3\mathbf{y}_2 + \mathbf{u}_2$$

$$\left(\mathbf{y}_1 \ddot{+} \mathbf{y}_2\right) = 5t^2\left(\mathbf{y}_1 \dot{+} \mathbf{y}_2\right) + 3\left(\mathbf{y}_1 \dot{+} \mathbf{y}_2\right) + (\mathbf{u}_1 + \mathbf{u}_2) \rightarrow \text{Additive}$$

$$\left(\alpha\ddot{\mathbf{y}}\right) = 5t^2\left(\alpha\dot{\mathbf{y}}\right) + 3\left(\alpha\mathbf{y}\right) + \alpha\mathbf{u} \rightarrow \text{Homogeneous}$$

$$\Rightarrow \underline{\text{Linear}}$$

Let $t = t - \tau$

$$\ddot{\mathbf{y}}(t-\tau) = 5(t-\tau)^2\dot{\mathbf{y}}(t-\tau) + 3\mathbf{y}(t-\tau) + \mathbf{u}(t-\tau) \neq 5t^2\dot{\mathbf{y}}(t-\tau) + 3\mathbf{y}(t-\tau) + \mathbf{u}(t-\tau)$$

$$\Rightarrow \underline{\text{Time varying}}$$

Tricks to check time-invariant systems: in $D_\tau(y(t)) = y(t-\tau)$, we will **replace all the $t$ with** $t-\tau$, while in $D_\tau(u(t)) = u(t-\tau)$, we only **add $-\tau$ in the parentheses of** $u(t)$, i.e. we will not change $t$ outside of $u(t)$. Then check whether $D_\tau(y(t)) =$ the output of $D_\tau(u(t))$

# Table of Contents

# State Space Equation

- Can we establish theories that can work on **all** linear control systems in automotive/aerospace engineering, chemistry reaction, petroleum industry, economy?
- First we need to have a universal way to describe these systems. Fortunately, we do.

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}\\ \mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}\end{aligned}$$

Rudolf Kálmán, 1930-2016

# A Brief Introduction to Me

L. Prandtl (1900)
"Father of modern aerodynamics"

T. von Karman (1908), founded JPL

X. Qian (1939)

H. W. Liepmann (1938), fluid mechanics

A. E. Bryson (1951), optimal control, backpropagation

R. E. Kalman (1957)

M. Tomizuka (1974)

J. K. Hedrick (1971)

H. Peng (1992)

H.

Ding Zhao

# A Brief Introduction to Me

L. Euler (1726)

P. S. Laplace (1769) J. L. Lagrange (1755)

J. Fourier (1790) S. D. Poisson (1800) C. F. Gauss (1798)

G. L. Dirichlet (1827)

R. Lipschitz (1853) J. Plucker (1823)

F. C. Klein (1868)

E. Schmidt (1905) → D. Hilbert (1885) → F. Lindemann (1873)

S. Bochner (1921) → H. Scarf (1954) → D. Iglehart (1961) →

P. Glynn (1982) → J. Blanchet (2004) → H. Lam (2011) →

L. Prandtl (1900)

T. von Kármán (1908)

X. Qian (1939) H. W. Liepmann (1938)

A. E. Bryson (1951) R. E. Kálmán (1957)

M. Tomizuka (1974) J. K. Hedrick (1971)

H. Peng (1992)

Ding Zhao

# Notation

An $m$ x $n$ matrix: $m$ rows and $n$ columns

$$\begin{array}{c c} & \begin{array}{cccc} 1 & 2 & \dots & n \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ \vdots \\ m \end{array} & \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ a_{31} & a_{32} & \dots & a_{3n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \end{array}$$

- $m = 1$ : **Row matrix**
- $n = 1$ : **Column matrix**
- $m = n$ : **Square matrix**
- $m = n = 1$ : **Scalar**

# Matrix Transpose

Consider $m$ x $n$ matrix $\mathbf{A} = [a_{ij}]$; $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n$
Transpose of $\mathbf{A}$ is an $n$ x $m$ matrix $\mathbf{A}^T = [a_{ji}]$ .

Example: $\begin{bmatrix} 0 & 4 \\ 7 & 0 \\ 3 & 1 \end{bmatrix}^T = \begin{bmatrix} 0 & 7 & 3 \\ 4 & 0 & 1 \end{bmatrix}$

- if $\mathbf{A} = \mathbf{A}^T$ : $\mathbf{A}$ is said to be symmetric
- if $\mathbf{A} = -\mathbf{A}^T$ : $\mathbf{A}$ is said to be skew-symmetric

# Conjugate and Associate Matrix

**Conjugate matrix**

The conjugate of $\mathbf{A}$, written $\overline{\mathbf{A}}$, is the matrix formed by replacing every element in $\mathbf{A}$ by its complex conjugate. Thus $\overline{\mathbf{A}} = [\overline{a}_{ij}]$.

- If all elements of $\mathbf{A}$ are real, then $\overline{\mathbf{A}} = \mathbf{A}$
- If all elements are purely imaginary, then $\overline{\mathbf{A}} = -\mathbf{A}$

**Associate matrix**

The associate matrix of $\mathbf{A}$ is the conjugate transpose of $\mathbf{A}$. The order of these two operations is immaterial.

- $\mathbf{A} = \overline{\mathbf{A}}^T \Rightarrow \mathbf{A}$: Hermitian
- $\mathbf{A} = -\overline{\mathbf{A}}^T \Rightarrow \mathbf{A}$: Skew-Hermitian

For real matrices, symmetric and Hermitian mean the same.

# Matrix Addition and Subtraction

Matrix addition and subtraction are performed on an element-by-element basis. That is, if $\mathbf{A} = [a_{ij}]$ and $\mathbf{B} = [b_{ij}]$ are both $m \times n$ matrices, then $\mathbf{A} + \mathbf{B} = \mathbf{C}$ and $\mathbf{A} - \mathbf{B} = \mathbf{D}$ indicate that the matrices $\mathbf{C} = [c_{ij}]$ and $\mathbf{D} = [d_{ij}]$ are also $m \times n$ matrices whose elements are given by $c_{ij} = a_{ij} + b_{ij}$ and $d_{ij} = a_{ij} - b_{ij}$ for $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n$.

Example: $\begin{bmatrix} 0 & 4 \\ 7 & 0 \\ 3 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 0 & 4 \end{bmatrix} = \begin{bmatrix} 1 & 6 \\ 9 & 3 \\ 3 & 5 \end{bmatrix}$

**Properties:**

- $\mathbf{A} + \mathbf{B} = \mathbf{B} + \mathbf{A}$ **Commutative**
- $(\mathbf{A} + \mathbf{B}) + \mathbf{C} = \mathbf{A} + (\mathbf{B} + \mathbf{C})$ **Associative**
- $(\mathbf{A} + \mathbf{B})^T = \mathbf{A}^T + \mathbf{B}^T$

# Scalar Multiplication

Multiplication of a matrix $\mathbf{A} = [a_{ij}]$ by an arbitrary scalar $\alpha \in \mathscr{F}$ amounts to multiplying every element in $\mathbf{A}$ by $\alpha$. That is, $\alpha\mathbf{A} = \mathbf{A}\alpha = [\alpha a_{ij}]$.

Example: $(-2)\begin{bmatrix} 1 & 6 \\ 9 & 3 \\ 6 & 0 \end{bmatrix} = \begin{bmatrix} -2 & -12 \\ -18 & -6 \\ -12 & 0 \end{bmatrix}$

**Properties:**

- $(\alpha + \beta)\mathbf{A} = \alpha\mathbf{A} + \beta\mathbf{A}$
- $(\alpha\beta)\mathbf{A} = (\alpha)(\beta\mathbf{A})$
- $\alpha(\mathbf{A} + \mathbf{B}) = \alpha\mathbf{A} + \alpha\mathbf{B}$

# Matrix Multiplication

Consider an $m \times n$ matrix $\mathbf{A} = [a_{ij}]$ and a $p \times q$ matrix $\mathbf{B} = [b_{ij}]$. This product is only defined when $\mathbf{A}$ has the same number of columns as $\mathbf{B}$ has rows i.e., when $n = p$. The elements of $\mathbf{C} = [c_{ij}]$ are then computed according to $c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$

Example:

$$\begin{bmatrix} 2 & 3 \\ 4 & 5 \end{bmatrix} \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 8 \end{bmatrix} = \begin{bmatrix} 2(1)+3(2) & 2(3)+3(4) & 2(5)+3(8) \\ 4(1)+5(2) & 4(3)+5(4) & 4(5)+5(8) \end{bmatrix} = \begin{bmatrix} 8 & 18 & 34 \\ 14 & 32 & 60 \end{bmatrix}$$

**Properties:**

- $(\mathbf{AB})\mathbf{C} = \mathbf{A}(\mathbf{BC}) = \mathbf{ABC}$
- $\alpha(\mathbf{AB}) = (\alpha\mathbf{A})\mathbf{B} = \mathbf{A}(\alpha\mathbf{B})$, where $\alpha$ is a scalar
- $\mathbf{A}(\mathbf{B}+\mathbf{C}) = \mathbf{AB} + \mathbf{AC}$, $(\mathbf{A}+\mathbf{B})\mathbf{C} = \mathbf{AC} + \mathbf{BC}$
- $(\mathbf{AB})^T = \mathbf{B}^T\mathbf{A}^T$
- $\mathbf{AB} \neq \mathbf{BA}$ Not commutative, this makes vectors/matrices different than scalars

# Field

Let $\mathscr{F}$ be a set with at least 2 elements, assume $\mathscr{F}$ has 2 operations:
"$+$": $\mathscr{F} \times \mathscr{F} \to \mathscr{F}$ (addition) and "$\cdot$" : $\mathscr{F} \times \mathscr{F} \to \mathscr{F}$ (multiplication). $\mathscr{F}$ is called a field iff:

- $\mathbf{A_0}$ : $\forall \alpha, \beta \in \mathscr{F}, \exists \alpha + \beta \in \mathscr{F} \Rightarrow$ **Closure under Addition**
- $\mathbf{A_1}$ : $\forall \alpha, \beta \in \mathscr{F}, \alpha + \beta = \beta + \alpha \Rightarrow$ **Commutativity**
- $\mathbf{A_2}$ : $\forall \alpha, \beta, \gamma \in \mathscr{F}, (\alpha + \beta) + \gamma = \alpha + (\beta + \gamma) \Rightarrow$ **Associativity**
- $\mathbf{A_3}$ : $\exists 0 \in \mathscr{F}, \forall \alpha \in \mathscr{F}, \alpha + 0 = \alpha \Rightarrow$ **Neutral**
- $\mathbf{A_4}$ : $\forall \alpha \in \mathscr{F}, \exists (-\alpha) \in \mathscr{F}, \alpha + (-\alpha) = 0 \Rightarrow$ **Inverse**

- $\mathbf{M_0}$ : $\forall \alpha, \beta \in \mathscr{F}, \exists \alpha \cdot \beta \in \mathscr{F} \Rightarrow$ **Closure under Multiplication**
- $\mathbf{M_1}$ : $\forall \alpha, \beta \in \mathscr{F}, \alpha \cdot \beta = \beta \cdot \alpha \Rightarrow$ **Commutativity**
- $\mathbf{M_2}$ : $\forall \alpha, \beta \in \mathscr{F}, (\alpha \cdot \beta) \cdot \gamma = \alpha \cdot (\beta \cdot \gamma) \Rightarrow$ **Associativity**
- $\mathbf{M_3}$ : $\exists 1 \in \mathscr{F}, \forall \alpha \in \mathscr{F}, \alpha \cdot 1 = \alpha \Rightarrow$ **Neutral**
- $\mathbf{M_4}$ : $\forall \alpha \neq 0, \exists \alpha^{-1}, \alpha \cdot \alpha^{-1} = 1 \Rightarrow$ **Inverse**

- $\mathbf{D}$ : $\alpha, \beta, \gamma \in \mathscr{F}, \alpha \cdot (\beta + \gamma) = \alpha \cdot \beta + \alpha \cdot \gamma \Rightarrow$ **Distributivity**

# Vector Space

Vector spaces $\rightarrow$ Linear spaces Let $\mathscr{F}$ be a field, let $\mathscr{V}$ be a set that has an "addition" operation "+": $\mathscr{V} \times \mathscr{V} \rightarrow \mathscr{V}$. $\mathscr{V}$ is called a vector space over $\mathscr{F}$ iff:

- $\mathbf{A_0} : \forall \mathbf{x}, \mathbf{y} \in \mathscr{V}, \exists \mathbf{x} + \mathbf{y} \in \mathscr{V}$ **Closure under Addition**
- $\mathbf{A_1} : \forall \mathbf{x}, \mathbf{y} \in \mathscr{V}, \mathbf{x} + \mathbf{y} = \mathbf{y} + \mathbf{x}$ **Commutativity**
- $\mathbf{A_2} : \forall \mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathscr{V}, (\mathbf{x} + \mathbf{y}) + \mathbf{z} = \mathbf{x} + (\mathbf{y} + \mathbf{z})$ **Associativity**
- $\mathbf{A_3} : \emptyset \in \mathscr{V}, \forall \mathbf{x} \in \mathscr{V}, \mathbf{x} + \emptyset = \mathbf{x}$ **Neutral**
- $\mathbf{A_4} : \forall \mathbf{x} \in \mathscr{V}, \exists(-\mathbf{x}) \in \mathscr{V}, \mathbf{x} + (-\mathbf{x}) = \emptyset$ **Inverse**

- $\mathbf{SM_0} : \forall \alpha \in \mathscr{F}, \forall \mathbf{x} \in \mathscr{V}, \exists \alpha \cdot \mathbf{x} \in \mathscr{V}$ **Closure under Scalar Multiplication**
- $\mathbf{SM_1} : \forall \alpha, \beta \in \mathscr{F}, \exists \mathbf{x} \in \mathscr{V}, (\alpha \cdot \beta)\mathbf{x} = \alpha(\beta \cdot \mathbf{x})$ **Scalar Associativity**
- $\mathbf{SM_2} : \forall \alpha \in \mathscr{F}, \forall \mathbf{x}, \mathbf{y}, \in \mathscr{V}, \alpha(\mathbf{x} + \mathbf{y}) = \alpha \mathbf{x} + \alpha \mathbf{y}$ **Scalar-Vector Distributivity**
- $\mathbf{SM_3} : \forall \alpha, \beta \in \mathscr{F}, \forall \mathbf{x} \in \mathscr{V}, (\alpha + \beta)\mathbf{x} = \alpha \mathbf{x} + \beta \mathbf{x}$ **Vector-Scalar Distributivity**
- $\mathbf{SM_4} : \forall \mathbf{x} \in \mathscr{V}, 1 \cdot \mathbf{x} = \mathbf{x}$ **Neutral**

Usually denoted as $(\mathscr{X}, \mathscr{F})$ or $(\mathscr{V}, \mathscr{F})$

# Dot Product

Consider two vectors $\mathbf{a} = [a_1, a_2, ..., a_n]$ and $\mathbf{b} = [b_1, b_2, ..., b_n]$, the dot product of two vectors is defined as: $\mathbf{a} \cdot \mathbf{b} = \mathbf{a}^T \mathbf{b} = \sum_{i=1}^{n} a_i b_i = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$

Example: $[1, 2, -5]\,[4, -3, -1]^T = (1 \times 4) + (2 \times -3) + (-5 \times -1) = 4 - 6 + 5 = 3$

**Inner product** generalizes the dot product (which in in Euclidean spaces) to vector space of any dimensions. An inner product space is a vector space $\mathscr{V}$ over the field $\mathscr{F}$, and can be represented with a map

$$\langle \cdot, \cdot \rangle : \mathscr{V} \times \mathscr{V} \to \mathscr{F}$$

that satisfies three properties: conjugate symmetry $\langle \mathbf{x}, \mathbf{y} \rangle = \overline{\langle \mathbf{y}, \mathbf{x} \rangle}$, linearity in the first argument $\langle a\mathbf{x}, \mathbf{y} \rangle = a \langle \mathbf{x}, \mathbf{y} \rangle$ and positive-definite ($\langle \mathbf{x}, \mathbf{x} \rangle \geq 0$).
Usually denoted as $(\mathscr{X}, \mathscr{F}, \langle \cdot, \cdot \rangle)$

# Angle between Vectors and Orthogonal

Inspired from the geometric space, the **angle** between two vectors is defined as:

$$\angle(\mathbf{x}, \mathbf{y}) = \arccos \frac{\mathbf{x}^T \ \mathbf{y}}{\sqrt{\mathbf{x}^T\mathbf{x}}\sqrt{\mathbf{y}^T\mathbf{y}}}$$

For $\mathbf{x}, \mathbf{y} \in \mathscr{V}$, we say that $\mathbf{x}$ and $\mathbf{y}$ are **orthogonal** if $\angle(\mathbf{x}, \mathbf{y}) = 90^\circ$

$$\mathbf{x}^T\mathbf{y} = \cos(90^\circ) = 0$$

A set of vectors $\mathbf{x} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ is called **orthonormal** if

$$\begin{cases} \mathbf{x}_i^T\mathbf{x}_j = 0, \ \forall i \neq j \\ \mathbf{x}_i^T\mathbf{x}_i = 1, \ 1 \leqslant i \leqslant n \end{cases}$$

# $p$-norm

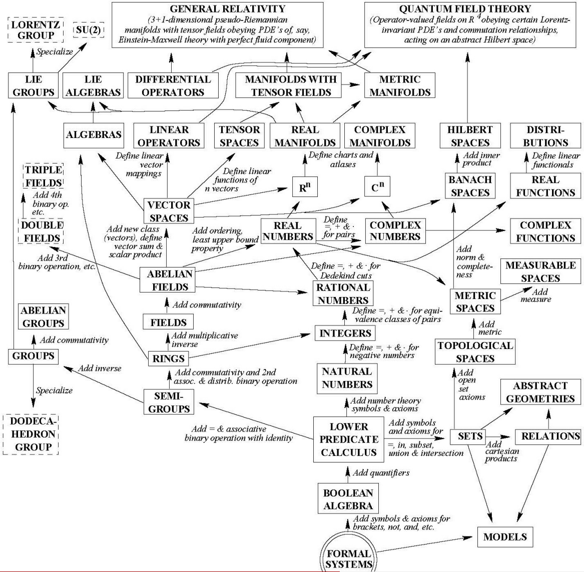
$\mathbf{x} = [x_1, \ldots, x_n]^T$

- $||\mathbf{x}||_2 = \left(\sum_{i=1}^{n} |x_i|^2\right)^{\frac{1}{2}} = \sqrt{\mathbf{x}^T\mathbf{x}}$ Euclidean norm - Distance
- $||\mathbf{x}||_p = (\sum_{i=1}^{n} |x_i|^p)^{\frac{1}{p}}$

- $||\mathbf{x}||_1 = \sum_{i=1}^{n} |x_i|$
- $||\mathbf{x}||_\infty \ = \ \max |x_i|$

You may interpret norm as the generalized linear space version of absolute value. It is an important concept because it is usually used as a measure of magnitude, which we will use extensively to describe the behaviors of a system, e.g. stability.

# Theory of Everything




Max Tegmark, "Is 'the theory of everything' merely the ultimate ensemble theory?", Annals of Physics, 1998

# Null Matrix and Unit Matrix

**Null Matrix**

The null matrix $\mathbf{0}$ is one that has all its elements equal to zero. The null matrix is however, not unique because the numbers of rows and columns it possesses can be any finite positive integers. Whenever necessary, a null matrix of size $m \times n$ is denoted by $\mathbf{0}_{mn}$.

- $\mathbf{A} + \mathbf{0} = \mathbf{0} + \mathbf{A} = \mathbf{A}$
- $\mathbf{0A} = \mathbf{A0} = \mathbf{0}$

Note: $\mathbf{AB} = \mathbf{0}$ does not imply that either $\mathbf{A}$ or $\mathbf{B}$ is a null matrix.

**Identity Matrix**

The identity or unit matrix $\mathbf{I}$ is a square matrix with elements on its diagonal ($i = j$positions) as ones and with all other elements as zeros. When necessary, an $n \times n$ unit matrix shall be denoted by $\mathbf{I}_n$.

- if $\mathbf{A}$ is $m \times n$, then $\mathbf{I}_m\mathbf{A} = \mathbf{A}$ and $\mathbf{AI}_n = \mathbf{A}$

# States

The concept of **state** occupies a central position in modern control theory. But what is **state**?

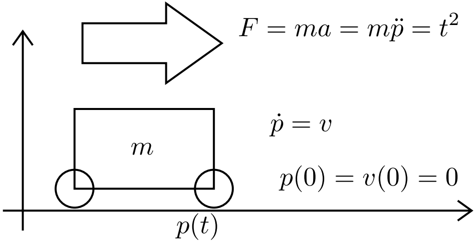
**State is a complete summary of the status of the system at a particular point in time**

- The state at any time $t_0$ is a set of the minimum number of parameters $x_i(t_0)$ which allows a unique output segment $Y_{[t_0,t]}$ to be associated with each input segment $\mathbf{u}_{[t_0,t]}$ for every $t_0 \in \mathscr{T}$ and for all $t > t_0$, $t \in \mathscr{T}$.

# What is the State

**State is a minimum complete summary of the status of the system at a particular point in time.**

Example: the State of Driving




$$F = ma = m\ddot{p} = t^2$$

$$\dot{p} = v$$

$$p(0) = v(0) = 0$$

$$x_1 = p \ , \ x_2 = \dot{p}, \ \mathbf{x} = [x_1, x_2]^T$$

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 1/m \end{bmatrix} F$$

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

Which is the state?:

$\{p, \dot{p}, \ddot{p}\}$, $\{\dot{p}, \ddot{p}\}$, $\{p, \dot{p}\}$, $\{p\}$, solve $p(t)$

# Our First Program

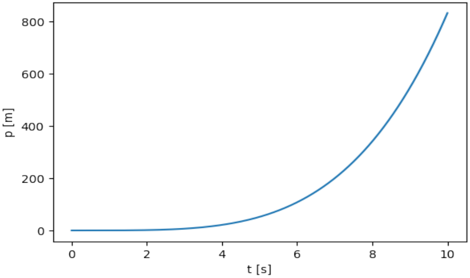
COLAB LINK

```
import numpy as np
from scipy.signal import StateSpace, lsim
import matplotlib.pyplot as plt

m = 1  # kg
A = np.asarray([[0., 1.],
                [0., 0.]])
B = np.asarray([[0.],
                [1. / m]])
C = np.asarray([[1., 0.]])
D = np.asarray([[0.]])
# define the continouse time linear system
cart_sys = StateSpace(A, B, C, D)
```

# In Python Code (cont'd)

```
# define simulation steps in time
t = np.arange(0, 10, 1e-3)
# define control input
F = t**2
# simulate the system
_, y, x = lsim(cart_sys, F, t, X0=[0., 0.])
# plot
plt.figure(dpi=100)
plt.plot(t, y)
plt.ylabel('p [m]')
plt.xlabel('t [s]')
plt.show()
```



# State Space Example

Find a state space model for $\ddot{y} = 7t^2\dot{y} + 2y + u$
Which is the state?: $\{y, \dot{y}, \ddot{y}\}$, $\{\dot{y}, \ddot{y}\}$, $\{y, \dot{y}\}$, $\{y\}$

Choose $x_1 = y$, $x_2 = \dot{y}$.
$2^{nd}$ order ODE to $1^{st}$ order ODE
$\dot{x}_1 = \dot{y} = x_2$
$\dot{x}_2 = \ddot{y} = 7t^2\dot{y} + 2y + u = 7t^2 x_2 + 2x_1 + u$
Now stack into a vector equation

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ 2x_1 + 7t^2 x_2 + u \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 2 & 7t^2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

For the output, want $y$, i.e

$$y = x_1 = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + 0 \cdot u$$

# The Choice of State is NOT Unique!

Choose $x_1 = 2y + 3\dot{y}$, $x_2 = \dot{y}, \Rightarrow y = (x_1 - 3x_2)/2$
Then,
$\dot{x}_1 = 2x_2 + 3\left(5t^2 x_2 + 3x_1 + u\right)$
$\dot{x}_2 = \ddot{y} = 5t^2\dot{y} + 3y + u = 5t^2 x_2 + 3(x_1 - 3x_2)/2 + u$
And,
$y = \frac{1}{2}x_1 - \frac{3}{2}x_2$
Hence, $\dot{\mathbf{x}} = \begin{bmatrix} 9 & 2 + 15t^2 \\ 3/2 & 5t^2 - 9/2 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 3 \\ 1 \end{bmatrix} u$
$\mathbf{y} = \begin{bmatrix} \frac{1}{2} & -\frac{3}{2} \end{bmatrix} \mathbf{x}$
- These both model the same system!
- Different choices of state variables will have difference advantages.

# Differential Equations to State Space - SISO System

$\frac{d^n y^a}{dt^n} + a_{n-1}\frac{d^{n-1}y}{dt^{n-1}} + \ldots + a_1\frac{dy}{dt} + a_0 y = b_0 u$

Define state variables: $x_1 = y, x_2 = \frac{dy}{dt}, x_n = \frac{dy^{n-1}}{dt^{n-1}}$

Then, $\dot{x}_1 = x_2, \dot{x}_2 = x_3, \cdots, \dot{x}_{n-1} = x_n$

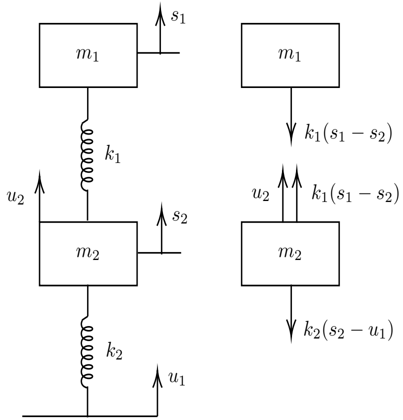
$\dot{x}_n = b_0 u - a_{n-1}x_{n-1} - a_{n-2}x_{n-2} - \cdots - a_0 x_1$

Write in matrix form

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

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0\ldots & 0 \end{bmatrix} \quad \mathbf{D} = 0$$

# Example: Automotive Suspension (Group Discussion)




Write the equations of motion in steady state

$m_1\ddot{s}_1 = -k_1\left(s_1 - s_2\right)$

$m_2\ddot{s}_2 = k_1\left(s_1 - s_2\right) - k_2\left(s_2 - u_1\right) + u_2$

Let $x_1 = s_1, x_2 = \dot{s}_1 \; x_3 = s_2, x_4 = \dot{s}_2 \Rightarrow$

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

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad D = 0 \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{k_2}{m_2} & \frac{1}{m_2} \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_1}{m_1} & 0 & \frac{k_1}{m_1} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{k_1}{m_2} & 0 & -\frac{k_1}{m_2} - \frac{k_2}{m_2} & 0 \end{bmatrix}$$

# Examples

Let's take a look at the last example and use the difference between positions as a state

$x_1' = s_1,\ x_2' = \dot{s_1},\ x_3' = s_1 - s_2,\ x_4' = \dot{s_1} - \dot{s_2},\ \mathbf{x}' = [x_1', x_2', x_3', x_4']^T$

Let $\mathbf{x} = \mathbf{M}\mathbf{x}', \dot{\mathbf{x}} = \mathbf{M}\dot{\mathbf{x}}' \Rightarrow \mathbf{M} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} = \mathbf{M}^{-1}$

Given $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \Rightarrow \mathbf{M}\dot{\mathbf{x}}' = \mathbf{A}\mathbf{M}\mathbf{x}' + \mathbf{B}\mathbf{u}$

$\Rightarrow \dot{\mathbf{x}}' = \mathbf{M}^{-1}\mathbf{A}\mathbf{M}\mathbf{x}' + \mathbf{M}^{-1}\mathbf{B}\mathbf{u}, \mathbf{y} = \mathbf{C}\mathbf{M}\mathbf{x}' + \mathbf{D}\mathbf{u}$

Let $\hat{\mathbf{A}} = \mathbf{M}^{-1}\mathbf{A}\mathbf{M}$ (**similarity transformation**), $\hat{\mathbf{B}} = \mathbf{M}^{-1}\mathbf{B}$, and $\hat{\mathbf{C}} = \mathbf{C}\mathbf{M}$

$\Rightarrow \hat{\mathbf{A}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -k_1/m_1 & 0 \\ 0 & 0 & 0 & 1 \\ k_2/m_2 & 0 & (-2k_1 - k_2)/m_2 & 0 \end{bmatrix} \hat{\mathbf{B}} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -k_2/m_2 & -1/m_2 \end{bmatrix}$

$\hat{\mathbf{C}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ Again, we can see the choice of state variables is not unique

# Summary of State Space Representation

- State space form writes high order and coupled DEQs as 1st order equations
- Many (infinite) different state equations represent the same DEQ
- State equations can be transformed

Motivation of using Linear Space Representations

- Can naturally deal with MIMO systems (Compare with "transfer function matrices". We will see later.)
- Provide a convenient, compact notation, and uniform representations.
- Allow the application of the powerful vector-matrix theory (you will see later in this class)
- An ideal format for computer programming. (Vectorization)

# Table of Contents

# Why Linearization

Ulam 1909-1984
Neumann 1903-1957
Metropolis 1915-1999

- Almost all real systems are nonlinear.
  *"Using a term like nonlinear science is like referring to the bulk of zoology as the study of non-elephant animals."* – Stanislaw Ulam
- In general, nonlinear dynamical systems are difficult to analyze. It is often advantageous to look for approximations to the complicated nonlinear systems.
- Under specific conditions, we can replace the nonlinear system with an approximate linear system. For example, we want to study whether small perturbations away from an equilibrium point of the nonlinear system grow or decay with time.

# Equilibrium Point

Consider the nonlinear time invariant system:

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}), f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$$

$\bar{\mathbf{x}} \in \mathbb{R}^n$ is an **equilibrium point** if

$$\exists \bar{\mathbf{u}} \in \mathbb{R}^m, \text{s.t.}\, f(\bar{\mathbf{x}}, \bar{\mathbf{u}}) = 0$$

i.e. $\dot{\mathbf{x}}(t) = 0$ at equilibrium points.

# Linearization

Jacobi, 1804-1851

Define the deviation variables:

$\delta_{\mathbf{x}} = \mathbf{x} - \bar{\mathbf{x}}, \delta_{\mathbf{u}} = \mathbf{u} - \bar{\mathbf{u}} \Rightarrow \mathbf{x} = \delta_{\mathbf{x}} + \bar{\mathbf{x}}, \mathbf{u} = \delta_{\mathbf{u}} + \bar{\mathbf{u}}, \dot{x} = \dot{\delta}_{\mathbf{x}}$

Taylor Series

$$\dot{x} = f(\bar{\mathbf{x}} + \delta_{\mathbf{x}}, \bar{\mathbf{u}} + \delta_{\mathbf{u}}) \approx f(\bar{\mathbf{x}}, \bar{\mathbf{u}}) + \frac{\partial f}{\partial \mathbf{x}}|_{\mathbf{x}=\bar{\mathbf{x}},\mathbf{u}=\bar{\mathbf{u}}}\delta_{\mathbf{x}} + \frac{\partial f}{\partial \mathbf{u}}|_{\mathbf{x}=\bar{\mathbf{x}},\mathbf{u}=\bar{\mathbf{u}}}\delta_{\mathbf{u}}$$

$$\dot{\delta}_{\mathbf{x}} = \frac{\partial f}{\partial \mathbf{x}}|_{\mathbf{x}=\bar{\mathbf{x}},\mathbf{u}=\bar{\mathbf{u}}}\delta_{\mathbf{x}} + \frac{\partial f}{\partial \mathbf{u}}|_{\mathbf{x}=\bar{\mathbf{x}},\mathbf{u}=\bar{\mathbf{u}}}\delta_{\mathbf{u}}$$

$$\mathbf{A} = \frac{\partial f}{\partial \mathbf{x}}|_{\mathbf{x}=\bar{\mathbf{x}},\mathbf{u}=\bar{\mathbf{u}}} \in \mathbb{R}^{n\times n}, \mathbf{B} = \frac{\partial f}{\partial \mathbf{u}}|_{\mathbf{x}=\bar{\mathbf{x}},\mathbf{u}=\bar{\mathbf{u}}} \in \mathbb{R}^{n\times m}, \frac{\partial f}{\partial \mathbf{x}} \text{ is called "Jacobian"}$$

# Example

$$\dot{\mathbf{x}} = \begin{bmatrix} x_1 - x_1^3 + x_1 x_2 \\ -x_2 + 2x_1 x_2 \end{bmatrix}$$

1. Find equilibria

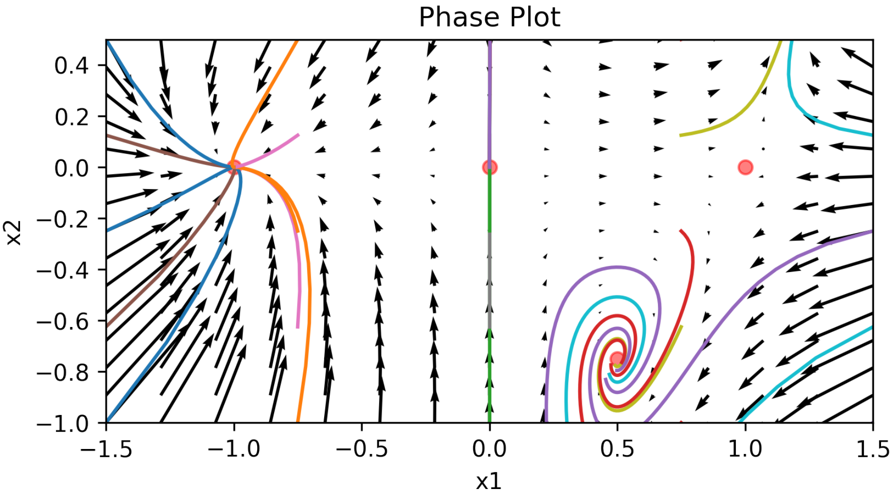
$$\begin{cases} x_1 - x_1^3 + x_1 x_2 = 0 \\ -x_2 + 2x_1 x_2 = 0 \end{cases} \quad \Rightarrow x_2(2x_1 - 1) = 0 \Rightarrow x_2 = 0 \text{ or } x_1 = \frac{1}{2}$$

$x_2 = 0 \Rightarrow x_1(1 - x_1^2) = 0 \Rightarrow x_1 = 0, x_1 = 1, x_1 = -1 \Rightarrow (0, 0), (1, 0), (-1, 0)$

$x_1 = \frac{1}{2} \Rightarrow \frac{1}{2} - \frac{1}{2^3} + \frac{1}{2}x_2 = 0 \Rightarrow x_2 = (-1 + \frac{1}{2^2}) \Rightarrow (\frac{1}{2}, -\frac{3}{4})$

# Phase Portrait Plot of the Nonlinear System

[COLAB LINK]



# Example

2. Linearization

$$\dot{\mathbf{x}} = \begin{bmatrix} x_1 - x_1^3 + x_1 x_2 \\ -x_2 + 2x_1 x_2 \end{bmatrix}, \mathbf{x} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} -1 \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{1}{2} \\ -\frac{3}{4} \end{bmatrix}$$

$$\frac{\partial f}{\partial \mathbf{x}} = \begin{bmatrix} 1 - 3x_1^2 + x_2 & x_1 \\ 2x_2 & 2x_1 - 1 \end{bmatrix}$$

$(0,0)$: $\dot{\delta}_{\mathbf{x}} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \delta_{\mathbf{x}}$

$(1,0)$: $\dot{\delta}_{\mathbf{x}} = \begin{bmatrix} -2 & 1 \\ 0 & 1 \end{bmatrix} \delta_{\mathbf{x}}$

$(-1,0)$: $\dot{\delta}_{\mathbf{x}} = \begin{bmatrix} -2 & -1 \\ 0 & -3 \end{bmatrix} \delta_{\mathbf{x}}$

$(\frac{1}{2}, -\frac{3}{4})$: $\dot{\delta}_{\mathbf{x}} = \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \\ -\frac{3}{2} & 0 \end{bmatrix} \delta_{\mathbf{x}}$

# Recap: State Space Equations

| | general (nonlinear) | linear |
|---|---|---|
| time-varying | $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, t)$ <br> $\mathbf{y} = g(\mathbf{x}, \mathbf{u}, t)$ | $\dot{\mathbf{x}} = \mathbf{A}(t)\mathbf{x} + \mathbf{B}(t)\mathbf{u}$ <br> $\mathbf{y} = \mathbf{C}(t)\mathbf{x} + \mathbf{D}(t)\mathbf{u}$ |
| time-invariant | $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$ <br> $\mathbf{y} = g(\mathbf{x}, \mathbf{u})$ | $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$ <br> $\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}$ |

where $\mathbf{u} \in \mathbb{R}^m$ is input, $\mathbf{x} \in \mathbb{R}^n$ is the states, and $\mathbf{y} \in \mathbb{R}^p$ is the output. In this course, we will focus on the linear SS problems.

# Recap: Linearization

Consider the nonlinear system:

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}), f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$$

$\bar{\mathbf{x}} \in \mathbb{R}^n$ is an **equilibrium point**:

$$\exists \bar{\mathbf{u}} \in \mathbb{R}^m, \text{s.t.} f(\bar{x}, \bar{\mathbf{u}}) = 0$$

Define the deviation variables:

$$\delta_{\mathbf{x}} = \mathbf{x} - \bar{\mathbf{x}}, \delta_{\mathbf{u}} = \mathbf{u} - \bar{\mathbf{u}} \Rightarrow x = \delta_{\mathbf{x}} + \bar{x}, \mathbf{u} = \delta_{\mathbf{u}} + \bar{\mathbf{u}}$$

$$\mathbf{A} = \frac{\partial f}{\partial \mathbf{x}}|_{\mathbf{x}=\bar{\mathbf{x}}, \mathbf{u}=\bar{\mathbf{u}}} \in \mathbb{R}^{n \times n}, \mathbf{B} = \frac{\partial f}{\partial \mathbf{u}}|_{\mathbf{x}=\bar{\mathbf{x}}, \mathbf{u}=\bar{\mathbf{u}}} \in \mathbb{R}^{n \times m}, \frac{\partial f}{\partial \mathbf{x}} \text{ is called "\textbf{Jacobian}"}$$