Typed Handout
ELEC60008/70089 - Control Engineering

Author: Dr. G. Scarciotti

Version: 0.1
Initial draft generated with Gemini 2.5 Pro
from a voice transcript of G. Scarciotti's delivery in 2024
https://www.youtube.com/watch?v=KZwEdeBw1ls

October 16, 2025

Contents

Chapter 1

Lecture 06/10: State-Space Models

1.1 Introduction to State Space

The traditional approach to control systems, often encountered in introductory courses, is based on the **transfer function**. A transfer function $G(s)$ describes the relationship between the Laplace transform of the input and the Laplace transform of the output of a system. While powerful, this approach is primarily suited for **Linear Time-Invariant (LTI)** systems that are **Single-Input, Single-Output (SISO)**.

Many real-world systems, however, are **Multiple-Input, Multiple-Output (MIMO)** and may be **nonlinear** or **time-varying**. The **state-space representation** is a more general and powerful mathematical framework for modeling such systems.

Definition 1.1.1 (Informal State-Space Model)**.** *A **state-space model** describes a dynamical system by a set of first-order differential (for continuous-time) or difference (for discrete-time) equations. It consists of:*

- *A **state vector**, $\mathbf{x}(t) \in \mathbb{R}^n$, which is the set of variables that, together with the input, completely determines the future behavior of the system. The state encapsulates the "memory" of the system; knowledge of the state at time t_0 and the input for $t \geq t_0$ is sufficient to determine the state and output for all $t \geq t_0$.*
- *A **state equation** that describes the evolution of the state vector over time.*
- *An **output equation** that relates the state vector and the input vector to the output vector.*

The key advantage of the state-space approach is its generality. It provides a unified framework for analyzing SISO and MIMO systems, linear and nonlinear systems, and time-invariant and time-varying systems.

1.1.1 Illustrative Examples

To understand the concept of state-space, let's explore a few examples.

Example 1.1.2 (Student Population Dynamics - Discrete Time)**.** Consider modeling the flow of students through a 3-year university program. We want to describe how the number of students in each year changes annually.

Variables:

- Let k be the time index representing the academic year (e.g., $k = 2025, 2026, \dots$).
- **State variables**:
  - $x_1(k)$: Number of students in Year 1 at the start of year k.
  - $x_2(k)$: Number of students in Year 2 at the start of year k.

– $x_3(k)$: Number of students in Year 3 at the start of year k.

The state vector is $\mathbf{x}(k) = \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \in \mathbb{R}^3$.

- **Input variable**: $u(k)$: Number of new students (intake) for year k. $\mathbf{u}(k) = u(k) \in \mathbb{R}$.
- **Output variable**: $y(k)$: Number of graduating students at the end of year k. $\mathbf{y}(k) = y(k) \in \mathbb{R}$.

Parameters:

- α_i: Promotion rate from Year i to Year $i+1$.
- $1 - \alpha_i$: Failure rate for Year i (students who repeat the year).

Model Equations: The number of students in each year for the next academic year, $k+1$, can be described as follows:

- **Year 1**: The students in Year 1 next year will be the new intake plus the students from this year's Year 1 who were not promoted.

$$x_1(k+1) = (1-\alpha_1)x_1(k) + u(k)$$

- **Year 2**: The students in Year 2 next year will be those promoted from Year 1 plus those from this year's Year 2 who were not promoted.

$$x_2(k+1) = \alpha_1 x_1(k) + (1-\alpha_2)x_2(k)$$

- **Year 3**: The students in Year 3 next year will be those promoted from Year 2 plus those from this year's Year 3 who were not promoted.

$$x_3(k+1) = \alpha_2 x_2(k) + (1-\alpha_3)x_3(k)$$

- **Output**: The graduating students are those successfully promoted from Year 3

$$y(k) = \alpha_3 x_3(k).$$

State-Space Matrix Form: These equations can be written in the standard LTI discrete-time state-space form

$$\begin{cases} \mathbf{x}(k+1) = A\mathbf{x}(k) + B\mathbf{u}(k) & \text{(State Equation)} \\ \mathbf{y}(k) = C\mathbf{x}(k) + D\mathbf{u}(k) & \text{(Output Equation)} \end{cases}$$

where

$$\underbrace{\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{bmatrix}}_{\mathbf{x}(k+1)} = \underbrace{\begin{bmatrix} 1-\alpha_1 & 0 & 0 \\ \alpha_1 & 1-\alpha_2 & 0 \\ 0 & \alpha_2 & 1-\alpha_3 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix}}_{\mathbf{x}(k)} + \underbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} u(k) \end{bmatrix}}_{\mathbf{u}(k)}$$

$$\underbrace{\begin{bmatrix} y(k) \end{bmatrix}}_{\mathbf{y}(k)} = \underbrace{\begin{bmatrix} 0 & 0 & \alpha_3 \end{bmatrix}}_{C} \underbrace{\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix}}_{\mathbf{x}(k)} + \underbrace{\begin{bmatrix} 0 \end{bmatrix}}_{D} \underbrace{\begin{bmatrix} u(k) \end{bmatrix}}_{\mathbf{u}(k)}.$$

The system can also be visualized as a state transition diagram (Figure 1.1).

Example 1.1.3 (Google PageRank Algorithm - Discrete Time)**.** The PageRank algorithm, which was foundational to Google's success, is a massive dynamical system. It assigns an "importance" score to every page on the World Wide Web.

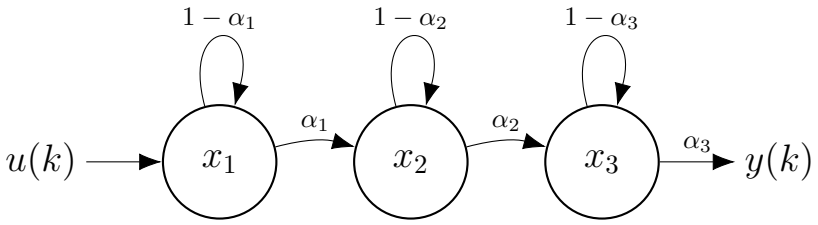

Figure 1.1: State transition diagram for the student population model.

The Model: The web is represented as a directed graph where pages are nodes and hyperlinks are edges. The rank of a page is determined by the ranks of the pages that link to it. The algorithm models a "random surfer" who either clicks a link on the current page or randomly "teleports" to any page on the web.

Variables and Parameters:

- Let N be the total number of pages on the web (in the billions).
- State vector $\mathbf{x}(k) \in \mathbb{R}^N$, where $x_i(k)$ is the PageRank score of page i at iteration k.
- Damping factor $p \in [0, 1]$ (typically $p \approx 0.15$): the probability that the surfer teleports to a random page.
- $1 - p$: the probability that the surfer follows an outgoing link.
- Let L_j be the number of outgoing links from page j.
- Let $\mathcal{N}_i$ be the set of pages that link to page i.

State Equation: The rank of page i at the next iteration, $k+1$, is a weighted sum of two components:

1. **Link Following**: For each page j that links to page i, there is a chance the surfer came from j. The rank of j, $x_j(k)$, is distributed evenly among its L_j outgoing links. The contribution from all incoming links is summed up. This component is weighted by $1 - p$.
2. **Teleportation**: The surfer lands on page i with uniform probability $1/N$. The rank of j, $x_j(k)$, is distributed evenly among all pages. This contributes $p \cdot \frac{1}{N} \cdot \sum_{j=1}^{N} x_j(k)$.

The update equation for a single page i is

$$x_i(k+1) = \frac{p}{N} \sum_{j=1}^{N} x_j(k) + (1-p) \sum_{j \in \mathcal{N}_i} \frac{x_j(k)}{L_j}.$$

This set of N equations can be written in matrix form

$$\mathbf{x}(k+1) = A\mathbf{x}(k).$$

The matrix A (the "Google matrix") is a combination of a modified adjacency matrix M and a teleportation matrix J:

$$A = (1-p)M + \frac{p}{N}J$$

where $M_{ij} = 1/L_j$ if page j links to page i (and 0 otherwise), and J is an $N \times N$ matrix of all ones.

Convergence: The matrix A is a special type of matrix (a column-stochastic, irreducible, and aperiodic matrix). The Perron-Frobenius theorem guarantees that:

1. A has a unique largest eigenvalue $\lambda_1 = 1$.
2. The corresponding eigenvector $\bar{\mathbf{x}}$ is strictly positive ($\forall i, \bar{x}_i > 0$).
3. For any initial state vector $\mathbf{x}(0)$ (e.g., uniform ranks), the system converges to this eigenvector: $\lim_{k \to \infty} \mathbf{x}(k) = \bar{\mathbf{x}}$.

The final PageRank vector $\bar{\mathbf{x}}$ is the steady-state solution to the eigenvalue problem $\bar{\mathbf{x}} = A\bar{\mathbf{x}}$. This means the algorithm finds a stable ranking irrespective of the starting point.

Example 1.1.4 (Predator-Prey Model - Continuous Time, Nonlinear)**.** The Lotka-Volterra equations model the population dynamics of two interacting species, such as rabbits (prey) and foxes (predators).

Variables:

- t: continuous time.
- State variables:
  - $x_1(t)$: Population density of prey (rabbits).
  - $x_2(t)$: Population density of predators (foxes).

  The state vector is $\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$.

Model Equations: The rate of change of each population is described by a differential equation:

- **Prey ($\dot{x}_1$)**: The prey population grows exponentially on its own (proportional to x_1) but decreases due to predation (proportional to the rate of encounters, $x_1 x_2$).

  $$\frac{dx_1}{dt} = \alpha x_1 - \beta x_1 x_2$$

  where α is the natural growth rate of prey and β is the predation rate.

- **Predators ($\dot{x}_2$)**: The predator population grows by consuming prey (proportional to $x_1 x_2$) and decreases due to natural death (proportional to x_2).

  $$\frac{dx_2}{dt} = \delta x_1 x_2 - \gamma x_2$$

  where δ is the rate of predator growth from predation and γ is the natural death rate of predators.

This system can be written in state-space form $\dot{\mathbf{x}} = f(\mathbf{x})$:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \alpha x_1 - \beta x_1 x_2 \\ \delta x_1 x_2 - \gamma x_2 \end{bmatrix} = f(\mathbf{x})$$

This is a **nonlinear** continuous-time system because the function $f(\mathbf{x})$ contains product terms like $x_1 x_2$.

Example 1.1.5 (COVID-19 Epidemiological Model - Continuous Time)**.** Compartmental models in epidemiology are excellent examples of state-space systems. A model published in *Nature* (2020) for the COVID-19 pandemic divided the population into several states.

States: The state vector $\mathbf{x}(t)$ could include proportions of the population in states such as:

- S: Susceptible
- I: Infected (asymptomatic, undetected)
- D: Diagnosed (asymptomatic, detected)
- A: Ailing (symptomatic, undetected)
- R: Recognized (symptomatic, detected)
- T: Threatened (acutely symptomatic, detected)
- H: Healed (recovered)
- E: Extinct (deceased)

The dynamics are described by a system of nonlinear ordinary differential equations (ODEs), $\dot{\mathbf{x}} = f(\mathbf{x})$, where each equation describes the flow of individuals between compartments (see Figure 1.2), namely

$$\begin{aligned} \tfrac{dS}{dt} &= -S(t)\left(\alpha I(t) + \beta D(t) + \gamma A(t) + \delta R(t)\right) \\ \tfrac{dI}{dt} &= S(t)\left(\alpha I(t) + \beta D(t) + \gamma A(t) + \delta R(t)\right) - (\varepsilon + \zeta + \lambda) I(t) \\ \tfrac{dD}{dt} &= \varepsilon I(t) - (\eta + \rho) D(t) \\ &\vdots \end{aligned} \tag{1.1}$$

where $\alpha, \beta, \gamma, \delta, \varepsilon, \zeta, \ldots$ are various rates (infection, detection, recovery, etc.) to go from one compartment to another. The full system consists of an ODE for each compartment, forming a high-dimensional, nonlinear state-space model.

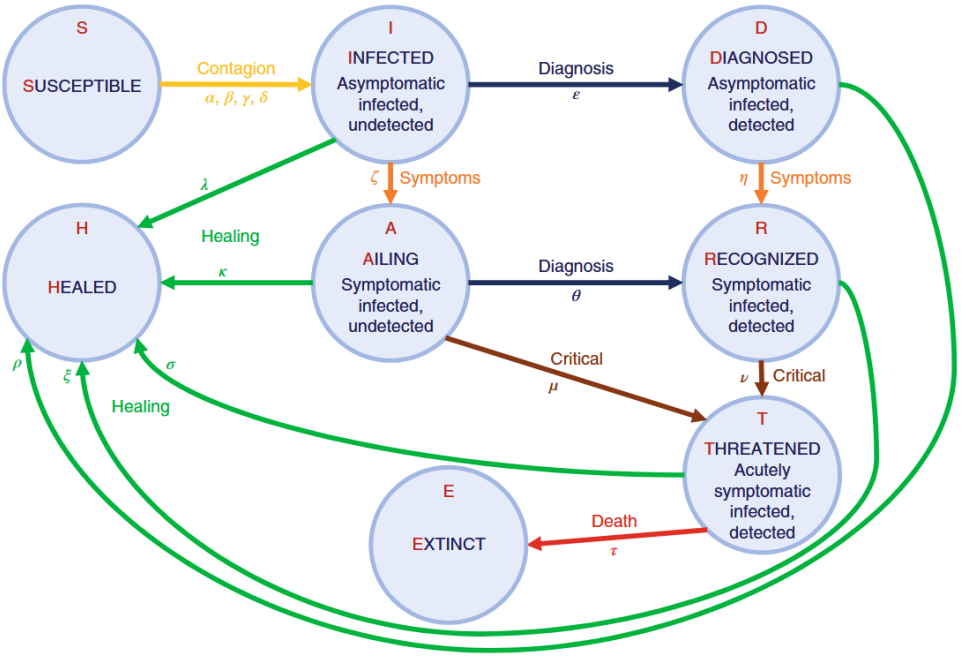

Figure 1.2: Flow diagram for a COVID-19 compartmental model (*Nature Medicine*, vol. 26, pages 855–860 (2020)).

1.2 Notation

We now formalize the notation used for state-space models throughout this course.

- $\mathbf{x} \in \mathbb{R}^n$: State vector ($n$ is the number of states, or system order).
- $\mathbf{u} \in \mathbb{R}^m$: Input or control vector (m is the number of inputs).
- $\mathbf{y} \in \mathbb{R}^p$: Output or measurement vector (p is the number of outputs).

1.2.1 Discrete-Time Systems

A discrete-time system evolves at distinct time steps $k \in \mathbb{Z}$.

Definition 1.2.1 (Discrete-Time System)**.** *A general discrete-time system is described by the equations*

$$\mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{u}(k)) \quad \textit{(State Equation)} \tag{1.2}$$

$$\mathbf{y}(k) = h(\mathbf{x}(k), \mathbf{u}(k)) \quad \textit{(Output Equation)} \tag{1.3}$$

where $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ *and* $h : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^p$.

A common shorthand notation is $\mathbf{x}^+ = f(\mathbf{x}, \mathbf{u})$, where $\mathbf{x}^+$ implies $\mathbf{x}(k+1)$. Given an initial state $\mathbf{x}(0)$ and an input sequence $\mathbf{u}(0), \mathbf{u}(1), \ldots$, the entire future trajectory of the state can be computed iteratively.

1.2.2 Continuous-Time Systems

A continuous-time system evolves continuously over time $t \in \mathbb{R}$.

Definition 1.2.2 (Continuous-Time System)**.** *A general continuous-time system is described by the equations*

$$\frac{d\mathbf{x}(t)}{dt} = f(\mathbf{x}(t), \mathbf{u}(t)) \quad \textit{(State Equation)} \tag{1.4}$$

$$\mathbf{y}(t) = h(\mathbf{x}(t), \mathbf{u}(t)) \quad \textit{(Output Equation)} \tag{1.5}$$

A common shorthand notation is $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$, where $\dot{\mathbf{x}}$ is the time derivative. Solving for the state trajectory $\mathbf{x}(t)$ requires integrating the differential equation.

1.2.3 Time-Invariant vs. Time-Varying Systems

Definition 1.2.3 (Time-Invariance)**.** *A system is **time-invariant** if the functions f and h do not explicitly depend on time. That is, $\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t))$. A system is **time-varying** if the functions f and h do explicitly depend on time, i.e., $\dot{\mathbf{x}}(t) = f(t, \mathbf{x}(t), \mathbf{u}(t))$.*

This course will primarily focus on time-invariant systems, which represent a vast and important class of physical systems.

1.2.4 Relation between continuous-time and discrete-time

Computers operate in discrete time. To simulate or control a continuous-time system with a digital computer, we must first discretize it. The most straightforward method is the **Forward Euler approximation**.

Given a continuous-time system $\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t))$, we can approximate the derivative over a small, constant sampling interval T_s

$$\frac{d\mathbf{x}(t)}{dt} \approx \frac{\mathbf{x}(t + T_s) - \mathbf{x}(t)}{T_s}$$

Let k be the discrete time index, such that $t_k = kT_s$. We define $\mathbf{x}(k) := \mathbf{x}(t_k)$. Then the approximation becomes

$$\frac{\mathbf{x}(k+1) - \mathbf{x}(k)}{T_s} \approx f(\mathbf{x}(k), \mathbf{u}(k)).$$

Rearranging gives the discrete-time state equation

$$\mathbf{x}(k+1) = \mathbf{x}(k) + T_s f(\mathbf{x}(k), \mathbf{u}(k)).$$

This is a discrete-time model that approximates the original continuous-time system. More sophisticated discretization methods exist (e.g., Runge-Kutta, Zero-Order Hold), but Euler's method illustrates the fundamental link.

1.3 A Formal Definition of State and State-Space

So far, we have used the term "state" intuitively. A formal definition is required for to make this concept completely clear. The core idea is that a **system** is a relationship between inputs and outputs. The **state** is the minimal internal information needed to summarize the entire past history of the system, such that this information, combined with future inputs, uniquely determines all future outputs.

1.3.1 System Response and State Transition

Consider a general continuous-time system. Its behavior can be captured by two functions.

1. **State Transition Function** or **State Response** (ϕ): This function describes how the state evolves over time.
2. **Output Function** or **Output Response** (η): This function maps the current state and input to the current output.

Definition 1.3.1 (System Response)**.** *The **state transition function**, or **state response**, ϕ computes the state of the system at time t, given an initial time t_0, an initial state $\mathbf{x}_0 = \mathbf{x}(t_0)$, and the input function $\mathbf{u}$ over the interval $[t_0, t]$. We denote this as*

$$\mathbf{x}(t) = \phi(t, t_0, \mathbf{x}_0, \mathbf{u}_{[t_0,t]})$$

*The **output response** η gives the output at time t as*

$$\mathbf{y}(t) = \eta(t, \mathbf{x}(t), \mathbf{u}(t)) = \eta(t, \phi(t, t_0, \mathbf{x}_0, \mathbf{u}_{[t_0,t]}), \mathbf{u}(t))$$

Formally, given a time set $\mathbb{T}$, sets of vectors $\mathbb{U}$, $\mathbb{X}$, $\mathbb{Y}$, and a set of input functions $\mathcal{U}$ with domain $\mathbb{T}$ and codomain $\mathbb{U}$, the functions ϕ and η are defined as

$$\phi : \mathbb{T} \times \mathbb{T} \times \mathbb{X} \times \mathcal{U} \to \mathbb{X} : (t, t_0, \mathbf{x}_0, \mathbf{u}_{[t_0,t]}) \mapsto \phi(t, t_0, \mathbf{x}_0, \mathbf{u}_{[t_0,t]})$$

$$\eta : \mathbb{T} \times \mathbb{X} \times \mathbb{U} \to \mathbb{Y} : (t, \mathbf{x}(t), \mathbf{u}(t)) \mapsto \eta(t, \mathbf{x}(t), \mathbf{u}(t)).$$

For the sets $\mathbb{T}$, $\mathbb{U}$, $\mathbb{X}$, $\mathbb{Y}$, $\mathcal{U}$, and functions ϕ and η to represent a valid state-space system, the function ϕ must satisfy certain axiomatic properties.

Axiomatic Properties of a State-Space System:

1. **Causality**: The state $\mathbf{x}(t)$ is determined by the initial state $\mathbf{x}(t_0)$ and the input history only from t_0 and up to time t, i.e., $\mathbf{u}_{[t_0,t]}$. It does not depend on past inputs ($\mathbf{u}(\tau)$ for $\tau < t_0$) and on future inputs ($\mathbf{u}(\tau)$ for $\tau > t$).
2. **Consistency**: If no time elapses, the state does not change. For any $t_0 \in \mathcal{T}$ and $\mathbf{x}_0 \in \mathbb{X}$

   $$\phi(t_0, t_0, \mathbf{x}_0, \cdot) = \mathbf{x}_0.$$

3. **Separation (Semigroup Property)**: The evolution from t_0 to t_2 can be broken down into intermediate steps. For any $t_0 \leq t_1 \leq t_2$:

   $$\mathbf{x}(t_2) = \phi(t_2, t_1, \mathbf{x}(t_1), \mathbf{u}_{[t_1,t_2]})$$

   where $\mathbf{x}(t_1) = \phi(t_1, t_0, \mathbf{x}_0, \mathbf{u}_{[t_0,t_1]})$. This is the crucial property of the state: the state at time t_1 fully encapsulates all the information from the past needed to predict the future. We don't need to know $\mathbf{x}_0$ or $\mathbf{u}_{[t_0,t_1]}$ once we know $\mathbf{x}(t_1)$.

Definition 1.3.2 (Formal State-Space Model)**.** *A **state-space model** is a mathematical construct represented by the tuple $(\mathbb{X}, \phi, \eta)$ where the functions ϕ and η satisfy the properties of causality, consistency, and separation. The set $\mathbb{X}$ is the **state space**, and its elements $\mathbf{x} \in \mathbb{X}$ are the **states**.*

Remark 1.3.3. The first-order differential/difference equations we use in this course, e.g.,

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t)), \quad \mathbf{y}(t) = h(\mathbf{x}(t), \mathbf{u}(t))$$

are a concrete realization of this abstract definition. The solution to the differential equation is precisely the state transition function ϕ. For our purposes, we will refer to this set of equations as the **state-space model** or **representation**.

1.3.2 From Higher-Order Models to State-Space Form

A key skill in state-space modeling is converting systems described by higher-order differential equations into the standard first-order state-space form. This is always possible.

Consider a general n-th order scalar differential equation:

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

where $y^{(k)}$ denotes the k-th derivative of the output y, and u is the input. The procedure to convert this to state-space form is as follows.

1. **Define State Variables**: Define n state variables starting from the output y and its successive derivatives, up to the $(n-1)$-th derivative.

$$\begin{aligned} x_1 &= y \\ x_2 &= \dot{y} = \dot{x}_1 \\ x_3 &= \ddot{y} = \dot{x}_2 \\ &\vdots \\ x_n &= y^{(n-1)} = \dot{x}_{n-1} \end{aligned}$$

2. **Write the State Equations**: Write the first-order differential equations for each state variable.
   - For the first $n-1$ states, the relationship is simply by definition:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ &\vdots \\ \dot{x}_{n-1} &= x_n \end{aligned}$$

   - For the last state, $\dot{x}_n$, we use the original n-th order differential equation

$$\dot{x}_n = y^{(n)} = -a_0 y - a_1 \dot{y} - \cdots - a_{n-1} y^{(n-1)} + u.$$

     Substituting our state variable definitions we obtain

$$\dot{x}_n = -a_0 x_1 - a_1 x_2 - \cdots - a_{n-1} x_n + u.$$

3. **Write the Output Equation**: The output is typically one of the state variables. In this standard choice, the output y is simply x_1

$$y = x_1.$$

4. **Assemble into Matrix Form**: Collect these equations into the matrix form $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$ and $\mathbf{y} = C\mathbf{x} + D\mathbf{u}$.

$$\underbrace{\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_{n-1} \\ \dot{x}_n \end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -a_0 & -a_1 & -a_2 & \dots & -a_{n-1} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} u \end{bmatrix}}_{\mathbf{u}}$$

$$\underbrace{\begin{bmatrix} y \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}}_{C} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} 0 \end{bmatrix}}_{D} \underbrace{\begin{bmatrix} u \end{bmatrix}}_{\mathbf{u}}$$

   This specific structure for matrix A is known as the **companion form**.

1.3.3 Example: Coupled Mass-Spring-Damper System

Let's apply this method to a more complex MIMO system. Consider two masses, m_1 and m_2, coupled by springs and dampers as in Fig. 1.3.

The equations of motion, derived from Newton's second law, are given as:

$$M_1 \ddot{y}_1 + (B + B_1)\dot{y}_1 - B\dot{y}_2 + (K + K_1)y_1 - Ky_2 = F_1 \tag{1.6}$$

$$M_2 \ddot{y}_2 + (B + B_2)\dot{y}_2 - B_1\dot{y}_1 + (K + K_2)y_2 - Ky_1 = -F_2. \tag{1.7}$$

This is a system of two coupled second-order ODEs. The system has 4 degrees of freedom that need to be captured in the state (2 positions, 2 velocities).

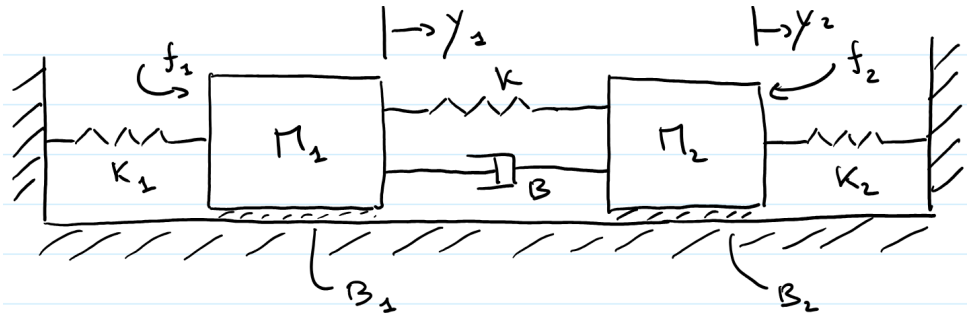

Figure 1.3: A coupled two-mass system.

1. Define State Variables: We choose the positions and velocities as our state variables.

$$\begin{aligned} x_1 &= y_1 && \text{(position of mass 1)} \\ x_2 &= \dot{y}_1 && \text{(velocity of mass 1)} \\ x_3 &= y_2 && \text{(position of mass 2)} \\ x_4 &= \dot{y}_2 && \text{(velocity of mass 2)} \end{aligned}$$

The state vector is $\mathbf{x} = [x_1, x_2, x_3, x_4]^T$. The input vector is $\mathbf{u} = [F_1, F_2]^T$.

2. Write State Equations:

- $\dot{x}_1 = \dot{y}_1 = x_2$
- $\dot{x}_3 = \dot{y}_2 = x_4$
- For $\dot{x}_2 = \ddot{y}_1$, we rearrange the first equation of motion:

$$\ddot{y}_1 = \frac{1}{M_1}\left(-(K + K_1)y_1 - (B + B_1)\dot{y}_1 + Ky_2 + B\dot{y}_2 + F_1\right)$$

  Substituting state variables:

$$\dot{x}_2 = \frac{1}{M_1}\left(-(K + K_1)x_1 - (B + B_1)x_2 + Kx_3 + Bx_4\right) + \frac{1}{M_1}F_1$$

- For $\dot{x}_4 = \ddot{y}_2$, we rearrange the second equation of motion:

$$\ddot{y}_2 = \frac{1}{M_2}\left(Ky_1 + B\dot{y}_1 - (K + K_2)y_2 - (B + B_2)\dot{y}_2 + F_2\right)$$

  Substituting state variables:

$$\dot{x}_4 = \frac{1}{M_2}\left(Kx_1 + Bx_2 - (K + K_2)x_3 - (B + B_2)x_4\right) + \frac{1}{M_2}F_2$$

3. Assemble into Matrix Form: The resulting state-space model $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$ is:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{K+K_1}{M_1} & -\frac{B+B_1}{M_1} & \frac{K}{M_1} & \frac{B}{M_1} \\ 0 & 0 & 0 & 1 \\ \frac{K}{M_2} & \frac{B}{M_2} & -\frac{K+K_2}{M_2} & -\frac{B+B_2}{M_2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{1}{M_1} & 0 \\ 0 & 0 \\ 0 & \frac{1}{M_2} \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}$$

If our outputs of interest are the positions, $\mathbf{y} = [y_1, y_2]^T$, then the output equation is:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{C} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}}_{D} \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}$$

Remark 1.3.4 (Non-Uniqueness of State-Space Representations)**.** An important concept is that the choice of state variables is not unique. For the same physical system, we can choose different (but valid) sets of state variables. For example, we could have defined the state variables for the mass-spring system in a different order (e.g. $x_1 = y_2$), or as linear combinations of the positions and velocities (e.g. $x_1 = y_1 + y_2$). This would result in a different set of matrices (A, B, C, D), but the new representation would describe the exact same physical system. All valid state-space representations of a given system are related by a linear transformation, called a **similarity transformation**. While there are infinitely many state-space representations for a single LTI system, they all share the same **transfer function**. The transfer function is a unique characteristic of the system's input-output behavior. We will explore this connection in more detail later in the course.

1.4 Conclusion

This lecture introduced the state-space representation as a versatile and powerful tool for modeling dynamical systems, overcoming the limitations of the transfer function approach. Through diverse examples, we have seen its applicability to discrete-time, continuous-time, linear, and nonlinear systems.

1.5 Homework

Exercise 1.5.1. Consider Fig. 1.2 in Example 1.1.5. Derive the rest of the equations to complete the state-space model in (1.1).

Exercise 1.5.2. A satellite of mass m is orbiting a planet. Its position is described by its radial distance r and angle θ. The system is subject to control forces u_1 (radial) and u_2 (tangential). The nonlinear equations of motion are given by

$$m(\ddot{r} - r\dot{\theta}^2) = -\frac{k}{r^2} + u_1$$
$$m(r\ddot{\theta} + 2\dot{r}\dot{\theta}) = u_2$$

where k is a gravitational constant.

Task: Write this system in the general state-space form $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$.

1. Identify the necessary state variables. (Hint: you will need four).
2. Write the four first-order differential equations for the state variables.
3. Express the final system as $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$, identifying the state vector $\mathbf{x}$, input vector $\mathbf{u}$, and the nonlinear function f.

Chapter 2

Lecture 10/10: Existence and Uniqueness of Solutions

In Section 1.3, we have seen that continuous-time and discrete-time systems can be described by differential equations, namely

$$\dot{x} = f(t, x, u) \qquad y = h(t, x, u) \tag{2.1}$$

and difference equations

$$x^+ = f(t, x, u) \qquad y = h(t, x, u), \tag{2.2}$$

where all signals have to be understood as evaluated at time t, and $t \in \mathbb{R}$ if the system is continuous-time, whereas $t \in \mathbb{Z}$ if the system is discrete-time. In what follows, whenever convenient and for compactness, we also use the notation

$$\sigma x = f(t, x, u) \qquad y = h(t, x, u), \tag{2.3}$$

where σx stands for $\dot{x}$ if the system is continuous-time, and σx stands for x^+ if the system is discrete-time.

2.1 Existence and Uniqueness of Solutions

For a given state-space model to be a useful representation of a physical system, we must be confident that its mathematical solution is well-behaved.

The simplest question that can be posed in the study of the equations (2.1) and (2.2) is the following.

Given an initial time t_0, an initial value of the state $x(t_0) = x_0$ and an input signal $u \in \mathcal{U}$, is it possible to obtain a *solution* of the equation (2.3)? By a solution we mean a function $x(t)$, defined for all $t \geq t_0$, and such that

$$\sigma x(t) = f(t, x(t), u(t))$$

for all $t \in \mathbb{T}$, or for all $t \in [t_0, \bar{t}) \subset \mathbb{T}$, for some $\bar{t} > t_0$.

The answer to this question is trivial in the case of discrete-time systems, provided the state set $\mathbb{X}$ coincides with $\mathbb{R}^n$ and the function f is continuous. In fact, if the function f is continuous and the input signal $u(t)$ is bounded for all finite $t \in \mathbb{T}$, then equation (2.2) describes a continuous mapping from $\mathbb{T} \times \mathbb{R}^n \times \mathbb{U}$ to $\mathbb{R}^n$, hence the solution $x(t)$ is unique and it is such that $x(t) \subset \mathbb{X} = \mathbb{R}^n$ for all finite $t \geq t_0$. Note however that, if the function f is not continuous, or if the state set $\mathbb{X}$ is not $\mathbb{R}^n$, then solutions of the equation (2.2) may not be defined for all $t \geq t_0$.

In the case of continuous-time systems the situation is much more involved, and continuity of f is not enough to guarantee existence and uniqueness of solutions of equation (2.1).

Thus, for continuous-time systems, this leads to two fundamental questions:

1. **Existence**: Does a solution $\mathbf{x}(t)$ exist for all time $t \geq t_0$?
2. **Uniqueness**: If a solution exists, is it the only one for a given initial condition?

Failure on either count suggests a potential flaw in the model or indicates a system with pathological behavior.

Remark 2.1.1. We stress again that these problems are specific to continuous-time systems. For a discrete-time system $\mathbf{x}(k+1) = f(\mathbf{x}(k))$, as long as the function f is continuous, a unique next state $\mathbf{x}(k+1)$ can always be computed from the current state $\mathbf{x}(k)$. The solution is generated by simple iteration, so existence and uniqueness are guaranteed.

Example 2.1.2 (Non-Existence for All Time - Finite Escape Time)**.** Consider the simple nonlinear ODE:

$$\dot{x} = x^3, \quad \text{with initial condition } x(0) = x_0 > 0$$

We solve this using separation of variables:

$$\begin{aligned}
\frac{dx}{x^3} &= dt \\
\int_{x_0}^{x(t)} s^{-3} ds &= \int_0^t d\tau \\
\left[-\frac{1}{2s^2}\right]_{x_0}^{x(t)} &= t \\
-\frac{1}{2x(t)^2} + \frac{1}{2x_0^2} &= t \\
x(t)^2 &= \frac{x_0^2}{1 - 2tx_0^2} \\
x(t) &= \frac{x_0}{\sqrt{1 - 2tx_0^2}}.
\end{aligned}$$

The solution $x(t)$ goes to infinity as the term in the denominator approaches zero. This occurs at the **finite escape time** $t_{esc} = \frac{1}{2x_0^2}$. The solution exists only for $t \in [0, t_{esc})$. For $t \geq t_{esc}$, the system "blows up" and a solution no longer exists.

Example 2.1.3 (Non-Uniqueness of Solutions)**.** Consider the ODE

$$\dot{x} = x^{1/3}, \quad \text{with initial condition } x(0) = 0$$

We can identify at least two distinct solutions that satisfy this initial value problem.

1. **The trivial solution**: $x(t) = 0$ for all $t \geq 0$. We can verify this: $\dot{x}(t) = 0$, and $x(t)^{1/3} = 0^{1/3} = 0$. The equation holds.
2. **A non-trivial solution**: We solve via separation of variables.

$$\begin{aligned}
\frac{dx}{x^{1/3}} &= dt \\
\int_0^{x(t)} s^{-1/3} ds &= \int_0^t d\tau \\
\left[\frac{3}{2} s^{2/3}\right]_0^{x(t)} &= t \\
\frac{3}{2} x(t)^{2/3} &= t \\
x(t) &= \left(\frac{2}{3} t\right)^{3/2} \quad \text{for } t \geq 0.
\end{aligned}$$

   This also satisfies the initial condition and the ODE.

Since there are at least two valid solutions, the solution is not unique. This is problematic for a model supposed to predict a system's behavior. The issue arises because the function $f(x) = x^{1/3}$ has an infinite slope at $x = 0$.

In general the equation $\dot{x} = f(x, u)$ may exhibit the following behaviours.

- **No solution exists.** For a particular initial condition x_0, the equation provides no valid path forward. The rules governing the system are contradictory at that specific point, making it impossible for a solution to even begin.
- **Multiple distinct solutions exist.** From a single initial condition x_0, the system can evolve along several different paths. The future is not uniquely determined by the initial state.
- **A unique solution exists, but stops at a finite time.** From the initial state x_0, there is one and only one path the system can take. However, this path does not continue forever. At a specific future time, the solution "blows up" (e.g., goes to infinity) or hits a boundary where it cannot continue.
- **A unique solution exists for all time.** This is the most well-behaved case. From the initial state x_0, there is one and only one path, and this path continues indefinitely into the future.

Remark 2.1.4. We now discretize the equations $\dot{x} = x^3$ and $\dot{x}^{\frac{1}{3}}$, obtaining with the Euler forward rule the models $x^+ = x + Tx^3$ and $x^+ = x + Tx^{\frac{1}{3}}$, respectively. In both cases, these discrete-time systems have a unique solution that exists for all times. This shows two key features of discrete-time systems.

- Existence and uniqueness are not issues for discrete-time systems. The system $x^+ = f(x, u, k)$ gives always a unique solution for any x_0, $u(k)$ and $k \geq k_0$, as long as f is well-defined.
- Consequently, any discrete-time model of a continuous-time system may not reveal all the properties of the underlying continuous time system. Next time you simulate something, keep in mind that your computer may be lying to you.

2.1.1 A Sufficient Condition: Lipschitz Continuity

The pathological behaviors in the examples above were caused by the function $f(x)$ growing too rapidly or having an infinite derivative. To guarantee well-behaved solutions, we need to impose a condition that limits how fast $f(x)$ can change. This condition is known as Lipschitz continuity.

Definition 2.1.5 (Lipschitz Continuity)**.** *A function $f : \mathbb{R}^n \to \mathbb{R}^p$ is said to be **(globally) Lipschitz continuous** if there exists a constant $L \geq 0$, called the **Lipschitz constant**, such that for all $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^n$:*

$$\|f(\mathbf{x}_1) - f(\mathbf{x}_2)\| \leq L\|\mathbf{x}_1 - \mathbf{x}_2\|$$

where $\|\cdot\|$ is any vector norm.

Intuition: The Lipschitz condition essentially bounds the "steepness" of the function f. It ensures that the slope of any secant line on the graph of f is bounded by L. If f is continuously differentiable, a sufficient condition for it to be Lipschitz is that its Jacobian matrix is bounded, that is

$$\left\| \frac{\partial f}{\partial \mathbf{x}} \right\| \leq L \quad \text{for all } \mathbf{x}.$$

The fundamental result that connects this property to our problem is the Picard-Lindelöf theorem.

Theorem 2.1.6 (Existence and Uniqueness)**.** *Consider the initial value problem $\dot{\mathbf{x}} = f(t, \mathbf{x})$ with $\mathbf{x}(t_0) = \mathbf{x}_0$. If the function f is continuous in t and (globally) Lipschitz continuous in $\mathbf{x}$, then there exists a **unique** solution $\mathbf{x}(t)$ for all $t \geq t_0$.*

Proof: See https://en.wikipedia.org/wiki/Picard%E2%80%93Lindel%C3%B6f_theorem.

2.1.2 Application to Linear Systems

Consider the following linear time-varying system (LTV)

$$\dot{\mathbf{x}} = f(\mathbf{x}, t) = A(t)\mathbf{x} + g(t)$$

where $\mathbf{x}(t) \in \mathbb{R}^n$, $A(t) \in \mathbb{R}^{n \times n}$, and $g(t) \in \mathbb{R}^n$. Assume that all entries of the matrix $A(t)$ are continuous and bounded functions of time. Bounded means that there exists $\bar{A}$ such that $||A(t)|| \leq \bar{A}$ for all times.

Let us check if this function is Lipschitz continuous.

$$\begin{aligned}\|f(\mathbf{x}_1,t)-f(\mathbf{x}_2,t)\| &= \|A(t)\mathbf{x}_1+g(t)-A(t)\mathbf{x}_2-g(t)\| = \|A(t)(\mathbf{x}_1-\mathbf{x}_2)\| \\ &\le \|\bar{A}(\mathbf{x}_1-\mathbf{x}_2)\| \quad \text{(by the boundedness of } A(t)\text{)} \\ &\le \|\bar{A}\|\cdot\|\mathbf{x}_1-\mathbf{x}_2\| \quad \text{(by property of induced matrix norms)}\end{aligned}$$

This inequality holds for all $\mathbf{x}_1, \mathbf{x}_2$. We have shown that $f(\mathbf{x},t) = A(t)\mathbf{x} + g(t)$ is globally Lipschitz continuous with a Lipschitz constant $L = \|\bar{A}\|$ (where $\|\bar{A}\|$ is the induced norm of the matrix A).

Remark 2.1.7 (Key Takeaway)**. For $A(t) \equiv A$ constant, all systems of the form $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$ (called linear time-invariant (LTI)) are guaranteed to have a unique solution that exists for all time.** This is a cornerstone of linear systems theory. It means we do not have to worry about the mathematical pathologies of non-existence or non-uniqueness when dealing with the LTI models that form the bulk of this course.

Recall: Matrix Norm

A function $\|\cdot\| : \mathbb{R}^{n\times m} \to \mathbb{R}$ is a **matrix norm** if for any matrices $A, B \in \mathbb{R}^{n\times m}$ and any scalar $\alpha \in \mathbb{R}$, it satisfies the following properties:

(i) **Non-negativity**: $\|A\| \ge 0$.
(ii) **Definiteness**: $\|A\| = 0$ if and only if A is the zero matrix.
(iii) **Homogeneity**: $\|\alpha A\| = |\alpha|\|A\|$.
(iv) **Triangle Inequality**: $\|A+B\| \le \|A\| + \|B\|$.

Furthermore, for matrices where multiplication is defined (e.g., square matrices), we often require the **sub-multiplicative property**:

(v) **Sub-multiplicativity**: $\|AB\| \le \|A\|\|B\|$.

This is the property we used to show that linear systems are Lipschitz continuous.

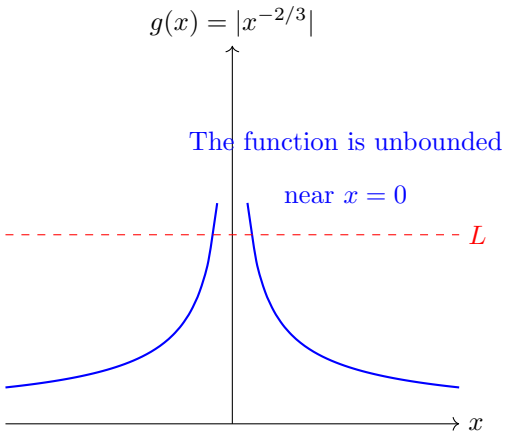

Figure 2.1: The function $|x^{-2/3}|$ is unbounded near the origin. No finite constant L can serve as an upper bound.

2.1.3 Revisiting Example 2.1.3: Failure of the Lipschitz Condition

Let's revisit the example where the solution was not unique:

$$\dot{x} = x^{1/3} = f(x), \quad x(0) = 0$$

To check if $f(x) = x^{1/3}$ is Lipschitz continuous, we must find a constant $L \ge 0$ such that for all $x, y \in \mathbb{R}$:

$$|f(x) - f(y)| \le L|x - y|$$

Since this must hold for *all* x and y, let's test it for the case where $y = 0$. The condition becomes:

$$|x^{1/3} - 0^{1/3}| \le L|x - 0| \implies |x^{1/3}| \le L|x|$$

For $x \neq 0$, we can divide by $|x|$:

$$\frac{|x^{1/3}|}{|x|} \leq L \implies |x^{-2/3}| \leq L$$

The question now is: can we find a constant L that is greater than or equal to $|x^{-2/3}|$ for all $x \in \mathbb{R}$? Let's examine the function $g(x) = |x^{-2/3}|$. As x approaches 0, $g(x)$ approaches infinity (see Figure 2.1). Since the function is unbounded, no such constant L exists. Therefore, $f(x) = x^{1/3}$ is **not** Lipschitz continuous around the origin. The failure of this condition is consistent with our earlier finding that the system's solution is not unique.

Remark 2.1.8. The Lipschitz condition is **sufficient but not necessary**. A system might fail the Lipschitz test but still have a unique solution. However, if the condition holds, uniqueness and existence are guaranteed.

2.2 Conclusion

This lecture addressed the critical issue of existence and uniqueness of solutions for continuous-time systems. While nonlinear systems can exhibit complex and sometimes pathological behavior, we have established that the linear systems that will be the focus of our study are inherently well-behaved, always possessing a unique solution for all time. This provides a solid mathematical foundation for the analysis and control techniques we will develop in the upcoming lectures.

Chapter 3

Lecture 13/10: Qualitative Behaviour and Linear Systems Fundamentals

3.1 Qualitative Behaviour of Dynamical Systems

The concepts of this sections are illustrated in the video titled "Qualitative Behaviour of Dynamical Systems" available on BlackBoard.

To understand how systems evolve, we use several graphical and conceptual tools. We consider again a system described by means of an input-state-output representation $\{\mathbb{X}, \phi, \eta\}$ and define a few typical dynamic behaviours of the system.

A *time history* is the set of points visited by $\{t, x_i(t)\}$ as a function of time starting from $x(0)$. For a system with state $x(t) = [x_1(t), x_2(t), x_3(t)]^T$, the time histories can be represented on separate plots of $x_1(t)$ vs t, $x_2(t)$ vs t, and $x_3(t)$ vs t.

For continuous-time systems, these are continuous curves. For discrete-time systems $x(k)$, they are sequences of points plotted against the discrete time index k.

Definition 3.1.1 (Trajectory)**.** *Consider a state-space model $\{\mathbb{X}, \phi, \eta\}$. A trajectory is the set*

$$\mathcal{T} = \{x \in \mathbb{X} \ : \ x = \phi(t, t_0, x_0, u)\} \subset \mathbb{X},$$

i.e. is the set of points in $\mathbb{X}$ reached by the state $x(t)$, for $t \geq t_0$, and for a specific initial state x_0 and input signal u.

For systems $\dot{x} = f(x)$ where f is sufficiently smooth (e.g., Lipschitz continuous, guaranteeing uniqueness of solutions), distinct trajectories *do not cross*. They can approach each other or converge to the same equilibrium point, but cannot intersect at a finite time if they start from different initial conditions.

For systems $x^+ = f(x)$, trajectories $x(k)$ are sequences of points. The lines sometimes drawn between points are merely visual aids and do not represent the system's state between samples. Differently from continuous-time systems, for discrete-time systems distinct trajectories can land on the same point after a finite number of steps (e.g., a matrix $A^k = 0$ for some finite k, known as a nilpotent matrix in discrete time, will drive the state to zero in a finite number of iterations).

A *phase portrait* is a collection of representative trajectories in the phase plane (for 2D systems) or state space (for higher dimensions, though hard to visualize beyond 3D). It provides a global view of the system's behavior from various initial conditions. Phase portraits can reveal equilibrium points, limit cycles, regions of stability/instability, etc.

Definition 3.1.2 (Motion)**.** *Consider a state-space model $\{\mathbb{X}, \phi, \eta\}$. A motion is the set*

$$\mathcal{M} = \{(t, x(t)) \in \mathbb{T} \times \mathbb{X} \ : \ x(t) = \phi(t, t_0, x_0, u)\} \subset \mathbb{T} \times \mathbb{X},$$

i.e. is the set of points in $\mathbb{T} \times \mathbb{X}$ taken by the pairs $(t, x(t))$, for $t \geq t_0$, and for a specific initial state x_0 and input signal u.

The main differences between a trajectory and a motion are that they leave in different spaces, and the motion is parameterized by t, whereas the trajectory does not contain any information on t. This means that the trajectory provides solely information on the points of the state space $\mathbb{X}$ visited by the system during his evolution, whereas the motion specifies in addition when each point has been visited. Note, however, that the (natural) projection of a motion along $\mathbb{T}$ yields a trajectory.

Definition 3.1.3 (Equilibrium)**.** *Consider a state-space model $\{\mathbb{X}, \phi, \eta\}$. Assume the input u is constant, i.e. $u(t) = u_0$ for all t and for some constant u_0. A state x_e is an equilibrium of the system associated to the input u_0 if*

$$x_e = \phi(t, t_0, x_e, u_0),$$

for all $t \geq t_0$, i.e. an equilibrium is a trajectory composed of a single point. In terms of motion, it's a vertical line in the (x, t)-space.

If the state-space model $\{\mathbb{X}, \phi, \eta\}$ can be described by means of the triple $\{\mathbb{X}, f, \eta\}$, then the computation of equilibria requires the solution of the system of equations

$$0 = f(t, x_e, u_0),$$

for continuous-time systems, and of the systems of equations

$$x_e = f(t, x_e, u_0),$$

for discrete-time systems.

Example 3.1.4 (Simple Pendulum)**.** A simple pendulum of mass m and length L subject to gravity g and an external torque U_{ext} can be described by $\ddot{\theta} = -\frac{g}{L}\sin\theta + \frac{1}{mL^2}U_{ext}$. Let $u = \frac{1}{mL^2}U_{ext}$ be the normalized input. The state equations with $x_1 = \theta, x_2 = \dot{\theta}$ are:

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{g}{L}\sin x_1 + u.\end{aligned}$$

To find equilibrium points for a constant input $u = u_{bar}$: Set $\dot{x}_1 = 0 \implies x_{2e} = 0$. Set $\dot{x}_2 = 0 \implies -\frac{g}{L}\sin x_{1e} + u_{bar} = 0$. This gives $\sin x_{1e} = \frac{L}{g}u_{bar}$.

The equilibrium points are $(x_{1e}, 0)$ where x_{1e} satisfies $\sin x_{1e} = \frac{Lu_{bar}}{g}$.

- If $|Lu_{bar}/g| > 1$: No real solutions for x_{1e}, so no equilibrium points. The torque is too large, causing continuous rotation.
- If $|Lu_{bar}/g| \leq 1$: There are solutions for x_{1e}. Let $K = Lu_{bar}/g$. Then $x_{1e} = \arcsin(K) + 2n\pi$ and $x_{1e} = \pi - \arcsin(K) + 2n\pi$ for $n \in \mathbb{Z}$.
  - For $u_{bar} = 0$: $\sin x_{1e} = 0 \implies x_{1e} = n\pi$. Equilibrium points are $(n\pi, 0)$. The points $(0, 0), (\pm 2\pi, 0), \ldots$ correspond to the pendulum hanging down (stable equilibria - defined later). The points $(\pm\pi, 0), (\pm 3\pi, 0), \ldots$ correspond to the pendulum balanced upwards (unstable equilibria - defined later).
  - For $0 < |Lu_{bar}/g| < 1$: Two distinct sets of solutions for x_{1e} within each 2π interval. For example, if $u_{bar} = g/(2L)$, then $\sin x_{1e} = 1/2$, so $x_{1e} = \pi/6 + 2n\pi$ and $x_{1e} = 5\pi/6 + 2n\pi$. One corresponds to a stable, deflected position, the other to an unstable one.
  - For $|Lu_{bar}/g| = 1$: Solutions merge, e.g., if $u_{bar} = g/L$, $\sin x_{1e} = 1 \implies x_{1e} = \pi/2 + 2n\pi$. This is the horizontal position.

Graphically, one can plot $y = \sin x_1$ and $y = \frac{L}{g}u_{bar}$ and find their intersections. Alternatively, plot $y = \frac{g}{L}\sin x_1$ and $y = u_{bar}$. The points of intersection give x_{1e}.

Example 3.1.5 (Predator-Prey Model - Lotka-Volterra)**.** The system is:

$$\begin{aligned}\dot{x}_1 &= \alpha x_1 - \beta x_1 x_2 \quad \text{(prey, e.g., rabbits)} \\ \dot{x}_2 &= \delta x_1 x_2 - \gamma x_2 \quad \text{(predators, e.g., foxes)}\end{aligned}$$

where $\alpha, \beta, \delta, \gamma$ are positive constants. Set derivatives to zero:

$$x_{1e}(\alpha - \beta x_{2e}) = 0$$
$$x_{2e}(\delta x_{1e} - \gamma) = 0$$

From the first equation: $x_{1e} = 0$ or $\alpha - \beta x_{2e} = 0 \implies x_{2e} = \alpha/\beta$. From the second equation: $x_{2e} = 0$ or $\delta x_{1e} - \gamma = 0 \implies x_{1e} = \gamma/\delta$.

Case 1: $x_{1e} = 0$. The first equation is satisfied. The second becomes $x_{2e}(0 - \gamma) = 0 \implies -\gamma x_{2e} = 0$. Since $\gamma > 0$, $x_{2e} = 0$. This gives the equilibrium point $E_1 = (0, 0)$. (No rabbits, no foxes).

Case 2: $x_{1e} \neq 0$. Then $\alpha - \beta x_{2e} = 0 \implies x_{2e} = \alpha/\beta$. Since $x_{2e} = \alpha/\beta \neq 0$ (as $\alpha, \beta > 0$), for the second equation to hold, we must have $\delta x_{1e} - \gamma = 0 \implies x_{1e} = \gamma/\delta$. This gives the equilibrium point $E_2 = (\gamma/\delta, \alpha/\beta)$. (Coexistence of populations).

3.2 Linear Time-Invariant (LTI) Systems: Analysis

We now start our journey into linear time-invariant systems.

3.2.1 Definition and Notation

An LTI system is typically represented as:

$$\text{Continuous-time: } \dot{x}(t) = Ax(t) + Bu(t) \tag{3.1}$$
$$y(t) = Cx(t) + Du(t) \tag{3.2}$$
$$\text{Discrete-time: } x(k+1) = Ax(k) + Bu(k) \tag{3.3}$$
$$y(k) = Cx(k) + Du(k) \tag{3.4}$$

where:

- $x \in \mathbb{R}^n$ is the state vector.
- $u \in \mathbb{R}^m$ is the input vector.
- $y \in \mathbb{R}^p$ is the output vector.
- $A \in \mathbb{R}^{n \times n}$ is the state matrix (or system matrix).
- $B \in \mathbb{R}^{n \times m}$ is the input matrix.
- $C \in \mathbb{R}^{p \times n}$ is the output matrix.
- $D \in \mathbb{R}^{p \times m}$ is the direct feedthrough (or feedforward) matrix.

The dimensions must be consistent. Sometimes these are compactly written as $\Sigma = (A, B, C, D)$. An alternative, less common, notation which is used in specific subdomains uses (F, G, H, J) for (A, B, C, D).

3.2.2 Linearity and Superposition

Linear systems obey the principle of superposition. We show this for simplicity for the state equation $\dot{x} = Ax$ (homogeneous case): If $x_a(t)$ is a solution (i.e., $\dot{x}_a = Ax_a$) and $x_b(t)$ is a solution (i.e., $\dot{x}_b = Ax_b$), then for any scalars $\alpha, \beta \in \mathbb{R}$, the linear combination $z(t) = \alpha x_a(t) + \beta x_b(t)$ is also a solution.

Proof.

$$\begin{aligned}
\dot{z}(t) &= \frac{d}{dt}(\alpha x_a(t) + \beta x_b(t)) \\
&= \alpha \dot{x}_a(t) + \beta \dot{x}_b(t) \quad \text{(linearity of differentiation)} \\
&= \alpha(Ax_a(t)) + \beta(Ax_b(t)) \quad \text{(since } x_a, x_b \text{ are solutions)} \\
&= A(\alpha x_a(t)) + A(\beta x_b(t)) \quad \text{(linearity of scalar-matrix multiplication)} \\
&= A(\alpha x_a(t) + \beta x_b(t)) \\
&= Az(t)
\end{aligned}$$

Thus, $z(t)$ satisfies $\dot{z} = Az$. The principle extends to systems with inputs and initial conditions. □

3.2.3 Equilibrium Points of LTI Systems

Consider a constant input $u(t) = u_{bar}$.

Continuous-Time Systems

Equilibrium points x_e satisfy $Ax_e + Bu_{bar} = 0$.

- **Homogeneous Case ($u_{bar} = 0$):** The equation becomes $Ax_e = 0$.
  - $x_e = 0$ is always an equilibrium point.
  - If A is invertible (non-singular, $\det(A) \neq 0$), then $x_e = 0$ is the *unique* equilibrium point.
  - If A is singular ($\det(A) = 0$), there are infinitely many equilibrium points, forming the null space (kernel) of A: $x_e \in \text{Null}(A)$.
- **General Case ($u_{bar} \neq 0$):** The equation is $Ax_e = -Bu_{bar}$. This is a linear system of algebraic equations.
  - If A is invertible: A unique equilibrium point exists: $x_e = -A^{-1}Bu_{bar}$.
  - If A is singular:
    - No equilibrium point exists if $-Bu_{bar} \notin \text{Im}(A)$ (i.e., $-Bu_{bar}$ is not in the column space/image of A).
    - Infinitely many equilibrium points exist if $-Bu_{bar} \in \text{Im}(A)$. The set of solutions is an affine subspace $x_p + \text{Null}(A)$, where x_p is any particular solution.

Discrete-Time Systems

Equilibrium points x_e satisfy $x_e = Ax_e + Bu_{bar}$, which rearranges to $(I - A)x_e = Bu_{bar}$. Let $A' = I - A$. The conditions are analogous to the continuous-time case with A' instead of A, and Bu_{bar} instead of $-Bu_{bar}$.

Example 3.2.1 (Double Integrator with specific input)**.** Consider $\dot{x} = Ax + Bu$ with

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Find equilibrium points for a constant input u_{bar}.

We need to solve $Ax_e + Bu_{bar} = 0$:

$$\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_{1e} \\ x_{2e} \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \end{pmatrix} u_{bar} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \Rightarrow \quad \begin{pmatrix} x_{2e} \\ 0 \end{pmatrix} + \begin{pmatrix} u_{bar} \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \Rightarrow \quad \begin{pmatrix} x_{2e} + u_{bar} \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

This gives two equations:

- $x_{2e} + u_{bar} = 0 \implies x_{2e} = -u_{bar}$
- $0 = 0$. The variable x_{1e} is free.

This shows that this system has infinitely many equilibrium points. We now show the same using the general theory developed above. The matrix A is singular ($\det(A) = 0$). The vector $-Bu_{bar} = \begin{pmatrix} -u_{bar} \\ 0 \end{pmatrix}$. The image of A is $\text{Im}(A) = \text{span}\left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\}$. So $-Bu_{bar}$ is always in $\text{Im}(A)$. Thus, infinitely many solutions exist. The equilibrium points are $x_e = \begin{pmatrix} c \\ -u_{bar} \end{pmatrix}$ for any $c \in \mathbb{R}$. This is a line in the state space parallel to the x_1-axis, shifted by $-u_{bar}$ along the x_2-axis. If $u_{bar} = 0$, the equilibrium points are $\begin{pmatrix} c \\ 0 \end{pmatrix}$, which is the x_1-axis. This is $\text{Null}(A)$.

3.2.4 Solution of Homogeneous LTI Systems ($\dot{x} = Ax$ or $x(k+1) = Ax(k)$)

We are now interested in determining $x(t)$ and $y(t)$ for all $t \geq 0$ given $x(0)$ and $u(t)$ for all $t \geq 0$. In this section we start with the homogenous case, which means that we assume that $u(t) = 0$ for all $t \geq 0$.

We know from basic calculus how to solve this problem when $n = 1$, that is, for the scalar equation

$$\dot{x} = \alpha x$$

where $x(t) \in \mathbb{R}$ and $\alpha \in \mathbb{R}$. In fact,

$$\frac{dx}{dt} = \alpha x \quad \Rightarrow \quad \int_{x(0)}^{x(t)} \frac{dx}{x} = \int_0^t \alpha d\tau \quad \Rightarrow \quad \ln \frac{x(t)}{x(0)} = \alpha t \quad \Rightarrow \quad x(t) = e^{\alpha t} x(0).$$

We can verify this solution by substituting it into the equation, namely

$$\frac{d}{dt} e^{\alpha t} x(0) = \alpha e^{\alpha t} x(0) = \alpha x$$

and

$$x(t)|_{t=0} = \left(e^{\alpha t} x(0)\right)_{t=0} = e^0 x(0) = x(0).$$

This suggests that in the general case in which $n \neq 1$ we may be able to solve the equation if we have an object with similar properties to that of $e^{\alpha} t$. Recall that one way of defining the exponential is by using its Taylor expansion, namely

$$e^{\alpha t} := I + \alpha t + \frac{(\alpha t)^2}{2!} + \frac{(\alpha t)^3}{3!} + \dots$$

We now define the matrix exponential by analogy. Given a (square) matrix A , the matrix exponential e^{At} is formally defined as

$$e^{At} := I + At + \frac{(At)^2}{2!} + \frac{(At)^3}{3!} + \dots$$

The matrix exponential so defined enjoys a fundamental derivative property which shows that this is a good definition.

Theorem 3.2.2. *The derivative of the matrix exponential is given by* $\frac{d}{dt} e^{At} = Ae^{At} = e^{At} A$.

Proof.

$$\begin{aligned} \tfrac{d}{dt} e^{At} &= \tfrac{d}{dt} \left(I + At + \tfrac{(At)^2}{2!} + \tfrac{(At)^3}{3!} + \dots \right) \\ &= At + 2\tfrac{A^2 t}{2!} + 3\tfrac{A^3 t^2}{3!} + \dots = A \left(I + At + \tfrac{(At)^2}{2!} + \tfrac{(At)^3}{3!} + \dots \right) = Ae^{At} \end{aligned}$$

The same proof can be carried out by factorizing A on the right, showing the other expression, and proving also that A and e^{At} commute. □

Theorem 3.2.3. *For* $\dot{x} = Ax$ *with* $x(0) = x_0$*, the solution is* $x(t) = e^{At} x_0$.

Proof. We first show that $\dot{x}(t) = \frac{d}{dt}(e^{At} x_0) = (\frac{d}{dt} e^{At}) x_0 = (Ae^{At}) x_0 = A(e^{At} x_0) = Ax(t)$. So the differential equation is satisfied. Then we show that $x(0) = e^{A \cdot 0} x_0 = e^0 x_0 = I x_0 = x_0$. So the initial condition is satisfied. □

Example 3.2.4. Compute e^{At} for $A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$. We find that $A^2 = 0$, so $A^k = 0$ for $k \geq 2$. Thus, $e^{At} = I + At + \frac{A^2 t^2}{2!} + \dots = I + At$, which is

$$e^{At} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} t = \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix}.$$

The solution to $\dot{x} = Ax$ for $x(0) = [x_{10}, x_{20}]^T$ and this A is

$$x(t) = \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} x(0) = \begin{pmatrix} x_{10} + t x_{20} \\ x_{20} \end{pmatrix}.$$

This can be verified with basic calculus by solving component-wise: $\dot{x}_1 = x_2$ and $\dot{x}_2 = 0$. From $\dot{x}_2 = 0$, we have $x_2(t) = x_{20}$ (constant). Substituting this into the first equation we obtain $\dot{x}_1 = x_{20}$. Integrating we obtain $x_1(t) = \int x_{20} dt = x_{20} t + C$. Using $x_1(0) = x_{10}$, we have that $x_{10} = x_{20}(0) + C \implies C = x_{10}$. So $x_1(t) = x_{10} + x_{20} t$. This matches.

Remark 3.2.5. The state-space approach provides a unified way to solve any system of linear **any**-order ODEs. Higher-order linear ODEs can always be converted to a system of first-order ODEs in state-space form (by the procedure of constructing the state-space model shown in Section 1.3.2). The solution is then always formally given by the matrix exponential, avoiding the need for case-by-case methods for finding homogeneous and particular solutions based on roots of characteristic equations.

3.2.5 Homework

Exercise 3.2.1 (Scholastic System)**.** Consider a discrete-time system modeling student progression through university years:

$$\begin{aligned} x_1(k+1) &= (1-\alpha_1)x_1(k) + u(k) \\ x_2(k+1) &= (1-\alpha_2)x_2(k) + \alpha_1 x_1(k) \\ x_3(k+1) &= (1-\alpha_3)x_3(k) + \alpha_2 x_2(k) \end{aligned}$$

where $x_i(k)$ is the number of students in year i at year k, α_i is the promotion rate from year i (so $1-\alpha_i$ is the repeat/dropout rate factor for those staying in year i), and $u(k)$ is the new intake of students into year 1. Assume $\alpha_i \in (0,1)$. Let $u(k) = \bar{u}$ (constant intake). Determine the equilibrium points $x_e = [x_{1e}, x_{2e}, x_{3e}]^T$.

Exercise 3.2.2. Consider a rectangular slice of air heated from below and cooled from above by edges kept at constant temperatures. This is our atmosphere in its simplest description. The bottom is heated by the earth and the top is cooled by the void of outer space. Within this slice, warm air rises and cool air sinks. In the model, as in the atmosphere, convection cells develop, transferring heat from bottom to top.

The state of the atmosphere in this model can be completely described by three variables, namely the convective flow x, the horizontal temperature distribution y, and the vertical temperature distribution z; by three parameters, namely the ratio of viscosity to thermal conductivity σ, the temperature difference between the top and bottom of the slice ρ, and the width to height ratio of the slice β, and by three differential equations describing the appropriate laws of fluid dynamics, namely

$$\dot{x} = \sigma(y-x) \qquad \dot{y} = \rho x - y - xz \qquad \dot{z} = xy - \beta z. \tag{3.5}$$

These equations were introduced by E.N. Lorenz in 1963, to model the *strange* behaviour of the atmosphere and to justify why weather forecast can be erroneous, and have been shown to play an important role on models of lasers and electrical generators. The Lorenz equations, which are one of the most famous and simplest system that generate *chaotic behaviours* are still at the basis of modern weather forecast algorithms.

Find the equilibrium points of the Lorenz model .

Chapter 4

Lecture 17/10: General Solution of Linear Systems

4.1 More on the Matrix Exponential

We have seen (Theorem 3.2.3) that for a linear time-invariant (LTI) system $\dot{x}(t) = Ax(t)$, where $x(t) \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n\times n}$, the solution is given by $x(t) = e^{At}x(0)$. The matrix exponential e^{At}, which we recall below, is central to understanding LTI systems.

Definition 4.1.1 (Matrix Exponential)**.** *The matrix exponential e^{At} for $A \in \mathbb{R}^{n\times n}$ and $t \in \mathbb{R}$ is defined by*

$$e^{At} := I + At + \frac{(At)^2}{2!} + \frac{(At)^3}{3!} + \cdots = \sum_{k=0}^{\infty} \frac{(At)^k}{k!} = \sum_{k=0}^{\infty} \frac{A^k t^k}{k!} \tag{4.1}$$

where I is the $n \times n$ identity matrix. This series converges for all A and t.

Recall (Theorem 3.2.2) that the derivative of the matrix exponential is given by $\frac{d}{dt}e^{At} = Ae^{At} = e^{At}A$.

We now look at additional properties of the matrix exponential.

Proposition 4.1.2. *For any $A \in \mathbb{R}^{n\times n}$ and $t_1, t_2 \in \mathbb{R}$:*

$$e^{A(t_1+t_2)} = e^{At_1}e^{At_2} \tag{4.2}$$

Proof.

$$\begin{aligned}
e^{At_1}e^{At_2} &= \left(\sum_{k=0}^{\infty} \frac{A^k t_1^k}{k!}\right)\left(\sum_{m=0}^{\infty} \frac{A^m t_2^m}{m!}\right) \\
&= \left(I + At_1 + \frac{A^2t_1^2}{2!} + \dots\right)\left(I + At_2 + \frac{A^2t_2^2}{2!} + \dots\right) \\
&= I \cdot I + I \cdot At_2 + At_1 \cdot I + I \cdot \frac{A^2t_2^2}{2!} + At_1 \cdot At_2 + \frac{A^2t_1^2}{2!} \cdot I + \dots \\
&= I + A(t_1 + t_2) + \frac{A^2t_1^2}{2!} + A^2t_1t_2 + \frac{A^2t_2^2}{2!} + \dots \\
&= I + A(t_1 + t_2) + \frac{A^2}{2!}(t_1^2 + 2t_1t_2 + t_2^2) + \dots \\
&= I + A(t_1 + t_2) + \frac{A^2(t_1 + t_2)^2}{2!} + \dots \\
&= \sum_{n=0}^{\infty} \frac{A^n(t_1 + t_2)^n}{n!} = e^{A(t_1+t_2)}.
\end{aligned}$$

The step involving collecting terms relies on the binomial expansion $(t_1 + t_2)^n = \sum_{k=0}^{n} \binom{n}{k} t_1^k t_2^{n-k}$ and the fact that all matrices involved are powers of A, so they commute. □

This property implies that the state transition matrix $\Phi(t, t_0) = e^{A(t-t_0)}$ satisfies the semigroup property $\Phi(t_2, t_0) = \Phi(t_2, t_1)\Phi(t_1, t_0)$.

Proposition 4.1.3. *For $A_1, A_2 \in \mathbb{R}^{n\times n}$ and $t \in \mathbb{R}$:*

$$e^{(A_1+A_2)t} = e^{A_1 t}e^{A_2 t} \quad \textit{if and only if } A_1A_2 = A_2A_1 \textit{ (i.e., } A_1 \textit{ and } A_2 \textit{ commute)} \tag{4.3}$$

Proof. Consider the series expansions:

$$\begin{aligned} e^{A_1 t}e^{A_2 t} &= \left(I + A_1 t + \frac{(A_1 t)^2}{2!} + \dots\right)\left(I + A_2 t + \frac{(A_2 t)^2}{2!} + \dots\right) \\ &= I + (A_1 + A_2)t + \left(\frac{A_1^2 t^2}{2!} + A_1 t A_2 t + \frac{A_2^2 t^2}{2!}\right) + \dots \\ &= I + (A_1 + A_2)t + \frac{t^2}{2!}(A_1^2 + 2A_1A_2 + A_2^2) + \dots \end{aligned}$$

And the expansion for $e^{(A_1+A_2)t}$:

$$\begin{aligned} e^{(A_1+A_2)t} &= I + (A_1 + A_2)t + \frac{((A_1 + A_2)t)^2}{2!} + \dots \\ &= I + (A_1 + A_2)t + \frac{t^2}{2!}(A_1 + A_2)(A_1 + A_2) + \dots \\ &= I + (A_1 + A_2)t + \frac{t^2}{2!}(A_1^2 + A_1A_2 + A_2A_1 + A_2^2) + \dots \end{aligned}$$

For these two expressions to be equal, the coefficients of $t^k/k!$ must match for all k. Comparing the $t^2/2!$ terms, we need: $A_1^2 + 2A_1A_2 + A_2^2 = A_1^2 + A_1A_2 + A_2A_1 + A_2^2$. This simplifies to $2A_1A_2 = A_1A_2 + A_2A_1$, which means $A_1A_2 = A_2A_1$. If A_1 and A_2 commute, then $(A_1+A_2)^k = \sum_{j=0}^{k} \binom{k}{j} A_1^j A_2^{k-j}$ by the binomial theorem This allows the general proof to proceed similarly to Proposition 4.1.2. □

Example 4.1.4. Compute e^{At} for $A = \begin{pmatrix} \lambda & 1 \\ 0 & \lambda \end{pmatrix}$ (this matrix is an example of a Jordan block https://en.wikipedia.org/wiki/Jordan_matrix.).

We can write $A = A_1 + A_2$, where $A_1 = \begin{pmatrix} \lambda & 0 \\ 0 & \lambda \end{pmatrix} = \lambda I$ and $A_2 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$. Check for commutation: $A_1A_2 = (\lambda I)A_2 = \lambda A_2$. $A_2A_1 = A_2(\lambda I) = \lambda A_2$. Since $A_1A_2 = A_2A_1$, they commute. Thus $e^{At} = e^{A_1 t}e^{A_2 t}$. Compute $e^{A_1 t}$:

$$e^{A_1 t} = e^{\lambda I t} = \sum_{k=0}^{\infty} \frac{(\lambda I t)^k}{k!} = \sum_{k=0}^{\infty} \frac{\lambda^k t^k I^k}{k!} = I \sum_{k=0}^{\infty} \frac{(\lambda t)^k}{k!} = e^{\lambda t} I = \begin{pmatrix} e^{\lambda t} & 0 \\ 0 & e^{\lambda t} \end{pmatrix}.$$

For $e^{A_2 t}$, we have already shown in Exercise 3.2.4 that

$$e^{\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} t} = \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix}.$$

Therefore,

$$e^{At} = e^{A_1 t}e^{A_2 t} = \begin{pmatrix} e^{\lambda t} & 0 \\ 0 & e^{\lambda t} \end{pmatrix}\begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} e^{\lambda t}\cdot 1 + 0\cdot 0 & e^{\lambda t}\cdot t + 0\cdot 1 \\ 0\cdot 1 + e^{\lambda t}\cdot 0 & 0\cdot t + e^{\lambda t}\cdot 1 \end{pmatrix} = \begin{pmatrix} e^{\lambda t} & te^{\lambda t} \\ 0 & e^{\lambda t}. \end{pmatrix}$$

The structure $e^{\lambda t}$ on the diagonal and $te^{\lambda t}$ on the superdiagonal is characteristic of the matrix exponential of a Jordan block.

Remark 4.1.5 (General Form of e^{At})**.** The elements of e^{At} are sums of terms of the form $p_{ij}(t)e^{\lambda_k t}$, where λ_k are eigenvalues of A and $p_{ij}(t)$ are polynomials in t of degree less than the multiplicity of λ_k in the minimal polynomial[1] of A (or less than n). More precisely, e^{At} can be expressed as:

$$e^{At} = \sum_{i=1}^{\nu} \sum_{h=0}^{m_i-1} \frac{t^h}{h!} e^{\lambda_i t} R_{ih} \tag{4.4}$$

[1] This is explained later in the course.

where $\lambda_1, \ldots, \lambda_\nu$ are the distinct eigenvalues of A, m_i is the multiplicity of λ_i as a root of the minimal polynomial of A (which is also the size of the largest Jordan block associated with λ_i), and R_{ih} are constant $n \times n$ matrices determined by A. It is then clear from (4.4) that the behavior of e^{At} as $t \to \infty$ is dictated by the eigenvalues λ_i. If all $\mathrm{Re}(\lambda_i) < 0$, then $e^{At} \to 0$. If any $\mathrm{Re}(\lambda_i) > 0$, some elements of e^{At} may grow unbounded.

Proposition 4.1.6. *The inverse of e^{At} is e^{-At}, namely*

$$(e^{At})^{-1} = e^{-At}. \tag{4.5}$$

Proof. Using Proposition 4.1.2: $e^{At}e^{-At} = e^{A(t-t)} = e^{A\cdot 0} = e^0 = I$. Similarly, $e^{-At}e^{At} = I$. Thus, e^{At} is always invertible, and its inverse is e^{-At}. □

This property is crucial for continuous-time systems, allowing one to "go backward in time" in principle, as the state transition matrix is always invertible.

Proposition 4.1.7 (Similarity Transformation)**.** *If T is an invertible matrix, then:*

$$e^{T^{-1}ATt} = T^{-1}e^{At}T \tag{4.6}$$

Proof. Let $M = T^{-1}AT$. Then

$$e^{Mt} = \sum_{k=0}^{\infty} \frac{(Mt)^k}{k!} = \sum_{k=0}^{\infty} \frac{M^k t^k}{k!}$$

Note that $M^k = (T^{-1}AT)(T^{-1}AT)\ldots(T^{-1}AT) = T^{-1}A(TT^{-1})A(TT^{-1})\ldots AT = T^{-1}A^kT$. So,

$$\begin{aligned} e^{T^{-1}ATt} &= \sum_{k=0}^{\infty} \frac{(T^{-1}A^kT)t^k}{k!} = T^{-1}\left(\sum_{k=0}^{\infty} \frac{A^k t^k}{k!}\right) T \\ &= T^{-1}e^{At}T. \end{aligned}$$

□

This property is useful, for instance, because we can transform A into a simpler form, such as a diagonal or Jordan form $J = T^{-1}AT$. Then $e^{At} = Te^{Jt}T^{-1}$, and e^{Jt} is easy to compute, see *e.g.* Example 4.1.4.

4.2 Solution of Linear Continuous-Time Systems

Consider the LTI system:

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(t_0) = x_0 \tag{4.7}$$

where $x(t) \in \mathbb{R}^n$ is the state, $u(t) \in \mathbb{R}^m$ is the input, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$. We now want to determine $x(t)$ for all $t \geq 0$ in the general case.

4.2.1 Special Case: $A = 0$

We first consider a special case that will help us with the general case. If $A = 0$, the system becomes $\dot{x}(t) = Bu(t)$. Integrating from t_0 to t:

$$\begin{aligned} \int_{t_0}^{t} \dot{x}(\tau)d\tau &= \int_{t_0}^{t} Bu(\tau)d\tau \\ x(t) - x(t_0) &= \int_{t_0}^{t} Bu(\tau)d\tau \\ x(t) &= x(t_0) + \int_{t_0}^{t} Bu(\tau)d\tau. \end{aligned}$$

4.2.2 General Solution

To solve (4.7), we use a change of variables. Let $z(t) = e^{-A(t-t_0)}x(t)$. Then $x(t) = e^{A(t-t_0)}z(t)$. Differentiate $z(t)$ with respect to t:

$$\begin{aligned}
\dot{z}(t) &= \frac{d}{dt}(e^{-A(t-t_0)})x(t) + e^{-A(t-t_0)}\dot{x}(t) \\
&= -Ae^{-A(t-t_0)}x(t) + e^{-A(t-t_0)}(Ax(t) + Bu(t)) \\
&= -Ae^{-A(t-t_0)}x(t) + e^{-A(t-t_0)}Ax(t) + e^{-A(t-t_0)}Bu(t) \\
&= -Ae^{-A(t-t_0)}x(t) + Ae^{-A(t-t_0)}x(t) + e^{-A(t-t_0)}Bu(t) \\
&= e^{-A(t-t_0)}Bu(t)
\end{aligned}$$

This is in the same form as the case $A = 0$. So, like in that case, we integrate $\dot{z}(t)$ from t_0 to t:

$$\begin{aligned}
\int_{t_0}^{t} \dot{z}(\tau)d\tau &= \int_{t_0}^{t} e^{-A(\tau-t_0)}Bu(\tau)d\tau \\
z(t) - z(t_0) &= \int_{t_0}^{t} e^{-A(\tau-t_0)}Bu(\tau)d\tau
\end{aligned}$$

We know $z(t_0) = e^{-A(t_0-t_0)}x(t_0) = e^0 x(t_0) = Ix(t_0) = x(t_0)$. So,

$$z(t) = x(t_0) + \int_{t_0}^{t} e^{-A(\tau-t_0)}Bu(\tau)d\tau$$

Substitute back $x(t) = e^{A(t-t_0)}z(t)$:

$$\begin{aligned}
x(t) &= e^{A(t-t_0)}\left(x(t_0) + \int_{t_0}^{t} e^{-A(\tau-t_0)}Bu(\tau)d\tau\right) \\
&= e^{A(t-t_0)}x(t_0) + e^{A(t-t_0)}\int_{t_0}^{t} e^{-A(\tau-t_0)}Bu(\tau)d\tau \\
&= e^{A(t-t_0)}x(t_0) + \int_{t_0}^{t} e^{A(t-t_0)}e^{-A(\tau-t_0)}Bu(\tau)d\tau \\
&= e^{A(t-t_0)}x(t_0) + \int_{t_0}^{t} e^{A(t-\tau)}Bu(\tau)d\tau.
\end{aligned}$$

This is the complete solution to the LTI system.

$$x(t) = \underbrace{e^{A(t-t_0)}x(t_0)}_{\text{Free response (or zero-input response)}} + \underbrace{\int_{t_0}^{t} e^{A(t-\tau)}Bu(\tau)d\tau}_{\text{Forced response (or zero-state response)}} \tag{4.8}$$

If $t_0 = 0$, the formula becomes:

$$x(t) = e^{At}x(0) + \int_{0}^{t} e^{A(t-\tau)}Bu(\tau)d\tau. \tag{4.9}$$

The term $e^{A(t-\tau)}$ is known as the state transition matrix, often denoted $\Phi(t,\tau)$. The integral term is a convolution integral: $(e^{At} * Bu(t))$.

Remark 4.2.1 (Superposition)**.** The solution demonstrates the principle of superposition for linear systems.

- The response due to the initial state $x(t_0)$ (with $u(t) = 0$) is $x_{zi}(t) = e^{A(t-t_0)}x(t_0)$.
- The response due to the input $u(t)$ (with $x(t_0) = 0$) is $x_{zs}(t) = \int_{t_0}^{t} e^{A(t-\tau)}Bu(\tau)d\tau$.

The total response is $x(t) = x_{zi}(t) + x_{zs}(t)$.

Remark 4.2.2 (Reversibility)**.** Since $e^{A(t-t_0)}$ is always invertible, we can express $x(t_0)$ in terms of $x(t)$:

$$\begin{aligned} e^{A(t-t_0)}x(t_0) &= x(t) - \int_{t_0}^{t} e^{A(t-\tau)}Bu(\tau)d\tau \\ x(t_0) &= e^{-A(t-t_0)}\left(x(t) - \int_{t_0}^{t} e^{A(t-\tau)}Bu(\tau)d\tau\right) \\ x(t_0) &= e^{A(t_0-t)}x(t) - e^{A(t_0-t)}\int_{t_0}^{t} e^{A(t-\tau)}Bu(\tau)d\tau \\ x(t_0) &= e^{A(t_0-t)}x(t) - \int_{t_0}^{t} e^{A(t_0-\tau)}Bu(\tau)d\tau \end{aligned}$$

This implies that, given $x(t)$ and the input history $u(\tau)$ for $\tau \in [t_0, t]$, one can uniquely determine the initial state $x(t_0)$. This is equivalent to running the system backward in time.

Example 4.2.3. Consider a continuous-time, time-invariant, linear system with $x \in \mathbb{R}^2$, $u(t) \in \mathbb{R}$ and $y(t) \in \mathbb{R}$. Suppose that $x(0) = [1, 1]'$, that $u(t) = 0$ for all t, and that

$$A = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$

Note that

$$A = T^{-1}\tilde{A}T = \begin{bmatrix} 1 & 1 \\ -1 & -2 \end{bmatrix}\begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix}\begin{bmatrix} 2 & 1 \\ -1 & -1 \end{bmatrix}.$$

By Proposition 4.1.7 we have

$$e^{At} = e^{(T^{-1}\tilde{A}T)t} = T^{-1}\begin{bmatrix} e^{-t} & 0 \\ 0 & e^{-2t} \end{bmatrix}T = e^{-t}\begin{bmatrix} 2 & 1 \\ -2 & -1 \end{bmatrix} + e^{-2t}\begin{bmatrix} -1 & -1 \\ 2 & 2 \end{bmatrix}.$$

Therefore

$$x(t) = \begin{bmatrix} -2e^{-2t} + 3e^{-t} \\ 4e^{-2t} - 3e^{-t} \end{bmatrix}$$

and $y(t) = -2e^{-2t} + 3e^{-t}$. Note that, the state and the output are linear combinations of exponential functions with exponents given by t times the eigenvalues of A.

4.3 Connection with the Laplace Transform

Consider the LTI system with output equation:

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0 \tag{4.10}$$
$$y(t) = Cx(t) + Du(t) \tag{4.11}$$

We use $t_0 = 0$ for simplicity in Laplace transform derivations. Let $X(s) = \mathcal{L}\{x(t)\}$, $U(s) = \mathcal{L}\{u(t)\}$, and $Y(s) = \mathcal{L}\{y(t)\}$, where $\mathcal{L}$ indicates the Laplace transform. Taking the Laplace transform of (4.10) yields

$$sX(s) - x(0) = AX(s) + BU(s).$$

Rearranging to solve for $X(s)$ we obtain

$$sX(s) - AX(s) = x(0) + BU(s) \quad \Rightarrow \quad (sI - A)X(s) = x(0) + BU(s).$$

If $(sI - A)$ is invertible (i.e., s is not an eigenvalue of A), then:

$$X(s) = (sI - A)^{-1}x(0) + (sI - A)^{-1}BU(s) \tag{4.12}$$

Taking the Laplace transform of (4.11) we obtain

$$Y(s) = CX(s) + DU(s)$$

Substituting (4.12) into this equation yields

$$\begin{aligned} Y(s) &= C\left((sI - A)^{-1}x(0) + (sI - A)^{-1}BU(s)\right) + DU(s) \\ Y(s) &= \underbrace{C(sI - A)^{-1}x(0)}_{\text{Laplace transform of zero-input response}} + \underbrace{\left(C(sI - A)^{-1}B + D\right)U(s)}_{\text{Laplace transform of zero-state response}} \end{aligned}$$

Definition 4.3.1 (Transfer Function Matrix)**.** *If the initial condition $x(0) = 0$ (zero-state response), then $Y(s) = (C(sI - A)^{-1}B + D)U(s)$. The matrix $G(s) = C(sI - A)^{-1}B + D$ is called the transfer function matrix of the system. It relates the Laplace transform of the input $U(s)$ to the Laplace transform of the output $Y(s)$ when initial conditions are zero.*

Remark 4.3.2. Comparing the time-domain solution (4.9) with the Laplace-domain solution:

- $\mathcal{L}\{e^{At}x(0)\} = (sI - A)^{-1}x(0)$. This implies $\mathcal{L}\{e^{At}\} = (sI - A)^{-1}$.
- $\mathcal{L}\left\{\int_0^t e^{A(t-\tau)}Bu(\tau)d\tau\right\} = (sI-A)^{-1}BU(s)$. This is due to the convolution theorem $\mathcal{L}\{f(t)*g(t)\} = F(s)G(s)$, where here $f(t) = e^{At}$ and $g(t) = Bu(t)$.

4.3.1 Computing the Matrix Exponential using Laplace Transform

The identity $\mathcal{L}\{e^{At}\} = (sI - A)^{-1}$ provides a method for computing e^{At}:

$$e^{At} = \mathcal{L}^{-1}\{(sI - A)^{-1}\} \tag{4.13}$$

This involves:

1. Forming the matrix $sI - A$.
2. Computing its inverse $(sI - A)^{-1}$. This is typically done using the adjugate matrix: $(sI - A)^{-1} = \frac{\text{adj}(sI-A)}{\det(sI-A)}$.
3. Taking the inverse Laplace transform of each element of $(sI - A)^{-1}$.

Example 4.3.3 (The Harmonic Oscillator)**.** Consider the system $\dot{x} = Ax$ with $A = \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix}$. This system describes simple harmonic motion. We want to compute e^{At}.

Method 1: Taylor Series Expansion

$$A = \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix}$$

$$A^2 = \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix}\begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix} = \begin{pmatrix} -\omega^2 & 0 \\ 0 & -\omega^2 \end{pmatrix} = -\omega^2 I$$

$$A^3 = A \cdot A^2 = A(-\omega^2 I) = -\omega^2 A = \begin{pmatrix} 0 & -\omega^3 \\ \omega^3 & 0 \end{pmatrix}$$

$$A^4 = (A^2)^2 = (-\omega^2 I)^2 = \omega^4 I = \begin{pmatrix} \omega^4 & 0 \\ 0 & \omega^4 \end{pmatrix}$$

In general, $A^{2k} = (-1)^k\omega^{2k}I$ and $A^{2k+1} = (-1)^k\omega^{2k}A$. Thus

$$\begin{aligned} e^{At} &= \sum_{k=0}^{\infty} \frac{A^k t^k}{k!} = \sum_{j=0}^{\infty} \frac{A^{2j}t^{2j}}{(2j)!} + \sum_{j=0}^{\infty} \frac{A^{2j+1}t^{2j+1}}{(2j+1)!} \\ &= \sum_{j=0}^{\infty} \frac{(-1)^j\omega^{2j}It^{2j}}{(2j)!} + \sum_{j=0}^{\infty} \frac{(-1)^j\omega^{2j}At^{2j+1}}{(2j+1)!} \\ &= I\sum_{j=0}^{\infty} \frac{(-1)^j(\omega t)^{2j}}{(2j)!} + A\frac{1}{\omega}\sum_{j=0}^{\infty} \frac{(-1)^j(\omega t)^{2j+1}}{(2j+1)!} \\ &= I\cos(\omega t) + \frac{A}{\omega}\sin(\omega t) \\ &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\cos(\omega t) + \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}\sin(\omega t) \\ &= \begin{pmatrix} \cos(\omega t) & \sin(\omega t) \\ -\sin(\omega t) & \cos(\omega t) \end{pmatrix}. \end{aligned}$$

Note that to solve this we have used the Taylor expansions of $\cos(\omega t)$ and $\sin(\omega t)$. If the expressions above seem too complicated, write $I + At + \frac{(At)^2}{2!} + \frac{(At)^3}{3!} + \frac{(At)^4}{4!} + \ldots$, obtaining

$$\begin{pmatrix} 1 - \frac{\omega^2 t^2}{2!} + \frac{\omega^4 t^4}{4!} + \ldots & \omega t - \frac{\omega^3 t^3}{3!} + \ldots \\ -\omega t + \frac{\omega^3 t^3}{3!} + \ldots & 1 - \frac{\omega^2 t^2}{2!} + \frac{\omega^4 t^4}{4!} + \ldots \end{pmatrix}$$

from which then the result follows from the Taylor expansions of $\cos(\omega t)$ and $\sin(\omega t)$

Method 2: Laplace Transform $sI - A = \begin{pmatrix} s & 0 \\ 0 & s \end{pmatrix} - \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix} = \begin{pmatrix} s & -\omega \\ \omega & s \end{pmatrix}$. $\det(sI - A) = s^2 - (-\omega)(\omega) = s^2 + \omega^2$. $(sI - A)^{-1} = \frac{1}{s^2+\omega^2} \begin{pmatrix} s & \omega \\ -\omega & s \end{pmatrix} = \begin{pmatrix} \frac{s}{s^2+\omega^2} & \frac{\omega}{s^2+\omega^2} \\ \frac{-\omega}{s^2+\omega^2} & \frac{s}{s^2+\omega^2} \end{pmatrix}$. Taking the inverse Laplace transform of each element: $\mathcal{L}^{-1}\left\{\frac{s}{s^2+\omega^2}\right\} = \cos(\omega t)$ $\mathcal{L}^{-1}\left\{\frac{\omega}{s^2+\omega^2}\right\} = \sin(\omega t)$ So, $e^{At} = \mathcal{L}^{-1}\{(sI-A)^{-1}\} = \begin{pmatrix} \cos(\omega t) & \sin(\omega t) \\ -\sin(\omega t) & \cos(\omega t) \end{pmatrix}$. This confirms the result from Method 1 and is often computationally simpler.

Example 4.3.4. Let

$$F_\lambda = \begin{bmatrix} \lambda & \omega \\ -\omega & \lambda \end{bmatrix} \qquad \text{with} \quad F_0 = \begin{bmatrix} 0 & \omega \\ -\omega & 0 \end{bmatrix}.$$

Note that $F_\lambda = \lambda I + F_0$, and that I and F_0 commutes, hence

$$e^{F_\lambda t} = e^{(\lambda I + F_0)t} = e^{\lambda t} e^{F_0 t}.$$

Thus, by Example 4.3.3

$$e^{F_\lambda t} = e^{\lambda t} \begin{bmatrix} \cos \omega t & \sin \omega t \\ -\sin \omega t & \cos \omega t \end{bmatrix}.$$

Chapter 5

Lecture 20/10: Discrete-time Systems, Realization, and Lyapunov Stability

Exercises

Suggested exercises for practice:

- Exercises 1 and 2 (Linear Algebra: computing eigenvalues/eigenvectors etc.).
- Exam 2022-2023: 1.b, 1.c part 1, 2.a.
- Exam 2021-2022: 1.a.

5.1 Response of Continuous-Time Linear Systems (Recap)

Last time, we derived the response of a continuous-time linear system $\dot{x}(t) = Ax(t) + Bu(t)$, $y(t) = Cx(t) + Du(t)$, starting from $t_0 = 0$ with initial condition $x(0)$:

$$x(t) = e^{At}x(0) + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau$$

A question from students arose about the general formula if we start from an arbitrary initial time t_0 with initial condition $x(t_0)$. In Section 4.2.2 we did provide the general formula. However, it is of interest to provide an alternative derivation as follows.

We know $x(t) = e^{At}x(0) + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau$. We want to express $x(t)$ in terms of $x(t_0)$ for $t > t_0 > 0$.

$$\begin{aligned}
x(t) &= e^{A(t-t_0+t_0)}x(0) + \int_0^{t_0} e^{A(t-\tau)}Bu(\tau)d\tau + \int_{t_0}^t e^{A(t-\tau)}Bu(\tau)d\tau \\
&= e^{A(t-t_0)}e^{At_0}x(0) + \int_0^{t_0} e^{A(t-t_0)}e^{A(t_0-\tau)}Bu(\tau)d\tau + \int_{t_0}^t e^{A(t-\tau)}Bu(\tau)d\tau \\
&= e^{A(t-t_0)}\left(e^{At_0}x(0) + \int_0^{t_0} e^{A(t_0-\tau)}Bu(\tau)d\tau\right) + \int_{t_0}^t e^{A(t-\tau)}Bu(\tau)d\tau
\end{aligned}$$

Recognizing that the term in the parenthesis is

$$x(t_0) = e^{At_0}x(0) + \int_0^{t_0} e^{A(t_0-\tau)}Bu(\tau)d\tau$$

we have:

$$x(t) = e^{A(t-t_0)}x(t_0) + \int_{t_0}^t e^{A(t-\tau)}Bu(\tau)d\tau.$$

5.2 Full Trajectories of Discrete-Time Systems

Consider the discrete-time LTI system:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k) \end{aligned}$$

Given $x(0)$ and the input sequence $u(0), u(1), \ldots$, we want to find $x(k)$ and $y(k)$ for $k \geq 0$.

Iterative calculations show:

- For $k = 0$: $x(1) = Ax(0) + Bu(0)$
- For $k = 1$: $x(2) = Ax(1) + Bu(1) = A(Ax(0) + Bu(0)) + Bu(1) = A^2x(0) + ABu(0) + Bu(1)$
- For $k = 2$: $x(3) = Ax(2) + Bu(2) = A(A^2x(0) + ABu(0) + Bu(1)) + Bu(2) = A^3x(0) + A^2Bu(0) + ABu(1) + Bu(2)$

Generalizing, we find $x(k)$:

$$\begin{aligned} x(k) &= A^k x(0) + A^{k-1}Bu(0) + A^{k-2}Bu(1) + \cdots + ABu(k-2) + Bu(k-1) \\ &= A^k x(0) + \sum_{j=0}^{k-1} A^{k-1-j}Bu(j) \end{aligned}$$

This is the state response for $k \geq 0$, assuming $x(0)$ is known. The sum is defined to be zero if $k = 0$.

The output $y(k)$ is then:

$$\begin{aligned} y(k) &= Cx(k) + Du(k) \\ &= C\left(A^k x(0) + \sum_{j=0}^{k-1} A^{k-1-j}Bu(j)\right) + Du(k) \\ &= CA^k x(0) + \sum_{j=0}^{k-1} CA^{k-1-j}Bu(j) + Du(k) \end{aligned}$$

The response can be split into:

- **Free response** (due to initial condition $x(0)$, assuming $u(k) = 0$ for all k):

$$x_{free}(k) = A^k x(0)$$

- **Forced response** (due to input $u(k)$, assuming $x(0) = 0$):

$$x_{forced}(k) = \sum_{j=0}^{k-1} A^{k-1-j}Bu(j) + Du(k)$$

The total response is $x(k) = x_{free}(k) + x_{forced}(k)$.

Remark 5.2.1. The expression of $x(k)$ can be rewritten as

$$x(k) = A^k x_0 + \begin{bmatrix} B & AB & \cdots & A^{k-1}B \end{bmatrix} \begin{bmatrix} u(k-1) \\ u(k-2) \\ \vdots \\ u(0) \end{bmatrix}.$$

This expression highlights the role of the matrix $\begin{bmatrix} B, & AB, \cdots, & A^{k-1}B \end{bmatrix}$ in the computation of $x(k)$.

5.2.1 Markov Parameters

The matrices CA^jB for $j \geq 0$ are known as **Markov parameters**. They are important in system identification. If $x(0) = 0$ and $u(k)$ is a unit impulse, i.e., $u(0) = 1$ and $u(j) = 0$ for $j > 0$:

- $y(0) = Cx(0) + Du(0) = D$
- $y(1) = Cx(1) + Du(1) = C(Ax(0) + Bu(0)) + Du(1) = CB$
- $y(2) = Cx(2) + Du(2) = C(Ax(1) + Bu(1)) + Du(2) = C(A(Bu(0))) = CAB$
- In general, $y(k) = CA^{k-1}B$ for $k \geq 1$, and $y(0) = D$.

The sequence $D, CB, CAB, CA^2B, \ldots$ is the impulse response of the system. The problem of finding (A, B, C, D) from the impulse response (or input/output data) is called **system realization** or **system identification**.

5.2.2 Reversibility

For CT systems, e^{At} is always invertible ($(e^{At})^{-1} = e^{-At}$). We can go backward in time. For DT systems, to find $x(0)$ from $x(k)$:

$$x(k) = A^k x(0) + \sum_{j=0}^{k-1} A^{k-1-j} Bu(j) \qquad \Rightarrow \qquad A^k x(0) = x(k) - \sum_{j=0}^{k-1} A^{k-1-j} Bu(j)$$

If A^k is invertible, then $x(0) = (A^k)^{-1} \left(x(k) - \sum_{j=0}^{k-1} A^{k-1-j} Bu(j) \right)$. A^k is invertible if and only if A is invertible. A is invertible if and only if $\det(A) \neq 0$, which means 0 is not an eigenvalue of A. If A is not invertible, we cannot uniquely determine $x(0)$ from $x(k)$; the system is not reversible. This is a key difference from CT systems.

5.2.3 Z-Transform and Transfer Function for DT Systems

The Z-transform is the discrete-time counterpart of the Laplace transform and it is defined as

$$\mathcal{Z}\{x(k)\} = X(z) = \sum_{k=0}^{\infty} x(k) z^{-k}.$$

Similarly to the Laplace transform, it enjoys a time shift property:

$$\mathcal{Z}\{x(k+1)\} = zX(z) - zx(0).$$

Applying the Z-transform to the state equation $x(k+1) = Ax(k) + Bu(k)$ yields

$$\begin{aligned} zX(z) - zx(0) &= AX(z) + BU(z) \quad \Rightarrow \quad (zI - A)X(z) = zx(0) + BU(z) \quad \Rightarrow \\ X(z) &= (zI - A)^{-1} zx(0) + (zI - A)^{-1} BU(z) \end{aligned}$$

Applying the Z-transform to the output equation $y(k) = Cx(k) + Du(k)$ we obtain

$$Y(z) = C(zI - A)^{-1} zx(0) + \left[C(zI - A)^{-1} B + D \right] U(z).$$

Assuming zero initial conditions ($x(0) = 0$), the **transfer function** $H(z)$ is:

$$H(z) = \frac{Y(z)}{U(z)} = C(zI - A)^{-1} B + D$$

This is analogous to the CT transfer function $H(s) = C(sI - A)^{-1}B + D$. Moreover, we also clearly see that

$$A^k = \mathcal{Z}^{-1} \left\{ (zI - A)^{-1} z \right\} \qquad \sum_{j=0}^{k-1} A^{k-1-j} Bu(j) = \mathcal{Z}^{-1} \left\{ (zI - A)^{-1} BU(z) \right\}.$$

5.3 Realization Problem (Continuous-Time)

In this section we solve the following problem: given a transfer function $H(s)$, find a state-space representation (A, B, C, D).

Let $H(s) = \frac{Y(s)}{U(s)} = \frac{N(s)}{D(s)}$, where $N(s)$ and $D(s)$ are polynomials in s. Then $D(s)Y(s) = N(s)U(s)$. Let $D(s) = s^n + \alpha_{n-1}s^{n-1} + \cdots + \alpha_1 s + \alpha_0$ (monic, degree n). Let $N(s) = \beta_{n-1}s^{n-1} + \cdots + \beta_1 s + \beta_0$ (degree at most $n-1$), that is $\deg(N(s)) < \deg(D(s))$. Note that this implies $D = 0$ in the state-space model. In fact, if $\deg(N(s)) = \deg(D(s))$, then $D \neq 0$. In this case we can find D by polynomial long division first, and then the resulting polynomial will have $\deg(N(s)) < \deg(D(s))$. Assume also that $N(s)$ and $D(s)$ are coprime (no common factors). Assume zero initial conditions for $y(t)$ and its derivatives, and for $u(t)$ and its derivatives. Recall that $\mathcal{L}\{y^{(k)}(t)\} = s^k Y(s)$ under zero initial conditions. Taking the inverse Laplace transform of $D(s)Y(s) = N(s)U(s)$ yields

$$y^{(n)}(t) + \alpha_{n-1}y^{(n-1)}(t) + \cdots + \alpha_0 y(t) = \beta_{n-1}u^{(n-1)}(t) + \cdots + \beta_0 u(t).$$

This is an n-th order differential equation.

To simplify, we can assume that $N(s) = \beta_0$ (i.e., $\beta_1 = \cdots = \beta_{n-1} = 0$) as this will still allow us to solve the problem as we will see. Now define the states

$$\begin{aligned} x_1 &= y \\ x_2 &= \dot{y} = \dot{x}_1 \\ &\vdots \\ x_n &= y^{(n-1)} = \dot{x}_{n-1} \end{aligned}$$

Then

$$\dot{x}_n = y^{(n)}(t) = -\alpha_0 y - \alpha_1 \dot{y} - \cdots - \alpha_{n-1}y^{(n-1)} + \beta_0 u(t) \quad \Rightarrow \quad \dot{x}_n = -\alpha_0 x_1 - \alpha_1 x_2 - \cdots - \alpha_{n-1}x_n + \beta_0 u(t)$$

which in matrix form ($\dot{x} = Ax + Bu, y = Cx + Du$) gives

$$A = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -\alpha_0 & -\alpha_1 & -\alpha_2 & \dots & -\alpha_{n-1} \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \beta_0 \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & 0 & \dots & 0 \end{pmatrix}, \quad D = 0$$

This is called the **controllable canonical form**.

Example 5.3.1. Consider the transfer function

$$H(s) = \frac{1}{s^2 + 3s + 2}.$$

and the problem of determining a realization. Here, $n = 2$. $D(s) = s^2 + 3s + 2 \implies \alpha_1 = 3, \alpha_0 = 2$. $N(s) = 1 \implies \beta_0 = 1$. For the specific example where $N(s) = \beta_0$:

$$A = \begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 \end{pmatrix}, \quad D = 0.$$

We can check that this expression is correct by computing the transfer function:

$$C(sI - A)^{-1}B + D = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} s & -1 \\ 2 & s+3 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \frac{1}{s^2 + 3s + 2}.$$

5.4 Introduction to Stability

Objective of stability theory: Establish properties of trajectories subject to perturbations in the initial state.

To study the qualitative behaviour of a system, hence to describe the properties of its trajectories for all $t \in \mathbb{T}$ and for $t \to \infty$, we introduce the notion of stability. In addition, this notion allows to study the behaviour of trajectories close to an equilibrium point or to a certain motion.

Remark 5.4.1. The notion of stability that we discuss has been introduced in 1882 by the Russian mathematician A.M. Lyapunov, in its Doctoral Thesis (hence it is often referred to as Lyapunov stability). There are other notions of stability due for example to Lagrange (Lagrange stability) or introduced in the past 20 years. Nevertheless, the concept of Lyapunov stability is the most commonly used in applications.

Consider a nominal trajectory $x_n(t)$ starting from $x_n(0)$ and a perturbed trajectory $x_p(t)$ starting from $x_p(0)$. We want to find the relation between $\|x_p(t) - x_n(t)\|$ and $\|x_p(0) - x_n(0)\|$. In the following, we will focus on the case in which the nominal trajectory is an equilibrium point x_e.

Definition 5.4.2. *An equilibrium point x_e is **Lyapunov stable** if for any $\epsilon > 0$, there exists a $\delta = \delta(\epsilon, t_0) > 0$ such that for any initial condition $x(t_0)$ satisfying*

$$\|x(t_0) - x_e\| < \delta,$$

the corresponding solution $x(t)$ satisfies

$$\|x(t) - x_e\| < \epsilon$$

for all $t \geq t_0$.

Important observations:

- ϵ is arbitrary: no matter how small a neighborhood (ball $B_\epsilon(x_e)$) around x_e you desire the trajectory to stay in, you can find an initial neighborhood (ball $B_\delta(x_e)$) such that starting within $B_\delta(x_e)$ guarantees remaining in $B_\epsilon(x_e)$.
- Note that $\delta \leq \epsilon$, or the condition is already violated at $t = t_0$.
- An equilibrium point is unstable if it is not stable.

In stability theory, the quantity $x(t_0) - x_e$ is called initial perturbation, and $x(t) = \phi(t, t_0, x_0, 0)$ is called perturbed evolution.

Therefore, an equilibrium x_e is stable if for any neighborhood of x_e (even very small) the perturbed evolution stays within this neighborhood for all initial perturbations belonging to a sufficiently small neighborhood of x_e.

The definition of stability can be interpreted as follows. An equilibrium point x_e is stable if however we select a *tolerable* deviation ϵ, there exists a (sufficiently small) region with the equilibrium x_e in its interior, such that all initial perturbations in this region give rise to trajectories which are within the *tolerable* deviation.

Remark 5.4.3. The constant δ is in general a function of t_0. If it is possible to define a δ which does not depend upon t_0, we say that the equilibrium is uniformly stable. Note that if an equilibrium of a time-invariant system is stable, it is also uniformly stable.

The property of stability dictates a condition on the free evolution of the system for all $t \geq t_0$. Note, however, that in the definition of stability we have not requested that the perturbed evolution converges, for $t \to \infty$, to x_e. This convergence property is very important in applications, as it allows to characterize the situation in which not only the perturbed evolution remains close to the unperturbed evolution, but it also converges to the initial (unperturbed) evolution. To capture this property we introduce new definitions.

Definition 5.4.4. *Given an equilibrium point x_e, there exists a $\delta_a > 0$ (often different from the δ for stability) such that if*

$$\|x(0) - x_e\| < \delta_a \quad \Rightarrow \quad \lim_{t \to \infty} x(t) = x_e.$$

The set of initial conditions for which trajectories converge to x_e is the **region of attraction**.

Definition 5.4.5. *An equilibrium point x_e is **asymptotically stable** if:*

1. *It is stable.*
2. *It is attractive.*

In summary, an equilibrium point is asymptotically stable if it is stable and whenever the initial perturbation is inside a certain neighborhood of x_e the perturbed evolution converges, as $t \to \infty$, to the equilibrium point, which is said to be attractive. From a physical point of view this means that all sufficiently small initial perturbations give rise to effects which can be a-priori bounded (stability) and these vanish as $t \to \infty$ (convergence).

It is important to realize that convergence does not imply stability: it is possible to have an equilibrium of a system which is not stable (i.e. it is unstable), yet for all initial perturbations the perturbed evolution converges to the equilibrium.

Stability is a property of an equilibrium point (or a motion), not of the system itself, unless the system is linear. For LTI systems, if one equilibrium point (typically the origin) is stable/asymptotically stable, then all equilibrium points (if others exist, which is rare for typical LTI systems unless $\det(A) = 0$) share this property. Effectively, the system behavior is uniform. Hence, for LTI systems, one often says "the system is stable".

The choice of norm (e.g., Euclidean ℓ_2-norm, ℓ_1-norm, ℓ_∞-norm) does not affect the stability definitions, due to the equivalence of norms in finite-dimensional spaces.

5.4.1 Examples

Example 1 (Stable but not Attractive - DT). Consider the system $x(k+1) = -x(k)$. For this system the equilibrium point satisfies $x_e = -x_e \implies 2x_e = 0 \implies x_e = 0$. If $x(0) \neq 0$, then $x(1) = -x(0)$, $x(2) = -x(1) = x(0)$, etc. The trajectory oscillates between $x(0)$ and $-x(0)$. Let check if the equilibrium $x_e = 0$ is stable. For any $\epsilon > 0$, choose $\delta = \epsilon$. If $|x(0) - 0| < \delta$, then $|x(k) - 0| = |x(0)| < \delta = \epsilon$ for all $k \geq 0$. So, $x_e = 0$ is stable. However, $\lim_{k\to\infty} x(k)$ does not converge to 0 (unless $x(0) = 0$). So, $x_e = 0$ is not attractive, and thus not asymptotically stable.

Example 2 (Attractive but not Stable - DT). Consider the system

$$x(k+1) = \begin{cases} 2x(k) & \text{if } |x(k)| < 1 \\ 0 & \text{if } |x(k)| \geq 1 \end{cases}$$

Equilibrium point: If $|x_e| < 1$: $x_e = 2x_e \implies x_e = 0$. If $|x_e| \geq 1$: $x_e = 0$. This is a contradiction since $0 < 1$. So, $x_e = 0$ is the only equilibrium point. Attractivity: If $x(0) \neq 0$ and $|x(0)| < 1$, then $x(k)$ will eventually become ≥ 1 (since it doubles). Once $|x(k)| \geq 1$, $x(k+1) = 0$, and $x(j) = 0$ for $j > k+1$. So, $\lim_{k\to\infty} x(k) = 0$. The origin is attractive. Stability: Consider $\epsilon < 1$, e.g., $\epsilon = 0.5$. For any $\delta > 0$ (no matter how small, e.g. $\delta < \epsilon$), if $0 < |x(0)| < \delta$, the trajectory $x(k) = 2^k x(0)$ will grow. Eventually $2^k x(0)$ will exceed ϵ (and even 1) before returning to 0. For example, if $x(0) = \delta/2$, then $x(1) = \delta, x(2) = 2\delta, \ldots$. Even if δ is small, $x(k)$ will eventually exceed $\epsilon = 0.5$. Thus, $x_e = 0$ is not stable.

Example 3 (CT - System with circular trajectories). Consider the system $\dot{x}_1 = \psi(x_1, x_2)x_2$ $\dot{x}_2 = -\psi(x_1, x_2)x_1$, where $\psi(x_1, x_2) > 0$ is some function. Equilibrium point: $\psi x_2 = 0$ and $-\psi x_1 = 0$; since $\psi > 0$, this implies $x_1 = 0, x_2 = 0$. So $x_e = (0,0)^T$ is the only equilibrium point. Consider $V(x_1, x_2) = \frac{1}{2}(x_1^2 + x_2^2)$. $\dot{V} = x_1\dot{x}_1 + x_2\dot{x}_2 = x_1(\psi x_2) + x_2(-\psi x_1) = \psi x_1 x_2 - \psi x_1 x_2 = 0$. Since $\dot{V} = 0$, $V(x_1(t), x_2(t))$ is constant along trajectories. So, $x_1^2(t) + x_2^2(t) = x_1^2(0) + x_2^2(0) = R^2$. Trajectories are circles around the origin. Stability: Given $\epsilon > 0$. Choose $\delta = \epsilon$. If $\|x(0)\| < \delta$, then $\|x(t)\| = \|x(0)\| < \delta = \epsilon$. So, $x_e = 0$ is stable. Not attractive: trajectories do not approach the origin (unless $x(0) = 0$).

Example 4 (CT - Asymptotically Stable System). Consider the system $\dot{x}_1 = \psi(x_1, x_2)x_2 - x_1$ $\dot{x}_2 = -\psi(x_1, x_2)x_1 - x_2$, where $\psi(x_1, x_2) > 0$ is some function. We can show again that $x_e = (0,0)^T$ is the only equilibrium point. Consider $V(x_1, x_2) = \frac{1}{2}(x_1^2 + x_2^2)$. $\dot{V} = x_1\dot{x}_1 + x_2\dot{x}_2 = x_1(\psi x_2 - x_1) + x_2(-\psi x_1 - x_2)$ $= \psi x_1 x_2 - x_1^2 - \psi x_1 x_2 - x_2^2 = -(x_1^2 + x_2^2) = -2V$. So $\dot{V} = -2V$. This implies $V(t) = V(0)e^{-2t}$. $x_1^2(t) + x_2^2(t) = (x_1^2(0) + x_2^2(0))e^{-2t}$. $\|x(t)\|_2^2 = \|x(0)\|_2^2 e^{-2t} \implies \|x(t)\|_2 = \|x(0)\|_2 e^{-t}$. The trajectory converges to the origin exponentially. This implies asymptotic stability. Stability: Given $\epsilon > 0$. Choose

$\delta = \epsilon$. If $\|x(0)\| < \delta$, then $\|x(t)\| = \|x(0)\| e^{-t} < \delta e^{-t} \leq \delta = \epsilon$. Stable. Attractivity: $\lim_{t\to\infty} \|x(t)\|_2 = \lim_{t\to\infty} \|x(0)\|_2 e^{-t} = 0$. Attractive. So, $x_e = 0$ is asymptotically stable.

Example 5 (CT - Unstable System: Double Integrator). Consider the system

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= 0. \end{aligned}$$

We recall that the equilibrium point is $x_e = (c, 0)^T$ for any constant c. Let's consider $x_e = (0, 0)^T$ and recall that the solution is $x_2(t) = x_2(0)$. $x_1(t) = x_1(0) + x_2(0)t$. If $x_2(0) \neq 0$, $x_1(t)$ grows unboundedly. Let ϵ be any positive value. No matter how small $\delta > 0$ is chosen, if we pick $x(0) = (0, \delta/2)^T$, then $\|x(0)\| = \delta/2 < \delta$. The trajectory is $x_1(t) = (\delta/2)t$, $x_2(t) = \delta/2$. $\|x(t)\| = \sqrt{((\delta/2)t)^2 + (\delta/2)^2}$. This will exceed ϵ for large enough t. So, $x_e = (0, 0)^T$ is unstable.

5.4.2 Further Stability Concepts

Definition 5.4.6. *An equilibrium point x_e is **globally asymptotically stable** if it is stable, and it is attractive for* all *initial conditions $x(0) \in \mathbb{R}^n$. (Region of attraction is $\mathbb{R}^n$).*

Obviously, the property of global asymptotic stability is much stronger than the property of asymptotic stability (which is often referred to as local asymptotic stability), as it requires that the effect of **all** initial perturbations (no matter how large) vanishes as $t \to \infty$.

Definition 5.4.7. *An equilibrium point x_e is **exponentially stable** if it is stable, and there exist $\lambda > 0$ and $M > 0$ (and $\delta_M > 0$) such that if $\|x(0) - x_e\| < \delta_M$, then $\|x(t) - x_e\| \leq M \|x(0) - x_e\| e^{-\lambda t}$ for all $t \geq 0$.*

The property of exponential stability implies the property of uniform asymptotic stability and provides a rate of convergence.

Definition 5.4.8. *Consider a system $\{X, \phi, \eta\}$ and a nominal motion*

$$\mathcal{M} = \{(t, x_N(t)) \in \mathbb{T} \times \mathbb{X} \ : \ t \in \mathbb{T}, \ x_N(t) = \phi(t, t_0, x_N(t_0), u)\}.$$

The motion is stable if if for any $\epsilon > 0$, there exists a $\delta(\epsilon, t_0) > 0$ such that if

$$\|x_P(0) - x_N(0)\| < \delta,$$

then

$$\|x_P(t) - x_N(t)\| < \epsilon \tag{5.1}$$

for all $t \geq t_0$.

This means the perturbed trajectory $x_P(t)$ remains within an ϵ-tube around the nominal trajectory $x_N(t)$. Asymptotic stability of a motion can be similarly defined (stability + attractivity of $x_P(t)$ towards $x_N(t)$).

The notion of stability of motion is substantially similar to the notion of stability of an equilibrium. The important issue is that the time-parameterization is important, i.e. a motion is stable if, for small initial perturbations, for any $t \geq t_0$ the perturbed evolution is close to the non-perturbed evolution. This does not mean that if the perturbed and un-perturbed trajectories are close then the motion is stable: in fact the trajectories may be close, but may be followed with different timing, which means that for some $t \geq t_0$ condition (5.1) may be violated.

Chapter 6

Homework Solutions

Exercise 1.5.1 (COVID-19 Epidemiological Model)**.** The dynamics are described by a system of nonlinear ordinary differential equations (ODEs), $\dot{\mathbf{x}} = f(\mathbf{x})$, where each equation describes the flow of individuals between compartments (see Figure 1.2), namely

$$\begin{aligned}
\tfrac{dS}{dt} &= -S(t)\left(\alpha I(t) + \beta D(t) + \gamma A(t) + \delta R(t)\right) \\
\tfrac{dI}{dt} &= S(t)\left(\alpha I(t) + \beta D(t) + \gamma A(t) + \delta R(t)\right) - (\varepsilon + \zeta + \lambda) I(t) \\
\tfrac{dD}{dt} &= \varepsilon I(t) - (\eta + \rho) D(t) \\
\tfrac{dA}{dt} &= \zeta I(t) - (\theta + \mu + \kappa) A(t) \\
\tfrac{dR}{dt} &= \eta D(t) + \theta A(t) - (\nu + \xi) R(t) \\
\tfrac{dT}{dt} &= \mu A(t) + \nu R(t) - (\sigma + \tau) T(t) \\
\tfrac{dH}{dt} &= \lambda I(t) + \rho D(t) + \kappa A(t) + \xi R(t) + \sigma T(t) \\
\tfrac{dE}{dt} &= \tau T(t)
\end{aligned} \tag{6.1}$$

where $\alpha, \beta, \gamma, \delta, \varepsilon, \zeta, \ldots$ are various rates (infection, detection, recovery, etc.) to go from one compartment to another. The full system consists of an ODE for each compartment, forming a high-dimensional, nonlinear state-space model.

Exercise 1.5.2 (Satellite Dynamics)**.** We need to capture all variables and their derivatives up to one less than the highest order. Let:

- $x_1 = \theta$ (angular position)
- $x_2 = \dot{\theta}$ (angular velocity)
- $x_3 = r$ (radial distance)
- $x_4 = \dot{r}$ (radial velocity)

The state vector is $x = [x_1, x_2, x_3, x_4]^T$.

Now we find the derivatives of these states:

$$\begin{aligned}
\dot{x}_1 &= \dot{\theta} = x_2 \\
\dot{x}_2 &= \ddot{\theta} = -\frac{2x_2 x_4}{x_3} + \frac{1}{m x_3} u_2 \\
\dot{x}_3 &= \dot{r} = x_4 \\
\dot{x}_4 &= \ddot{r} = x_3 x_2^2 - \frac{k}{m x_3^2} + \frac{1}{m} u_1
\end{aligned}$$

This system is non-linear due to terms like $x_2 x_4 / x_3$, $1/x_3$, $x_3 x_2^2$, and $1/x_3^2$. It cannot be written in the simple matrix form $\dot{x} = Ax + Bu$.

However, we can separate the terms involving the state from those involving the input. This is often called an input-affine form: $\dot{x} = f(x) + g(x)u$. Let $u = [u_1, u_2]^T$.

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{pmatrix} = \underbrace{\begin{pmatrix} x_2 \\ -\frac{2x_2x_4}{x_3} \\ x_4 \\ x_3x_2^2 - \frac{k}{mx_3^2} \end{pmatrix}}_{f(x)} + \underbrace{\begin{pmatrix} 0 & 0 \\ 0 & \frac{1}{mx_3} \\ 0 & 0 \\ \frac{1}{m} & 0 \end{pmatrix}}_{g(x)} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$$

This form is common for many non-linear control problems. "Affine" refers to the way the input u enters the equations linearly, though $g(x)$ itself can be a non-linear function of the state x.

The choice of state variables is not unique. For example, swapping $x_1 \leftrightarrow x_3$ and $x_2 \leftrightarrow x_4$ would lead to a different, but equivalent, state-space representation (a change of coordinates).

Exercise 3.2.1 (Scholastic System)**.** To determine the equilibrium points for $u(k) \equiv \bar{u}$, we set the condition $x^+ = x$, yielding

$$\begin{aligned} x_1 &= (1-\alpha_1)x_1 + \bar{u} \\ x_2 &= (1-\alpha_2)x_2 + \alpha_1 x_1 \\ x_3 &= (1-\alpha_3)x_3 + \alpha_2 x_2 \end{aligned}$$

Solving the first equation yields

$$x_1 = \frac{\bar{u}}{\alpha_1}.$$

By replacing this value in the second equation and solving with respect to x_2 yields

$$x_2 = \frac{\bar{u}}{\alpha_2}.$$

Repeating this computation for the last equation yields

$$x_3 = \frac{\bar{u}}{\alpha_3}.$$

Thus, the equilibrium point is $(\frac{\bar{u}}{\alpha_1}, \frac{\bar{u}}{\alpha_2}, \frac{\bar{u}}{\alpha_3})$.

Exercise 3.2.2 (Lorenz System)**.** Equilibrium points are found by setting the time derivatives to zero, namely

$$\sigma(y - x) = 0 \tag{6.2}$$
$$\rho x - y - xz = 0 \tag{6.3}$$
$$xy - \beta z = 0 \tag{6.4}$$

From (6.2), assuming $\sigma \neq 0$, we have

$$y = x \tag{6.5}$$

Substituting (6.5) into (6.3) yields

$$\rho x - x - xz = 0 \quad \Rightarrow \quad x(\rho - 1 - z) = 0$$

This equation gives two cases.

Case 1: $x = 0$**.** If $x = 0$, then from (6.5), $y = 0$. Substitute $x = 0, y = 0$ into (6.4):

$$\begin{aligned} 0 \cdot 0 - \beta z &= 0 \\ -\beta z &= 0 \end{aligned}$$

Since $\beta > 0$, we must have $z = 0$. Thus, one equilibrium point is $E_0 = (0, 0, 0)$.

Case 2: $\rho - 1 - z = 0$**.** This implies

$$z = \rho - 1. \tag{6.6}$$

For this to be physically meaningful in the context of z representing a deviation from linear temperature profile, z is typically non-negative, so $\rho \geq 1$. Now substitute $y = x$ and $z = \rho - 1$ into (6.4):

$$\begin{aligned} x(x) - \beta(\rho - 1) &= 0 \\ x^2 - \beta(\rho - 1) &= 0 \\ x^2 &= \beta(\rho - 1) \end{aligned}$$

For real solutions for x, we require $\beta(\rho - 1) \geq 0$. Since $\beta > 0$, this means $\rho - 1 \geq 0$, or $\rho \geq 1$. If $\rho > 1$:

$$x = \pm\sqrt{\beta(\rho - 1)} \tag{6.7}$$

Since $y = x$, we have $y = \pm\sqrt{\beta(\rho - 1)}$. Thus, for $\rho > 1$, there are two additional equilibrium points:

$$\begin{aligned} E_1 &= \left(\sqrt{\beta(\rho - 1)}, \sqrt{\beta(\rho - 1)}, \rho - 1\right) \\ E_2 &= \left(-\sqrt{\beta(\rho - 1)}, -\sqrt{\beta(\rho - 1)}, \rho - 1\right) \end{aligned}$$

If $\rho = 1$, then $x = 0, y = 0, z = 0$, which collapses to the E_0 equilibrium point.

Summary of Equilibrium Points:

- If $0 < \rho \leq 1$: One equilibrium point $E_0 = (0, 0, 0)$.
- If $\rho > 1$: Three equilibrium points E_0, E_1, E_2.

The change in the number of equilibrium points as the parameter ρ varies is an example of a *bifurcation*. The Lorenz system is famous for exhibiting chaotic behavior for certain parameter values (e.g., $\sigma = 10, \beta = 8/3, \rho = 28$). Trajectories evolve around the non-origin equilibrium points but do not converge to them, displaying sensitive dependence on initial conditions. You can read more about this at https://en.wikipedia.org/wiki/Lorenz_system