Control Instrumentation System (DI03000181)

GTU Course Presentation

Table of Contents I

1. 1.1 Background
2. 1.2 Important Term Definition
3. 1.3 Classification of Control System
4. Lecture 4: 1.4 Open Loop Control System
5. 1.5 Closed Loop Control System
6. Lecture 6: 1.6 Comparison of Open loop and Closed loop Control System
7. 1.7 Requirements of an Ideal Control System

Table of Contents II

8. 2.1 Transfer Function

9. 2.1 Transfer Function

10. Lecture 9: Transfer Function

11. 2.2 Methods of finding Transfer function

12. Lecture 11: Methods of finding Transfer function

13. 2.3 Block Diagram Reduction (BDR)

14. Lecture 13: 2.3 Block Diagram Reduction (BDR)

Table of Contents III

15. Lecture 14: 2.4 Signal Flow Graph (SFG)
16. Lecture 15: 2.4 Signal flow graph(SFG)
17. 2.4 Signal Flow Graph (SFG)
18. Lecture 17: Comparison of BDR & SFG
19. 2.5 Comparison of BDR & SFG
20. Lecture 19: 2.6 System analogy
21. Lecture 20: 2.6 System Analogy

Table of Contents IV

22 Lecture 21: Time response: Transient & Steady State Response

23 3.1 Time Response: Transient and Steady State Response

24 3.2 Standard test signals

25 3.2 Standard test signals

26 Lecture 25: 3.3 Time response of first order system to step input

27 Lecture 26: Time Response of First Order System to Step Input

28 Lecture 27: 3.4 Time response of second order system to step input

Table of Contents V

29. Lecture 28: 3.4 Time response of second order system to step input

30. 3.5 Steady state error & error constant

31. Lecture 30: 3.5 Steady state error & error constant

32. Lecture 31: 4.1 Stability

33. 4.1 Stability

34. Lecture 33: 4.2 Routh stability criteria

35. Lecture 34: 4.2 Routh Stability Criteria

Table of Contents VI

36 4.4 Introduction to Root Locus Concept

37 Lecture 36: 4.4 Introduction to Root Locus Concept

38 4.5 Rules for Construction of Root Locus

39 Lecture 38: Rules for Construction of Root Locus

40 Lecture 39: 4.6 Simple examples of Root Locus

41 Lecture 40: 5.1 Introduction to Frequency Response Analysis

42 5.2 Concept of Bode plot in brief

Table of Contents VII

43 Lecture 42: Log Scales & Standard Form of $GH(j\omega)$

44 5.5 Bode Plot of Standard Factors

45 5.6 Frequency Domain Specification

46 Lecture 45: 5.7 Steps for solving bode plots

Introduction to Control Instrumentation Systems

- Welcome to **Control Instrumentation System (DI03000181)**.
- Modern engineering relies heavily on automation to operate complex processes safely and efficiently.
- This course explores how we measure physical quantities (Instrumentation) and use those measurements to influence the behavior of dynamic systems (Control).

Why study this?

Every modern industry—from aerospace and automotive to chemical processing and robotics—depends on precise control and accurate instrumentation to function.

What is an Instrumentation System?

Instrumentation

Instrumentation refers to the art and science of measurement and control of process variables within a production, laboratory, or manufacturing area.

Core Functions:

- **Measurement:** Acquiring data from the physical world using sensors (e.g., temperature, pressure, flow).
- **Transmission:** Sending the measured signal to a remote location or processing unit.
- **Display/Recording:** Presenting the information for human operators or data logging.

What is a Control System?

Control System

A **Control System** is an interconnection of components forming a system configuration that will provide a desired system response.

Objective: To regulate the output of a system (the *plant*) to match a desired reference input (the *setpoint*), despite the presence of external disturbances.

- Without instrumentation, a control system is "blind." It cannot measure the current state of the process.
- Without control, instrumentation only monitors; it cannot actively improve or correct the process.

Basic Block Diagram

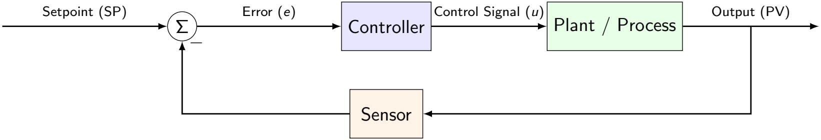


Figure: Closed-Loop Control System Architecture

Key Terminology

Understanding the language of control instrumentation is crucial:

- **Process Variable (PV):** The actual physical parameter being measured and controlled (e.g., room temperature, fluid pressure, motor speed).
- **Setpoint (SP):** The desired target value for the process variable.
- **Error (e):** The difference between the setpoint and the measured process variable ($e = SP - PV$).
- **Manipulated Variable (MV):** The parameter adjusted by the controller (via an actuator) to correct the error (e.g., valve position, heater voltage).
- **Disturbance:** An unwanted external signal or variation that affects the process output (e.g., opening a window in a heated room).

Historical Context: Evolution of Control I

The field of control systems has a rich history:

- **Antiquity:** Float regulator mechanisms in water clocks (Greece, $\sim$300 BC).
- **Industrial Revolution:** James Watt's centrifugal flyball governor (1788) for steam engine speed control. This was a purely mechanical, proportional controller.
- **Early 20th Century:** Development of pneumatic controllers (PID control logic) for the process industry.
- **Modern Era:** Digital control using microcontrollers, PLCs (Programmable Logic Controllers), and DCS (Distributed Control Systems).

Historical Context: Evolution of Control II

Shift in Paradigm

We have moved from purely mechanical and analog electronic instrumentation to highly networked, intelligent digital sensors and controllers.

Example: Home Heating System

Let's analyze a common household example:

- **System/Plant:** The interior of the house.
- **Process Variable (PV):** The actual room temperature.
- **Setpoint (SP):** The temperature you set on the thermostat (e.g., 22°C).
- **Sensor (Instrumentation):** A thermistor or thermocouple inside the thermostat.
- **Controller:** The comparator circuit or microchip in the thermostat.
- **Actuator:** The relay and the gas valve/heater in the furnace.

Mechanism: The sensor measures PV. The controller computes $e = SP - PV$. If $e > 0$ (too cold), it signals the actuator to turn on the heater (Manipulated Variable).

Open-Loop vs. Closed-Loop

Open-Loop Systems

- No feedback. The control action is independent of the output.
- Example: A toaster or a washing machine running on a set timer.
- **Pros:** Simple in design, cheaper, inherently stable.
- **Cons:** Cannot handle disturbances. If bread is frozen, it might not toast enough in the set time.

Closed-Loop Systems

- Uses feedback. Control action is dependent on the output.
- Example: Cruise control in a car.
- **Pros:** Highly accurate, can automatically correct for external disturbances (e.g., going up a hill).
- **Cons:** More complex, expensive, and can become unstable if not properly tuned.

Summary

- **Instrumentation** provides the sensory inputs required to understand physical processes.
- **Control engineering** uses these inputs to intelligently manipulate the process towards a desired state.
- Together, they form closed-loop systems that can adapt to changing conditions and disturbances.
- The fundamental architecture consists of a Plant, Sensor, Controller, and Actuator.
- As we progress through this course, we will mathematically model these components and design controllers to achieve desired performance metrics.

System and Plant

System

A system is a combination of components that act together and perform a certain objective. It is not limited to physical ones; it can refer to dynamic phenomena in economics, biology, etc.

Plant

A plant is a piece of equipment, perhaps just a set of machine parts functioning together, the purpose of which is to perform a particular operation. In control systems, it is the process or system to be controlled.

- **Example:** A chemical reactor, an airplane, or a heating furnace.

Control System and Controller

Control System

A control system is an interconnection of components forming a system configuration that will provide a desired system response.

Controller

A controller is a component or a group of components that acts on the system (plant) to achieve the desired output. It generates an appropriate control signal based on the error.

Inputs, Outputs, and Disturbances I

Input (Reference)

The desired response or the reference signal that the system is expected to follow. It is the stimulus or excitation applied to a control system from an external source.

Output

The actual response obtained from a control system. It may or may not be equal to the specified response implied by the input.

Disturbance

A disturbance is a signal that tends to adversely affect the value of the output of a system. If it is generated within the system, it is an *internal disturbance*; if generated outside, it is an *external disturbance*.

System Variables I

Controlled Variable

The controlled variable is the quantity or condition that is measured and controlled. It is the output of the plant which we desire to maintain at a specified value.

Manipulated Variable

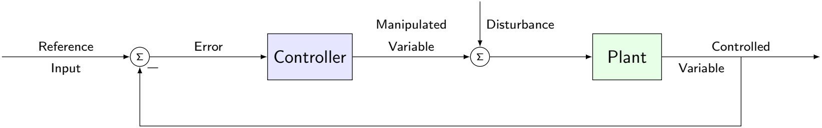
The manipulated variable is the quantity or condition that is varied by the controller so as to affect the value of the controlled variable.

System Variables II

Control Process

Control generally means measuring the value of the controlled variable and applying the manipulated variable to the plant to correct or limit deviation from the desired value.

Block Diagram of a General Control System



- The **Reference Input** sets the desired target.
- The **Controller** adjusts the **Manipulated Variable** based on the difference between target and actual output.
- **Disturbances** act on the plant, causing variations.
- The **Plant** produces the **Controlled Variable** (Output).

Classification of Control Systems

Control systems can be classified into several categories based on various criteria such as design, operation, mathematical models, and variable types.

- Natural and Man-made Systems
- Manual and Automatic Systems
- Open-loop and Closed-loop Systems
- Linear and Non-linear Systems
- Time-variant and Time-invariant Systems
- Continuous-data and Discrete-data Systems
- Deterministic and Stochastic Systems
- SISO and MIMO Systems

Natural and Man-made Systems I

Natural Control Systems

Systems created by nature to perform specific functions and maintain equilibrium within the environment or living organisms.

- **Example:** The human body temperature control system (perspiration to cool down, shivering to warm up).
- **Example:** Solar system, universe, human respiratory system.

Natural and Man-made Systems II

Man-made Control Systems

Systems designed, manufactured, and operated by humans to achieve a specific desired output.

- **Example:** Automobiles, aeroplanes, air conditioners, chemical plants.

Manual Control Systems

Control systems where human intervention or action is directly involved in the control action. The operator acts as the controller and feedback element.

- **Example:** Driving a car (human adjusts the steering wheel to stay on the road).
- **Example:** Manually regulating water flow in a pipe using a valve.

Manual and Automatic Systems II

Automatic Control Systems

Control systems where operations are performed automatically without continuous human intervention. The human operator's role is replaced by instruments, controllers, and actuators.

- **Example:** Room temperature control using a thermostat.
- **Example:** Automatic washing machine, cruise control in cars.

Open-loop and Closed-loop Systems I

Open-loop Control Systems

Systems in which the control action is entirely independent of the output. There is no feedback mechanism.

- Simple construction, cost-effective, and easy to maintain.
- Highly sensitive to external disturbances and environmental changes.
- **Example:** Traffic lights, electric toaster, automatic washing machine.

Closed-loop (Feedback) Control Systems

Systems in which the control action is dependent on the output. A portion of the output is fed back to the input to reduce the error.

- Complex construction and expensive.
- Highly robust against external disturbances and parameter variations.
- **Example:** Air conditioner, auto-pilot in an aircraft, human reaching for an object.

Linear and Non-linear Systems I

Linear Control Systems

A system is linear if it obeys the principles of **superposition** and **homogeneity** (proportionality). The relationship between input and output is strictly proportional. The differential equations describing the system are linear.

- **Superposition:** If input $x_1(t)$ yields $y_1(t)$ and $x_2(t)$ yields $y_2(t)$, then $x_1(t) + x_2(t)$ yields $y_1(t) + y_2(t)$.
- **Homogeneity:** If input $x(t)$ yields $y(t)$, then $a \cdot x(t)$ yields $a \cdot y(t)$ for any scalar a.

Linear and Non-linear Systems II

Perfectly linear systems do not exist in practice; all practical systems are linear only within a specific limited operating range.

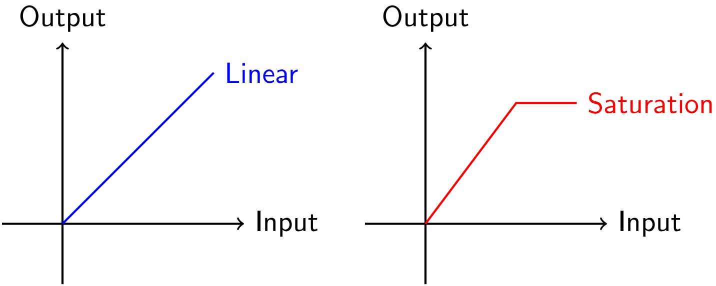
Non-linear Systems I

Non-linear Control Systems

A system that does **not** obey the principles of superposition and homogeneity. The relationship between input and output is not strictly proportional.

- The mathematical model involves non-linear differential equations (e.g., variables squared, trigonometric functions).
- System parameters may change with the magnitude of variables.
- Common non-linearities: Saturation, dead-zone, friction, backlash in gears.

Non-linear Systems II



Time-variant and Time-invariant Systems I

Time-invariant (Constant Parameter) Systems

A system whose parameters do not change with time. The response to a given input does not depend on the time at which the input is applied.

- Described by linear differential equations with **constant coefficients**.
- If input $x(t)$ gives output $y(t)$, then input $x(t-T)$ gives output $y(t-T)$.
- **Example:** An RLC electrical circuit with fixed component values.

Time-variant and Time-invariant Systems II

Time-variant Systems

A system whose parameters vary with time. The system's response depends on the exact time the input is applied.

- Described by differential equations with **time-varying coefficients**.
- **Example:** A spacecraft during flight (its mass decreases continuously as fuel is consumed).

Continuous-data and Discrete-data Systems I

Continuous-data (Analog) Control Systems

Systems where all variables (signals) are continuous functions of time. The signals exist at every instant of time.

- Often implemented using analog components (amplifiers, resistors, capacitors).
- Analyzed using Laplace transforms and differential equations.

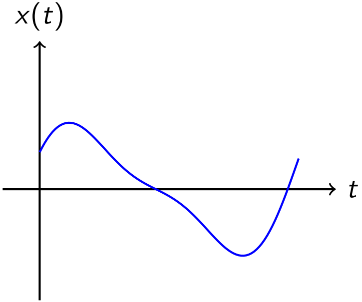
Discrete-data (Digital/Sampled-data) Control Systems

Systems where signals are discrete in time. Variables are known only at distinct, discrete instants of time.

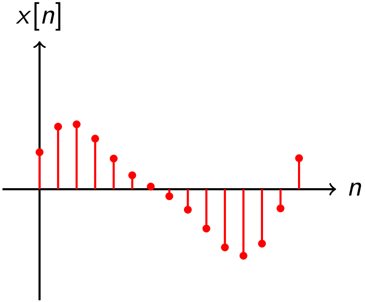
Continuous-data and Discrete-data Systems II

- Usually involves digital computers, microprocessors, or A/D and D/A converters.
- Systems are analyzed using Z-transforms and difference equations.
- **Example:** Computer Numerical Control (CNC) machines, digital flight controllers.

Signal Representations



Continuous-time



Discrete-time

Deterministic and Stochastic Systems I

Deterministic Control Systems

Systems in which the response to a specific input is completely predictable and repeatable.

- The system model is known exactly, and there are no uncertainties in the parameters or inputs.
- Most introductory control systems analysis assumes deterministic models.

Deterministic and Stochastic Systems II

Stochastic (Probabilistic) Control Systems

Systems subjected to random, unpredictable inputs, disturbances, or parameter variations.

- The system variables and signals are random and must be described using probability distributions and statistics.
- **Example:** Radar tracking system affected by thermal noise and atmospheric turbulence.

SISO and MIMO Systems I

SISO (Single-Input Single-Output) Systems

Systems possessing only one input and one output.

- Easiest to analyze and design.
- Transfer functions are typically used for SISO systems.
- **Example:** An electric motor where input is voltage and output is speed.

MIMO (Multiple-Input Multiple-Output) Systems

Systems possessing more than one input and more than one output.

SISO and MIMO Systems II

- There is often interaction or "coupling" between various inputs and outputs.
- State-space models are more suitable for analyzing MIMO systems.
- **Example:** A boiler where inputs are fuel rate and feedwater flow, and outputs are steam pressure and temperature.

Introduction to Open Loop Control Systems

Open Loop Control System

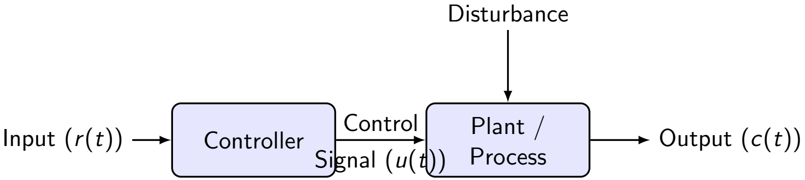
An **Open Loop Control System** (or non-feedback control system) is a system in which the control action is entirely independent of the output (or desired result). The output is neither measured nor fed back for comparison with the input.

Key Characteristics:

- The system operates on a preset timing or input sequence.
- There is no mechanism to correct errors if the output deviates from the desired value.
- It heavily relies on accurate calibration to achieve the desired performance.

Block Diagram of an Open Loop System

General Block Diagram:



- **Controller:** Generates the control signal based solely on the reference input.
- **Plant / Process:** The physical system being controlled.
- **Disturbance:** External factors that can affect the output. In an open-loop system, disturbances are not compensated for.

Advantages of Open Loop Systems I

- **Simplicity & Cost-Effectiveness:**
  - They have fewer components as they lack sensors for measurement and feedback loops.
  - Consequently, they are less expensive to design, manufacture, and maintain.
- **Stability:**
  - Generally more stable than closed-loop systems. There is no feedback mechanism that could inadvertently cause oscillations or instability if not properly tuned.
- **Ease of Construction & Maintenance:**
  - Straightforward architecture makes troubleshooting and repairs easier.

Advantages of Open Loop Systems II

When to use?

Open loop systems are ideal when the relationship between input and output is well-known, and external disturbances are negligible.

Disadvantages of Open Loop Systems

- **Inaccuracy:**
  - The system cannot correct for internal variations or external disturbances. If a disturbance alters the output, the system will not adjust.
- **Requires Recalibration:**
  - Component wear, aging, and environmental changes (like temperature drift) affect performance. Frequent manual calibration is required to maintain accuracy.
- **Sensitivity to Disturbances:**
  - Highly susceptible to both internal parameter variations (e.g., a motor losing torque) and external noise.

Real-Time Engineering Examples

1. **Electric Washing Machine**
   - **Operation:** The machine runs through wash, rinse, and spin cycles for pre-set times based on the user's initial selection.
   - **Why Open Loop?** It does not measure how clean the clothes actually are. If clothes are still dirty after the cycle, the machine won't automatically extend the wash time.
2. **Traffic Light System (Time-Based)**
   - **Operation:** Lights change colors at fixed time intervals regardless of the actual traffic volume.
   - **Limitation:** During heavy traffic or off-peak hours, the fixed timing may cause unnecessary delays since it doesn't adapt to real-time vehicle flow.

Real-Time Engineering Examples (Cont.)

3. **Stepper Motor Control**
   - **Operation:** The controller sends a specific number of pulses to rotate the motor by a desired angle.
   - **Why Open Loop?** Standard stepper setups do not use an encoder to verify if the motor actually reached the position (e.g., if it missed steps due to high load).

4. **Bread Toaster**
   - **Operation:** Heating elements are energized for a specific time based on the browning dial setting.
   - **Limitation:** It doesn't sense the actual color of the toast. If the bread is already slightly warm or dry, it might burn, but the toaster will still run for the full preset time.

Closed Loop Control System

Closed Loop (Feedback) Control System

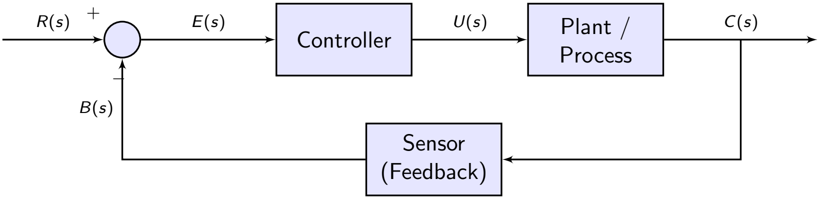
A system in which the control action is dependent on the output is called a closed loop control system. It utilizes feedback to compare the actual output with the desired output (reference input) and generates an error signal to reduce the deviation.

Key Characteristics:

- Also known as **Feedback Control Systems**.
- The output is continuously monitored and fed back to the input.
- The system automatically corrects for external disturbances.
- More accurate and reliable than open loop systems, but at the cost of increased complexity.

Block Diagram of a Closed Loop System I

The general block diagram consists of several key components: the reference input, error detector, controller, plant (process), and the feedback element.



Block Diagram of a Closed Loop System II

Error Signal

Equation: $E(s) = R(s) - B(s)$, where $R(s)$ is the reference input and $B(s)$ is the feedback signal. The controller adjusts its output $U(s)$ to drive $E(s)$ to zero.

Advantages and Disadvantages

Advantages:

- **Accuracy:** Highly accurate due to feedback mechanism.
- **Robustness:** Less sensitive to external disturbances and parameter variations.
- **Bandwidth:** Increased bandwidth, providing faster response to changes.
- **Automation:** Operates automatically without continuous human intervention.

Disadvantages:

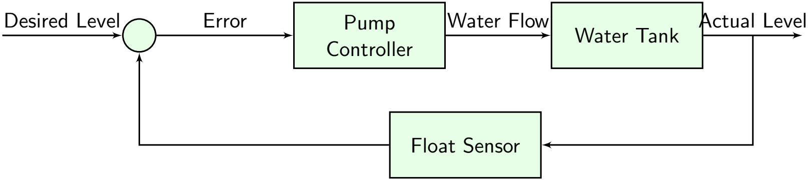
- **Complexity:** More complex design compared to open loop systems.
- **Cost:** More expensive due to additional components like sensors and error detectors.
- **Stability:** Feedback can introduce oscillatory behavior, making the system potentially unstable if not designed properly.
- **Maintenance:** Requires more maintenance due to higher component count.

Real Time Examples

Closed loop systems are ubiquitous in modern engineering and daily life.

1. **Air Conditioner (AC)**
   - **Goal:** Maintain a desired room temperature.
   - **Feedback:** A thermostat measures current room temperature.
   - **Action:** If the room is warmer than the setpoint, the compressor turns on. Once the setpoint is reached, it turns off.
2. **Cruise Control System in Automobiles**
   - **Goal:** Maintain a constant vehicle speed.
   - **Feedback:** Speed sensor measures actual speed.
   - **Action:** The controller adjusts the engine throttle based on the difference between desired speed and actual speed, overcoming gradients or wind resistance.

3. Automatic Water Level Controller



- **Operation:** The float sensor continuously monitors the water level. When it drops below the threshold, the pump is turned on; when full, it is turned off.

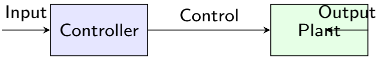
Introduction to System Comparison

- Control systems are broadly classified into **Open-loop** and **Closed-loop** systems.
- The fundamental difference lies in the presence or absence of **feedback**.
- Choosing the right control strategy depends on the required accuracy, stability, environmental disturbances, and cost constraints.

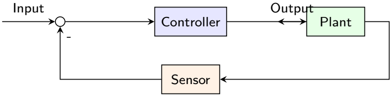
Feedback Mechanism

Feedback is the process of measuring the output and returning a portion of it to the input for comparison, forming a closed loop to correct errors automatically.

Open-Loop System (Without Feedback)



Closed-Loop System (With Feedback)



Comparison: Construction & Stability

Feature	Open-Loop System	Closed-Loop System
Feedback	Absent	Present
Construction	Simple and easy to build.	Complex due to feedback components.
Cost	Generally cheaper.	More expensive.
Stability	Inherently stable.	Prone to instability if feedback is not properly tuned.

Comparison: Accuracy & Maintenance

Feature	Open-Loop System	Closed-Loop System
Accuracy	Inaccurate; depends on calibration.	Highly accurate due to error correction.
Disturbances	Highly sensitive to internal/external disturbances.	Less sensitive; can reject disturbances automatically.
Maintenance	Easy to maintain.	Difficult to maintain due to complexity.
Example	Toaster, automatic washing machine.	Air conditioner, cruise control.

Decision Matrix: Which to choose?

- **Use Open-Loop when:**
  - Disturbances are minimal or predictable.
  - High precision is not strictly required.
  - Cost and simplicity are the primary constraints.
- **Use Closed-Loop when:**
  - The environment is unpredictable (wind, temperature changes).
  - High accuracy and reliability are non-negotiable.
  - The system must autonomously correct itself (automation).

What Makes an Ideal Control System?

An ideal control system must satisfy a strict set of performance criteria to guarantee safety, reliability, and desired operation.

Ideal Control System

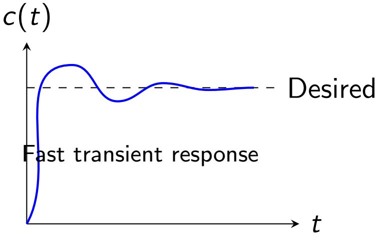
An ideal control system is one that perfectly tracks the desired input command with zero delay, zero error, and absolute immunity to noise and parameter variations. (In reality, we only approximate this ideal).

Key Requirements: Accuracy & Stability

- **Accuracy:**
  - The system should reach and maintain the desired output value precisely.
  - The steady-state error must be ideally zero.
  - High accuracy often demands high open-loop gain in closed-loop systems.
- **Stability:**
  - The most critical requirement. A system is stable if its output remains bounded for a bounded input (BIBO stability).
  - The system must not oscillate uncontrollably or diverge over time.
  - Increasing accuracy can sometimes compromise stability (the fundamental trade-off in control theory).

Key Requirements: Speed of Response I

- **Speed of Response (Transient Response):**
  - The system should react quickly to changes in the input command.
  - Measured by metrics such as **Rise Time (t_r)** and **Settling Time (t_s)**.
  - **Example:** In missile guidance, a sluggish response results in failure. In an elevator, too fast a response causes passenger discomfort.



Key Requirements: Sensitivity & Noise Immunity

- **Sensitivity:**
  - An ideal system should be highly *insensitive* to internal parameter variations (due to aging, temperature, wear and tear).
  - A closed-loop system is mathematically proven to reduce sensitivity by a factor of $(1 + GH)$.
- **Noise Immunity:**
  - Control systems operate in electrically noisy environments.
  - The system must reject unwanted signals (sensor noise, thermal noise) without affecting the output.

Key Requirements: Bandwidth

- **Bandwidth:**
  - Defined as the range of frequencies over which the system can track the input signal effectively.
  - A large bandwidth allows the system to track rapidly varying inputs (high speed of response).
  - However, excessive bandwidth makes the system highly susceptible to high-frequency noise.

The Engineering Compromise

Designing an ideal control system is impossible. We must balance conflicting requirements: maximizing speed and accuracy while maintaining robust stability and minimizing noise susceptibility.

Transfer Function

The **Transfer Function** of a linear time-invariant (LTI) system is defined as the ratio of the Laplace transform of the output (response) variable to the Laplace transform of the input (driving) variable, under the assumption that all initial conditions are zero.

If the system input is $r(t)$ and output is $c(t)$, their Laplace transforms are $R(s)$ and $C(s)$ respectively:

$$G(s) = \frac{\mathcal{L}\{c(t)\}}{\mathcal{L}\{r(t)\}} = \left.\frac{C(s)}{R(s)}\right|_{\text{initial conditions}=0}$$

Introduction to Transfer Function II

- It is a mathematical model representing the input-output relationship of a system in the s-domain.
- The transfer function $G(s)$ completely defines the dynamic behavior of the system, acting as a functional block connecting input to output.

Properties of Transfer Function I

Key Properties

- The transfer function is a property of the system itself, entirely independent of the magnitude and nature of the input or driving function.
- It relates the input to the output algebraically; however, it does not provide any information concerning the internal physical structure of the system.
- It is strictly defined only for **linear time-invariant (LTI)** systems. It is not defined for nonlinear or time-varying systems.
- The highest power of s in the denominator polynomial defines the **order** of the system.
- Once the transfer function is known, the system's response can be easily analyzed for various standard inputs (step, impulse, ramp).

Advantages and Disadvantages I

Advantages

- Converts complex time-domain differential equations into simple algebraic equations in the s-domain.
- Allows straightforward stability analysis by examining the roots of the denominator (poles).
- Facilitates the analysis of complex interconnected systems using block diagram reduction.
- Provides both transient and frequency response information simultaneously.

Disadvantages

- Cannot be applied to nonlinear or time-varying systems.
- Neglects initial conditions, which may be critical for certain system analyses.
- Treats the system as a "black box," offering no insight into the internal state variables.

Advantages and Disadvantages II

- Primarily limited to Single-Input Single-Output (SISO) systems, though expandable via matrices.

Poles, Zeros, and Characteristic Equation I

A transfer function can typically be expressed as a rational function of two polynomials in s:

$$G(s) = \frac{N(s)}{D(s)} = \frac{K(s - z_1)(s - z_2)\dots(s - z_m)}{(s - p_1)(s - p_2)\dots(s - p_n)}$$

- **Zeros (z_i):** The roots of the numerator polynomial $N(s) = 0$. At these values of s, the transfer function evaluates to zero.
- **Poles (p_i):** The roots of the denominator polynomial $D(s) = 0$. At these values of s, the transfer function evaluates to infinity.
- **Characteristic Equation:** The equation formed by setting the denominator polynomial to zero, i.e., $D(s) = 0$.

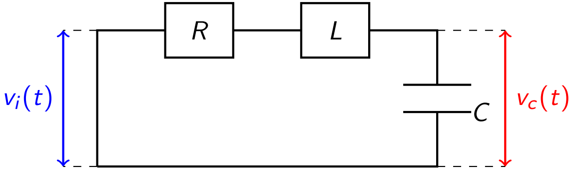
Poles, Zeros, and Characteristic Equation II

System Stability

The location of the poles on the complex s-plane dictates system stability. For an LTI system to be stable, all poles must lie strictly in the **left half** of the s-plane.

Transfer Function of Electrical Systems: RLC Series I

Consider a standard RLC series circuit with input voltage $v_i(t)$ and output voltage $v_c(t)$ taken across the capacitor.



Applying Kirchhoff's Voltage Law (KVL) around the loop:

$$v_i(t) = Ri(t) + L\frac{di(t)}{dt} + \frac{1}{C}\int i(t)dt$$

Since the current $i(t) = C\frac{dv_c(t)}{dt}$, we substitute it into the KVL equation.

Transfer Function of RLC Series Circuit I

Substituting $i(t)$ into the KVL equation yields the governing differential equation:

$$v_i(t) = RC\frac{dv_c(t)}{dt} + LC\frac{d^2v_c(t)}{dt^2} + v_c(t)$$

Taking the Laplace transform with zero initial conditions:

$$V_i(s) = RCsV_c(s) + LCs^2V_c(s) + V_c(s)$$

$$V_i(s) = \left(LCs^2 + RCs + 1\right) V_c(s)$$

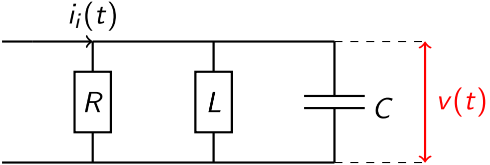
The Transfer Function is the ratio of output to input:

$$G(s) = \frac{V_c(s)}{V_i(s)} = \frac{1}{LCs^2 + RCs + 1} = \frac{\frac{1}{LC}}{s^2 + \frac{R}{L}s + \frac{1}{LC}}$$

- Characteristic equation: $s^2 + \frac{R}{L}s + \frac{1}{LC} = 0$
- This represents a standard second-order system.

Transfer Function of RLC Parallel Circuit I

Consider an RLC parallel circuit driven by a current source $i_i(t)$ with output voltage $v(t)$.



Applying Kirchhoff's Current Law (KCL):

$$i_i(t) = i_R(t) + i_L(t) + i_C(t) = \frac{v(t)}{R} + \frac{1}{L}\int v(t)dt + C\frac{dv(t)}{dt}$$

Taking the Laplace transform with zero initial conditions:

$$I_i(s) = \left(\frac{1}{R} + \frac{1}{sL} + sC\right)V(s) = \left(\frac{sL + R + s^2RLC}{sRL}\right)V(s)$$

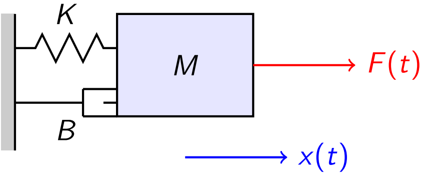
Transfer Function of RLC Parallel Circuit II

The Transfer Function (Impedance) is:

$$G(s) = \frac{V(s)}{I_i(s)} = \frac{sRL}{RLCs^2 + sL + R} = \frac{\frac{s}{C}}{s^2 + \frac{1}{RC}s + \frac{1}{LC}}$$

Mechanical Translational Systems I

Translational mechanical systems are modeled using three fundamental components: Mass (M), Spring (K), and Damper (B).



By Newton's Second Law of Motion:

$$\sum F = Ma \implies F(t) - F_K - F_B = M\frac{d^2x(t)}{dt^2}$$

$$F(t) - Kx(t) - B\frac{dx(t)}{dt} = M\frac{d^2x(t)}{dt^2}$$

Transfer Function of Translational System I

The differential equation governing the translational system is:

$$M\frac{d^2x(t)}{dt^2} + B\frac{dx(t)}{dt} + Kx(t) = F(t)$$

Taking the Laplace transform with zero initial conditions:

$$\left(Ms^2 + Bs + K\right) X(s) = F(s)$$

The Transfer Function relating displacement $X(s)$ to force $F(s)$ is:

$$G(s) = \frac{X(s)}{F(s)} = \frac{1}{Ms^2 + Bs + K}$$

Transfer Function of Translational System II

Force-Voltage Analogy

Notice the mathematical similarity to the RLC series circuit. This forms the basis of analogous systems:
Force $F \leftrightarrow$ Voltage V, Mass $M \leftrightarrow$ Inductance L,
Damper $B \leftrightarrow$ Resistance R, Spring $K \leftrightarrow$ Inverse Capacitance $1/C$.

Rotational mechanical systems involve angular motion and consist of: Moment of Inertia (J), Torsional Spring (K), and Rotational Damper (B).
Variables: Applied Torque $T(t)$, Angular displacement $\theta(t)$.
Newton's Second Law for Rotation:

$$\sum \text{Torques} = J\alpha \implies T(t) - B\frac{d\theta(t)}{dt} - K\theta(t) = J\frac{d^2\theta(t)}{dt^2}$$

Rearranging gives the differential equation:

$$J\frac{d^2\theta(t)}{dt^2} + B\frac{d\theta(t)}{dt} + K\theta(t) = T(t)$$

Taking the Laplace transform with zero initial conditions:

$$\left(Js^2 + Bs + K\right)\Theta(s) = T(s)$$

Transfer Function:

$$G(s) = \frac{\Theta(s)}{T(s)} = \frac{1}{Js^2 + Bs + K}$$

Transfer Function: Definition I

Transfer Function

The **transfer function** of a linear, time-invariant (LTI) system is defined as the ratio of the Laplace transform of the output (response) variable to the Laplace transform of the input (driving) variable, under the assumption that all initial conditions are zero.

Mathematically, if $x(t)$ is the input and $y(t)$ is the output of an LTI system:

- Input: $\mathcal{L}\{x(t)\} = X(s)$
- Output: $\mathcal{L}\{y(t)\} = Y(s)$

Transfer Function: Definition II

The Transfer Function $G(s)$ is given by:

$$G(s) = \left.\frac{Y(s)}{X(s)}\right|_{\text{initial conditions}=0}$$

Properties of Transfer Function I

- The transfer function is defined only for a **linear, time-invariant (LTI)** system. It is not defined for non-linear systems.
- It is defined strictly under **zero initial conditions**.
- The transfer function is completely independent of the input applied to the system. It is a property of the system itself.
- It is expressed as a ratio of two polynomials in the complex variable s (i.e., $s = \sigma + j\omega$).
- If the transfer function of a system is known, the output response for any given input can be easily determined.
- The roots of the denominator polynomial give the system's **poles**, which dictate the stability of the system.

Advantages and Disadvantages I

Advantages

- Simplifies complex differential equations into algebraic equations in the s-domain.
- Provides a clear representation of system dynamics and stability through poles and zeros.
- Enables block diagram reduction techniques for complex interconnected systems.
- Allows for easy calculation of system response to various standard test inputs (step, ramp, impulse).

Disadvantages

- Applicable only to LTI systems; fails for non-linear or time-varying systems.
- Assumes zero initial conditions, which may not always be practically true.
- Does not provide information about the internal state of the system, only the input-output relationship.
- Primarily limited to single-input single-output (SISO) systems, though extendable to MIMO with matrix formulations.

Poles, Zeros, and Characteristic Equation I

Let a rational transfer function be represented as:

$$G(s) = \frac{Y(s)}{X(s)} = \frac{N(s)}{D(s)} = K\frac{(s-z_1)(s-z_2)\cdots(s-z_m)}{(s-p_1)(s-p_2)\cdots(s-p_n)}$$

- **Zeros (z_i):** The roots of the numerator polynomial $N(s) = 0$. At these frequencies, the transfer function becomes zero.
- **Poles (p_i):** The roots of the denominator polynomial $D(s) = 0$. At these frequencies, the transfer function tends to infinity.

Poles, Zeros, and Characteristic Equation II

Characteristic Equation

The equation obtained by setting the denominator polynomial of the closed-loop transfer function to zero is called the **characteristic equation**.

$$D(s) = 0$$

The roots of the characteristic equation are the poles of the system, which determine system stability and transient response.

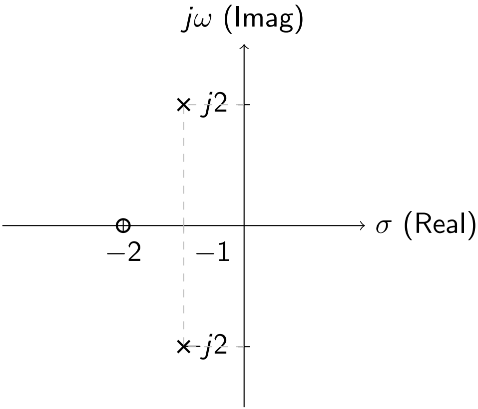
Pole-Zero Plot Example I

Consider the transfer function:

$$G(s) = \frac{s+2}{s^2+2s+5} = \frac{s+2}{(s+1-j2)(s+1+j2)}$$

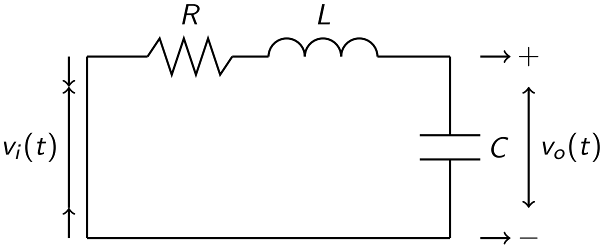
- **Zero:** $s = -2$
- **Poles:** $s = -1 \pm j2$

Pole-Zero Plot Example II



Standard RLC Series Circuit I

Consider a series RLC circuit with input voltage $v_i(t)$ and output voltage $v_o(t)$ taken across the capacitor.



Applying KVL in the time domain:

$$v_i(t) = Ri(t) + L\frac{di(t)}{dt} + \frac{1}{C}\int i(t)dt \quad \text{and} \quad v_o(t) = \frac{1}{C}\int i(t)dt$$

Standard RLC Series Circuit (Contd.) I

Taking the Laplace transform with zero initial conditions:

$$\begin{aligned} V_i(s) &= RI(s) + sLI(s) + \frac{1}{sC}I(s) \\ &= I(s)\left(R + sL + \frac{1}{sC}\right) \end{aligned}$$

And for the output:

$$V_o(s) = \frac{1}{sC}I(s) \implies I(s) = sCV_o(s)$$

Substituting $I(s)$ into the $V_i(s)$ equation:

$$V_i(s) = sCV_o(s)\left(R + sL + \frac{1}{sC}\right) = V_o(s)\left(s^2LC + sRC + 1\right)$$

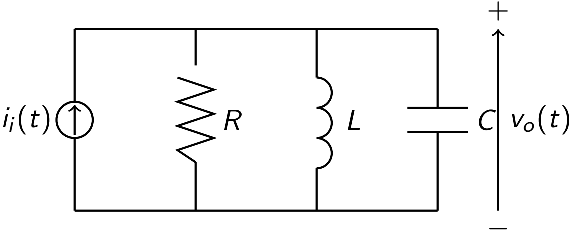
Standard RLC Series Circuit (Contd.) II

The transfer function is:

$$\frac{V_o(s)}{V_i(s)} = \frac{1}{LCs^2 + RCs + 1} = \frac{\frac{1}{LC}}{s^2 + \frac{R}{L}s + \frac{1}{LC}}$$

Standard RLC Parallel Circuit I

Consider a parallel RLC circuit excited by a current source $i_i(t)$, with output voltage $v_o(t)$.



By KCL: $i_i(t) = i_R(t) + i_L(t) + i_C(t)$

$$i_i(t) = \frac{v_o(t)}{R} + \frac{1}{L}\int v_o(t)dt + C\frac{dv_o(t)}{dt}$$

Taking the Laplace transform with zero initial conditions:

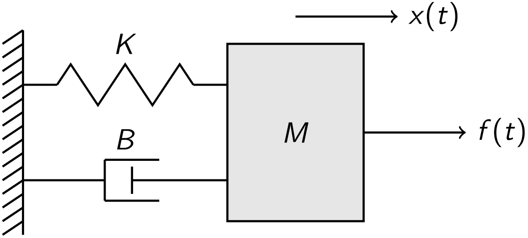
$$\begin{aligned} I_i(s) &= \frac{V_o(s)}{R} + \frac{V_o(s)}{sL} + sCV_o(s) \\ &= V_o(s)\left(\frac{1}{R} + \frac{1}{sL} + sC\right) \\ &= V_o(s)\left(\frac{sL + R + s^2RLC}{sRL}\right) \end{aligned}$$

The transfer function (Impedance $Z(s)$) is:

$$\frac{V_o(s)}{I_i(s)} = \frac{sRL}{s^2RLC + sL + R} = \frac{\frac{s}{C}}{s^2 + \frac{1}{RC}s + \frac{1}{LC}}$$

Mechanical Translational System I

Consider a standard mass-spring-damper system. A force $f(t)$ is applied to a mass M, which is attached to a fixed wall via a spring of stiffness K and a viscous damper with friction coefficient B.



The opposing forces are:

- Inertia Force: $f_m(t) = M\frac{d^2x(t)}{dt^2}$
- Damping Force: $f_b(t) = B\frac{dx(t)}{dt}$

- Spring Force: $f_k(t) = Kx(t)$

Mechanical Translational System (Contd.) I

Applying D'Alembert's Principle, the algebraic sum of external applied forces and opposing forces is zero:

$$f(t) = f_m(t) + f_b(t) + f_k(t)$$

$$f(t) = M\frac{d^2x(t)}{dt^2} + B\frac{dx(t)}{dt} + Kx(t)$$

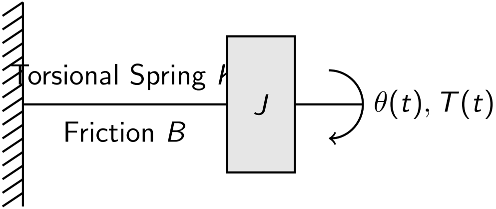
Taking the Laplace transform with zero initial conditions:

$$F(s) = Ms^2X(s) + BsX(s) + KX(s)$$
$$F(s) = X(s)(Ms^2 + Bs + K)$$

The transfer function is:

$$\frac{X(s)}{F(s)} = \frac{1}{Ms^2 + Bs + K}$$

Consider a rotational system with applied torque $T(t)$, Moment of Inertia J, rotational friction coefficient B, and torsional spring stiffness K. The angular displacement is $\theta(t)$.



The opposing torques are:

- Inertia Torque: $T_j(t) = J\frac{d^2\theta(t)}{dt^2}$
- Damping Torque: $T_b(t) = B\frac{d\theta(t)}{dt}$
- Spring Torque: $T_k(t) = K\theta(t)$

Applying Newton's Second Law for Rotation:

$$T(t) = T_j(t) + T_b(t) + T_k(t)$$

$$T(t) = J\frac{d^2\theta(t)}{dt^2} + B\frac{d\theta(t)}{dt} + K\theta(t)$$

Taking the Laplace transform with zero initial conditions:

$$T(s) = Js^2\Theta(s) + Bs\Theta(s) + K\Theta(s)$$
$$T(s) = \Theta(s)(Js^2 + Bs + K)$$

The transfer function is:

$$\frac{\Theta(s)}{T(s)} = \frac{1}{Js^2 + Bs + K}$$

Force-Voltage Analogy I

Notice the similarity between the Mechanical Translational System and the Electrical RLC Series Circuit:

Mechanical Translational	**Electrical Series**
Force, $f(t)$	Voltage, $v(t)$
Mass, M	Inductance, L
Viscous Friction, B	Resistance, R
Spring Stiffness, K	Reciprocal Capacitance, $1/C$
Displacement, $x(t)$	Charge, $q(t)$
Velocity, $v(t) = \dot{x}(t)$	Current, $i(t) = \dot{q}(t)$

What is a Transfer Function?

Transfer Function

The **Transfer Function** of a linear, time-invariant (LTI) system is defined as the ratio of the Laplace transform of the output variable to the Laplace transform of the input variable, under the assumption that all initial conditions are zero.

Mathematically, if $c(t)$ is the output and $r(t)$ is the input:

$$G(s) = \left.\frac{C(s)}{R(s)}\right|_{\text{initial conditions}=0}$$

- $G(s)$: Transfer function of the system
- $C(s) = \mathcal{L}\{c(t)\}$: Laplace transform of output
- $R(s) = \mathcal{L}\{r(t)\}$: Laplace transform of input

Properties of Transfer Function I

Properties of Transfer Function II

Key Properties

- The transfer function is a property of a system itself, independent of the magnitude and nature of the input or driving function.
- It includes the units necessary to relate the input to the output; however, it does not provide any information concerning the physical structure of the system.
- The transfer function is defined only for a linear time-invariant system. It is not defined for non-linear systems.
- All initial conditions of the system are assumed to be zero.
- If the transfer function is known, the output response can be studied for various types of inputs.

Advantages and Disadvantages I

Advantages:

- **Simplicity:** Converts complex time-domain differential equations into simple algebraic equations in the s-domain.
- **Stability Analysis:** The stability of the system can be easily determined using the denominator of the transfer function (poles).
- **System Response:** It is easy to find the output response for any given reference input.

Disadvantages:

- **Applicability:** Applicable only to Linear Time-Invariant (LTI) systems.
- **Initial Conditions:** It assumes all initial conditions are zero, which is not always true in real-world applications.

Advantages and Disadvantages II

- **Internal States:** It only provides an input-output relation and ignores the internal states of the system (unlike State-Space modeling).

Poles and Zeros

A general transfer function can be expressed as a ratio of two polynomials in s:

$$G(s) = \frac{C(s)}{R(s)} = \frac{K(s - z_1)(s - z_2)\dots(s - z_m)}{(s - p_1)(s - p_2)\dots(s - p_n)}$$

Zeros

Zeros $(z_1, z_2, \dots, z_m)$ are the roots of the numerator polynomial. When $s = z_i$, the transfer function becomes zero.

Poles

Poles $(p_1, p_2, \dots, p_n)$ are the roots of the denominator polynomial. When $s = p_i$, the transfer function approaches infinity.

Characteristic Equation

Characteristic Equation

The equation obtained by equating the denominator polynomial of a transfer function to zero is called the **Characteristic Equation** of the system.

For the transfer function:

$$G(s) = \frac{N(s)}{D(s)}$$

The characteristic equation is:

$$D(s) = 0$$

Significance: The roots of the characteristic equation are the poles of the system, which strictly determine the system's dynamic behavior and absolute stability.

Transfer Function of RLC Series Circuit

Consider a series RLC circuit with input voltage $v_i(t)$ and output voltage taken across the capacitor $v_c(t)$.

Time domain equations:

$$v_i(t) = Ri(t) + L\frac{di(t)}{dt} + \frac{1}{C}\int i(t)dt$$

$$v_c(t) = \frac{1}{C}\int i(t)dt$$

Taking Laplace Transform (assuming zero initial conditions):

$$V_i(s) = RI(s) + sLI(s) + \frac{1}{sC}I(s) = I(s)\left(R + sL + \frac{1}{sC}\right)$$

$$V_c(s) = \frac{1}{sC}I(s) \implies I(s) = sCV_c(s)$$

Transfer Function of RLC Series Circuit (Cont.)

Substituting $I(s)$ into the input equation:

$$V_i(s) = sCV_c(s)\left(R + sL + \frac{1}{sC}\right)$$

$$V_i(s) = V_c(s)\left(sRC + s^2LC + 1\right)$$

The transfer function is:

Series RLC Transfer Function

$$\frac{V_c(s)}{V_i(s)} = \frac{1}{s^2LC + sRC + 1} = \frac{\frac{1}{LC}}{s^2 + s\frac{R}{L} + \frac{1}{LC}}$$

Notice the standard second-order system format!

Transfer Function of RLC Parallel Circuit I

Consider a parallel RLC circuit driven by an input current source $i_{in}(t)$ and output voltage $v(t)$ across the common nodes.
KCL in Time domain:

$$i_{in}(t) = \frac{v(t)}{R} + C\frac{dv(t)}{dt} + \frac{1}{L}\int v(t)dt$$

Taking Laplace Transform:

$$I_{in}(s) = \frac{V(s)}{R} + sCV(s) + \frac{1}{sL}V(s)$$

$$I_{in}(s) = V(s)\left(\frac{1}{R} + sC + \frac{1}{sL}\right)$$

The Transfer Function (Impedance $Z(s) = \frac{V(s)}{I_{in}(s)}$) is:

Transfer Function of RLC Parallel Circuit II

Parallel RLC Transfer Function

$$G(s) = \frac{V(s)}{I_{in}(s)} = \frac{1}{sC + \frac{1}{R} + \frac{1}{sL}} = \frac{s}{C(s^2 + s\frac{1}{RC} + \frac{1}{LC})}$$

Mechanical Translational System

Basic elements of a mechanical translational system:

- **Mass (M):** Opposes acceleration. Force $f_M = M\frac{d^2x}{dt^2}$
- **Spring (K):** Opposes displacement. Force $f_K = Kx$
- **Damper/Dashpot (B):** Opposes velocity. Force $f_B = B\frac{dx}{dt}$

For a basic spring-mass-damper system with input force $f(t)$ and output displacement $x(t)$:

$$f(t) = M\frac{d^2x}{dt^2} + B\frac{dx}{dt} + Kx(t)$$

Taking Laplace Transform:

$$F(s) = Ms^2X(s) + BsX(s) + KX(s)$$

$$F(s) = X(s)(Ms^2 + Bs + K)$$

Mechanical Translational System (Cont.)

The transfer function is:

Translational System Transfer Function

$$\frac{X(s)}{F(s)} = \frac{1}{Ms^2 + Bs + K} = \frac{\frac{1}{M}}{s^2 + s\frac{B}{M} + \frac{K}{M}}$$

Analogy to Electrical System: Notice the mathematical similarity to the series RLC circuit ($\frac{1}{s^2LC+sRC+1}$).

- Mass (M) $\leftrightarrow$ Inductance (L)
- Damper (B) $\leftrightarrow$ Resistance (R)
- Spring (K) $\leftrightarrow$ Elastance ($1/C$)

Mechanical Rotational System

Basic elements of a mechanical rotational system:

- **Moment of Inertia (J):** Opposes angular acceleration. Torque $T_J = J\frac{d^2\theta}{dt^2}$
- **Torsional Spring (K):** Opposes angular displacement. Torque $T_K = K\theta$
- **Rotational Damper (B):** Opposes angular velocity. Torque $T_B = B\frac{d\theta}{dt}$

For a rotational system with input torque $T(t)$ and output angular displacement $\theta(t)$:

$$T(t) = J\frac{d^2\theta}{dt^2} + B\frac{d\theta}{dt} + K\theta(t)$$

Taking Laplace Transform:

$$T(s) = Js^2\Theta(s) + Bs\Theta(s) + K\Theta(s)$$

$$T(s) = \Theta(s)(Js^2 + Bs + K)$$

The transfer function is:

Rotational System Transfer Function

$$\frac{\Theta(s)}{T(s)} = \frac{1}{Js^2 + Bs + K}$$

Introduction to Block Diagram Reduction

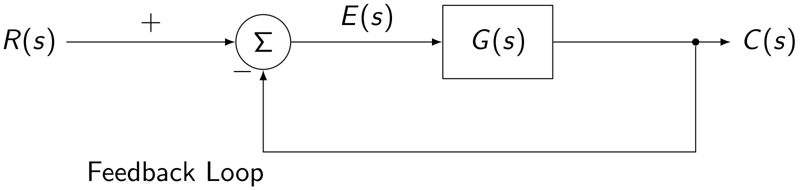
Block Diagram

A block diagram is a pictorial representation of the functions performed by each component and of the flow of signals within a control system.

- In a block diagram, all system variables are linked to each other through functional blocks.
- The functional block (or simply block) is a symbol for the mathematical operation on the input signal to the block that produces the output.
- **Why BDR?** Complex systems consist of many interconnected subsystems. Block Diagram Reduction simplifies these connections into a single block representing the **Closed-Loop Transfer Function**.

Basic Elements of Block Diagrams I

1. **Block:** Represents the transfer function of a component.
2. **Arrows:** Represent the direction of signal flow.
3. **Summing Point:** Compares two or more signals (addition or subtraction).
4. **Take-off Point:** A point from which a signal is taken to be fed to one or more other blocks.



Rules of Block Diagram Reduction

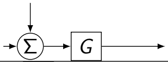
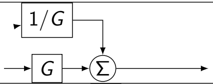
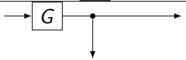
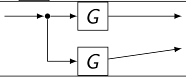
Key Rules

1. **Blocks in Cascade (Series):** $G_{eq} = G_1 \cdot G_2 \ldots G_n$
2. **Blocks in Parallel:** $G_{eq} = G_1 \pm G_2 \pm \cdots \pm G_n$
3. **Eliminating a Feedback Loop:**

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 \pm G(s)H(s)}$$

(Use '+' for negative feedback, '-' for positive feedback).

Rules for Shifting Points

Operation	**Original**	**Equivalent**
Moving summing point ahead of a block		
Moving take-off point behind a block		

Note: The goal is to isolate loops to apply the feedback rule!

Introduction to Signal Flow Graphs (SFG)

Signal Flow Graph

A Signal Flow Graph is a graphical representation of the algebraic equations representing a system. It consists of a network of nodes connected by directed branches.

- Originally developed by S. J. Mason.
- Offers a more straightforward method to find the transfer function compared to BDR, especially for highly interconnected, complex systems.
- **Node:** Represents a system variable (signal).
- **Branch:** Directed line connecting two nodes. It has a gain or transmittance.

SFG Terminology

- **Input Node (Source):** A node with only outgoing branches.
- **Output Node (Sink):** A node with only incoming branches.
- **Path:** A continuous, unidirectional succession of branches along which no node is passed more than once.
- **Forward Path:** A path from an input node to an output node.
- **Forward Path Gain (P_k):** Product of branch gains along the forward path.
- **Loop:** A closed path that originates and terminates on the same node, passing no node more than once.
- **Loop Gain:** Product of branch gains of a loop.
- **Non-touching Loops:** Loops that do not share any common nodes.

Mason's Gain Formula

Mason's Gain Formula is used to calculate the overall transfer function directly from the SFG.

Mason's Gain Formula

$$T = \frac{C(s)}{R(s)} = \frac{1}{\Delta} \sum_k P_k \Delta_k$$

Where:

- T: Overall Transfer Function
- P_k: Gain of the k-th forward path
- $\Delta = 1 - (\sum L_i) + (\sum L_i L_j) - (\sum L_i L_j L_k) + \dots$
- L_i: Gain of individual loop.
- $L_i L_j$: Product of gains of two non-touching loops.
- Δ_k: The value of Δ for the part of the graph not touching the k-th forward path.

Comparing BDR and SFG

Block Diagram Reduction	Signal Flow Graph
Visualizes both mathematical operations and hardware blocks.	Strictly visualizes mathematical algebraic equations.
Requires sequential reduction using multiple rules.	Provides a direct, single-step formula (Mason's Gain).
Can be tedious and error-prone for complex systems.	Highly systematic and easier for complex systems.
Variables are represented by lines, operations by blocks.	Variables are represented by nodes, operations by branches.

Complex Systems I

- In real-world engineering, systems are rarely composed of a single element.
- They consist of multiple subsystems interconnected in various ways (series, parallel, feedback).
- Finding the overall transfer function $T(s) = \frac{C(s)}{R(s)}$ of a complex system can be challenging if done purely by algebraic manipulation of system equations.

Key Methods

To simplify complex interconnected systems and find their overall transfer function, we primarily use two graphical methods:

1. **Block Diagram Reduction (BDR)**
2. **Signal Flow Graph (SFG)**

Block Diagram Fundamentals

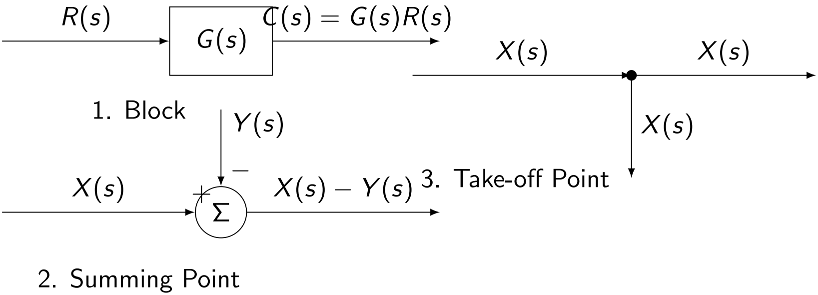
Block Diagram

A block diagram is a pictorial representation of the cause-and-effect relationship between the input and output of a physical system.

Basic Elements:

- **Block:** Represents the mathematical operation (transfer function) on the input signal.
- **Summing Point:** Compares algebraic sums of two or more signals.
- **Take-off Point:** A point from which a signal is taken and fed to other blocks concurrently.

Basic Elements of Block Diagram

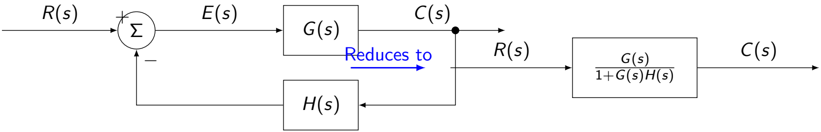


Rules of Block Diagram Reduction

BDR involves systematically simplifying the diagram using established rules until it reduces to a single block.

- **Series Connection:** Blocks connected in cascade (series) are multiplied.
- **Parallel Connection:** Blocks connected in parallel (feed-forward) are algebraically added.
- **Feedback Loop:** A negative feedback loop with forward gain $G(s)$ and feedback gain $H(s)$ reduces to $\frac{G(s)}{1+G(s)H(s)}$. (For positive feedback, the denominator is $1 - G(s)H(s)$).
- **Moving Points:** Rules exist for moving summing points and take-off points ahead of or behind blocks, multiplying or dividing by the block's transfer function as appropriate.

Example: Feedback Loop Reduction



Limitations of Block Diagram Reduction

- BDR is intuitive but can become very tedious and error-prone for highly complex systems with multiple overlapping loops.
- The sequence of reduction steps is not unique, meaning different engineers might take different paths to simplify the same diagram.

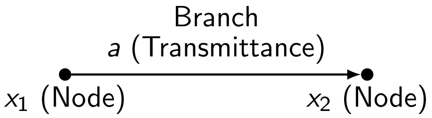
Signal Flow Graph (SFG)

Developed by S.J. Mason, an SFG is a network of nodes connected by directed branches. It provides a more systematic approach to finding the transfer function without step-by-step reduction, using **Mason's Gain Formula**.

Elements of Signal Flow Graph I

Nodes and Branches

- **Node:** Represents a system variable (signal). Nodes are shown as small circles.
- **Branch:** A directed line segment connecting two nodes. It indicates the dependency of one variable on another.
- **Transmittance:** The gain or transfer function of a branch.



Equation Representation: $x_2 = a \cdot x_1$

SFG Terminology

- **Input Node (Source):** A node with only outgoing branches.
- **Output Node (Sink):** A node with only incoming branches.
- **Path:** A continuous, unidirectional succession of branches along which no node is passed more than once.
- **Forward Path:** A path from an input node to an output node.
- **Loop:** A closed path originating and terminating on the same node, passing through no other node more than once.
- **Non-touching Loops:** Two or more loops that do not share any common node.

Mason's Gain Formula

The overall transfer function (gain) $T = \frac{C(s)}{R(s)}$ of a system represented by an SFG is given by Mason's Gain Formula:

Mason's Gain Formula

$$T = \frac{1}{\Delta} \sum_{k=1}^{N} P_k \Delta_k$$

Where:

- N = Total number of forward paths.
- P_k = Gain of the k^{th} forward path.
- Δ = Determinant of the graph.
- Δ_k = Path factor (cofactor of the k^{th} forward path determinant).

The Determinant Δ is calculated as:

$$
\begin{aligned}
\Delta = 1 &- (\text{Sum of all individual loop gains}) \\
&+ \begin{pmatrix}\text{Sum of gain products of all possible} \\ \text{combinations of 2 non-touching loops}\end{pmatrix} \\
&- \begin{pmatrix}\text{Sum of gain products of all possible} \\ \text{combinations of 3 non-touching loops}\end{pmatrix} \\
&+ \dots
\end{aligned}
$$

The Path Factor Δ_k:

- Δ_k is evaluated from Δ by removing the terms containing loop gains of loops that touch the k^{th} forward path.
- In other words, $\Delta_k = \Delta$ for that part of the graph which is *not touching* the k^{th} forward path.
- If all loops touch the k^{th} path, then $\Delta_k = 1$.

Comparison: BDR vs. SFG

Block Diagram Reduction	Signal Flow Graph
Uses blocks, summing points, and take-off points.	Uses nodes and directed branches.
Requires stepwise reduction which can be tedious.	Gives the transfer function directly using a single formula.
Self-loops do not appear naturally without specific reduction steps.	Self-loops can be directly handled in Mason's formula.
Difficult to check for errors halfway through the process.	Systematic; identifying paths and loops reduces algebraic errors.

Summary

- Both Block Diagram Reduction and Signal Flow Graphs are essential tools in control systems engineering.
- **BDR** offers an intuitive visual step-by-step simplification of system components.
- **SFG**, coupled with Mason's Gain Formula, provides a robust, systematic, and often quicker method to determine the transfer function of complicated, highly interconnected systems.
- Engineers generally prefer converting complex block diagrams to signal flow graphs to avoid the pitfalls of manual algebraic reduction.

Introduction to Block Diagrams I

Block Diagram

A block diagram of a system is a pictorial representation of the functions performed by each component and of the flow of signals. Such a diagram depicts the interrelationships that exist among the various components.

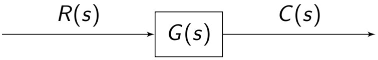
Key Elements of a Block Diagram:

- **Blocks:** Represent the cause-and-effect mathematical relationship between the input and output of a system component. Often characterized by a Transfer Function ($G(s)$).
- **Arrows:** Indicate the direction of signal flow.

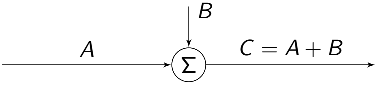
Introduction to Block Diagrams II

- **Summing Points:** Points where two or more signals are algebraically added or subtracted.
- **Take-off Points:** Points from which a signal is tapped off to be used concurrently in multiple branches.

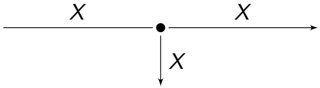
Components of a Block Diagram



Block



Summing Point



Take-off Point

Advantages and Disadvantages I

Advantages of Block Diagrams

- **Simplicity:** Simplifies the analysis and design of complex control systems.
- **Visual Clarity:** Provides a clear, graphical representation of signal flow and interconnections.
- **Modularity:** Individual components can be evaluated independently and modified without affecting the overall structure.
- **Mathematical Convenience:** Easy to derive the overall transfer function of the system.

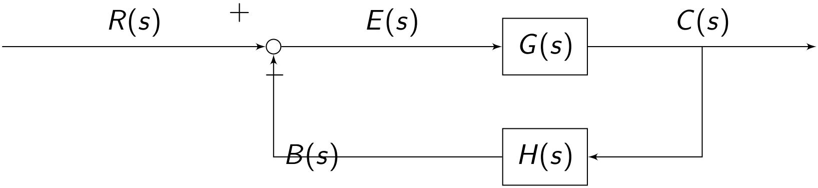
Disadvantages:

Advantages and Disadvantages II

- Does not contain any information about the physical construction of the system.
- A single system can be represented by multiple valid block diagrams.
- For highly complex systems with multiple inputs and outputs, block diagrams can become cluttered (Signal Flow Graphs are often preferred).

Canonical Form of a Feedback Control System I

The canonical (standard) form of a closed-loop feedback control system consists of a forward path and a feedback path.



- $R(s)$: Reference Input
- $C(s)$: Controlled Output
- $E(s)$: Error Signal ($E(s) = R(s) - B(s)$)
- $B(s)$: Feedback Signal ($B(s) = C(s)H(s)$)
- $G(s)$: Forward Path Transfer Function
- $H(s)$: Feedback Path Transfer Function

Transfer Function of a Simple Closed-Loop System I

Let's derive the overall transfer function $T(s) = \frac{C(s)}{R(s)}$ for the canonical form.

From the block diagram, we can write the following equations:

$$C(s) = E(s) \cdot G(s) \tag{1}$$
$$E(s) = R(s) - B(s) \tag{2}$$
$$B(s) = C(s) \cdot H(s) \tag{3}$$

Substitute (3) into (2):

$$E(s) = R(s) - C(s)H(s) \tag{4}$$

Transfer Function of a Simple Closed-Loop System II

Substitute (4) into (1):

$$\begin{aligned}
C(s) &= [R(s) - C(s)H(s)] \cdot G(s) \\
C(s) &= R(s)G(s) - C(s)G(s)H(s) \\
C(s) + C(s)G(s)H(s) &= R(s)G(s) \\
C(s)[1 + G(s)H(s)] &= R(s)G(s)
\end{aligned}$$

Transfer Function of a Simple Closed-Loop System

Closed-Loop Transfer Function

Dividing by $R(s)$ and $[1 + G(s)H(s)]$, we get:

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)}$$

Note:

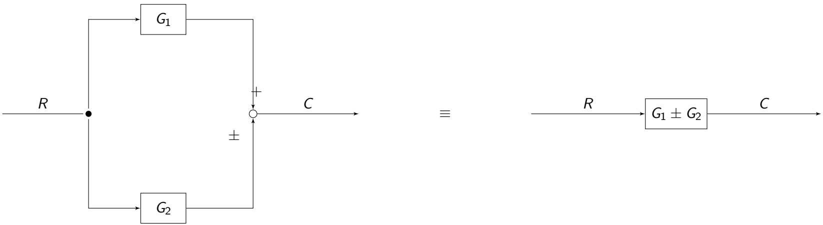
- For **negative feedback**, the denominator is $1 + G(s)H(s)$.
- For **positive feedback** ($E(s) = R(s) + B(s)$), the denominator is $1 - G(s)H(s)$.
- The expression $G(s)H(s)$ is known as the **Open-Loop Transfer Function**.
- The expression $1 \pm G(s)H(s) = 0$ is called the **Characteristic Equation** of the system.

Block Diagram Reduction Rules

To find the overall transfer function of complex systems, we simplify the block diagrams using standard rules.

Rule 1: Blocks in Series (Cascade) When blocks are connected in series, their transfer functions are multiplied.

Rule 2: Blocks in Parallel When blocks are connected in parallel, their transfer functions are added algebraically.



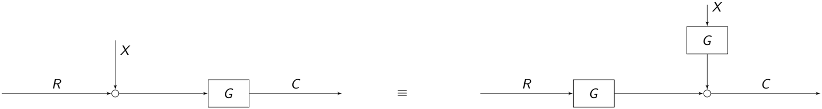
Rule 3: Moving a Summing Point Ahead of a Block To move a summing point ahead of a block, insert a block with transfer function $\frac{1}{G}$ in the path coming to the summing point.

Original

Equivalent

Block Diagram Reduction Rules

Rule 4: Moving a Summing Point Beyond a Block To move a summing point behind a block, insert a block with transfer function G in the path coming to the summing point.



Rule 5: Moving a Take-off Point Ahead of a Block To move a take-off point ahead of a block, add a block with transfer function G in the take-off path.

Block Diagram Reduction Rules

Rule 6: Moving a Take-off Point Beyond a Block To move a take-off point behind a block, add a block with transfer function $\frac{1}{G}$ in the take-off path.

Summary of BDR Strategy

When reducing a complex block diagram to a single transfer function block:

1. Combine all cascade (series) blocks.
2. Combine all parallel blocks.
3. Eliminate all minor feedback loops.
4. Shift take-off points to the right and summing points to the left if needed to untangle loops.
5. Repeat steps 1 to 4 until a single canonical form is obtained.
6. Compute the final overall transfer function.

Tip for Problem Solving

Always try to avoid moving take-off points and summing points past each other unless absolutely necessary, as it can introduce complicated cross-coupling paths. Focus on internal loops first!

Introduction to Block Diagrams I

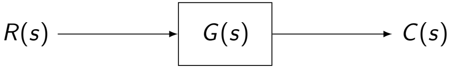
Block Diagram

A **block diagram** of a system is a pictorial representation of the functions performed by each component and of the flow of signals. Such a diagram depicts the interrelationships that exist among the various components.

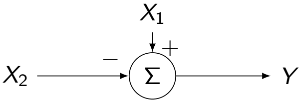
- Differing from a purely abstract mathematical representation, a block diagram emphasizes signal flow and functional relationships.
- Used extensively in control theory to simplify complex systems into manageable visual and mathematical forms.

Basic Components of a Block Diagram I

- **Blocks:** Represent the transfer function of a component. The output is the product of the input signal and the transfer function.
- **Summing Point:** A circle with a cross (or just a circle) where two or more signals are algebraically added or subtracted.
- **Take-off Point (Branch Point):** A point from which a signal goes concurrently to more than one block. The signal is identical in all branches.



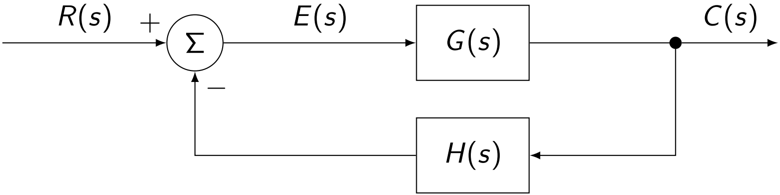
Block



Summing Point

Simple Closed-Loop System (Canonical Form) I

A typical feedback control system is represented in its **canonical form** as follows:



- $R(s)$: Reference input (Command)
- $C(s)$: Controlled output
- $E(s)$: Error signal ($E(s) = R(s) \mp C(s)H(s)$)
- $G(s)$: Forward path transfer function
- $H(s)$: Feedback path transfer function

Transfer Function of a Closed-Loop System I

Let's derive the overall transfer function $\frac{C(s)}{R(s)}$ for the negative feedback system:

$$C(s) = E(s) \cdot G(s)$$
$$E(s) = R(s) - B(s)$$
$$B(s) = C(s) \cdot H(s)$$

Substituting $B(s)$ into the error equation:

$$E(s) = R(s) - C(s)H(s)$$

Substitute $E(s)$ into the first equation:

$$\begin{aligned} C(s) &= [R(s) - C(s)H(s)] \cdot G(s) \\ C(s) &= R(s)G(s) - C(s)H(s)G(s) \\ C(s)[1 + G(s)H(s)] &= R(s)G(s) \end{aligned}$$

Closed-Loop Transfer Function

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 \pm G(s)H(s)}$$

(Use $+$ for negative feedback, and $-$ for positive feedback)

Advantages and Disadvantages of Block Diagrams I

Advantages:

- **Visual Clarity:** Provides a clear, graphical representation of signal flow and interactions between system components.
- **Modularity:** Complex systems can be broken down into individual functional blocks.
- **Mathematical Simplification:** Overall transfer function can be easily evaluated using Block Diagram Reduction (BDR) rules.

Disadvantages:

- **Lack of Physical Insight:** Does not provide any information about the physical construction or internal state variables of the system.
- **Non-Unique:** A single control system can be represented by multiple valid, yet different, block diagrams.

Advantages and Disadvantages of Block Diagrams II

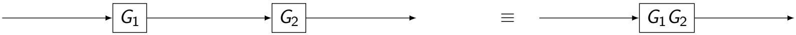
- **Source Loading:** Assumes no loading effects between blocks (i.e., output of one block is not affected by connecting another block to it).

Block Diagram Reduction Rules (1/3) I

To simplify complex diagrams into a single block, we use BDR rules.

Rule 1: Blocks in Series (Cascade)

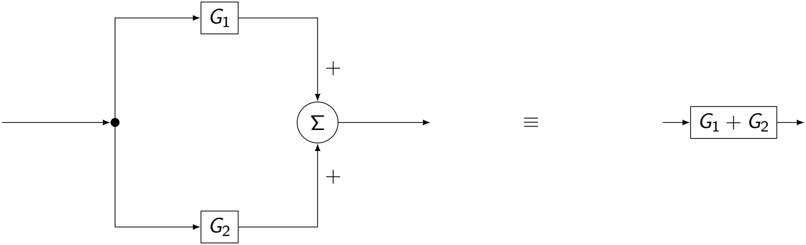
- Transfer functions are multiplied.



Rule 2: Blocks in Parallel

- Transfer functions are algebraically added.

Block Diagram Reduction Rules (1/3) II

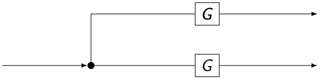


Rule 3: Shifting a Summing Point AFTER a Block

Rule 4: Shifting a Summing Point BEFORE a Block

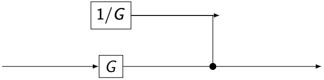
Rule 5: Shifting a Take-off Point BEFORE a Block

$\equiv$



Rule 6: Shifting a Take-off Point AFTER a Block

$\equiv$



General Procedure for Block Diagram Reduction I

1. Combine all blocks in cascade (series).
2. Combine all blocks in parallel.
3. Reduce any minor feedback loops into single blocks.
4. Shift summing points to the left and take-off points to the right of the major loop blocks.
5. Repeat Steps 1 to 4 until a single block (canonical form) is obtained.
6. Evaluate the overall closed-loop transfer function.

Tip for Reduction

Try to avoid moving take-off points past summing points, as it introduces complicated cross-coupling and makes the diagram harder to reduce. Focus on moving take-off points past blocks.

Signal Flow Graph (SFG)

A Signal Flow Graph is a graphical representation of a set of simultaneous linear algebraic equations. It consists of a network of directed branches connected at nodes, representing the flow of signals from one point of a system to another.

- Originally developed by S. J. Mason in the 1950s.
- It serves as an alternative to block diagrams for representing control systems.
- While block diagram reduction can be tedious for complex systems, SFG allows the calculation of the overall transfer function directly using **Mason's Gain Formula**.

Basic Definitions in SFG I

- **Node:** A point representing a system variable or signal.
- **Branch:** A directed line segment joining two nodes. The arrow indicates the direction of signal flow.
- **Transmittance (Branch Gain):** The gain or transmission function between two nodes, written along the branch.
- **Input Node (Source Node):** A node that has *only outgoing* branches. It represents an independent input variable.
- **Output Node (Sink Node):** A node that has *only incoming* branches. It represents an output variable.
- **Mixed Node:** A node that has *both* incoming and outgoing branches.

$$x_1 \xrightarrow{\quad a \quad} x_2 \xrightarrow{\quad b \quad} x_3$$

Input Node — Mixed Node — Output Node

Paths and Loops I

- **Path:** A continuous, unidirectional succession of branches along which no node is traversed more than once.
- **Forward Path:** A path from an input node to an output node.
- **Forward Path Gain (P_k):** The product of branch gains along a forward path.
- **Loop:** A closed path that originates and terminates on the same node, and along which no other node is traversed more than once.
- **Loop Gain (L):** The product of branch transmittances of a loop.
- **Non-touching Loops:** Two or more loops are said to be non-touching if they do not share any common node.

Advantages and Disadvantages of SFG I

Advantages

- Gives the overall transfer function directly in one step using Mason's Gain Formula.
- Eliminates the need for complex and tedious block diagram reduction rules.
- Provides better visualization of the signal flow and feedback loops within the system.
- Easier to construct from system equations.

Advantages and Disadvantages of SFG II

Disadvantages

- Applicable **only** to Linear Time-Invariant (LTI) systems (since it relies on linear algebraic equations).
- Can become heavily cluttered and difficult to read for highly complex systems with many interconnections.

Rules of Signal Flow Graphs I

1. **Node Value Rule:** The value of the variable represented by a node is equal to the algebraic sum of all signals entering the node.
2. **Transmission Rule:** A signal travels along a branch in the direction of the arrow.
3. **Multiplication Rule:** The signal transmitted along a branch is multiplied by the branch gain (transmittance).
4. **Distribution Rule:** The value of the variable represented by any node is transmitted on all branches leaving that node.

Rules of Signal Flow Graphs II

Example System Equations

If $x_2 = ax_1 + bx_3$, then node x_2 has incoming branches from x_1 (gain a) and x_3 (gain b).

Mason's Gain Formula

The overall transfer function (gain) $T = \frac{C(s)}{R(s)}$ of a system represented by a signal flow graph is given by:

$$T = \frac{1}{\Delta} \sum_{k=1}^{N} P_k \Delta_k$$

Where:

- N = Total number of forward paths.
- P_k = Gain of the k^{th} forward path.
- Δ = Determinant of the graph.
- Δ_k = Path factor corresponding to the k^{th} forward path.

Determinant of the Graph (Δ) I

The determinant of the graph, Δ, is calculated as:

$$\Delta = 1 - \sum L_i + \sum L_i L_j - \sum L_i L_j L_k + \cdots$$

Where:

- $\sum L_i =$ Sum of the gains of all individual loops.
- $\sum L_i L_j =$ Sum of the products of gains of all possible combinations of **two non-touching** loops.
- $\sum L_i L_j L_k =$ Sum of the products of gains of all possible combinations of **three non-touching** loops.
- And so on...

Path Factor (Δ_k)

The path factor Δ_k is the value of the determinant Δ for the part of the signal flow graph that **does not touch** the k^{th} forward path.

- To find Δ_k, eliminate all nodes that are part of the k^{th} forward path from the original graph.
- Recalculate the determinant Δ for the remaining sub-graph.
- If all loops in the graph touch the k^{th} forward path, then the remaining sub-graph has no loops, and $\Delta_k = 1$.

Steps for Solving SFG using Mason's Gain Formula I

To find the transfer function of a system using Mason's Gain Formula, systematically follow these steps:

1. **Locate Forward Paths:** Identify all possible forward paths from input to output and calculate their gains $(P_1, P_2, \ldots, P_N)$.
2. **Locate Individual Loops:** Identify all individual closed loops and calculate their loop gains $(L_1, L_2, \ldots)$.
3. **Identify Non-touching Loops:** Find all combinations of two non-touching loops, three non-touching loops, etc., and calculate their gain products.
4. **Calculate Determinant (Δ):** Substitute the loop gains into the formula for Δ.
5. **Calculate Path Factors (Δ_k):** Evaluate Δ_k for each forward path.

Steps for Solving SFG using Mason's Gain Formula II

6. **Apply Formula:** Substitute P_k, Δ_k, and Δ into Mason's Gain Formula to obtain the transfer function T.

Summary of Block Diagram vs SFG I

Feature	**Block Diagram**	**Signal Flow Graph**
Basic Element	Blocks, summing points, takeoff points	Nodes and directed branches
Complexity	Difficult to reduce for complex systems	Easy to solve using Mason's formula
Mathematical Basis	Algebraic equations with graphical blocks	Direct graphical representation of equations
Self-loops	Rare or explicitly drawn as feedback blocks	Commonly used and directly supported

Introduction to Signal Flow Graphs

- A **Signal Flow Graph (SFG)** is a graphical representation of the algebraic equations that describe a control system.
- Introduced by S. J. Mason, it offers an alternative to block diagram reduction techniques.
- Instead of reducing a block diagram step-by-step, SFGs allow for the direct computation of the overall transfer function using a single mathematical formula.

Key Characteristics

In an SFG, the variables (signals) of the system are represented by **nodes**, and the mathematical relationships between these variables are represented by directed **branches**.

SFG Basic Terminology

Nodes and Branches

- **Node:** Represents a system variable or signal.
- **Branch:** A directed line connecting two nodes, indicating signal flow.
- **Transmittance (Gain):** The multiplying factor associated with a branch.

Types of Nodes:

- **Input Node (Source):** A node with only outgoing branches (independent variable).
- **Output Node (Sink):** A node with only incoming branches (dependent variable).
- **Mixed Node:** A node with both incoming and outgoing branches.

SFG Paths and Loops

- **Path:** A continuous, unidirectional succession of branches traversed from one node to another.
- **Forward Path:** A path from an input node to an output node wherein no node is crossed more than once.
- **Forward Path Gain (P_k):** The product of all branch gains along a forward path.
- **Feedback Loop:** A path that starts and ends at the same node, without crossing any node more than once.
- **Loop Gain:** The product of the branch transmittances of a loop.
- **Non-touching Loops:** Two or more loops that do not share any common node.

Properties of Signal Flow Graphs

1. **Signal Direction:** Signals travel along branches only in the direction of the arrows.
2. **Multiplication by Transmittance:** A signal is multiplied by the transmittance (gain) of the branch as it travels through it.
3. **Addition at Nodes:** The value of the variable at a node is the algebraic sum of all signals entering that node.
4. **Transmission from Nodes:** The value of the variable at a node is transmitted unaltered on all branches leaving that node.

Algebraic Representation

For a branch from node X_1 to X_2 with gain G, the relationship is $X_2 = G \cdot X_1$. If multiple branches enter X_2 with signals A and B, then $X_2 = A + B$.

Advantages and Disadvantages of SFGs

Advantages

- **Efficiency:** Calculates the overall transfer function in a single step using Mason's gain formula.
- **Simplicity:** Avoids the cumbersome step-by-step reduction required in block diagrams.
- **Visual Clarity:** Provides a clear picture of the cause-and-effect relationship between variables.

Disadvantages

- **Abstract Nature:** Can be less physically intuitive than block diagrams, which map more directly to hardware components.
- **Complexity:** For highly interconnected systems with numerous nodes, identifying all loops and non-touching combinations can become error-prone.

Mason's Gain Formula

Mason's Gain Formula provides a direct method to evaluate the transfer function of a system from its signal flow graph.

Mason's Gain Formula

$$T = \frac{C(s)}{R(s)} = \frac{1}{\Delta} \sum_{k=1}^{N} P_k \Delta_k$$

Where:

- T: Overall system transmittance (Transfer Function).
- N: Total number of forward paths.
- P_k: Path gain of the k-th forward path.
- Δ: Graph determinant (characteristic function).
- Δ_k: Path factor of the k-th forward path.

Components of Mason's Gain Formula

Graph Determinant (Δ):

$$\Delta = 1 - \sum L_1 + \sum L_2 - \sum L_3 + \dots$$

- $\sum L_1$: Sum of all individual loop gains.
- $\sum L_2$: Sum of gain products of all possible pairs of non-touching loops.
- $\sum L_3$: Sum of gain products of all possible triplets of non-touching loops.

Path Factor (Δ_k):

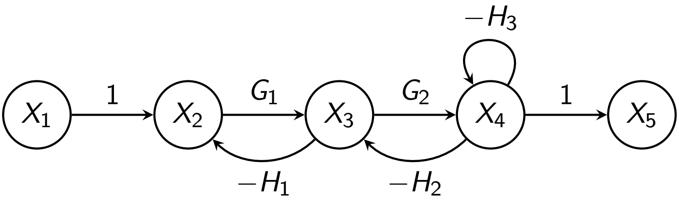
- Δ_k is the value of Δ for that part of the graph that is **not touching** the k-th forward path.
- It is obtained by removing all loops in Δ that share one or more nodes with the k-th forward path.

Steps to Apply Mason's Gain Formula

1. **Identify Forward Paths:** Find all forward paths from input to output and calculate their gains $(P_1, P_2, \ldots, P_N)$.
2. **Identify Individual Loops:** Find all individual feedback loops and calculate their gains $(L_1, L_2, \ldots)$.
3. **Identify Non-Touching Loops:** Determine combinations of 2 non-touching loops, 3 non-touching loops, etc., and find their gain products.
4. **Calculate Δ:** Use the determinant formula with the values obtained in Steps 2 and 3.
5. **Calculate Δ_k:** Evaluate the path factor for each forward path identified in Step 1.
6. **Compute Final Transfer Function:** Substitute P_k, Δ_k, and Δ into Mason's Gain Formula to find T.

Example Application

Consider the following Signal Flow Graph:



Step 1: Forward Paths

- $P_1 = 1 \cdot G_1 \cdot G_2 \cdot 1 = G_1 G_2$

Example Application (Continued)

Step 2: Individual Loops

- $L_1 = G_1(-H_1) = -G_1H_1$ (Loop between X_2 and X_3)
- $L_2 = G_2(-H_2) = -G_2H_2$ (Loop between X_3 and X_4)
- $L_3 = -H_3$ (Self-loop at X_4)

Step 3: Non-Touching Loops

- L_1 and L_3 are non-touching because they do not share any nodes.
- Pair product: $L_{13} = L_1 \cdot L_3 = (-G_1H_1)(-H_3) = G_1H_1H_3$

Example Application (Final Calculation)

Step 4: Calculate Determinant (Δ)

$$\Delta = 1 - (L_1 + L_2 + L_3) + (L_{13})$$
$$\Delta = 1 - (-G_1H_1 - G_2H_2 - H_3) + (G_1H_1H_3)$$
$$\Delta = 1 + G_1H_1 + G_2H_2 + H_3 + G_1H_1H_3$$

Step 5: Calculate Path Factors (Δ_k)

- For P_1, it passes through all nodes (X_1 to X_5). Therefore, it touches all loops.
- Thus, $\Delta_1 = 1 - 0 = 1$.

Step 6: Transfer Function (T)

$$T = \frac{P_1\Delta_1}{\Delta} = \frac{G_1G_2}{1 + G_1H_1 + G_2H_2 + H_3 + G_1H_1H_3}$$

Signal Flow Graph (SFG)

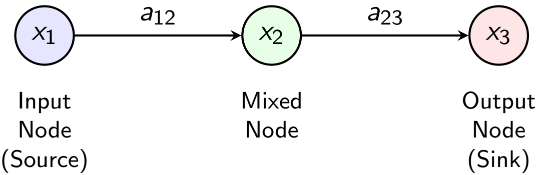
A Signal Flow Graph (SFG) is a graphical representation of a set of simultaneous linear algebraic equations representing a system. It consists of nodes representing system variables and directed branches representing multipliers (transmittances) relating the variables.

- Introduced by S. J. Mason in 1953.
- It is an alternative to block diagram reduction for finding the transfer function of a complex control system.
- SFGs are particularly useful for systems with multiple interrelated feedback loops.
- In an SFG, the system is viewed as a network where signals flow from one point to another along directed branches.

Basic Definitions in SFG I

- **Node:** Represents a system variable (e.g., a signal or an equation variable).
- **Branch:** A directed line segment joining two nodes. The direction specifies the signal flow, and the branch has a gain (transmittance).
- **Input Node (Source):** A node with only outgoing branches. Represents an independent variable.
- **Output Node (Sink):** A node with only incoming branches. Represents a dependent variable.
- **Mixed Node:** A node that has both incoming and outgoing branches.

Basic Definitions in SFG II



Path and Loop Definitions

- **Path:** A continuous, unidirectional succession of branches along which no node is passed more than once.
- **Forward Path:** A path from an input node to an output node.
- **Forward Path Gain:** The product of the branch gains along a forward path.
- **Loop:** A closed path that originates and terminates on the same node, and along which no other node is passed more than once.
- **Loop Gain:** The product of the branch gains of a loop.
- **Non-touching Loops:** Two or more loops that do not share any common node.

Advantages and Disadvantages of SFG I

Advantages

- Graphical representation provides clear insight into system flow and interconnections.
- Direct application of Mason's Gain Formula avoids step-by-step block diagram reduction.
- Easier to construct directly from a set of linear differential or algebraic equations.

Advantages and Disadvantages of SFG II

Disadvantages

- Applicable primarily to linear time-invariant (LTI) systems.
- Requires the system to be accurately modeled as a set of linear algebraic equations.
- Can become visually cluttered for extremely large and complex systems if not drawn carefully.

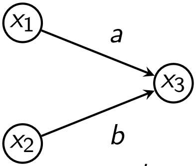
Rules of Signal Flow Graphs I

1. **Addition Rule:** The value of a node is the algebraic sum of all incoming signals to that node.

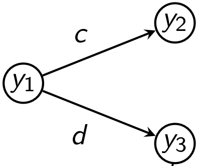
$$x_k = \sum_j a_{jk} x_j$$

2. **Transmission Rule:** The signal travels along a branch only in the direction of the arrow. The signal gets multiplied by the branch gain.
3. **Distribution Rule:** A node transmits its signal to all outgoing branches. The value of the signal remains unchanged as it leaves the node.

Rules of Signal Flow Graphs II



$x_3 = ax_1 + bx_2$



$y_2 = cy_1, \; y_3 = dy_1$

Mason's Gain Formula

Mason's gain formula provides a generic expression to compute the overall transfer function (gain) of a system represented by a Signal Flow Graph.

$$T = \frac{C(s)}{R(s)} = \frac{1}{\Delta} \sum_k P_k \Delta_k$$

Where:

- T = Overall transfer function (gain) from input node to output node.
- P_k = Gain of the k-th forward path.
- Δ = Graph determinant.
- Δ_k = Forward path cofactor for the k-th forward path.

Components of Mason's Gain Formula

- **Graph Determinant (Δ):**

$$\Delta = 1 - \sum L_1 + \sum L_2 - \sum L_3 + \dots$$

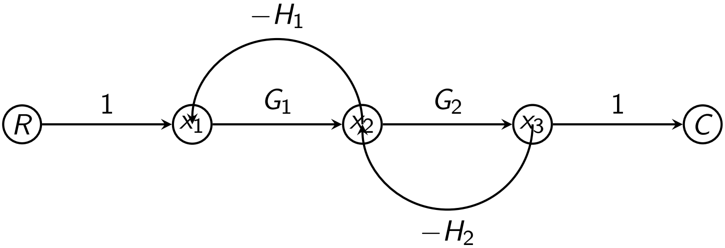
  - $\sum L_1$ = Sum of gains of all individual loops.
  - $\sum L_2$ = Sum of gain products of all possible combinations of two non-touching loops.
  - $\sum L_3$ = Sum of gain products of all possible combinations of three non-touching loops.
  - ... and so on.

- **Path Cofactor (Δ_k):** The value of Δ for the part of the graph that is **not touching** the k-th forward path. In other words, compute Δ_k by eliminating all loops in Δ that share at least one node with the k-th forward path. If all loops touch the k-th path, then $\Delta_k = 1$.

Steps for Solving SFG Using Mason's Formula

1. **Identify Forward Paths:** Find all forward paths from the input node to the output node and calculate their gains $(P_1, P_2, \ldots)$.
2. **Identify Individual Loops:** Find all individual closed loops in the graph and calculate their gains. Let the sum be $\sum L_1$.
3. **Identify Non-touching Loops:**
   - Find all pairs of loops that do not share any node. Calculate the product of their gains. Sum them to find $\sum L_2$.
   - Find all triplets of mutually non-touching loops. Sum their products to find $\sum L_3$, etc.
4. **Calculate Graph Determinant (Δ):** Compute $\Delta = 1 - \sum L_1 + \sum L_2 - \sum L_3 + \ldots$
5. **Calculate Path Cofactors (Δ_k):** For each forward path P_k, find Δ_k by removing the loops from Δ that touch path k.
6. **Apply Formula:** Substitute all values into $T = \frac{1}{\Delta} \sum P_k \Delta_k$.

Example: SFG Application I



- **Forward Path:** $P_1 = 1 \cdot G_1 \cdot G_2 \cdot 1 = G_1 G_2$
- **Loops:** $L_1 = -G_1 H_1$ (nodes x_1, x_2), $L_2 = -G_2 H_2$ (nodes x_2, x_3)
- **Non-touching loops:** None (they share node x_2). So, $\sum L_2 = 0$.
- **Determinant:** $\Delta = 1 - (L_1 + L_2) = 1 + G_1 H_1 + G_2 H_2$

Example: SFG Application II

- **Cofactor:** $\Delta_1 = 1$ (all loops touch P_1)
- **Transfer Function:** $T = \frac{P_1 \Delta_1}{\Delta} = \frac{G_1 G_2}{1 + G_1 H_1 + G_2 H_2}$

Overview

- Both **Block Diagram Reduction (BDR)** and **Signal Flow Graphs (SFG)** are graphical methods used to represent and analyze control systems.
- The ultimate goal of both methods is to find the **transfer function** of a complex system.
- While BDR uses blocks, summing points, and take-off points, SFG uses nodes and directed branches.

Key Takeaway

BDR is intuitive for understanding the physical layout and components of a system, whereas SFG provides a more streamlined, algebraic approach suited for complex systems.

Block Diagram Reduction (BDR): A Recap

- **Elements:** Blocks (transfer functions), Summing points (algebraic addition), Take-off points (signal splitting).
- **Process:** Relies on applying a set of reduction rules step-by-step (e.g., combining blocks in series, parallel, or eliminating feedback loops).
- **Shifting Rules:** Often requires moving summing points or take-off points behind or ahead of blocks to simplify the diagram.

Pros & Cons:

- **(+)** Clear physical representation of system components.
- **(-)** Reduction rules can be tedious and prone to errors for highly interconnected systems.

Signal Flow Graph (SFG): A Recap

- **Elements:** Nodes (variables/signals) and Directed Branches (transmittances/gains).
- **Process:** Translates algebraic equations directly into a graphical form.
- **Mason's Gain Formula:** Used to calculate the overall transfer function directly from the graph without step-by-step reduction.

$$T = \frac{1}{\Delta} \sum_k P_k \Delta_k$$

Pros & Cons:

- **(+)** Direct and systematic calculation using a single formula. No need for iterative reduction.
- **(-)** Abstract representation; loses the direct visual link to physical hardware components.

Comparison: BDR vs. SFG

Feature	Block Diagram	Signal Flow Graph
Basic Elements	Blocks, Summing points, Take-off points	Nodes and directed branches
Variables	Represented by signals on lines	Represented by nodes
Transfer Function	Written inside the blocks	Represented by branch transmittances
Feedback	Shown via summing points and feedback loops	Shown by closed loops of branches
Analysis Method	Step-by-step reduction using rules	Mason's Gain Formula (single step)

Comparison: BDR vs. SFG (Contd.)

Feature	**Block Diagram**	**Signal Flow Graph**
Complexity	Becomes very complicated for large systems	Easier to handle for complex systems
Self-Loops	Not explicitly used (handled as feedback)	Commonly used and easily evaluated
Physical Meaning	Preserves physical meaning of components	Mathematical/algebraic abstraction
Error Prone	High chance of error during block shifting	Lower chance of error (systematic)

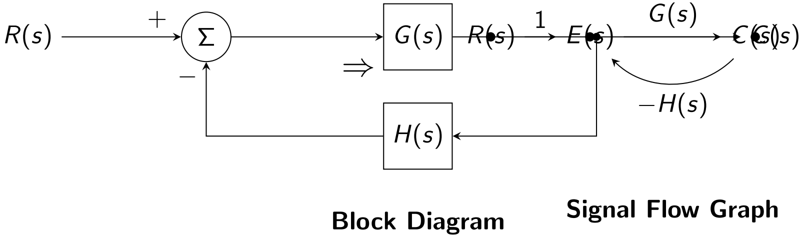
Conversion Rules

A Block Diagram can be easily converted into a Signal Flow Graph using these mappings:

1. All **variables**, **summing points**, and **take-off points** become **Nodes** in the SFG.
2. **Blocks** become **Branches**, with the block's transfer function as the branch transmittance (gain).
3. The direction of signal flow is preserved as the direction of the branch.

Note: Converting a complex Block Diagram to an SFG and then applying Mason's Gain Formula is often the most efficient way to find a transfer function.

Visualizing the Conversion



Summary

- **Use Block Diagrams** when you want to emphasize the physical components, subsystem interactions, and functional layout of a control system.
- **Use Signal Flow Graphs** when you need to solve for the transfer function of a highly complex, multi-loop system efficiently without tedious algebraic manipulation.
- In practice, engineers often sketch the block diagram for understanding and convert it to an SFG for mathematical analysis.

System Representation Techniques

- In Control Systems Engineering, analyzing the behavior of complex systems requires simplifying them to find the overall transfer function.
- Two primary graphical methods are widely used for this purpose:
  - **Block Diagram Reduction (BDR)**
  - **Signal Flow Graph (SFG)**
- While both methods aim to evaluate the closed-loop transfer function of the system, their approaches, mathematical rigor, and visual representations differ significantly.

Block Diagram vs. Signal Flow Graph

Block Diagram (BDR)

A block diagram is a pictorial representation of the cause-and-effect relationship between the input and output of a physical system, consisting of blocks (transfer functions), summing points, and take-off points.

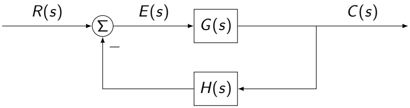
Signal Flow Graph (SFG)

A signal flow graph is a graphical representation of a set of linear algebraic equations, consisting of nodes (representing variables) and directed branches (representing transmittances or gains).

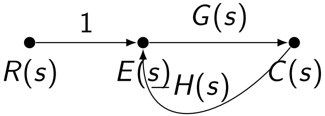
Comparison of Elements

Feature	Block Diagram	Signal Flow Graph
Variables	Represented by signals (lines).	Represented by nodes.
System gains	Represented by blocks containing transfer functions.	Represented by branches connecting nodes.
Signal flow	Direction indicated by arrows on lines.	Direction indicated by arrows on branches.
Addition/Subtraction	Requires explicit summing points (Σ).	Implicitly handled at nodes (sum of incoming signals).
Signal Distribution	Requires take-off points.	Handled naturally as multiple branches leaving a node.

Block Diagram Representation:



Signal Flow Graph Equivalent:



Comparison of Reduction Methods I

Block Diagram Reduction

Requires applying a set of rules (shifting take-off points, moving summing points, combining blocks) iteratively. The process can be tedious, time-consuming, and prone to algebraic errors, especially for multi-loop, multi-variable systems. Furthermore, there is no unique sequence of steps to follow.

Comparison of Reduction Methods II

Signal Flow Graph (Mason's Gain Formula)

Provides a direct mathematical approach. By identifying forward paths, individual loops, and non-touching loops from the graph, the overall transfer function is calculated in a **single step** using Mason's Gain Formula:

$$T = \frac{1}{\Delta} \sum_k P_k \Delta_k$$

Advantages of SFG over BDR

- **Simplicity of Representation:** SFG uses a minimalist representation (only nodes and branches), making it visually cleaner than BDR which requires bulky blocks, summing points, and take-off points.
- **Direct Solution:** The transfer function can be obtained directly using Mason's Gain Formula without requiring any structural modifications to the graph itself.
- **Algorithmic Nature:** Finding loops and forward paths is algorithmic and straightforward, reducing the chance of human error compared to BDR rules which must be applied in a heuristical sequence.
- **Complex Systems:** For highly complex systems with multiple interconnecting loops, BDR becomes extremely difficult to trace and reduce. SFG remains relatively manageable and methodical.

Summary: When to use which?

- **Use Block Diagrams when:**
  - You want to illustrate the physical components of a system and their practical interactions clearly.
  - The system is relatively simple (e.g., standard feedback loop).
  - You are presenting the system architecture to non-specialists, as functional blocks are intuitive.
- **Use Signal Flow Graphs when:**
  - The primary goal is to compute the overall transfer function efficiently.
  - The system contains multiple complex interconnected loops and cross-couplings.
  - You are working with purely mathematical or algebraic models.

Introduction to System Analogy I

System Analogy

Two systems are said to be analogous to each other if the differential equations governing their dynamic behavior are identical in form, even if the systems themselves belong to different physical domains (e.g., mechanical and electrical).

- In engineering, mathematical models (transfer functions or differential equations) for systems in different fields (mechanical, electrical, thermal, fluid) often look remarkably similar.
- Variables in one system can be directly mapped to corresponding variables in another system.

Introduction to System Analogy II

- This powerful concept allows us to analyze complex mechanical or non-electrical systems using well-established electrical circuit analysis techniques.

Advantages of System Analogy I

- **Unified Analysis:** Allows engineers to use the same mathematical tools and software for electrical, mechanical, hydraulic, and thermal systems.
- **Ease of Simulation:** Mechanical systems can be difficult and expensive to prototype. By converting them into their electrical analogs, we can easily simulate them using electrical circuit simulators.
- **Cross-Disciplinary Understanding:** Bridges the gap between different engineering disciplines, fostering better communication and system-level design in mechatronics and control systems.
- **Simplified Modeling:** Complex mechanical systems can be represented as electrical networks, making it easier to determine the overall transfer function using techniques like block diagram reduction.

Mechanical vs. Electrical Systems I

Let's recall the basic building blocks of mechanical (translational) and electrical systems.

Translational Mechanical System:

- **Mass (M):** Opposes change in velocity.
  $f_M(t) = M\frac{d^2x}{dt^2} = M\frac{dv}{dt}$
- **Damper/Friction (B):** Opposes velocity.
  $f_B(t) = B\frac{dx}{dt} = Bv$
- **Spring (K):** Opposes displacement. $f_K(t) = Kx = K\int v\,dt$

Electrical System:

- **Inductor (L):** Opposes change in current. $v_L(t) = L\frac{di}{dt}$
- **Resistor (R):** Opposes current. $v_R(t) = Ri$
- **Capacitor (C):** Opposes change in voltage.
  $v_C(t) = \frac{1}{C}\int i\,dt$

Concept

In the Force-Voltage analogy, we compare the differential equations of a mechanical system (derived using Newton's Second Law) with an electrical series circuit (derived using Kirchhoff's Voltage Law, KVL).

Consider a basic mass-spring-damper system subjected to a force $f(t)$:

$$f(t) = M\frac{d^2x}{dt^2} + B\frac{dx}{dt} + Kx$$

In terms of velocity $v(t) = \frac{dx}{dt}$:

$$f(t) = M\frac{dv}{dt} + Bv + K\int v\,dt$$

Force-Voltage (F-V) Analogy: Electrical Counterpart I

Consider a series RLC circuit excited by a voltage source $e(t)$:

By KVL:

$$e(t) = v_L(t) + v_R(t) + v_C(t)$$

$$e(t) = L\frac{di}{dt} + Ri + \frac{1}{C}\int i\,dt$$

In terms of charge $q(t) = \int i\,dt$:

$$e(t) = L\frac{d^2q}{dt^2} + R\frac{dq}{dt} + \frac{1}{C}q$$

Comparing the equations: The mechanical equation and the electrical equation are mathematically identical!

Force-Voltage (F-V) Analogy Mapping I

Mechanical System	**Symbol**	**Electrical System**	**Symbol**
Force	f	Voltage	e
Mass	M	Inductance	L
Viscous Friction (Damper)	B	Resistance	R
Spring Stiffness	K	Reciprocal of Capacitance	$1/C$
Displacement	x	Charge	q
Velocity	$v = \dot{x}$	Current	$i = \dot{q}$

Force-Current (F-I) Analogy I

Concept

In the Force-Current analogy, we compare the mechanical system equations with an electrical parallel circuit (derived using Kirchhoff's Current Law, KCL).

Mechanical equation (in terms of velocity):

$$f(t) = M\frac{dv}{dt} + Bv + K\int v\,dt$$

Consider a parallel RLC circuit excited by a current source $i(t)$. By KCL:

$$i(t) = i_C(t) + i_R(t) + i_L(t)$$

Force-Current (F-I) Analogy II

$$i(t) = C\frac{dv}{dt} + \frac{1}{R}v + \frac{1}{L}\int v\,dt$$

where $v(t)$ is the node voltage.

Force-Current (F-I) Analogy: Flux Linkage I

Alternatively, in terms of magnetic flux linkage $\psi(t) = \int v\,dt$:

$$i(t) = C\frac{d^2\psi}{dt^2} + \frac{1}{R}\frac{d\psi}{dt} + \frac{1}{L}\psi$$

Comparing this with the mechanical equation in terms of

displacement x:

$$f(t) = M\frac{d^2x}{dt^2} + B\frac{dx}{dt} + Kx$$

We again see identical mathematical forms, where displacement x

is analogous to magnetic flux linkage ψ.

Force-Current (F-I) Analogy Mapping I

Mechanical System	**Symbol**	**Electrical System**	**Symbol**
Force	f	Current	i
Mass	M	Capacitance	C
Viscous Friction (Damper)	B	Conductance $(1/R)$	G
Spring Stiffness	K	Reciprocal of Inductance	$1/L$
Displacement	x	Magnetic Flux Linkage	ψ
Velocity	$v = \dot{x}$	Voltage	$v = \dot{\psi}$

Rotational System Analogies I

The analogies extend naturally to rotational mechanical systems. The basic equation for a rotational mass-spring-damper system is:

$$T(t) = J\frac{d^2\theta}{dt^2} + B\frac{d\theta}{dt} + K\theta$$

where T is Torque, J is moment of inertia, and θ is angular displacement.

Torque-Voltage Analogy:

- Torque $T \rightarrow$ Voltage e
- Inertia $J \rightarrow$ Inductance L
- Angular Vel. $\omega \rightarrow$ Current i

Torque-Current Analogy:

- Torque $T \rightarrow$ Current i
- Inertia $J \rightarrow$ Capacitance C
- Angular Vel. $\omega \rightarrow$ Voltage v

Concept of System Analogy

Analogous Systems

Systems that belong to entirely different physical domains (e.g., mechanical, electrical, thermal, fluid) but are governed by the **same mathematical model** (identical forms of differential equations) are called *analogous systems*.

- When two different physical systems have equations of the same mathematical form, the solution for one system applies directly to the other.
- This mathematical similarity is the foundation of system analogy.
- For instance, a mechanical system consisting of a mass, spring, and damper behaves mathematically exactly like an electrical circuit containing an inductor, capacitor, and resistor.

Advantages of System Analogy

- **Cross-Domain Familiarity:** Engineers often have strong intuition in one domain (e.g., electrical circuits) and can use analogies to understand complex systems in other domains (e.g., mechanical vibrations).
- **Unified Analysis:** Allows the use of common analytical tools (like Laplace transforms, Bode plots, and block diagrams) across multiple disciplines.
- **Simulation and Modeling:** Mechanical, thermal, or fluid systems can be simulated using equivalent electrical circuits, which are often easier and cheaper to build and modify (foundation of analog computers).
- **Electromechanical Systems:** Essential for analyzing transducers and hybrid systems (like motors and sensors) where mechanical and electrical parts interact.

Mathematical Basis: Translational Mechanical System

Consider a basic mechanical system with Mass (M), viscous friction (B), and a spring (K), subjected to a force $f(t)$. Let x be the displacement and $v = \dot{x}$ be the velocity.

Applying Newton's Second Law:

$$f(t) = f_m + f_b + f_k \tag{5}$$

Substituting the component equations:

$$M\frac{d^2x}{dt^2} + B\frac{dx}{dt} + Kx = f(t) \tag{6}$$

Expressing in terms of velocity ($v = \frac{dx}{dt}$):

$$M\frac{dv}{dt} + Bv + K\int v\,dt = f(t) \tag{7}$$

Mathematical Basis: Series Electrical Circuit

Consider an electrical circuit with a resistor (R), inductor (L), and capacitor (C) connected in series with a voltage source $v(t)$. Let $i(t)$ be the current and $q(t)$ be the charge.

Applying Kirchhoff's Voltage Law (KVL):

$$v(t) = v_L + v_R + v_C \tag{8}$$

Substituting the component equations:

$$L\frac{d^2q}{dt^2} + R\frac{dq}{dt} + \frac{1}{C}q = v(t) \tag{9}$$

Expressing in terms of current ($i = \frac{dq}{dt}$):

$$L\frac{di}{dt} + Ri + \frac{1}{C}\int i\,dt = v(t) \tag{10}$$

Mathematical Basis: Parallel Electrical Circuit

Consider an electrical circuit with a resistor (R), inductor (L), and capacitor (C) connected in parallel with a current source $i(t)$. Let $v(t)$ be the node voltage and $\phi(t)$ be the magnetic flux linkage.

Applying Kirchhoff's Current Law (KCL):

$$i(t) = i_C + i_R + i_L \tag{11}$$

Substituting the component equations:

$$C\frac{d^2\phi}{dt^2} + \frac{1}{R}\frac{d\phi}{dt} + \frac{1}{L}\phi = i(t) \tag{12}$$

Expressing in terms of voltage ($v = \frac{d\phi}{dt}$):

$$C\frac{dv}{dt} + \frac{1}{R}v + \frac{1}{L}\int v\,dt = i(t) \tag{13}$$

Force-Voltage Analogy (Direct Analogy)

Comparing the mechanical equation (7) with the series electrical circuit equation (10):

$$M\frac{dv}{dt} + Bv + K\int v\,dt = f(t) \quad \text{(Mechanical)}$$
$$L\frac{di}{dt} + Ri + \frac{1}{C}\int i\,dt = v(t) \quad \text{(Series Electrical)}$$

We can observe a direct mathematical similarity between the two integro-differential equations. This one-to-one mapping forms the basis of the **Force-Voltage** analogy.

Force-Voltage Analogy Table I

Translational Mechanical System	**Electrical System (Series)**
Force (f)	Voltage (v)
Mass (M)	Inductance (L)
Viscous Friction Coefficient (B)	Resistance (R)
Spring Stiffness (K)	Elastance ($1/C$)
Displacement (x)	Charge (q)
Velocity ($v = \dot{x}$)	Current ($i = \dot{q}$)

Force-Voltage Analogy Table II

Key Insight

In the Force-Voltage analogy, mechanical elements that share the same **velocity** correspond to electrical elements connected in **series** (sharing the same current).

Force-Current Analogy (Inverse Analogy)

Comparing the mechanical equation (7) with the parallel electrical circuit equation (13):

$$
\begin{aligned}
M\frac{dv}{dt} + Bv + K\int v\,dt &= f(t) \quad \text{(Mechanical)} \\
C\frac{dv}{dt} + \frac{1}{R}v + \frac{1}{L}\int v\,dt &= i(t) \quad \text{(Parallel Electrical)}
\end{aligned}
$$

The forms are mathematically identical but map differently. This leads to the **Force-Current** analogy.

Force-Current Analogy Table I

Translational Mechanical System	**Electrical System (Parallel)**
Force (f)	Current (i)
Mass (M)	Capacitance (C)
Viscous Friction Coefficient (B)	Conductance ($1/R$)
Spring Stiffness (K)	Reciprocal of Inductance ($1/L$)
Displacement (x)	Magnetic Flux Linkage (ϕ)
Velocity ($v = \dot{x}$)	Voltage ($v = \dot{\phi}$)

Force-Current Analogy Table II

Key Insight

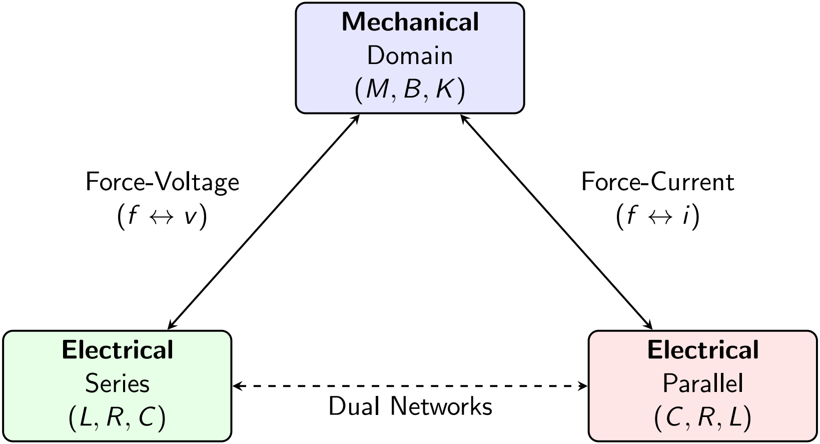
In the Force-Current analogy, mechanical elements that share the same **velocity** correspond to electrical elements connected in **parallel** (sharing the same node voltage).

Rotational Analogies

The same concepts apply to rotational mechanical systems, replacing Force with Torque (T) and mass with moment of inertia (J).

Rotational Mechanical	**Torque-Voltage**	**Torque-Current**
Torque (T)	Voltage (v)	Current (i)
Moment of Inertia (J)	Inductance (L)	Capacitance (C)
Rotational Friction (B)	Resistance (R)	Conductance ($1/R$)
Torsional Stiffness (K)	Elastance ($1/C$)	Reciprocal Ind. ($1/L$)
Angular Velocity (ω)	Current (i)	Voltage (v)
Angular Displacement (θ)	Charge (q)	Magnetic Flux (ϕ)

Visualizing the Analogy Mapping I



Duality

The Force-Voltage and Force-Current analogies result in electrical circuits that are **duals** of each other. A series RLC circuit is the exact dual of a parallel RLC circuit.

Introduction to Time Response

- The **time response** of a control system is how the output of a system changes with respect to time when subjected to a specific input.
- Analyzing the time response is crucial for understanding the dynamic behavior and performance of a system before it reaches its final, steady condition.
- When an input is applied, the system cannot respond instantaneously due to its inherent inertia, energy storage elements (like capacitors and inductors), or physical constraints.
- Therefore, the total response is analyzed in two distinct parts:
  1. Transient Response
  2. Steady-State Response

$$c(t) = c_{tr}(t) + c_{ss}(t)$$

Where $c(t)$ is total time response, $c_{tr}(t)$ is transient response, and $c_{ss}(t)$ is steady-state response.

Transient Response

Transient Response, $c_{tr}(t)$

The part of the time response that goes to zero as time becomes very large is called the **transient response**. Mathematically, $\lim_{t\to\infty} c_{tr}(t) = 0$.

- It reveals how a system transitions from its initial state to its final state.
- The nature of the transient response depends entirely on the system poles (the roots of the characteristic equation).
- Key characteristics observed during the transient period include:
  - **Speed of response:** How fast the system reacts to the input.
  - **Overshoot:** Whether the response exceeds its final target before settling.
  - **Oscillations:** Whether the response fluctuates around the final value.

Steady-State Response I

Steady-State Response, $c_{ss}(t)$

The part of the time response that remains after the transient response has completely died out is called the **steady-state response**.

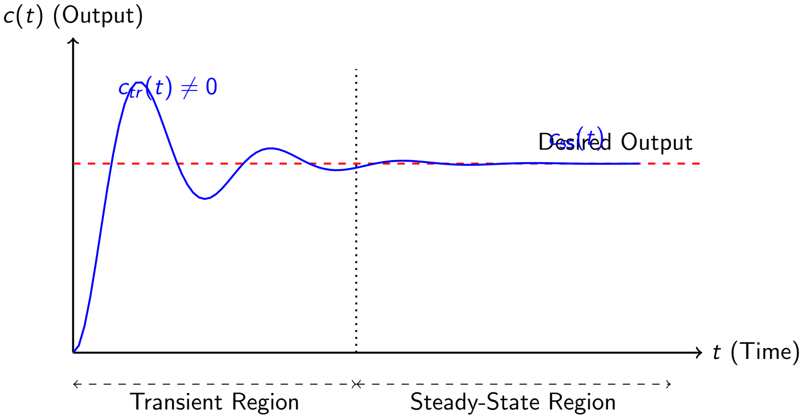
- It indicates the final accuracy of the control system when dealing with a sustained input.
- The steady-state response depends on both the system dynamics and the type of input signal applied.
- **Steady-State Error (e_{ss}):** The difference between the desired output (reference input) and the actual steady-state output. Minimizing this error is a primary goal of control system design.

Steady-State Response II

Stability Condition

A system must be strictly **stable** for a meaningful steady-state response to exist. If the system is unstable, the transient response grows indefinitely instead of decaying to zero.

Graphical Representation I



- **Transient Region:** The output fluctuates and approaches the final value.

Graphical Representation II

- **Steady-State Region:** The output has settled to its final value (within a small tolerance band).

Standard Test Signals

To evaluate and compare the time response of different systems, standard test signals are used as inputs. They represent typical real-world disturbances or commands.

1. Step Signal ($u(t)$)

- Represents a sudden change in reference input.
- e.g., turning on a switch.

2. Ramp Signal ($t \cdot u(t)$)

- Represents a constant velocity input.
- e.g., tracking a target moving at constant speed.

3. Parabolic Signal ($t^2/2 \cdot u(t)$)

- Represents a constant acceleration input.
- e.g., tracking an accelerating missile.

4. Impulse Signal ($\delta(t)$)

- Represents a sudden shock or disturbance.
- e.g., a lightning strike or a hammer blow.

Example: Step Response of a First-Order System

Consider a simple RC circuit where the input is voltage $v_i(t)$ and output is voltage across the capacitor $v_c(t)$. The transfer function is $G(s) = \frac{1}{RCs+1}$.
Let $T = RC$ (time constant). If a unit step input $V_i(s) = 1/s$ is applied:

$$V_c(s) = \frac{1}{s(Ts+1)} = \frac{1}{s} - \frac{T}{Ts+1} = \frac{1}{s} - \frac{1}{s+1/T}$$

$$v_c(t) = \left(1 - e^{-t/T}\right) u(t)$$

Separating the response:

- Total Response: $c(t) = 1 - e^{-t/T}$
- **Transient Response:** $c_{tr}(t) = -e^{-t/T}$ (Note that as $t \to \infty$, $c_{tr}(t) \to 0$)
- **Steady-State Response:** $c_{ss}(t) = 1$

Introduction to Time Response

- The **time response** of a control system is the variation of its output with respect to time when subjected to a given input.
- The overall time response of any linear control system consists of two distinct components:

$$c(t) = c_t(t) + c_{ss}(t)$$

Where:

- $c(t)$ = Total time response
- $c_t(t)$ = **Transient response**
- $c_{ss}(t)$ = **Steady-state response**

Transient Response

Transient Response

The part of the time response that goes to zero as time becomes very large is called the transient response.

- Mathematically, the transient response must satisfy:

$$\lim_{t \to \infty} c_t(t) = 0$$

- **Significance:** It reveals the dynamic behavior of the system, determining how fast the system responds, how much it oscillates, and how quickly it settles.
- It depends primarily on the system components and their layout (poles of the transfer function), not just the input.

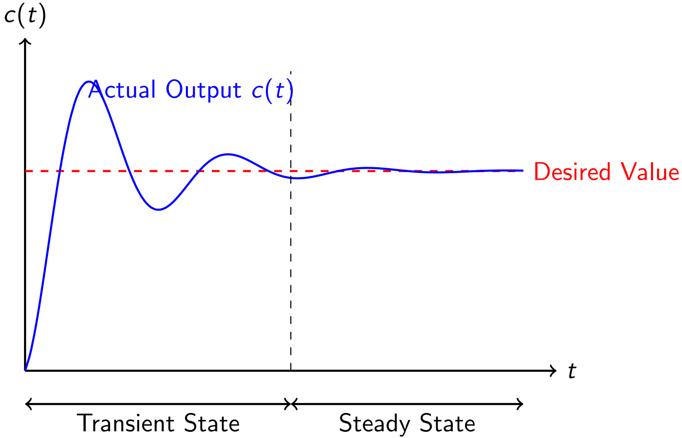
Steady-State Response

Steady-State Response

The part of the time response that remains even after the transient response has died out is called the steady-state response.

- It represents the final value of the output as $t \to \infty$.
- **Significance:** It indicates the final accuracy of the system and helps in evaluating the *steady-state error*.
- The steady-state error is the difference between the desired output (reference input) and the actual steady-state output.
- It depends on both the system dynamics and the type of input applied.

Graphical Illustration of Time Response



Standard Test Signals

Why Standard Test Signals?

To predict the time response of a system, it is necessary to know the input signal. Since practical inputs are often random or unpredictable, we use standard mathematical signals to analyze and compare the performance of various systems.

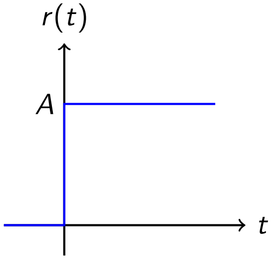
- The commonly used standard test signals are:
  1. **Step Signal** (Sudden change)
  2. **Ramp Signal** (Constant velocity)
  3. **Parabolic Signal** (Constant acceleration)
  4. **Impulse Signal** (Sudden shock)

1. Step Signal

Step Signal

A signal whose value changes from zero to a constant level A at $t = 0$.

$$r(t) = \begin{cases} A, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

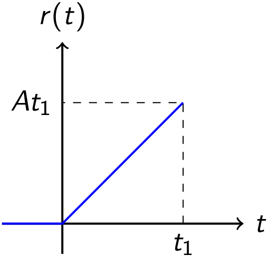


- When $A = 1$, it is a **Unit Step Signal**, denoted by $u(t)$.
- Laplace Transform: $R(s) = \frac{A}{s}$

Ramp Signal

A signal whose value starts at zero and increases linearly with time.

$$r(t) = \begin{cases} At, & t \geq 0 \\ 0, & t < 0 \end{cases}$$



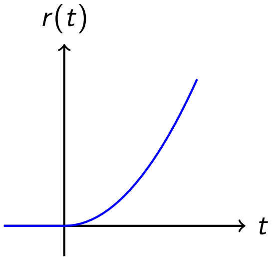
- Represents a constant velocity input.
- When $A = 1$, it is a **Unit Ramp Signal**.
- Laplace Transform: $R(s) = \frac{A}{s^2}$

Parabolic Signal

A signal that represents a constant acceleration input.

$$r(t) = \begin{cases} \frac{A}{2}t^2, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

- When $A = 1$, it is a **Unit Parabolic Signal**.
- Laplace Transform: $R(s) = \frac{A}{s^3}$

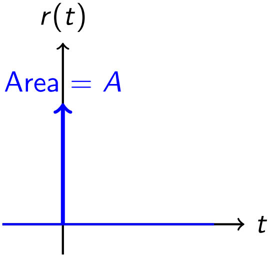


Impulse Signal

A signal of infinite magnitude and zero duration, with a finite area A.

$$r(t) = \begin{cases} \infty, & t = 0 \\ 0, & t \neq 0 \end{cases}$$

- Condition: $\int_{-\epsilon}^{\epsilon} r(t)dt = A$
- When $A = 1$, it is a **Unit Impulse Signal**, $\delta(t)$.
- Laplace Transform: $R(s) = A$



Standard Test Signals in Control Systems

- In control systems, standard test signals are used to evaluate the performance and characteristics of a system.
- Since the actual input signals in real-world systems (like human input or random disturbances) are often unpredictable and difficult to express mathematically, we use standard, well-defined test signals.
- The response of a system to these standard signals provides valuable insights into its steady-state and transient behavior.

Why use standard test signals?

By analyzing a system's response to standard inputs like step, ramp, or impulse, we can predict how it will behave under typical operating conditions. It provides a common basis for comparing different control systems.

Step Signal (Position Function) I

- A step signal represents a sudden change in the reference input.
- It is analogous to turning on a switch or applying a sudden constant force or voltage.

Step Signal

The step signal $r(t)$ with amplitude A is defined as:

$$r(t) = \begin{cases} A, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

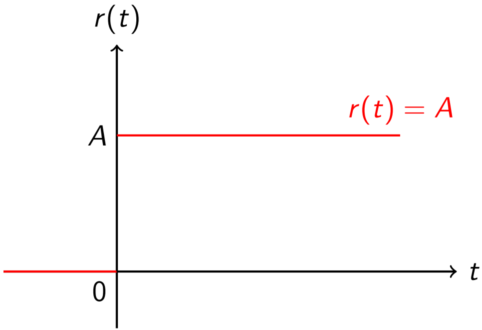
If $A = 1$, it is called a **unit step signal**, denoted by $u(t)$.

Step Signal (Position Function) II

- Laplace Transform of a unit step signal:

$$R(s) = \mathcal{L}\{u(t)\} = \int_0^\infty 1 \cdot e^{-st} dt = \frac{1}{s}$$

Step Signal Visualization I



- Used to study the **steady-state error** and **transient response** of systems tracking a constant position.

Ramp Signal (Velocity Function) I

- A ramp signal starts at zero and increases linearly with time.
- It represents a constant velocity input, such as an object moving at a constant speed or a reference level changing at a constant rate.

Ramp Signal

The ramp signal $r(t)$ with slope A is defined as:

$$r(t) = \begin{cases} At, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

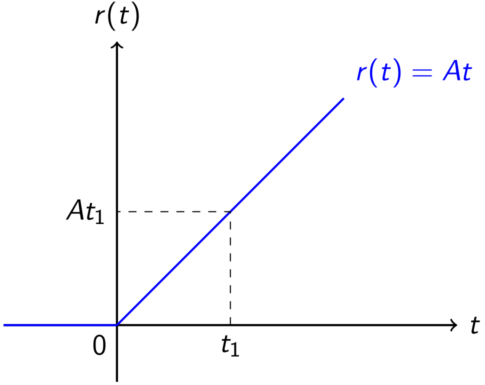
If $A = 1$, it is called a **unit ramp signal**, where $r(t) = t$ for $t \geq 0$.

Ramp Signal (Velocity Function) II

- Laplace Transform of a unit ramp signal:

$$R(s) = \mathcal{L}\{t\} = \frac{1}{s^2}$$

Ramp Signal Visualization I



- Used to determine the ability of a control system to track a constantly moving target (e.g., radar tracking a satellite).

Parabolic Signal (Acceleration Function) I

- A parabolic signal increases quadratically with time.
- It represents a constant acceleration input, where the rate of change of velocity is constant.

Parabolic Signal

The parabolic signal $r(t)$ is defined as:

$$r(t) = \begin{cases} \frac{A}{2}t^2, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

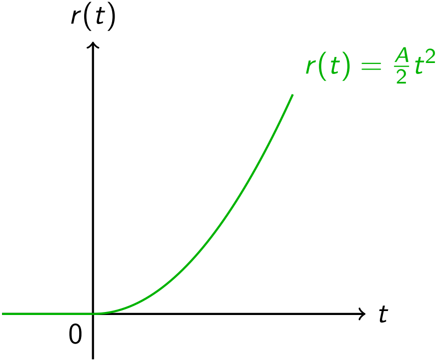
If $A = 1$, it is called a **unit parabolic signal**, where $r(t) = \frac{t^2}{2}$ for $t \geq 0$.

Parabolic Signal (Acceleration Function) II

- Laplace Transform of a unit parabolic signal:

$$R(s) = \mathcal{L}\left\{\frac{t^2}{2}\right\} = \frac{1}{s^3}$$

Parabolic Signal Visualization I



- Used to evaluate a system's tracking capability under constant acceleration inputs (e.g., a missile tracking an accelerating target).

Impulse Signal I

- An impulse signal represents an infinitely large input applied for an infinitely small duration of time.
- It is analogous to a sudden shock, a lightning strike, or a hammer blow.

Unit Impulse Signal (Dirac Delta Function)

The unit impulse signal $\delta(t)$ is defined such that:

$$\delta(t) = \begin{cases} \infty, & t = 0 \\ 0, & t \neq 0 \end{cases}$$

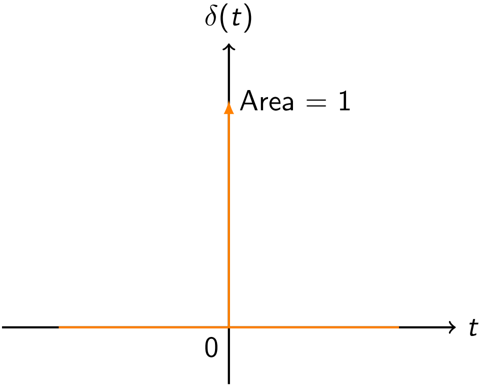
And the area under the curve is unity:

$$\int_{-\infty}^{\infty} \delta(t) dt = 1$$

- Laplace Transform of a unit impulse signal:

$$R(s) = \mathcal{L}\{\delta(t)\} = 1$$

Impulse Signal Visualization I



Relationship between Signals

The standard test signals are related by integration and differentiation:

$$\text{Impulse} \xrightarrow{\int} \text{Step} \xrightarrow{\int} \text{Ramp} \xrightarrow{\int} \text{Parabolic}$$

Summary of Standard Test Signals

Signal Type	**Time Domain** $r(t)$ **for** $t \geq 0$	**Laplace Domain** $R(s)$
Unit Impulse	$\delta(t)$	1
Unit Step	1	$\frac{1}{s}$
Unit Ramp	t	$\frac{1}{s^2}$
Unit Parabolic	$\frac{t^2}{2}$	$\frac{1}{s^3}$

- These relationships make analyzing control systems systematically simpler, as multiplying by $1/s$ in the Laplace domain corresponds to integration in the time domain.

Introduction to Standard Test Signals

- The actual input to a control system is often unpredictable or complex (e.g., wind gusts on an antenna, temperature variations in a furnace).
- To analyze and design control systems systematically, we subject them to specific, simple deterministic signals known as **standard test signals**.
- By studying the time response of a system to these standard inputs, we can infer its behavior under typical operating conditions.

Key Concept

The characteristics of an actual input signal can often be approximated by a combination of standard test signals. If a control system performs well under these test signals, it will generally perform well under actual operating conditions.

Common Standard Test Signals

The four most commonly used standard test signals in control systems are:

1. **Step Signal**: Represents a sudden change in input.
2. **Ramp Signal**: Represents a constant velocity or linearly changing input.
3. **Parabolic Signal**: Represents a constant acceleration or quadratically changing input.
4. **Impulse Signal**: Represents a sudden shock or impact.

Step Signal I

Step Signal

A step signal is a signal whose value changes from zero to a constant level A at $t = 0$ and remains constant thereafter. Mathematically, it is defined as:

$$r(t) = \begin{cases} A, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

- If $A = 1$, the signal is called a **unit step signal**, denoted as $u(t)$.
- **Physical significance:** It represents a sudden application of input, like switching on a constant voltage source, or instantly changing the setpoint of a thermostat.

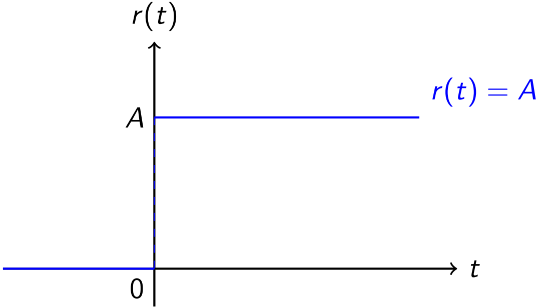
Step Signal II

Laplace Transform

The Laplace transform of a step signal $r(t) = Au(t)$ is:

$$R(s) = \frac{A}{s}$$

Step Signal: Visualization I



- The signal exhibits a discontinuity at $t = 0$.
- It is the most common test signal used to evaluate the **transient response** of a system.

Ramp Signal I

Ramp Signal

A ramp signal is a signal whose value increases linearly with time from $t = 0$. Mathematically, it is defined as:

$$r(t) = \begin{cases} At, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

- Here, A represents the slope of the signal.
- If $A = 1$, it is called a **unit ramp signal**.
- **Physical significance:** It represents an input moving at a **constant velocity**. E.g., a radar tracking an aircraft flying at a constant speed.

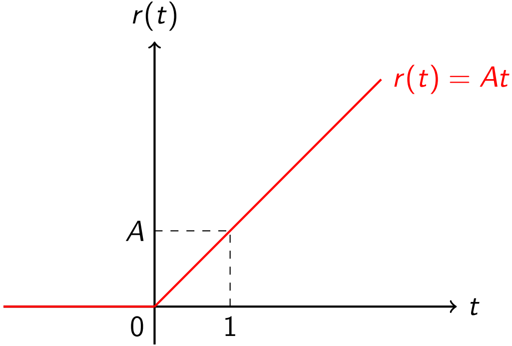
Ramp Signal II

Laplace Transform

The Laplace transform of a ramp signal $r(t) = At$ is:

$$R(s) = \frac{A}{s^2}$$

Ramp Signal: Visualization I



- Used to evaluate how well a system can track a constantly changing target (i.e., steady-state error for velocity inputs).

Parabolic Signal I

Parabolic Signal

A parabolic signal is a signal whose value varies quadratically with time from $t = 0$. Mathematically, it is defined as:

$$r(t) = \begin{cases} \frac{At^2}{2}, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

- If $A = 1$, it is called a **unit parabolic signal**.
- **Physical significance:** It represents a signal with **constant acceleration**. E.g., a control system tracking a missile accelerating at a constant rate.

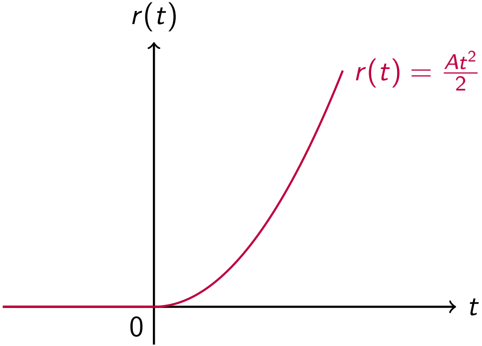
Parabolic Signal II

Laplace Transform

The Laplace transform of a parabolic signal $r(t) = \frac{At^2}{2}$ is:

$$R(s) = \frac{A}{s^3}$$

Parabolic Signal: Visualization I



- Used to determine the system's ability to track constantly accelerating inputs.

Impulse Signal

An ideal impulse signal is defined as a signal that has infinite magnitude at $t = 0$ and zero magnitude everywhere else, with the area under the curve equal to A. Mathematically:

$$r(t) = \begin{cases} \infty, & t = 0 \\ 0, & t \neq 0 \end{cases} \quad \text{and} \quad \int_{-\infty}^{\infty} r(t)dt = A$$

- If $A = 1$, it is the **unit impulse signal**, often represented by the Dirac delta function $\delta(t)$.
- **Physical significance:** Represents a sudden, very brief disturbance, like a hammer blow, lightning strike, or an electrical noise spike.

Impulse Signal II

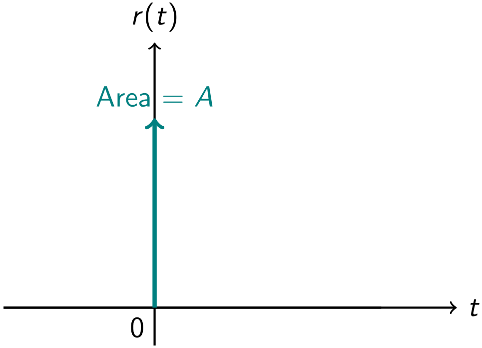
Laplace Transform

The Laplace transform of a unit impulse signal $\delta(t)$ is:

$$R(s) = 1$$

For $r(t) = A\delta(t)$, the transform is $R(s) = A$.

Impulse Signal: Visualization I



- The impulse response of a system is a fundamental property. The Laplace transform of a system's impulse response is identical to its Transfer Function.

Relationships Between Standard Signals

The standard test signals are mathematically related by integration and differentiation:

$$\begin{tikzcd}
\boxed{\text{Impulse}} \arrow[r, bend left, "\int dt"] & \boxed{\text{Step}} \arrow[l, bend left, "\frac{d}{dt}"] \arrow[r, bend left, "\int dt"] & \boxed{\text{Ramp}} \arrow[l, bend left, "\frac{d}{dt}"] \arrow[r, bend left, "\int dt"] & \boxed{\text{Parabolic}} \arrow[l, bend left, "\frac{d}{dt}"]
\end{tikzcd}$$

Summary Table

Signal Type	**Time Domain** $r(t)$	**Laplace Domain** $R(s)$
Impulse Signal	$A\delta(t)$	A
Step Signal	$Au(t)$	$\frac{A}{s}$
Ramp Signal	$At\,u(t)$	$\frac{A}{s^2}$
Parabolic Signal	$\frac{At^2}{2}u(t)$	$\frac{A}{s^3}$

Order of a System

Order of a System

The **order** of a control system is determined by the highest power of the complex variable s in the denominator of its closed-loop transfer function (or characteristic equation).

Consider a closed-loop transfer function:

$$\frac{C(s)}{R(s)} = \frac{b_m s^m + b_{m-1} s^{m-1} + \cdots + b_0}{a_n s^n + a_{n-1} s^{n-1} + \cdots + a_0}$$

where $n \geq m$.

- The highest power of s in the denominator is n.
- Therefore, the order of this system is n.
- It represents the number of independent energy storage elements in the physical system.

Type of a System

Type of a System

The **type** of a control system is determined by the number of open-loop poles located at the origin ($s = 0$) of the s-plane.

Consider the open-loop transfer function (OLTF) $G(s)H(s)$:

$$G(s)H(s) = \frac{K(1 + T_1 s)(1 + T_2 s)\dots}{s^N(1 + T_a s)(1 + T_b s)\dots}$$

- Here, N represents the number of poles at the origin.
- If $N = 0$, it is a **Type 0** system.
- If $N = 1$, it is a **Type 1** system.
- If $N = 2$, it is a **Type 2** system, and so on.

For a typical closed-loop system with forward path transfer function $G(s)$ and feedback path transfer function $H(s)$, the overall transfer function is:

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 \pm G(s)H(s)}$$

Characteristic Equation

The equation obtained by equating the denominator polynomial of the closed-loop transfer function to zero is called the **characteristic equation**.

For a negative feedback system, the characteristic equation is:

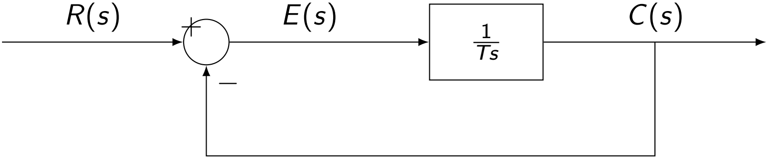
$$1 + G(s)H(s) = 0$$

Characteristic Equation II

The roots of the characteristic equation are the **closed-loop poles** of the system, which dictate the system's dynamic response and stability.

Standard First Order System I

A first order system is one whose order is $n = 1$. The standard block diagram of a first-order closed-loop system with unity negative feedback is shown below:



The closed-loop transfer function is:

$$\frac{C(s)}{R(s)} = \frac{\frac{1}{Ts}}{1 + \frac{1}{Ts}} = \frac{1}{Ts + 1}$$

Here, T is the **time constant** of the system.

Time Response to Step Input

Let's find the time response $c(t)$ of the first order system when subjected to a unit step input.

1. Define the input: For a unit step input, $r(t) = u(t)$. Taking the Laplace transform, we get:

$$R(s) = \frac{1}{s}$$

2. Find the output in s-domain: We know that $C(s) = R(s) \cdot \frac{1}{Ts+1}$. Substituting $R(s)$:

$$C(s) = \frac{1}{s} \cdot \frac{1}{Ts+1} = \frac{1}{s(Ts+1)}$$

To find the time response, we need to take the inverse Laplace transform. First, let's use partial fraction expansion.

Partial Fraction Expansion

$$C(s) = \frac{1}{s(Ts+1)} = \frac{A}{s} + \frac{B}{Ts+1}$$

Solving for A and B:

$$1 = A(Ts+1) + B(s)$$

- Put $s = 0$: $1 = A(0+1) \implies A = 1$
- Put $s = -1/T$: $1 = A(0) + B(-1/T) \implies B = -T$

Substitute A and B back into the equation:

$$C(s) = \frac{1}{s} - \frac{T}{Ts+1}$$

$$C(s) = \frac{1}{s} - \frac{1}{s + \frac{1}{T}}$$

Now, take the inverse Laplace transform of $C(s)$ to find $c(t)$:

$$c(t) = \mathcal{L}^{-1}\left\{\frac{1}{s}\right\} - \mathcal{L}^{-1}\left\{\frac{1}{s+\frac{1}{T}}\right\}$$

Using standard Laplace transform pairs:

- $\mathcal{L}^{-1}\left\{\frac{1}{s}\right\} = 1 \cdot u(t)$
- $\mathcal{L}^{-1}\left\{\frac{1}{s+a}\right\} = e^{-at} \cdot u(t)$

Therefore, the time response is:

Step Response of First Order System

$$c(t) = 1 - e^{-\frac{t}{T}} \quad \text{for } t \geq 0$$

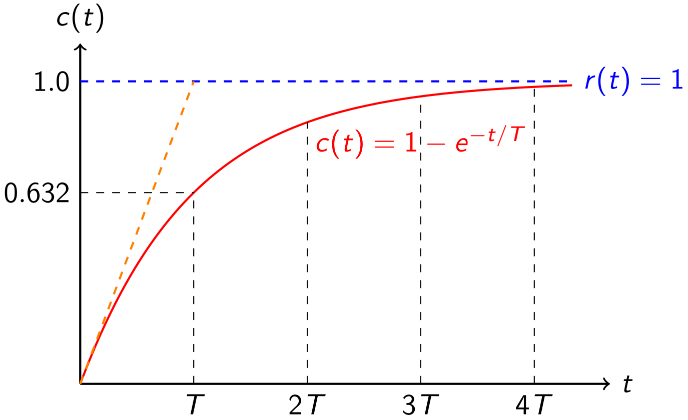
Graphical Representation

Let's evaluate $c(t)$ at different multiples of the time constant T:

Time (t)	**Value of** $e^{-t/T}$	**Response** $c(t) = 1 - e^{-t/T}$
0	1	0
T	0.368	0.632 (63.2%)
$2T$	0.135	0.865 (86.5%)
$3T$	0.050	0.950 (95.0%)
$4T$	0.018	0.982 (98.2%)
$5T$	0.0067	0.993 (99.3%)
∞	0	1 (100%)

- The response reaches 63.2% of its final value in one time constant T.
- The system is generally considered to have reached its steady state at $t = 4T$ (2% error band) or $t = 5T$ (1% error band).

Plot of the Step Response I



The initial slope of the response curve is $1/T$. If the response maintained this initial rate of change, it would reach the final value in time $t = T$.

Error Signal I

The error signal $e(t)$ is the difference between the input signal and the output signal.

$$E(s) = R(s) - C(s)$$

Substituting the expressions for $R(s)$ and $C(s)$:

$$E(s) = \frac{1}{s} - \left(\frac{1}{s} - \frac{1}{s + 1/T}\right) = \frac{1}{s + 1/T}$$

Taking the inverse Laplace transform:

$$e(t) = e^{-\frac{t}{T}} \quad \text{for } t \geq 0$$

Steady-State Error (e_{ss}): The steady-state error is the value of the error signal as $t \to \infty$.

$$e_{ss} = \lim_{t \to \infty} e(t) = \lim_{t \to \infty} e^{-t/T} = 0$$

Error Signal II

A Type 1, first-order system has zero steady-state error for a step input. It tracks a step input perfectly in the steady state.

System Order

The **order** of a control system is defined as the highest power of the complex variable s in the denominator polynomial (characteristic polynomial) of its closed-loop transfer function. It corresponds to the number of independent energy-storing elements in the system.

System Type

The **type** of a control system is determined by the number of poles of the open-loop transfer function $G(s)H(s)$ located at the origin ($s = 0$).

$$G(s)H(s) = \frac{K(1 + T_a s)(1 + T_b s) \cdots}{s^N (1 + T_1 s)(1 + T_2 s) \cdots}$$

Here, N represents the **type** of the system.

Characteristic Equation I

For a closed-loop control system with open-loop transfer function $G(s)$ and feedback path transfer function $H(s)$, the closed-loop transfer function is given by:

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)}$$

Characteristic Equation

The equation obtained by setting the denominator polynomial of the closed-loop transfer function to zero is called the **characteristic equation**.

$$1 + G(s)H(s) = 0$$

Characteristic Equation II

- The roots of the characteristic equation are the **closed-loop poles** of the system.
- These roots entirely determine the transient response and stability of the system.

A first order system is one where the highest power of s in the denominator of the transfer function is 1.
The general form of the transfer function of a first order system is:

$$\frac{C(s)}{R(s)} = \frac{K}{\tau s + 1}$$

where:

- K is the DC gain of the system.
- τ is the **time constant** of the system.

Let's consider a standard first order system with unity DC gain ($K = 1$):

$$\frac{C(s)}{R(s)} = \frac{1}{\tau s + 1}$$

The characteristic equation is $\tau s + 1 = 0 \implies s = -1/\tau$. (It has one pole, hence it is a 1st order system).

Step Response of a First Order System

We want to find the time response $c(t)$ to a **unit step input** $r(t) = u(t)$.

- The Laplace transform of the unit step input is $R(s) = \frac{1}{s}$.

Substituting $R(s)$ into the transfer function:

$$C(s) = R(s) \cdot \frac{1}{\tau s + 1} = \frac{1}{s(\tau s + 1)}$$

To find the inverse Laplace transform, we use partial fraction expansion:

$$C(s) = \frac{1}{s(\tau s + 1)} = \frac{A}{s} + \frac{B}{\tau s + 1}$$

Solving for A and B:

$$1 = A(\tau s + 1) + Bs$$

At $s = 0 \implies A = 1$. At $s = -1/\tau \implies 1 = B(-1/\tau) \implies B = -\tau$.

Time Response Equation

Substituting the partial fraction coefficients back:

$$C(s) = \frac{1}{s} - \frac{\tau}{\tau s + 1} = \frac{1}{s} - \frac{1}{s + 1/\tau}$$

Taking the inverse Laplace transform:

$$c(t) = \mathcal{L}^{-1}\left\{\frac{1}{s}\right\} - \mathcal{L}^{-1}\left\{\frac{1}{s + 1/\tau}\right\}$$

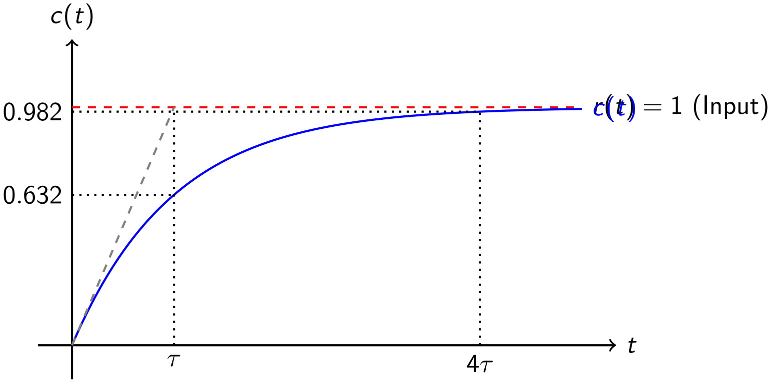
Unit Step Response

The time response of a standard first order system to a unit step input is:

$$c(t) = 1 - e^{-t/\tau} \quad \text{for } t \geq 0$$

The term 1 represents the **steady-state response**, and the term $-e^{-t/\tau}$ represents the **transient response**.

Graphical Representation of Step Response I



- The slope of the response curve at $t = 0$ is $1/\tau$.
- If the initial rate of change were maintained, the response would reach the final steady-state value in exactly one time constant τ.

Significance of the Time Constant (τ) I

The **time constant** (τ) is a measure of how fast the system responds to a change in input.

Time (t)	$c(t) = 1 - e^{-t/\tau}$	**% of Final Value**
1τ	$1 - e^{-1} = 0.632$	63.2%
2τ	$1 - e^{-2} = 0.865$	86.5%
3τ	$1 - e^{-3} = 0.950$	95.0%
4τ	$1 - e^{-4} = 0.982$	98.2%
5τ	$1 - e^{-5} = 0.993$	99.3%

- **Settling Time (t_s):** The time required for the response to reach and stay within a specified tolerance band of its final value.
- For a 2% tolerance band, $t_s \approx 4\tau$. For a 5% tolerance band, $t_s \approx 3\tau$.
- A smaller τ indicates a faster system response.

Error Analysis for Step Input

The error signal $e(t)$ is the difference between the input $r(t)$ and the output $c(t)$:

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

For a unit step input, $r(t) = 1$ for $t \geq 0$.

$$e(t) = 1 - \left(1 - e^{-t/\tau}\right) = e^{-t/\tau}$$

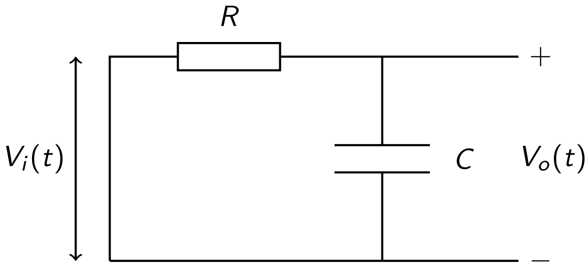
Steady-State Error (e_{ss})

The steady-state error is the error as time approaches infinity:

$$e_{ss} = \lim_{t\to\infty} e(t) = \lim_{t\to\infty} e^{-t/\tau} = 0$$

- This specific 1st order closed-loop system accurately tracks a step input with zero steady-state error.

Engineering Example: RC Low-Pass Filter I



Applying Kirchhoff's Voltage Law (KVL):

$$V_i(t) = Ri(t) + V_o(t) \quad \text{and} \quad i(t) = C\frac{dV_o(t)}{dt}$$

Substituting $i(t)$ and taking the Laplace transform:

$$V_i(s) = RCsV_o(s) + V_o(s) \implies \frac{V_o(s)}{V_i(s)} = \frac{1}{RCs + 1}$$

Engineering Example: RC Low-Pass Filter II

This is a standard first order transfer function with time constant $\tau = RC$.

Introduction to Second-Order Systems I

- The standard form of a closed-loop transfer function for a second-order system is given by:

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

- Two fundamental parameters define the behavior of the system:

Natural Frequency (ω_n)

The frequency at which the system would oscillate if there were no damping ($\zeta = 0$). It is measured in rad/sec.

Introduction to Second-Order Systems II

Damping Ratio (ζ)

A dimensionless measure that describes how the system's oscillations decay over time after a disturbance.

Step Response of a Second-Order System

- For a unit step input, $r(t) = u(t)$, the Laplace transform is $R(s) = \frac{1}{s}$.
- The system output response in the s-domain becomes:

$$C(s) = \frac{\omega_n^2}{s(s^2 + 2\zeta\omega_n s + \omega_n^2)}$$

- The dynamic behavior is governed by the roots of the characteristic equation ($s^2 + 2\zeta\omega_n s + \omega_n^2 = 0$), completely depending on the value of ζ:

- **Undamped ($\zeta = 0$)**: Continuous, undiminished oscillations.
- **Underdamped ($0 < \zeta < 1$)**: Decaying oscillations (most practical control systems).
- **Critically damped ($\zeta = 1$)**: Fastest return to steady state without oscillation.
- **Overdamped ($\zeta > 1$)**: Sluggish, purely exponential response.

The Underdamped Case ($0 < \zeta < 1$) I

- Most control systems are designed to be slightly underdamped to ensure a rapid approach to the final steady-state value.
- Taking the inverse Laplace transform of $C(s)$, the time response $c(t)$ is:

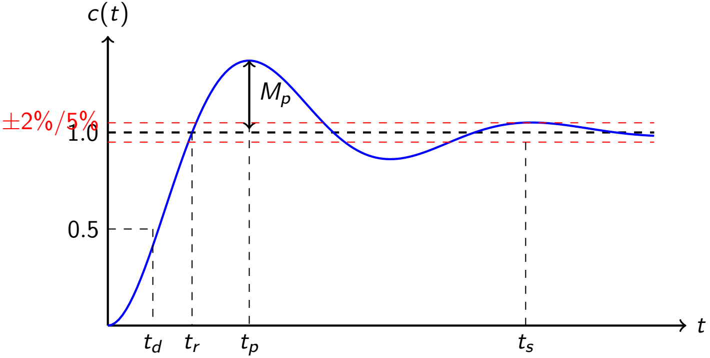
$$c(t) = 1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}} \sin(\omega_d t + \theta) \quad \text{for } t \geq 0$$

- Where the parameters are defined as:
  - **Damped Natural Frequency**: $\omega_d = \omega_n\sqrt{1-\zeta^2}$
  - **Phase Angle**: $\theta = \tan^{-1}\left(\frac{\sqrt{1-\zeta^2}}{\zeta}\right) = \cos^{-1}(\zeta)$

Transient Response Nature

The response contains an oscillation frequency of ω_d and is constrained within an exponentially decaying envelope bounded by $1 \pm \frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}}$.

When characterizing the transient response of an underdamped second-order system to a step input, we rely on standard performance metrics:



Delay Time (t_d)

The time required for the step response to reach **50%** of its final steady-state value for the very first time.

- For practical purposes, an empirical approximation is frequently used:

$$t_d \approx \frac{1 + 0.7\zeta}{\omega_n}$$

Rise Time (t_r)

The time required for the response to rise from **0% to 100%** of its final value for the first time (for underdamped systems).

Delay Time (t_d) and Rise Time (t_r) II

- Analytically, it is found by setting $c(t_r) = 1$:

$$t_r = \frac{\pi - \theta}{\omega_d}$$

- Note: θ must be evaluated in radians.

Peak Time (t_p) I

Peak Time (t_p)

The time required for the response to reach the **first peak** of the overshoot.

- At the peak, the slope of the response curve is zero ($\frac{dc}{dt} = 0$).
- Differentiating $c(t)$ and equating it to zero yields:

$$t_p = \frac{\pi}{\omega_d} = \frac{\pi}{\omega_n \sqrt{1 - \zeta^2}}$$

Peak Time (t_p) II

Oscillation Cycle

The peak time represents the time of the first half-cycle of the damped oscillation. Subsequent overshoot and undershoot peaks occur at integer multiples of t_p ($n \cdot t_p$ for $n = 1, 2, 3 \ldots$).

Maximum Overshoot (M_p)

The maximum peak value of the response curve measured from unity (the desired steady-state final value).

- It is obtained by evaluating the time response $c(t)$ exactly at the peak time $t = t_p$:

$$M_p = c(t_p) - 1 = e^{-\frac{\pi\zeta}{\sqrt{1-\zeta^2}}}$$

- Often expressed as Percentage Overshoot ($\%OS$):

$$\%OS = M_p \times 100\% = e^{-\frac{\pi\zeta}{\sqrt{1-\zeta^2}}} \times 100\%$$

Maximum Overshoot (M_p or $\%OS$) II

Damping Dependence

The maximum overshoot is a function **exclusively** of the damping ratio ζ. A smaller ζ (less damping) yields a larger overshoot.

Settling Time (t_s)

The time required for the response curve to reach and permanently stay within a specified tolerance band (usually 2% or 5%) of the final steady-state value.

- The response is bounded by the decaying exponential envelope. Since $\sqrt{1-\zeta^2} \approx 1$ for small ζ, we equate the envelope $e^{-\zeta\omega_n t}$ to the tolerance limits.
- **For 2% Tolerance Band:**

$$e^{-\zeta\omega_n t_s} \approx 0.02 \implies t_s \approx \frac{4}{\zeta\omega_n}$$

- **For 5% Tolerance Band:**

$$e^{-\zeta\omega_n t_s} \approx 0.05 \implies t_s \approx \frac{3}{\zeta\omega_n}$$

Summary of Time Domain Specifications I

Specification	**Symbol**	**Formula**
Delay Time	t_d	$\approx \frac{1+0.7\zeta}{\omega_n}$
Rise Time	t_r	$\frac{\pi-\theta}{\omega_d}$
Peak Time	t_p	$\frac{\pi}{\omega_d}$
Max Overshoot	M_p	$e^{-\frac{\pi\zeta}{\sqrt{1-\zeta^2}}}$
Settling Time (2%)	t_s	$\approx \frac{4}{\zeta\omega_n}$

Summary of Time Domain Specifications II

Trade-offs in Design

Decreasing ζ reduces rise time (resulting in a faster system) but inevitably increases the overshoot (leading to more oscillation). A common design compromise is $\zeta \approx 0.7$, providing a balanced response.

Introduction to Second Order Systems

- A second-order control system is one whose dynamics are described by a linear ordinary differential equation of the second order.
- The standard form of the closed-loop transfer function is given by:

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

- The dynamic behavior is completely governed by two parameters:
  - **Natural Frequency (ω_n)**
  - **Damping Ratio (ζ)**

Key Parameters: ω_n and ζ

Natural Frequency (ω_n)

The frequency at which the system would oscillate if all damping were removed. It is a measure of the speed of the system's response. Its unit is rad/sec.

Damping Ratio (ζ)

A dimensionless measure describing how oscillations in a system decay after a disturbance. It is the ratio of actual damping to critical damping:

$$\zeta = \frac{\text{Actual Damping}}{\text{Critical Damping}}$$

Step Response of a Second-Order System

When a unit step input ($R(s) = \frac{1}{s}$) is applied, the characteristic equation is:

$$s^2 + 2\zeta\omega_n s + \omega_n^2 = 0$$

The roots (poles) of the system are:

$$s_{1,2} = -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1}$$

Based on the value of ζ, the system behavior is classified into four cases:

- $\zeta > 1$: **Overdamped** (real, distinct negative roots)
- $\zeta = 1$: **Critically Damped** (real, equal negative roots)
- $0 < \zeta < 1$: **Underdamped** (complex conjugate roots)
- $\zeta = 0$: **Undamped** (purely imaginary roots)

Underdamped Response ($0 < \zeta < 1$)

Most practical control systems are designed to be underdamped because they reach the steady state faster, even though they exhibit initial oscillations.
For a unit step input, the time domain response $c(t)$ is derived as:

$$c(t) = 1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}} \sin\left(\omega_d t + \phi\right)$$

Where:

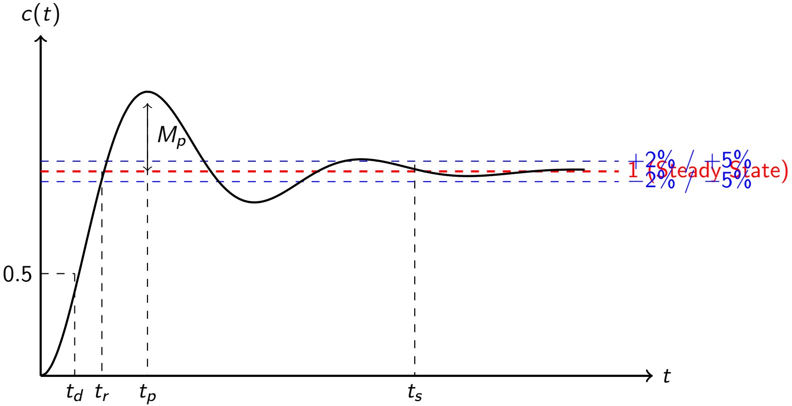
- $\omega_d = \omega_n\sqrt{1-\zeta^2}$ is the **damped natural frequency**
- $\phi = \tan^{-1}\left(\frac{\sqrt{1-\zeta^2}}{\zeta}\right)$ is the phase angle

Transient Response Specifications

The transient response of a practical control system often exhibits damped oscillations before reaching steady state.
To evaluate and specify the desired performance characteristics of a system, the following time-domain specifications are defined based on the step response of an underdamped second-order system:

- Delay Time (t_d)
- Rise Time (t_r)
- Peak Time (t_p)
- Maximum Overshoot (M_p)
- Settling Time (t_s)

Time Response Specifications: Graphical View



Delay Time (t_d) and Rise Time (t_r) I

Delay Time (t_d)

The time required for the response to reach **50%** of its final value for the very first time.

$$t_d \approx \frac{1 + 0.7\zeta}{\omega_n}$$

Rise Time (t_r)

The time required for the response to rise from **0% to 100%** of its final value for an underdamped system.

By setting $c(t_r) = 1$ in the time response equation:

$$t_r = \frac{\pi - \phi}{\omega_d} = \frac{\pi - \tan^{-1}\left(\frac{\sqrt{1-\zeta^2}}{\zeta}\right)}{\omega_n\sqrt{1-\zeta^2}}$$

Peak Time (t_p) and Maximum Overshoot (M_p) I

Peak Time (t_p)

The time required for the response to reach the first peak of the overshoot. It is found by differentiating $c(t)$ with respect to t and equating to zero:

$$t_p = \frac{\pi}{\omega_d} = \frac{\pi}{\omega_n\sqrt{1-\zeta^2}}$$

Peak Time (t_p) and Maximum Overshoot (M_p) II

Maximum Overshoot (M_p)

The maximum peak value of the response curve measured from unity (steady state). It indicates the relative stability of the system.

$$\%M_p = \frac{c(t_p) - c(\infty)}{c(\infty)} \times 100\% = e^{-\left(\frac{\zeta\pi}{\sqrt{1-\zeta^2}}\right)} \times 100\%$$

Note that M_p depends entirely on the damping ratio ζ.

Settling Time (t_s) I

Settling Time (t_s)

The time required for the response curve to reach and stay within a specified percentage (typically 2% or 5%) of the final steady-state value.

The settling time relates to the time constant $\tau = \frac{1}{\zeta\omega_n}$ of the exponentially decaying envelope $e^{-\zeta\omega_n t}$.

For 2% tolerance band: The envelope decays to 0.02.

$$t_s \approx \frac{4}{\zeta\omega_n} = 4\tau$$

For 5% tolerance band: The envelope decays to 0.05.

$$t_s \approx \frac{3}{\zeta\omega_n} = 3\tau$$

Summary of Time Domain Specifications

Specification	Formula
Delay Time (t_d)	$\approx \frac{1+0.7\zeta}{\omega_n}$
Rise Time (t_r)	$\frac{\pi-\phi}{\omega_d}$
Peak Time (t_p)	$\frac{\pi}{\omega_d}$
Max Overshoot ($\%M_p$)	$e^{-\left(\frac{\zeta\pi}{\sqrt{1-\zeta^2}}\right)} \times 100\%$
Settling Time (t_s) for 2%	$\frac{4}{\zeta\omega_n}$
Settling Time (t_s) for 5%	$\frac{3}{\zeta\omega_n}$

Introduction to Steady State Error

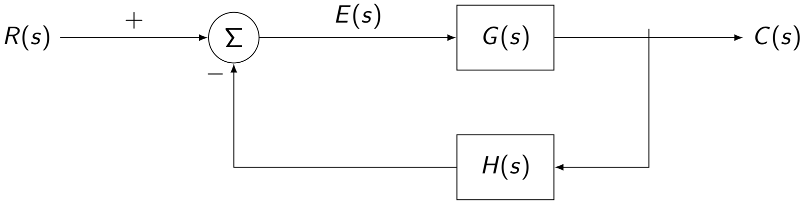
Steady State Error (e_{ss})

The steady state error is defined as the difference between the desired reference input $r(t)$ and the actual output $c(t)$ of a control system as time t approaches infinity. It serves as a measure of system accuracy.

- In a practical control system, it is difficult to achieve a zero error for all types of inputs.
- Some systems may have no steady-state error to a step input but exhibit a large or infinite error to a ramp input.
- Steady state error depends on two main factors:
  1. The type of reference input (e.g., step, ramp, parabolic).
  2. The type of the system (i.e., the number of poles at the origin in the open-loop transfer function).

Derivation of Error Signal I

Consider a standard closed-loop control system:



From the block diagram, the error signal $E(s)$ is:

$$E(s) = R(s) - B(s) = R(s) - C(s)H(s)$$

Since the output is $C(s) = E(s)G(s)$, we can substitute:

$$E(s) = R(s) - E(s)G(s)H(s) \implies E(s)[1 + G(s)H(s)] = R(s)$$

Derivation of Error Signal II

$$E(s) = \frac{R(s)}{1 + G(s)H(s)}$$

Calculation of Steady State Error

To find the steady state error in the time domain, we apply the **Final Value Theorem** to the Laplace transform of the error signal $E(s)$:

$$e_{ss} = \lim_{t \to \infty} e(t) = \lim_{s \to 0} sE(s)$$

Substituting the expression for $E(s)$ from the previous slide:

$$e_{ss} = \lim_{s \to 0} \frac{sR(s)}{1 + G(s)H(s)}$$

This fundamental equation reveals that the steady state error is solely determined by:

- The test input signal $R(s)$.
- The open-loop transfer function $G(s)H(s)$.

Static Error Constants

- In evaluating steady-state performance, we use standard test inputs: step, ramp, and parabolic signals.
- For these standard inputs, we define **Static Error Constants**. These constants are figures of merit for control systems indicating how well the system follows a specific input.
- Generally, the higher the error constants, the smaller the steady state error.

There are three main static error constants:

1. **Position Error Constant (K_p)** for a Step Input.
2. **Velocity Error Constant (K_v)** for a Ramp Input.
3. **Acceleration Error Constant (K_a)** for a Parabolic Input.

1. Static Position Error Constant (K_p)

Consider a step input of magnitude A:
$r(t) = Au(t) \implies R(s) = \frac{A}{s}$.
The steady state error is:

$$e_{ss} = \lim_{s \to 0} \frac{s\left(\frac{A}{s}\right)}{1 + G(s)H(s)} = \frac{A}{1 + \lim_{s \to 0} G(s)H(s)}$$

We define the **Static Position Error Constant (K_p)** as:

$$\boxed{K_p = \lim_{s \to 0} G(s)H(s)}$$

Thus, the steady state error for a step input becomes:

$$e_{ss} = \frac{A}{1 + K_p}$$

Consider a ramp input of magnitude A:
$r(t) = Atu(t) \implies R(s) = \frac{A}{s^2}$.
The steady state error is:

$$e_{ss} = \lim_{s\to 0} \frac{s\left(\frac{A}{s^2}\right)}{1+G(s)H(s)} = \lim_{s\to 0} \frac{A}{s+sG(s)H(s)} = \frac{A}{\lim_{s\to 0} sG(s)H(s)}$$

We define the **Static Velocity Error Constant (K_v)** as:

$$K_v = \lim_{s\to 0} sG(s)H(s)$$

Thus, the steady state error for a ramp input becomes:

$$e_{ss} = \frac{A}{K_v}$$

3. Static Acceleration Error Constant (K_a)

Consider a parabolic input of magnitude A:
$r(t) = \frac{At^2}{2}u(t) \implies R(s) = \frac{A}{s^3}$.
The steady state error is:

$$e_{ss} = \lim_{s\to 0} \frac{s\left(\frac{A}{s^3}\right)}{1+G(s)H(s)} = \lim_{s\to 0} \frac{A}{s^2+s^2G(s)H(s)} = \frac{A}{\lim_{s\to 0} s^2G(s)H(s)}$$

We define the **Static Acceleration Error Constant (K_a)** as:

$$\boxed{K_a = \lim_{s\to 0} s^2G(s)H(s)}$$

Thus, the steady state error for a parabolic input becomes:

$$e_{ss} = \frac{A}{K_a}$$

Summary of Error Constants

Input Type	**Input** $R(s)$	**Error Constant Formula**	**Error** e_{ss}
Step	$\frac{A}{s}$	$K_p = \lim_{s \to 0} G(s)H(s)$	$e_{ss} = \frac{A}{1+K_p}$
Ramp	$\frac{A}{s^2}$	$K_v = \lim_{s \to 0} sG(s)H(s)$	$e_{ss} = \frac{A}{K_v}$
Parabolic	$\frac{A}{s^3}$	$K_a = \lim_{s \to 0} s^2 G(s)H(s)$	$e_{ss} = \frac{A}{K_a}$

- **Note**: These error constants apply only to **stable systems**. If a closed-loop system is unstable, the steady-state error calculation using the Final Value Theorem is invalid because the response grows unbounded.

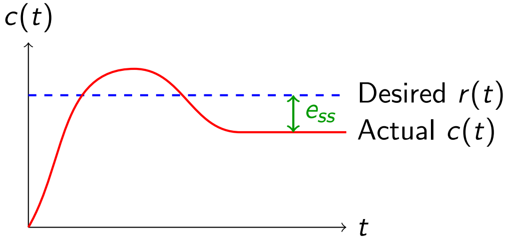
Introduction to Steady-State Error I

Steady-State Error (e_{ss})

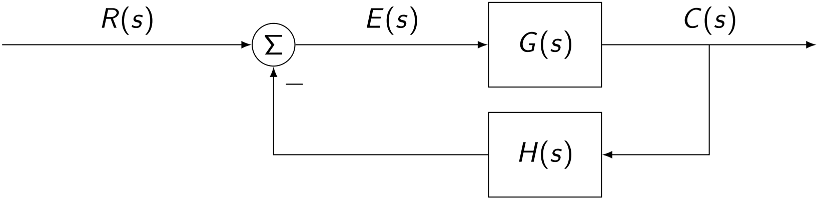
The steady-state error is the difference between the desired output (reference) and the actual output of a control system as time approaches infinity. It serves as a measure of system accuracy.

- In a practical control system, it is highly desirable to have the steady-state error as small as possible, ideally zero.
- The steady-state error depends on two main factors:
  1. The type of input applied to the system (e.g., step, ramp, or parabolic).
  2. The **system type** (the number of poles at the origin of the open-loop transfer function).

Introduction to Steady-State Error II



Consider a standard closed-loop control system:



The actuating error signal $E(s)$ is given by:

$$\begin{aligned} E(s) &= R(s) - C(s)H(s) \\ &= R(s) - E(s)G(s)H(s) \end{aligned}$$

Solving for $E(s)$, we obtain:

$$E(s) = \frac{R(s)}{1 + G(s)H(s)}$$

Mathematical Formulation II

Note: For a unity feedback system ($H(s) = 1$), $E(s)$ is the true error $R(s) - C(s)$.

Applying the Final Value Theorem

Final Value Theorem

If $sE(s)$ does not have poles in the right half of the s-plane or on the imaginary axis, the steady-state error is given by:

$$e_{ss} = \lim_{t \to \infty} e(t) = \lim_{s \to 0} sE(s)$$

Substituting the expression for $E(s)$, we get:

$$e_{ss} = \lim_{s \to 0} \frac{sR(s)}{1 + G(s)H(s)}$$

This shows that the steady-state error depends on the open-loop transfer function $G(s)H(s)$ and the input signal $R(s)$. To quantify this performance, we define **Static Error Constants**.

Static Error Constants Overview

Static error constants are figures of merit used to determine the steady-state accuracy of a control system for standard test inputs.

There are three main static error constants:

1. **Position Error Constant (K_p):** Evaluates error for a step (position) input.
2. **Velocity Error Constant (K_v):** Evaluates error for a ramp (velocity) input.
3. **Acceleration Error Constant (K_a):** Evaluates error for a parabolic (acceleration) input.

These constants provide a quick way to compute steady-state error without performing the full limit calculation every time.

Static Position Error Constant (K_p) I

Let the input be a step signal: $r(t) = Au(t) \implies R(s) = \frac{A}{s}$.
The steady-state error is:

$$e_{ss} = \lim_{s \to 0} \frac{s\left(\frac{A}{s}\right)}{1 + G(s)H(s)}$$
$$= \frac{A}{1 + \lim_{s \to 0} G(s)H(s)}$$

Static Position Error Constant (K_p) II

Position Error Constant

The static position error constant is defined as:

$$K_p = \lim_{s \to 0} G(s)H(s)$$

Thus, the steady-state error for a step input is:

$$e_{ss} = \frac{A}{1 + K_p}$$

Let the input be a ramp signal: $r(t) = Atu(t) \implies R(s) = \frac{A}{s^2}$.
The steady-state error is:

$$
\begin{aligned}
e_{ss} &= \lim_{s \to 0} \frac{s\left(\frac{A}{s^2}\right)}{1 + G(s)H(s)} = \lim_{s \to 0} \frac{A}{s + sG(s)H(s)} \\
&= \frac{A}{\lim_{s \to 0} sG(s)H(s)}
\end{aligned}
$$

Static Velocity Error Constant (K_v) II

Velocity Error Constant

The static velocity error constant is defined as:

$$K_v = \lim_{s \to 0} sG(s)H(s)$$

Thus, the steady-state error for a ramp input is:

$$e_{ss} = \frac{A}{K_v}$$

Let the input be a parabolic signal: $r(t) = \frac{At^2}{2}u(t) \implies R(s) = \frac{A}{s^3}$.
The steady-state error is:

$$e_{ss} = \lim_{s\to 0} \frac{s\left(\frac{A}{s^3}\right)}{1 + G(s)H(s)} = \lim_{s\to 0} \frac{A}{s^2 + s^2 G(s)H(s)}$$
$$= \frac{A}{\lim_{s\to 0} s^2 G(s)H(s)}$$

Static Acceleration Error Constant (K_a) II

Acceleration Error Constant

The static acceleration error constant is defined as:

$$K_a = \lim_{s \to 0} s^2 G(s) H(s)$$

Thus, the steady-state error for a parabolic input is:

$$e_{ss} = \frac{A}{K_a}$$

System Type and Steady-State Error

The **Type** of a system refers to the number of poles at the origin ($s = 0$) in the open-loop transfer function $G(s)H(s)$. For example, $G(s)H(s) = \frac{K(s+z)}{s^N(s+p)}$ is a Type N system.

Table: Steady-State Error e_{ss} for Different System Types (for $A = 1$)

System Type	**Step Input** $\mathbf{R(s) = 1/s}$	**Ramp Input** $\mathbf{R(s) = 1/s^2}$	**Parabolic Input** $\mathbf{R(s) = 1/s^3}$
Type 0	$\frac{1}{1+K_p}$	∞	∞
Type 1	0	$\frac{1}{K_v}$	∞
Type 2	0	0	$\frac{1}{K_a}$

Example Problem

Problem: A unity feedback control system has an open-loop transfer function:

$$G(s) = \frac{20(s+2)}{s(s+1)(s+3)}$$

Determine the static error constants (K_p, K_v, K_a) and the steady-state error when the input is $r(t) = 3u(t) + 2tu(t)$.

Solution: Since $H(s) = 1$, the open-loop transfer function has one pole at the origin, so it is a **Type 1 system**.

1. **Position Error Constant:**

$$K_p = \lim_{s \to 0} G(s) = \lim_{s \to 0} \frac{20(s+2)}{s(s+1)(s+3)} = \infty$$

Example Problem (Continued)

2. **Velocity Error Constant:**

$$
\begin{aligned}
K_v &= \lim_{s\to 0} sG(s) \\
&= \lim_{s\to 0} s\left[\frac{20(s+2)}{s(s+1)(s+3)}\right] \\
&= \lim_{s\to 0} \frac{20(s+2)}{(s+1)(s+3)} = \frac{20(2)}{(1)(3)} = \frac{40}{3}
\end{aligned}
$$

3. **Acceleration Error Constant:**

$$
\begin{aligned}
K_a &= \lim_{s\to 0} s^2 G(s) \\
&= \lim_{s\to 0} s^2\left[\frac{20(s+2)}{s(s+1)(s+3)}\right] = 0
\end{aligned}
$$

Example Problem (Continued) I

The input is $r(t) = 3u(t) + 2tu(t)$. This is a superposition of a step input (magnitude $A_1 = 3$) and a ramp input (magnitude $A_2 = 2$). By the principle of superposition, the total steady-state error is:

$$\begin{aligned} e_{ss} &= e_{ss(\text{step})} + e_{ss(\text{ramp})} \\ &= \frac{A_1}{1 + K_p} + \frac{A_2}{K_v} \\ &= \frac{3}{1 + \infty} + \frac{2}{40/3} \\ &= 0 + \frac{6}{40} = 0.15 \end{aligned}$$

Interpretation

The system can track a step input perfectly with zero steady-state error. However, when tracking a moving target (ramp input), it always lags behind by a steady-state error of 0.15 units.

Stability of a Control System I

- Stability is the most critical property of any control system.
- A control system must be stable to be useful in practice; an unstable system is generally useless and potentially dangerous.
- In this lecture, we will explore the fundamental concepts of stability, define different types of stability, and learn how to determine a system's stability based on the location of its poles.

Bounded-Input Bounded-Output (BIBO) Stability I

BIBO Stability

A linear time-invariant (LTI) system is said to be **Bounded-Input Bounded-Output (BIBO) stable** if for every bounded input, the system produces a bounded output.

- An input signal $r(t)$ is **bounded** if there exists a positive finite real number M such that $|r(t)| \leq M < \infty$ for all $t \geq 0$.
- An output signal $c(t)$ is **bounded** if there exists a positive finite real number N such that $|c(t)| \leq N < \infty$ for all $t \geq 0$.
- If applying a finite, bounded signal (like a step input) results in an output that grows towards infinity, the system is **unstable**.

Concept of Stability: Physical Analogy I

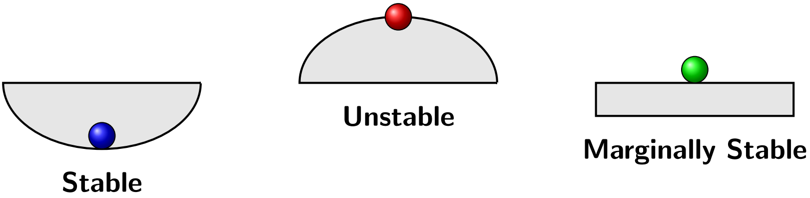


Figure: Physical analogy of stability using a ball on different surfaces.

- **Stable:** The system returns to its equilibrium state after a small temporary disturbance.
- **Unstable:** The system moves further away from equilibrium after a small disturbance.

Concept of Stability: Physical Analogy II

- **Marginally Stable:** The system stays in its new displaced state after a disturbance, neither returning to the original state nor diverging further.

Impulse Response and Stability I

BIBO Stability

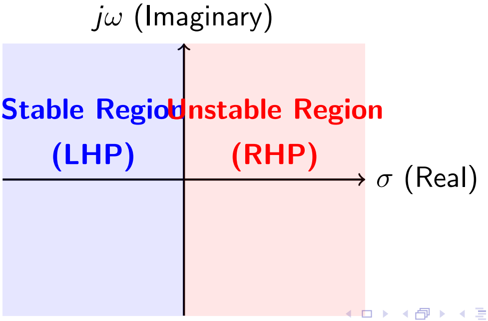
For an LTI system to be BIBO stable, its impulse response $h(t)$ must be absolutely integrable:

$$\int_0^\infty |h(t)|dt < \infty$$

- **Stable System:** The impulse response $h(t)$ decays to zero as $t \to \infty$. Example: $h(t) = e^{-2t}$.
- **Marginally Stable System:** The impulse response $h(t)$ does not decay to zero but remains bounded (e.g., constant or sustained oscillations) as $t \to \infty$. Example: $h(t) = \sin(\omega t)$.
- **Unstable System:** The impulse response $h(t)$ grows unbounded as $t \to \infty$. Example: $h(t) = e^{2t}$ or $t \sin(\omega t)$.

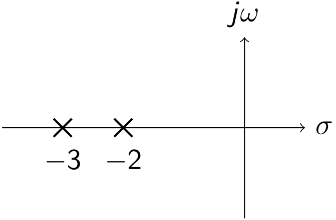
Location of Poles and Stability I

- The transfer function of a closed-loop LTI system is given by $T(s) = \frac{C(s)}{R(s)} = \frac{N(s)}{D(s)}$.
- The roots of the characteristic equation $D(s) = 0$ are the **poles** of the closed-loop system.
- The stability of a system is strictly dictated by the location of its poles in the complex s-plane ($s = \sigma + j\omega$).



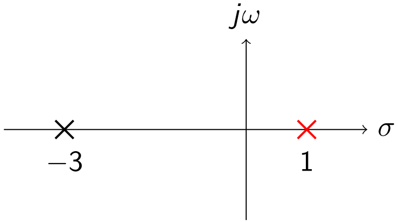
Stable Systems I

- A system is **stable** if all its poles lie strictly in the left half of the s-plane (LHP).
- This means every pole has a *negative real part* ($\sigma < 0$).
- **Example:** $T(s) = \frac{1}{(s+2)(s+3)}$. Poles are at $s_1 = -2, s_2 = -3$.
- The time response contains terms like Ae^{-2t} and Be^{-3t}, which decay to zero as $t \to \infty$.



Unstable Systems I

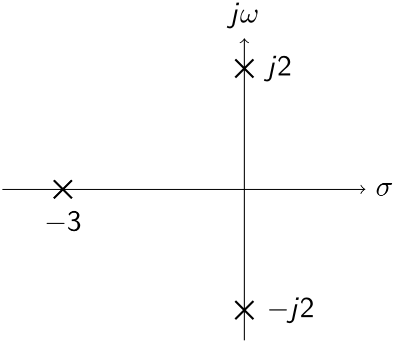
- A system is **unstable** if at least one pole lies in the right half of the *s*-plane (RHP).
- This means at least one pole has a *positive real part* ($\sigma > 0$).
- A system is also unstable if there are *multiple (repeated) poles at the same location on the imaginary axis*.
- **Example:** $T(s) = \frac{1}{(s-1)(s+3)}$. Poles are at $s_1 = 1, s_2 = -3$.
- The response contains an exponentially growing term e^{1t}.



Marginally Stable Systems I

- A system is **marginally stable** if there are no poles in the RHP, and there are only *simple (non-repeated) poles on the imaginary axis* ($j\omega$-axis).
- The response exhibits sustained, non-decaying oscillations.
- **Example:** $T(s) = \frac{10}{(s^2+4)(s+3)}$. Poles are at $s_1 = -3$ and $s_{2,3} = \pm j2$.
- The response will contain a decaying term and terms like $A\sin(2t + \phi)$.

Marginally Stable Systems II



Summary: Location of Poles & Stability I

Location of Poles	**Stability**	**Impulse Response** $(t \to \infty)$
All poles in LHP $(\sigma < 0)$	Stable	Decays to 0
At least one pole in RHP $(\sigma > 0)$	Unstable	Approaches ∞
Simple pole(s) on $j\omega$-axis $(\sigma = 0)$, all others in LHP	Marginally Stable	Constant or sustained oscillations
Multiple poles at the same location on $j\omega$-axis	Unstable	Approaches ∞

Necessary Conditions for Stability I

- Consider the characteristic equation of a system:

$$q(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0 = 0$$

- For a system to be stable (all roots in the LHP), there are two **necessary (but not sufficient)** conditions:

Necessary Conditions

1. **All coefficients must have the same sign:** $a_n, a_{n-1}, \ldots, a_0 > 0$.
2. **No missing terms:** All coefficients from s^n to s^0 must exist (i.e., non-zero).

- If either condition is violated, the system has at least one root in the RHP or on the imaginary axis, making it **unstable** or **marginally stable**.

Necessary Conditions for Stability II

- *Note:* If both conditions are satisfied, it does **not** guarantee stability. Further mathematical tests (e.g., Routh-Hurwitz criterion) are required.

Introduction to Stability

- Stability is arguably the most critical characteristic of any control system.
- An unstable system cannot be used for any meaningful control applications because its output will eventually grow out of bounds and may damage the system or its surroundings.
- In this lecture, we will formally define stability, understand the necessary conditions, and explore how the location of a system's poles dictates its overall stability.

Bounded-Input Bounded-Output (BIBO) Stability I

BIBO Stability

A linear time-invariant (LTI) system is strictly stable if, for any bounded input $r(t)$, the resulting output $c(t)$ is also bounded. This is known as **Bounded-Input Bounded-Output (BIBO) stability**.

- A signal $r(t)$ is considered **bounded** if there exists a positive real number M such that $|r(t)| \leq M < \infty$ for all time t.
- If the output diverges to infinity for a bounded input, the system is **unstable**.

- Mathematically, for a system with impulse response $h(t)$, the necessary and sufficient condition for BIBO stability is that the impulse response is absolutely integrable:

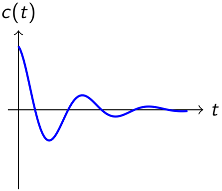
$$\int_{-\infty}^{\infty} |h(t)| dt < \infty$$

System Classification Based on Stability

We can classify control systems into three main categories based on their zero-input response (natural response) or their response to a bounded input:

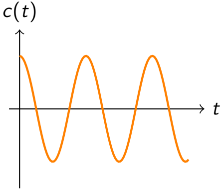
- **Stable System**: The natural response decays to zero as $t \to \infty$. Every bounded input produces a bounded output.
- **Unstable System**: The natural response grows without bound as $t \to \infty$. A bounded input can produce an unbounded output.
- **Marginally Stable (or Critically Stable) System**: The natural response neither decays nor grows, but remains constant or oscillates with a constant amplitude as $t \to \infty$.

Stable System



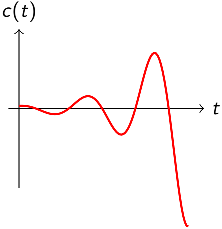
Response decays over time.

Marginally Stable



Response oscillates with constant amplitude.

Unstable System



Response grows without bound.

Location of Poles and Stability

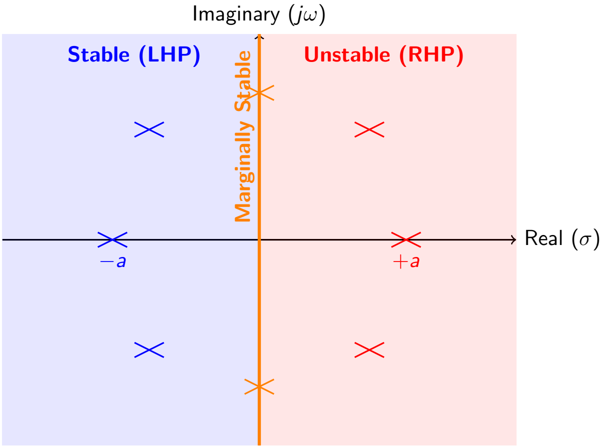
The stability of an LTI control system is entirely determined by the location of the roots of the characteristic equation, which are the **poles of the closed-loop transfer function**, in the complex s-plane ($s = \sigma + j\omega$).

Let the transfer function be $T(s) = \frac{C(s)}{R(s)} = \frac{N(s)}{D(s)}$. The characteristic equation is given by setting the denominator to zero: $D(s) = 0$.

Key Rule for Stability

For a system to be strictly stable, **all** poles of its transfer function must lie in the strict Left-Half of the s-Plane (LHP). This means every pole $s = \sigma + j\omega$ must have a strictly negative real part ($\sigma < 0$).

Mapping Pole Locations to the s-plane I



- **Poles in Left-Half Plane (LHP) ($\sigma < 0$):**
  - Real poles lead to an exponentially decaying response ($e^{-\sigma t}$).
  - Complex conjugate pairs lead to a decaying sinusoidal response ($e^{-\sigma t} \sin(\omega t)$).
  - The system is **stable**.

- **Poles on the Imaginary Axis ($j\omega$ axis, $\sigma = 0$):**
  - A single pole at the origin ($s = 0$) results in a constant step response.
  - A single pair of complex conjugate poles ($\pm j\omega$) results in sustained oscillations of constant amplitude.
  - The system is **marginally stable**.
  - **Crucial Exception:** If there are *repeated* roots on the $j\omega$ axis (e.g., $s^2 = 0$ or $(s^2 + \omega^2)^2 = 0$), the response grows as t, $t \sin(\omega t)$, making the system **unstable**.

- **Poles in Right-Half Plane (RHP) ($\sigma > 0$):**
  - Real poles lead to an exponentially growing response ($e^{\sigma t}$).
  - Complex conjugate pairs lead to an exponentially growing sinusoidal response ($e^{\sigma t} \sin(\omega t)$).
  - The presence of **even a single pole** in the RHP makes the entire system **unstable**.

Detailed Impact of Pole Locations (Cont.) II

Necessary Condition for Stability

Consider the characteristic equation polynomial:

$$a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0 = 0$$

A **necessary (but not sufficient)** condition for the system to be stable is that all coefficients a_i must have the **same sign** (usually positive) and **none of the coefficients can be zero**. If any coefficient is missing or negative, the system is automatically unstable or at best marginally stable.

Example: Assessing Stability from Transfer Functions

Consider the following systems:

System 1: $T_1(s) = \dfrac{10}{(s+2)(s+5)}$

- Poles are at $s = -2$ and $s = -5$.
- Both poles have negative real parts (LHP). **Result: Stable.**

System 2: $T_2(s) = \dfrac{5}{(s-1)(s+3)}$

- Poles are at $s = 1$ and $s = -3$.
- Pole $s = 1$ has a positive real part (RHP). **Result: Unstable.**

System 3: $T_3(s) = \dfrac{2}{s^2+4} = \dfrac{2}{(s+j2)(s-j2)}$

- Poles are at $s = \pm j2$.
- Simple roots on the imaginary axis. **Result: Marginally Stable.**

Introduction to Routh-Hurwitz Criterion

What is Routh Stability Criterion?

The Routh-Hurwitz stability criterion is an analytical method used to determine whether a linear time-invariant (LTI) system is stable, without actually solving for the roots of the characteristic equation.

- Developed independently by E.J. Routh (1877) and A. Hurwitz (1895).
- It determines the number of closed-loop poles that lie in the right half of the s-plane.
- For a system to be stable, **all** roots of its characteristic equation must lie in the left half of the s-plane (LHP).

Necessary Conditions for Stability

Let the characteristic equation of an LTI system be:

$$q(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0 = 0$$

where $a_n, a_{n-1}, \ldots, a_0$ are real coefficients.

Necessary (but not sufficient) Conditions

1. All the coefficients of the polynomial must have the **same sign** (all positive or all negative).
2. **None** of the coefficients can be **zero** (i.e., no missing powers of s).

If either condition is violated, the system is **unstable** (or marginally stable), and we do not need to construct the Routh array.

Construction of Routh Array I

If the necessary conditions are satisfied, we arrange the coefficients in a tabular form called the **Routh Array**:

$$
\begin{array}{c|cccc}
s^n & a_n & a_{n-2} & a_{n-4} & \dots \\
s^{n-1} & a_{n-1} & a_{n-3} & a_{n-5} & \dots \\
\hline
s^{n-2} & b_1 & b_2 & b_3 & \dots \\
s^{n-3} & c_1 & c_2 & c_3 & \dots \\
\vdots & \vdots & \vdots & \vdots & \\
s^0 & a_0 & 0 & 0 &
\end{array}
$$

Where:

$$
b_1 = \frac{a_{n-1}a_{n-2} - a_n a_{n-3}}{a_{n-1}}, \quad b_2 = \frac{a_{n-1}a_{n-4} - a_n a_{n-5}}{a_{n-1}}
$$

Construction of Routh Array II

$$c_1 = \frac{b_1 a_{n-3} - a_{n-1} b_2}{b_1}, \quad \ldots$$

Statement of Routh Stability Criterion

Routh Criterion

The necessary and sufficient condition for stability is that **all the elements in the first column of the Routh array must have the same sign**.

Key Insights:

- If there is any sign change in the first column, the system is **unstable**.
- The **number of sign changes** in the first column is exactly equal to the **number of roots in the right half of the** s**-plane** (RHP).

Example 1: Determination of Stability I

Problem: Determine the stability of the system with characteristic equation: $s^4 + 2s^3 + 3s^2 + 4s + 5 = 0$. **Solution:** Construct the Routh array.

$$
\begin{array}{c|ccc}
s^4 & 1 & 3 & 5 \\
s^3 & 2 & 4 & 0 \\
s^2 & \frac{(2)(3)-(1)(4)}{2} = 1 & \frac{(2)(5)-(1)(0)}{2} = 5 & 0 \\
s^1 & \frac{(1)(4)-(2)(5)}{1} = -6 & 0 & 0 \\
s^0 & 5 & 0 & 0
\end{array}
$$

Conclusion: There are **two sign changes** (from $+1$ to -6, and -6 to $+5$) in the first column. The system is **unstable** and has 2 roots in the RHP.

Difficulties in Formulation of Routh Table

While constructing the Routh table, we may encounter two special cases (difficulties) that prevent normal continuation:

Special Case 1

The first element of any row is **zero**, but at least one other element in that row is **non-zero**.

Special Case 2

All the elements of a row are **zero**.

Difficulty 1: First Element is Zero

If the first element of a row is zero, division by zero occurs when calculating the next row. **Remedy:**

- Replace the zero with a small positive number $\epsilon > 0$.
- Complete the Routh array in terms of ϵ.
- Take the limit as $\epsilon \to 0$ for the elements in the first column to determine the sign changes.

Alternative method: Substitute $s = 1/z$ into the characteristic equation and construct the Routh array for the new polynomial in z.

Example: Case 1 I

$q(s) = s^4 + s^3 + 2s^2 + 2s + 3 = 0$

$$
\begin{array}{c|ccc}
s^4 & 1 & 2 & 3 \\
s^3 & 1 & 2 & 0 \\
s^2 & 0 \approx \epsilon & 3 & 0 \\
s^1 & \frac{2\epsilon-3}{\epsilon} & 0 & 0 \\
s^0 & 3 & 0 & 0
\end{array}
$$

As $\epsilon \to 0^+$, the term $\frac{2\epsilon-3}{\epsilon} \to -\infty$.

- Signs in 1st column: $+, +, +, -, +$.
- Two sign changes $\implies$ Unstable (2 poles in RHP).

Difficulty 2: Entire Row is Zero

This occurs when the characteristic equation has roots that are symmetrically located with respect to the origin:

- A pair of purely imaginary roots $(\pm j\omega) \implies$ Oscillatory/Marginally stable.
- Real roots with opposite signs $(\pm\sigma)$.
- Complex conjugate roots forming a rectangle $(\pm\sigma \pm j\omega)$.

Remedy:

1. Form the **Auxiliary Polynomial**, $A(s)$, using the coefficients of the row just *above* the row of zeros.
2. Differentiate $A(s)$ with respect to s, i.e., $\frac{dA(s)}{ds}$.
3. Replace the row of zeros with the coefficients of $\frac{dA(s)}{ds}$.
4. Continue the Routh array.

$q(s) = s^5 + 2s^4 + 24s^3 + 48s^2 - 25s - 50 = 0$

s^5	1	24	-25
s^4	2	48	-50
s^3	0	0	0

Auxiliary equation from s^4 row: $A(s) = 2s^4 + 48s^2 - 50 = 0$.
Differentiating: $\frac{dA(s)}{ds} = 8s^3 + 96s$. Replace s^3 row with 8 and 96,

then continue:

s^3	8	96	0
s^2	24	-50	0
s^1	112.7	0	0
s^0	-50	0	0

Example: Case 2 (Row of Zeros) II

One sign change in the first column $\implies$ Unstable (1 pole in RHP).

In control systems, stability is a fundamental requirement. A linear time-invariant (LTI) system is stable if and only if all roots of the characteristic equation have negative real parts (i.e., they lie in the left half of the s-plane).

Characteristic Equation

For a closed-loop control system with open-loop transfer function $G(s)H(s)$, the characteristic equation is given by:

$$1 + G(s)H(s) = 0$$

which can be expressed as a polynomial in s:

$$a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0 = 0$$

Finding the exact roots of higher-order polynomials ($n \geq 3$) algebraically can be tedious. The **Routh-Hurwitz criterion** provides an analytical method to determine stability without solving for the exact roots.

Routh-Hurwitz Stability Criterion

Statement of Routh Stability Criterion

The Routh-Hurwitz criterion states that the number of roots of the characteristic equation with positive real parts is equal to the number of sign changes in the first column of the Routh array.

For a system to be strictly stable, there must be **no sign changes** in the first column of the Routh array, and all coefficients of the characteristic equation must be positive and non-zero.

Thus, the criterion serves two purposes:

- Determines absolute stability of the system.
- Finds the number of roots in the right half of the s-plane (unstable roots).

Necessary Conditions for Stability

Before constructing the Routh array, we can check the necessary (but not sufficient) conditions for stability.
Consider the characteristic equation:

$$a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0 = 0$$

- **Condition 1:** All coefficients of the polynomial must have the same sign (typically positive). If any coefficient has a different sign, the system is **unstable**.
- **Condition 2:** No coefficient can be zero (no missing terms in the polynomial). If a term is missing, it implies that there is a root with zero real part (on the imaginary axis) or a positive real part, making the system either marginally stable or unstable.

If these necessary conditions are satisfied, we proceed to formulate the **Routh array** to check the sufficient condition.

Formulation of the Routh Array I

Given the characteristic equation
$a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0 = 0$, the Routh table is constructed as follows:

$$
\begin{array}{c|cccc}
s^n & a_n & a_{n-2} & a_{n-4} & \dots \\
s^{n-1} & a_{n-1} & a_{n-3} & a_{n-5} & \dots \\
\hline
s^{n-2} & b_1 & b_2 & b_3 & \dots \\
s^{n-3} & c_1 & c_2 & c_3 & \dots \\
\vdots & \vdots & \vdots & \vdots & \ddots \\
s^0 & a_0 & & &
\end{array}
$$

Where the coefficients for the subsequent rows are computed using cross-multiplication:

$$b_1 = \frac{a_{n-1}a_{n-2} - a_n a_{n-3}}{a_{n-1}}, \quad b_2 = \frac{a_{n-1}a_{n-4} - a_n a_{n-5}}{a_{n-1}}$$

Formulation of the Routh Array II

$$c_1 = \frac{b_1 a_{n-3} - a_{n-1} b_2}{b_1}, \quad c_2 = \frac{b_1 a_{n-5} - a_{n-1} b_3}{b_1}$$

Example of Routh Array Formulation I

Example: Check the stability of the system with the characteristic equation:

$$s^4 + 2s^3 + 3s^2 + 4s + 5 = 0$$

Solution: All coefficients are present and positive. Let's form the Routh array:

$$\begin{array}{c|ccc}
s^4 & 1 & 3 & 5 \\
s^3 & 2 & 4 & 0 \\
\hline
s^2 & \frac{(2)(3)-(1)(4)}{2} = 1 & \frac{(2)(5)-(1)(0)}{2} = 5 & 0 \\
s^1 & \frac{(1)(4)-(2)(5)}{1} = -6 & 0 & 0 \\
s^0 & 5 & &
\end{array}$$

Example of Routh Array Formulation II

Looking at the first column: $1, 2, 1, -6, 5$.
There are **two sign changes** (from $+1$ to -6, and -6 to $+5$).
Therefore, the system is **unstable** and has two roots in the right half of the s-plane.

Difficulties in Routh Table Formulation

While constructing the Routh array, two special cases (difficulties) may occur that cause the normal procedure to break down:

- **Special Case 1:** The first element of a row becomes exactly zero, but the remaining elements in that row are non-zero.
  - *Consequence:* Division by zero occurs when calculating the elements of the next row.
- **Special Case 2:** All elements of a row become exactly zero.
  - *Consequence:* The array cannot be continued because the subsequent row calculation will result in indeterminate forms (0/0).
  - *Meaning:* This indicates the presence of roots that are symmetrically located about the origin (e.g., pairs of conjugate imaginary roots, or equal and opposite real roots).

Resolution of Special Case 1: Zero in the First Column

When only the first element of a row is zero, the normal Routh tabulation stops because of division by zero.

Remedy 1: Using a small parameter ϵ

- Replace the zero with a small positive number ϵ.
- Complete the rest of the Routh array in terms of ϵ.
- Take the limit as $\epsilon \to 0^+$ to determine the signs of the elements in the first column.

Remedy 2: Reverse polynomial method

- Substitute $s = 1/z$ in the original characteristic equation.
- Multiply by z^n to get a new polynomial in z.
- Construct the Routh array for the new polynomial. The number of RHP roots remains the same.

Example: Special Case 1 I

Consider: $s^5 + 2s^4 + 3s^3 + 6s^2 + 5s + 3 = 0$

$$
\begin{array}{c|ccc}
s^5 & 1 & 3 & 5 \\
s^4 & 2 & 6 & 3 \\
s^3 & 0 \to \epsilon & 3.5 & 0 \\
s^2 & \frac{6\epsilon - 7}{\epsilon} & 3 & 0 \\
s^1 & \frac{3.5(6\epsilon-7)/\epsilon - 3\epsilon}{(6\epsilon-7)/\epsilon} & 0 & 0 \\
s^0 & 3 & &
\end{array}
$$

Taking limit $\epsilon \to 0^+$:

- s^3 row: $\epsilon \to +$
- s^2 row: $\lim_{\epsilon \to 0} \frac{6\epsilon - 7}{\epsilon} = -\infty \implies -$
- s^1 row: $\lim_{\epsilon \to 0} \frac{21\epsilon - 24.5 - 3\epsilon^2}{6\epsilon - 7} = \frac{-24.5}{-7} = 3.5 \implies +$

First column signs: $+, +, +, -, +, +$. Two sign changes $\implies$ **Unstable** (2 RHP roots).

Resolution of Special Case 2: Entire Row is Zero I

When an entire row becomes zero, it implies symmetrically located roots (e.g., $\pm j\omega$).

Auxiliary Equation

An auxiliary polynomial $A(s)$ is formed using the coefficients of the row just **above** the row of zeros.

- The order of $A(s)$ is always even.
- The roots of $A(s)$ are also roots of the original characteristic equation.

Remedy for Special Case 2:

1. Form the auxiliary equation $A(s) = 0$ from the row above the zero row.
2. Differentiate $A(s)$ with respect to s: $\frac{dA(s)}{ds}$.

Resolution of Special Case 2: Entire Row is Zero II

3. Replace the row of zeros with the coefficients of the derivative $\frac{dA(s)}{ds}$.
4. Continue the Routh array as normal.

Example: Special Case 2 I

Consider: $s^5 + 2s^4 + 24s^3 + 48s^2 - 25s - 50 = 0$

$$\begin{array}{c|cccl} s^5 & 1 & 24 & & -25 \\ s^4 & 2 & 48 & & -50 \\ s^3 & 0 & 0 & 0 & \leftarrow \text{Row of zeros} \end{array}$$

Auxiliary equation from s^4 row: $A(s) = 2s^4 + 48s^2 - 50 = 0$.
Derivative: $\frac{dA(s)}{ds} = 8s^3 + 96s$.
Replace s^3 row with 8 and 96:

$$\begin{array}{c|ccc} s^5 & 1 & 24 & -25 \\ s^4 & 2 & 48 & -50 \\ s^3 & 8 & 96 & 0 \\ s^2 & 24 & -50 & 0 \\ s^1 & 112.67 & 0 & 0 \\ s^0 & -50 & & \end{array}$$

Example: Special Case 2 II

One sign change in the first column $\implies$ **Unstable** (1 RHP root).

Introduction to Root Locus

- The **Root Locus** technique was introduced by W. R. Evans in 1948.
- It is a graphical method for examining how the roots of a system's characteristic equation change with variation of a certain system parameter, commonly the loop gain K.
- The roots of the characteristic equation are the **closed-loop poles** of the system.
- Therefore, the root locus shows the trajectory of the closed-loop poles in the s-plane as the gain K is varied from 0 to ∞.

Why Root Locus?

It allows us to intuitively predict the absolute and relative stability, as well as the transient response of a closed-loop system, directly from its open-loop transfer function.

The Characteristic Equation

Consider a standard closed-loop control system with a forward path transfer function $G(s)$ and a feedback path transfer function $H(s)$. The closed-loop transfer function is given by:

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)}$$

Characteristic Equation

The characteristic equation of the closed-loop system is obtained by setting the denominator of the closed-loop transfer function to zero:

$$1 + G(s)H(s) = 0$$

The roots of this equation are the closed-loop poles. Since $G(s)H(s)$ typically contains a variable gain K, the roots depend on the value of K.

Formulation for Root Locus

Let the open-loop transfer function be expressed as:

$$G(s)H(s) = \frac{K \cdot N(s)}{D(s)}$$

where K is the variable parameter (gain), $N(s)$ is the numerator polynomial, and $D(s)$ is the denominator polynomial.
The characteristic equation becomes:

$$1 + \frac{K \cdot N(s)}{D(s)} = 0$$
$$\implies D(s) + K \cdot N(s) = 0$$

- When $K \to 0$, the roots of the characteristic equation approach the roots of $D(s) = 0$ (open-loop poles).
- When $K \to \infty$, the roots of the characteristic equation approach the roots of $N(s) = 0$ (open-loop zeros).

Angle and Magnitude Conditions

From the characteristic equation $1 + G(s)H(s) = 0$, we can write:

$$G(s)H(s) = -1 = -1 + j0$$

Since s is a complex variable ($s = \sigma + j\omega$), $G(s)H(s)$ is a complex quantity. The equation $G(s)H(s) = -1$ can be split into two fundamental conditions:

Fundamental Conditions

1. **Angle Condition**: For a point s to lie on the root locus, the angle of $G(s)H(s)$ evaluated at that point must be an odd multiple of 180°.
2. **Magnitude Condition**: For a point s on the root locus, the magnitude of $G(s)H(s)$ must be unity to determine the corresponding value of K.

1. The Angle Condition

The angle of $G(s)H(s)$ must satisfy:

$$\angle G(s)H(s) = \pm(2q+1)180^\circ \quad \text{where } q = 0, 1, 2, \ldots$$

If $G(s)H(s) = K\frac{\prod_{i=1}^{m}(s+z_i)}{\prod_{j=1}^{n}(s+p_j)}$, then the angle condition evaluated at a test point s is:

$$\sum_{i=1}^{m} \angle(s+z_i) - \sum_{j=1}^{n} \angle(s+p_j) = \pm(2q+1)180^\circ$$

- The angle condition is used to **verify** whether a given point s in the complex plane lies on the root locus.
- It is entirely independent of the value of K (assuming $K > 0$).

2. The Magnitude Condition

The magnitude of $G(s)H(s)$ must satisfy:

$$|G(s)H(s)| = 1$$

Using the pole-zero form:

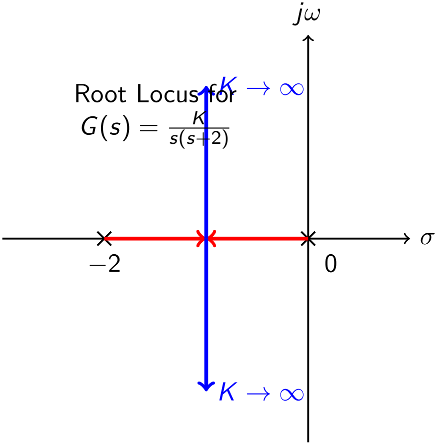
$$\frac{K \prod_{i=1}^{m} |s + z_i|}{\prod_{j=1}^{n} |s + p_j|} = 1$$

Solving for K:

$$K = \frac{\prod_{j=1}^{n} |s+p_j|}{\prod_{i=1}^{m} |s+z_i|} = \frac{\text{Product of vector lengths from open-loop poles to } s}{\text{Product of vector lengths from open-loop zeros to } s}$$

- The magnitude condition is used to **calculate** the value of K at a specific point on the root locus.

Visualizing the Root Locus I



- Starting points ($K = 0$): Open-loop poles at $s = 0, s = -2$.

Visualizing the Root Locus II

- Meeting point (breakaway): $s = -1$.
- As K increases, roots become complex conjugates, moving vertically, decreasing relative stability.

Summary of the Concept

- **System Design**: The root locus provides a graphical representation of the system's dynamic behavior. By observing the locus, an engineer can select a value of K that places the closed-loop poles at desired locations (e.g., for specific damping ratio or settling time).
- **Starting Points**: The root locus branches start at the open-loop poles ($K = 0$).
- **Ending Points**: The branches end at the open-loop zeros ($K = \infty$), which may include zeros at infinity.
- **Number of Branches**: The number of root locus branches equals the number of open-loop poles (assuming poles $\geq$ zeros).

Introduction to Root Locus

- The root locus technique was introduced by W. R. Evans in 1948.
- It is a graphical method for examining how the roots of a system change with variation of a certain system parameter, commonly a gain within a feedback system.
- For a closed-loop system, the transient response characteristics depend heavily on the location of the closed-loop poles.
- Thus, evaluating the movement of these poles when a parameter like the loop gain K varies from 0 to ∞ is extremely useful in control system design.

Definition of Root Locus

Root Locus

The root locus is defined as the path or locus traced out by the roots of the characteristic equation (i.e., closed-loop poles) in the s-plane as a specific system parameter (typically open-loop gain K) varies from zero to infinity.

Why do we need it?

- Determines absolute and relative stability.
- Helps in finding the range of K for which the system is stable, unstable, or marginally stable.
- Assists in designing controllers and compensators to achieve desired transient and steady-state performance.

Mathematical Background

Consider a standard negative feedback control system with forward path transfer function $G(s)$ and feedback path transfer function $H(s)$.

The closed-loop transfer function is:

$$T(s) = \frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)}$$

The characteristic equation is obtained by setting the denominator to zero:

$$1 + G(s)H(s) = 0$$

$$\implies G(s)H(s) = -1$$

Since s is a complex variable ($s = \sigma + j\omega$), the above equation yields two conditions:

1. **Magnitude Condition**
2. **Angle Condition**

Angle and Magnitude Conditions I

Angle Condition

For any point in the s-plane to lie on the root locus, the phase angle of the open-loop transfer function $G(s)H(s)$ at that point must be an odd multiple of 180°.

$$\angle G(s)H(s) = \pm(2q+1)180^\circ \quad \text{for } q = 0, 1, 2, \ldots$$

Angle and Magnitude Conditions II

Magnitude Condition

Once a point is verified to be on the root locus using the angle condition, the value of the parameter K at that point can be found using the magnitude condition.

$$|G(s)H(s)| = 1 \implies K = \frac{1}{|G'(s)H'(s)|}$$

where $G(s)H(s) = K \cdot G'(s)H'(s)$.

General Steps for Root Locus Construction

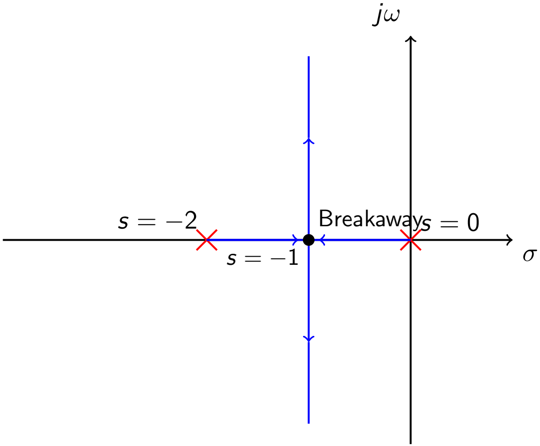
While plotting manually, several rules are derived from the angle condition:

1. **Starting and Ending Points:** Root locus starts at open-loop poles ($K = 0$) and ends at open-loop zeros or infinity ($K \to \infty$).
2. **Number of Branches:** Equal to the number of open-loop poles P (if $P \geq Z$).
3. **Symmetry:** The locus is always symmetrical with respect to the real axis.
4. **Real Axis Segments:** A point on the real axis lies on the root locus if the sum of total number of poles and zeros to its right on the real axis is odd.

Illustrative Example

Consider $G(s)H(s) = \frac{K}{s(s+2)}$.

- **Open-loop poles:** $s = 0$ and $s = -2$.
- **Open-loop zeros:** None.
- The root locus starts at 0 and -2.
- Locus on the real axis exists between 0 and -2 (odd number of poles to the right).
- The branches must leave the real axis and go towards infinity since there are no finite zeros. They meet halfway at -1 and break away.


$j\omega$
$s = -2$
Breakaway
$s = 0$
$s = -1$
σ

Introduction to Root Locus Rules I

- The **Root Locus** method, developed by W. R. Evans, provides a graphical way to determine the roots of the closed-loop characteristic equation as a system parameter (typically the gain K) varies from 0 to ∞.
- Instead of repeatedly solving the characteristic equation $1 + KG(s)H(s) = 0$ for different values of K, we can sketch the locus of the roots using a set of empirical rules.
- These rules are derived from the **angle condition** and **magnitude condition**.

Introduction to Root Locus Rules II

Fundamental Conditions

For a point s to lie on the root locus, it must satisfy:

- **Angle Condition**: $\angle G(s)H(s) = (2q+1)180^\circ$ (for $K > 0$)
- **Magnitude Condition**: $|G(s)H(s)| = \frac{1}{K}$

where $q = 0, \pm 1, \pm 2, \ldots$

Rule 1 & 2: Symmetry and Start/End Points I

Rule 1: Symmetry

The root locus is symmetric with respect to the real axis. This is because complex roots of polynomials with real coefficients always occur in conjugate pairs.

Rule 2: Starting and Ending Points

The root locus branches **start** (at $K = 0$) at the open-loop poles and **end** (at $K = \infty$) at the open-loop zeros (or at infinity).

- Characteristic equation:
  $1 + K\frac{N(s)}{D(s)} = 0 \implies D(s) + KN(s) = 0$
- When $K \to 0$, $D(s) = 0 \implies$ roots are open-loop poles.

Rule 1 & 2: Symmetry and Start/End Points II

- When $K \to \infty$, $1/K \to 0 \implies N(s) = 0 \implies$ roots are open-loop zeros.

Rule 3: Number of Branches

The number of branches of the root locus is equal to the number of closed-loop poles, which is equal to the number of open-loop poles P or open-loop zeros Z, whichever is greater.

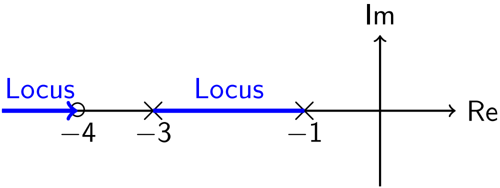
- Let $P =$ number of open-loop poles.
- Let $Z =$ number of open-loop zeros.
- The number of branches N is given by: $N = \max(P, Z)$.
- For physically realizable systems, $P \geq Z$, so the number of branches is typically equal to P.
- Out of P branches, Z branches end at the finite open-loop zeros, and the remaining $P - Z$ branches terminate at infinity.

Rule 4: Real Axis Segment

A point on the real axis lies on the root locus if the total number of open-loop poles and zeros to the **right** of that point is **odd**.

- This rule comes directly from the angle condition.
- Complex conjugate poles/zeros contribute a net angle of 360° (or 0°) to any point on the real axis, so they do not affect the odd/even count.
- Poles and zeros to the *left* of the point contribute 0°.
- Only poles and zeros to the *right* contribute 180° each. Thus, their sum must be odd for the total angle to be an odd multiple of 180°.

Rule 4: Root Locus on the Real Axis II



Rule 5: Asymptotes and Centroid I

- The $P - Z$ branches that go to infinity approach straight-line asymptotes.

Rule 5a: Angle of Asymptotes

The angles θ_A of the asymptotes with respect to the real axis are:

$$\theta_A = \frac{(2q+1)180^\circ}{P - Z}$$

for $q = 0, 1, 2, \ldots, (P - Z - 1)$.

Rule 5: Asymptotes and Centroid II

Rule 5b: Centroid

All asymptotes intersect on the real axis at a common point called the centroid (σ_A):

$$\sigma_A = \frac{\sum(\text{Real parts of Poles}) - \sum(\text{Real parts of Zeros})}{P - Z}$$

- Note: The centroid is the "center of gravity" of the pole-zero configuration.

Rule 6: Breakaway and Break-in Points I

- **Breakaway points**: Points where two (or more) locus branches converge and then separate (usually leaving the real axis).
- **Break-in points**: Points where branches coming from the complex plane enter the real axis.

Rule 6: Finding Break Points

Breakaway and break-in points occur at roots of the equation:

$$\frac{dK}{ds} = 0$$

where K is expressed as a function of s from the characteristic equation: $K = -\frac{D(s)}{N(s)}$

Rule 6: Breakaway and Break-in Points II

- **Validity Test**: Not all roots of $dK/ds = 0$ are valid break points. A root is valid only if it yields a positive, real value of K, or simply if it lies on a root locus segment determined by Rule 4.

Rule 7: $j\omega$-axis Crossing

The point where the root locus crosses the imaginary axis and the critical gain K_{cr} at that crossing can be found using the **Routh-Hurwitz criterion**.

1. Construct the Routh array for the closed-loop characteristic equation.
2. Find the value of K (called K_{cr} or K_{mar}) that makes an entire row of the Routh array zero (usually the s^1 row).
3. This marginal gain K_{cr} causes poles to lie exactly on the $j\omega$-axis.
4. Use the auxiliary equation from the row above (usually the s^2 row) to find the exact frequency ω of intersection: $As^2 + B = 0 \implies s = \pm j\omega$.

Rule 8: Angle of Departure and Arrival I

- Required when the open-loop transfer function has **complex** poles or zeros.

Rule 8a: Angle of Departure

The angle from which a branch departs from a complex pole is:

$$\theta_d = 180^\circ + \sum \phi_z - \sum \phi_p$$

where $\sum \phi_z$ is the sum of angles from all zeros to the complex pole, and $\sum \phi_p$ is the sum of angles from all other poles to the complex pole.

Rule 8: Angle of Departure and Arrival II

Rule 8b: Angle of Arrival

The angle at which a branch arrives at a complex zero is:

$$\theta_a = 180^\circ - \sum \phi_z + \sum \phi_p$$

Comprehensive Example

Problem: Sketch the root locus for a unity feedback system with:

$$G(s) = \frac{K}{s(s+2)(s+4)}$$

Solution Steps:

1. **Poles and Zeros:** $P = 3$ (poles at $s = 0, -2, -4$), $Z = 0$.
2. **Number of Branches:** $N = \max(3, 0) = 3$. All 3 branches end at ∞.
3. **Real Axis Segments:** By Rule 4, loci exist on the real axis between 0 and -2, and from -4 to $-\infty$.
4. **Asymptotes:**
   - Number of asymptotes $= P - Z = 3$.
   - Angles: $\theta_A = \frac{(2q+1)180^\circ}{3} \implies 60^\circ, 180^\circ, 300^\circ$.
   - Centroid: $\sigma_A = \frac{(0-2-4)-0}{3} = -2$.

Comprehensive Example (Cont.) I

5. **Breakaway Point:** Characteristic eq: $s^3 + 6s^2 + 8s + K = 0 \implies K = -(s^3 + 6s^2 + 8s)$.

$$\frac{dK}{ds} = -(3s^2 + 12s + 8) = 0$$

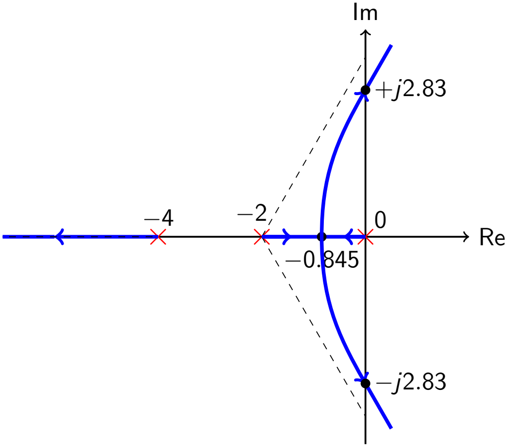
Roots: $s_1 = -0.845$, $s_2 = -3.155$. Since -3.155 does not lie on the real axis root locus segment, the valid breakaway point is $s = -0.845$.

6. **Imaginary Axis Crossing:** Characteristic eq: $s^3 + 6s^2 + 8s + K = 0$. Routh Array:

$$\begin{array}{c|cc} s^3 & 1 & 8 \\ s^2 & 6 & K \\ s^1 & \frac{48-K}{6} & 0 \\ s^0 & K & \end{array}$$

Marginal gain: $48 - K = 0 \implies K_{cr} = 48$.
Auxiliary eq: $6s^2 + 48 = 0 \implies s^2 = -8 \implies s = \pm j2.83$.

Sketch of the Root Locus I



Introduction to Root Locus Rules

- The root locus method, developed by W. R. Evans, provides a graphical method to determine the roots of the characteristic equation.
- The closed-loop transfer function is given by:

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)}$$

- The characteristic equation is:

$$1 + G(s)H(s) = 0 \implies G(s)H(s) = -1$$

- Let the open-loop transfer function be:

$$G(s)H(s) = \frac{K(s+z_1)(s+z_2)\dots(s+z_m)}{(s+p_1)(s+p_2)\dots(s+p_n)}$$

where K is the gain parameter, z_i are the open-loop zeros, and p_i are the open-loop poles.
- The root locus plots the path of closed-loop poles as K varies from 0 to ∞.

Rule 1: Number of Branches

The number of branches of the root locus is equal to the number of closed-loop poles, which is equal to the order of the characteristic equation.

- Let P be the number of open-loop poles and Z be the number of open-loop zeros.
- The number of branches $N = \max(P, Z)$.
- For physical systems, usually $P \geq Z$, so $N = P$.

Rule 1: Number of Branches & Rule 2: Symmetry II

Rule 2: Symmetry

The root locus is always symmetrical with respect to the real axis. This is because complex roots of a polynomial with real coefficients always occur in conjugate pairs.

Rule 3: Starting and Ending Points I

Starting Points ($K = 0$)

The root locus branches start (for $K = 0$) from the open-loop poles.

$$\lim_{K \to 0} [1 + G(s)H(s)] = 0 \implies \text{Roots are at } s = p_i$$

Ending Points ($K \to \infty$)

The root locus branches end (for $K \to \infty$) either at the open-loop zeros or at infinity.

$$\lim_{K \to \infty} \frac{1}{K} + \frac{(s + z_1)\ldots}{(s + p_1)\ldots} = 0 \implies \text{Roots are at } s = z_i \text{ or } \infty$$

Rule 3: Starting and Ending Points II

Note: If $P > Z$, Z branches end at the finite open-loop zeros, and the remaining $P - Z$ branches terminate at infinity.

Rule 4: Root Locus on the Real Axis I

Existence on the Real Axis

A point on the real axis lies on the root locus if the total number of open-loop poles and zeros on the real axis to the right of this point is odd.

- This rule is derived from the **angle condition**: $\angle G(s)H(s) = (2q+1)180^\circ$ for $q = 0, 1, 2, \ldots$
- Complex poles and zeros do not affect the root locus on the real axis because their net angle contribution to any point on the real axis is zero.

Example: Let open-loop poles be at $0, -2, -4$.

- Segment between 0 and -2 is on the root locus (1 pole to the right).

Rule 4: Root Locus on the Real Axis II

- Segment between -2 and -4 is not on the root locus (2 poles to the right).
- Segment from -4 to $-\infty$ is on the root locus (3 poles to the right).

Rule 5: Asymptotes of the Root Locus I

Branches tending to infinity do so along straight-line asymptotes. The number of asymptotes is equal to $P - Z$.

Angle of Asymptotes (ϕ_A)

The angles of the asymptotes with the real axis are given by:

$$\phi_A = \frac{(2q+1)180^\circ}{P - Z} \quad \text{for } q = 0, 1, 2, \ldots, (P - Z - 1)$$

Rule 5: Asymptotes of the Root Locus II

Centroid (σ_A)

The asymptotes intersect the real axis at a common point called the centroid, calculated as:

$$\sigma_A = \frac{\sum(\text{Re}(\text{poles})) - \sum(\text{Re}(\text{zeros}))}{P - Z}$$

Rule 6: Breakaway and Break-in Points I

Breakaway and break-in points correspond to multiple roots of the characteristic equation. They occur when two or more branches of the root locus meet and depart (break away) or arrive (break in).

Calculation

To find these points:

1. Express K in terms of s from the characteristic equation: $K = -\frac{1}{G(s)H(s)}$.
2. Differentiate K with respect to s and equate to zero: $\frac{dK}{ds} = 0$.
3. The roots of this equation give the potential breakaway/break-in points.

Rule 6: Breakaway and Break-in Points II

Validity Check: Substitute the roots back into the equation for K. Only roots that yield a positive, real value for K ($K > 0$) are valid breakaway or break-in points.

Rule 7: Angle of Departure and Arrival I

These angles are required when the open-loop transfer function has complex conjugate poles or zeros.

Angle of Departure (ϕ_d)

The angle at which a root locus branch departs from a complex open-loop pole is:

$$\phi_d = 180^\circ - (\Sigma\phi_P - \Sigma\phi_Z)$$

where $\Sigma\phi_P$ is the sum of angles contributed by other poles, and $\Sigma\phi_Z$ is the sum of angles contributed by all zeros to the complex pole in question.

Angle of Arrival (ϕ_a)

The angle at which a root locus branch arrives at a complex open-loop zero is:

$$\phi_a = 180^\circ + (\Sigma\phi_P - \Sigma\phi_Z)$$

Rule 8: Intersection with the Imaginary Axis I

The points where the root locus branches cross the imaginary axis represent the marginal stability conditions of the closed-loop system.

Rule 8: Intersection with the Imaginary Axis II

Finding the Intersection Points

The frequency of sustained oscillation (ω_n) and the corresponding value of K (marginal gain, K_{mar}) can be found by applying the **Routh-Hurwitz criterion**:

- Construct the Routh array from the characteristic equation.
- Set the terms in a row to zero to find the critical value of K for marginal stability.
- Form the auxiliary equation using the row directly above the zero row.
- Solve the auxiliary equation for $s = j\omega$ to find the intersection points on the imaginary axis.

Summary of Construction Steps

1. Locate open-loop poles and zeros on the s-plane.
2. Determine the root locus segments on the real axis (Rule 4).
3. Calculate the number of asymptotes and their angles, and locate the centroid (Rule 5).
4. Determine valid breakaway and break-in points (Rule 6).
5. If there are complex poles/zeros, calculate the angles of departure/arrival (Rule 7).
6. Find the points of intersection with the imaginary axis (Rule 8).
7. Sketch the complete root locus using the obtained information.

Introduction to Root Locus Examples

- In this lecture, we will apply the rules of root locus construction to simple systems.
- We will focus on systems up to **Type 1** and **Order 2**.
- **Type** refers to the number of poles at the origin ($s = 0$).
- **Order** refers to the highest power of s in the denominator polynomial of the open-loop transfer function (total number of open-loop poles).

Examples Covered

1. **Type 0, Order 1**: $G(s)H(s) = \frac{K}{s+a}$
2. **Type 0, Order 2**: $G(s)H(s) = \frac{K}{(s+p_1)(s+p_2)}$
3. **Type 1, Order 2**: $G(s)H(s) = \frac{K}{s(s+p_1)}$

Review of Key Root Locus Rules

Before diving into examples, let's briefly recall the essential rules for drawing the root locus:

1. **Starting and Ending Points**: Locus starts at open-loop poles ($K = 0$) and ends at open-loop zeros or infinity ($K \to \infty$).
2. **Real Axis Locus**: Exists at a point on the real axis if the sum of poles and zeros to its right is odd.
3. **Asymptotes**: Number of asymptotes $N_A = P - Z$. Angles: $\theta_A = \frac{(2k+1)180^\circ}{P-Z}$ for $k = 0, 1, \ldots, P - Z - 1$.
4. **Centroid**: $\sigma_A = \frac{\sum \text{Real parts of Poles} - \sum \text{Real parts of Zeros}}{P-Z}$
5. **Breakaway/Break-in Points**: Found by solving $\frac{dK}{ds} = 0$, where $K(s)$ is derived from the characteristic equation $1 + G(s)H(s) = 0$.

Example 1: Type 0, Order 1 System

Let's analyze a simple first-order system.

Open-Loop Transfer Function

$$G(s)H(s) = \frac{K}{s+2}$$

Step 1: Locate Poles and Zeros

- Number of Poles $(P) = 1$, at $s = -2$.
- Number of Zeros $(Z) = 0$.

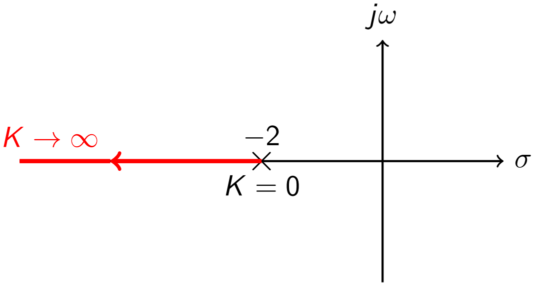
Step 2: Root Locus on the Real Axis

- Test points on the real axis.
- To the right of $s = -2$, there are 0 poles/zeros (even). No locus.
- To the left of $s = -2$, there is 1 pole (odd). The root locus exists for $s < -2$.

Example 1: Construction and Plot I

Step 3: Asymptotes and Ending Points

- Since $P - Z = 1 - 0 = 1$, one branch ends at infinity.
- Asymptote angle: $\theta_A = \frac{180^\circ}{1} = 180^\circ$.
- The single branch moves from $s = -2$ along the negative real axis towards $-\infty$ as K increases.



Example 2: Type 0, Order 2 System

Consider a system with two real poles and no zeros.

Open-Loop Transfer Function

$$G(s)H(s) = \frac{K}{(s+1)(s+3)}$$

Step 1: Locate Poles and Zeros

- Poles at $p_1 = -1$ and $p_2 = -3$. $(P = 2)$
- No finite zeros. $(Z = 0)$

Step 2: Real Axis Locus

- Between 0 and -1: 0 poles to the right (even) $\implies$ No locus.
- Between -1 and -3: 1 pole to the right (odd) $\implies$ Locus exists!
- Beyond -3: 2 poles to the right (even) $\implies$ No locus.

Example 2: Asymptotes and Breakaway Point

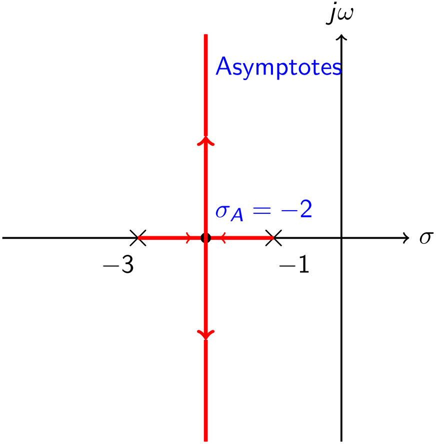
Step 3: Asymptotes

- Number of asymptotes $= P - Z = 2$.
- Angles: $\theta_A = \frac{180^\circ \pm k360^\circ}{2} = 90^\circ, 270^\circ$.
- Centroid: $\sigma_A = \frac{(-1)+(-3)-0}{2} = \frac{-4}{2} = -2$.

Step 4: Breakaway Point

- Characteristic equation:
  $1 + \frac{K}{(s+1)(s+3)} = 0 \implies K = -(s^2 + 4s + 3)$
- Differentiate with respect to s: $\frac{dK}{ds} = -(2s + 4) = 0$
- Solving gives: $s = -2$.
- Since $s = -2$ lies on the real axis locus segment between -1 and -3, it is a valid breakaway point.

Example 2: Root Locus Plot I



Example 3: Type 1, Order 2 System

Now consider a system with a pole at the origin (Type 1) and another real pole.

Open-Loop Transfer Function

$$G(s)H(s) = \frac{K}{s(s+4)}$$

Step 1: Locate Poles and Zeros

- Poles at $p_1 = 0$ and $p_2 = -4$. $(P = 2)$
- No finite zeros. $(Z = 0)$

Step 2: Real Axis Locus

- Between 0 and -4: 1 pole to the right $(s = 0) \implies$ Locus exists!
- The two branches move towards each other from $s = 0$ and $s = -4$.

Example 3: Asymptotes and Breakaway Point

Step 3: Asymptotes

- Number of asymptotes $= P - Z = 2$.
- Angles: $\theta_A = 90^\circ, 270^\circ$.
- Centroid: $\sigma_A = \frac{0+(-4)-0}{2} = -2$.

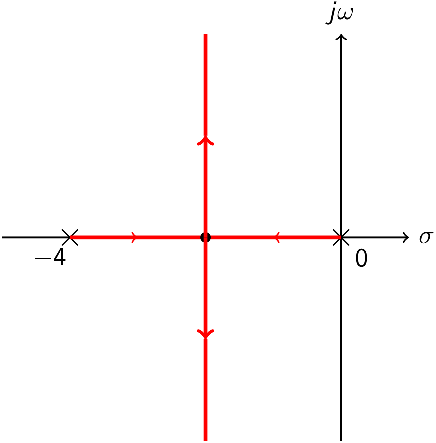
Step 4: Breakaway Point

- Characteristic equation: $1 + \frac{K}{s^2+4s} = 0 \implies K = -(s^2 + 4s)$
- $\frac{dK}{ds} = -(2s + 4) = 0 \implies s = -2$.
- Valid breakaway point as it lies between 0 and -4.

Step 5: Intersection with Imaginary Axis

- Characteristic equation: $s^2 + 4s + K = 0$.
- Substitute $s = j\omega$:
  $-\omega^2 + 4j\omega + K = 0 \implies (K - \omega^2) + j(4\omega) = 0$.
- Equating imaginary part to zero: $4\omega = 0 \implies \omega = 0$.
- At $\omega = 0$, $K = 0$. Branches don't cross the $j\omega$-axis for $K > 0$.

Example 3: Root Locus Plot I



Summary of Simple Systems

Type/Order	Open-Loop T.F.	Breakaway	Asymptotes
Type 0, Order 1	$G(s)H(s) = \frac{K}{s+a}$	None	1 (Angle: 180°)
Type 0, Order 2	$G(s)H(s) = \frac{K}{(s+a)(s+b)}$	$\frac{-(a+b)}{2}$	2 (Angles: $90^\circ, 270^\circ$)
Type 1, Order 2	$G(s)H(s) = \frac{K}{s(s+a)}$	$\frac{-a}{2}$	2 (Angles: $90^\circ, 270^\circ$)

Key Observation

For any second-order system with two distinct real poles and no zeros, the root locus always breaks away exactly halfway between the two poles and follows vertical asymptotes.

What is Frequency Response Analysis?

Frequency Response

The **frequency response** of a system is the steady-state response of a system to a sinusoidal input signal. It describes how a system reacts to inputs of different frequencies.

- In **time-domain analysis**, we study the system's transient and steady-state response to specific test signals like step, impulse, or ramp inputs.
- In **frequency-domain analysis**, we analyze the system's steady-state response to sinusoidal inputs across a continuous range of frequencies (ω).
- Frequency response is a powerful and highly practical tool for the design and analysis of Linear Time-Invariant (LTI) control systems.

Core Concept: Sinusoidal Response I

LTI System Property

When a stable **Linear Time-Invariant (LTI)** system is subjected to a sinusoidal input, the steady-state output is also a sinusoidal wave of the **same frequency**, but with a different **amplitude** and a **phase shift**.

Suppose the input to a system with transfer function $G(s)$ is:

$$r(t) = A\sin(\omega t)$$

The steady-state output $c(t)$ will be:

$$c(t) = B\sin(\omega t + \phi)$$

Where:

Core Concept: Sinusoidal Response II

- A is the input amplitude and B is the new output amplitude.
- ω is the angular frequency (remains unchanged!).
- ϕ is the phase shift (can be positive for phase lead or negative for phase lag).

To find the frequency response from a transfer function $G(s)$, we substitute $s = j\omega$ (since we are evaluating the steady-state sinusoidal response, the real part $\sigma = 0$ in the s-plane).

$$G(j\omega) = G(s)|_{s=j\omega}$$

$G(j\omega)$ is a complex number and can be expressed in polar form:

$$G(j\omega) = |G(j\omega)|e^{j\angle G(j\omega)}$$

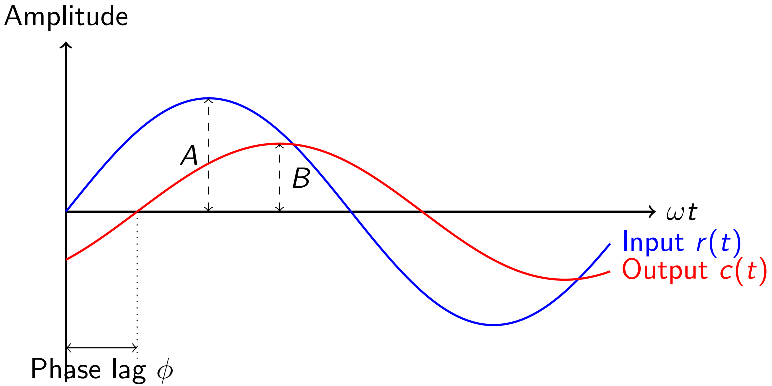
Magnitude (Amplitude Ratio):

$$M = |G(j\omega)| = \frac{B}{A}$$

Phase Shift:

$$\phi = \angle G(j\omega)$$

Graphical Representation: Input vs Output



Advantages of Frequency Response Analysis I

- **Experimental Determination:** Transfer functions of complex, physical components can be determined experimentally using frequency response tests (sine wave generators and oscilloscopes), even if their mathematical models are unknown or highly non-linear.
- **Design and Compensation:** It provides a straightforward method for designing control systems to meet specific performance criteria using compensators (Lead, Lag, Lead-Lag).
- **Stability Analysis:** We can determine both absolute and relative stability (Gain Margin and Phase Margin) of closed-loop systems from the open-loop frequency response without calculating the exact roots of the characteristic equation.

Advantages of Frequency Response Analysis II

- **Noise Filtering:** It gives clear insight into how the system will respond to high-frequency noise versus low-frequency reference signals.

Graphical Tools for Frequency Response

Frequency response characteristics are typically represented using three main graphical methods:

1. **Bode Plots:** Plots of magnitude (in decibels) and phase angle (in degrees) versus frequency (on a logarithmic scale).
2. **Polar Plots (Nyquist Plots):** A plot of the magnitude versus phase angle in polar coordinates as frequency varies from zero to infinity.
3. **Log-Magnitude vs Phase Plots (Nichols Charts):** A plot of logarithmic magnitude versus phase angle on rectangular coordinates.

Note: We will explore these tools in detail in the upcoming lectures.

Introduction to Frequency Response I

- The **frequency response** of a system is defined as the steady-state response of the system to a sinusoidal input signal.
- For a linear time-invariant (LTI) system with transfer function $G(s)$, if the input is $u(t) = A\sin(\omega t)$, the steady-state output is $y_{ss}(t) = B\sin(\omega t + \phi)$.
- The amplitude ratio B/A and the phase shift ϕ are functions of the input frequency ω.
- Specifically, $B/A = |G(j\omega)|$ and $\phi = \angle G(j\omega)$.

Bode Plot

A **Bode plot** (named after Hendrik W. Bode) is a graphical representation of the frequency response of a system. It consists of two separate plots:

1. **Magnitude Plot:** A plot of the magnitude of $G(j\omega)$ in decibels (dB) versus frequency ω on a logarithmic scale.
2. **Phase Plot:** A plot of the phase angle of $G(j\omega)$ in degrees versus frequency ω on a logarithmic scale.

Why use Logarithmic Scales and Decibels? I

- Frequency ranges in control systems can span several decades (e.g., from 0.1 rad/s to 1000 rad/s). A logarithmic scale allows a wide range of frequencies to be plotted on a single graph without compressing the lower frequencies.
- **Decibel (dB):** The magnitude $|G(j\omega)|$ is converted to decibels using the formula:

$$\text{Magnitude in dB} = 20\log_{10}|G(j\omega)|$$

Why use Logarithmic Scales and Decibels? II

Advantage of dB Scale

When a transfer function consists of multiple factors (e.g., $G(s) = \frac{G_1(s)G_2(s)}{G_3(s)}$), the overall magnitude in dB is simply the algebraic sum of the individual magnitudes in dB:

$$20\log_{10}|G| = 20\log_{10}|G_1| + 20\log_{10}|G_2| - 20\log_{10}|G_3|$$

Similarly, phase angles add linearly: $\angle G = \angle G_1 + \angle G_2 - \angle G_3$. This makes plotting significantly easier using asymptotic approximations.

Advantages of Bode Plots

- **Addition over Multiplication:** The use of logarithmic scales converts the multiplication of magnitudes into addition, simplifying the process of sketching the frequency response of complex systems.
- **Asymptotic Approximations:** The magnitude plot can often be approximated by straight-line asymptotes, which are very easy to draw by hand.
- **Wide Frequency Range:** Both low and high-frequency behaviors of a system can be observed on the same plot.
- **Stability Analysis:** Relative stability parameters like *Gain Margin (GM)* and *Phase Margin (PM)* can be easily determined directly from the Bode plot without calculating the roots of the characteristic equation.

Standard Factors in a Transfer Function

The general open-loop transfer function $G(j\omega)H(j\omega)$ can be factored into a combination of basic components. The Bode plot is drawn by adding the individual plots of these factors.

The basic factors are:

1. **Constant gain:** K
2. **Integral and derivative factors (Poles and Zeros at origin):** $(j\omega)^{\pm n}$
3. **First-order factors (Simple poles and zeros):** $(1 + j\omega T)^{\pm 1}$
4. **Quadratic factors (Complex conjugate poles and zeros):** $\left[1 + 2\zeta(j\omega/\omega_n) + (j\omega/\omega_n)^2\right]^{\pm 1}$

Bode Plot for a First-Order System (Simple Pole)

Consider a simple pole factor: $G(s) = \frac{1}{1+sT}$. Substituting $s = j\omega$:

$$G(j\omega) = \frac{1}{1 + j\omega T}$$

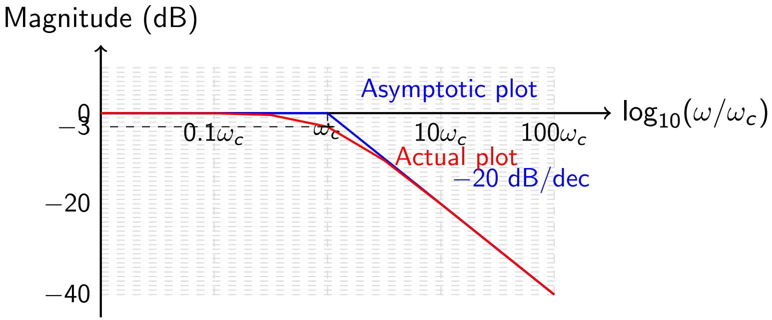
Magnitude: $|G(j\omega)|_{\text{dB}} = -20\log_{10}\sqrt{1 + (\omega T)^2}$

- For low frequencies ($\omega \ll 1/T$):
  $\omega T \to 0 \implies |G(j\omega)|_{\text{dB}} \approx -20\log_{10}(1) = 0$ dB.
  (Low-frequency asymptote is a horizontal line at 0 dB).
- For high frequencies ($\omega \gg 1/T$):
  $\omega T \gg 1 \implies |G(j\omega)|_{\text{dB}} \approx -20\log_{10}(\omega T)$. (High-frequency asymptote is a straight line with a slope of -20 dB/decade).
- The intersection of these asymptotes is at $\omega = 1/T$, known as the **corner frequency** or **break frequency** (ω_c).

Phase: $\phi(\omega) = -\tan^{-1}(\omega T)$.

- At $\omega = 0, \phi = 0°$. At $\omega = 1/T, \phi = -45°$. At $\omega \to \infty, \phi = -90°$.

Visualizing Asymptotic Bode Plot of a Simple Pole



- The maximum error between the asymptotic and actual plot occurs at the corner frequency $\omega_c = 1/T$ and is exactly -3 dB.

Stability Analysis via Bode Plot I

Bode plots are highly useful for determining the relative stability of a closed-loop system from its open-loop frequency response.

- **Gain Crossover Frequency (ω_{gc}):** The frequency at which the magnitude is 0 dB (i.e., $|G(j\omega)H(j\omega)| = 1$).
- **Phase Crossover Frequency (ω_{pc}):** The frequency at which the phase angle is -180°.

Stability Analysis via Bode Plot II

Gain Margin (GM) and Phase Margin (PM)

- **Gain Margin (GM):** The amount of gain in dB that can be added to the system before it becomes unstable. Evaluated at ω_{pc}.

$$GM = 0 \text{ dB} - |G(j\omega_{pc})H(j\omega_{pc})|_{\text{dB}}$$

- **Phase Margin (PM):** The additional phase lag required to make the system unstable. Evaluated at ω_{gc}.

$$PM = 180^\circ + \angle G(j\omega_{gc})H(j\omega_{gc})$$

For a stable minimum phase system, both GM and PM must be positive.

Introduction to Logarithmic Scales I

- In frequency response analysis (like Bode plots), we analyze systems over a very wide range of frequencies (e.g., from 0.1 rad/s to $10,000$ rad/s).
- Using a linear scale for frequency would compress low-frequency details into a tiny space while stretching out high-frequency regions.
- **Solution:** We use a **logarithmic scale** for the frequency axis (ω).

Introduction to Logarithmic Scales II

Advantages of Logarithmic Frequency Scale

- Allows plotting a wide range of frequencies on a single graph.
- Simplifies the drawing of frequency response plots (Bode plots become straight-line asymptotes).
- Relative frequency changes (like doubling or multiplying by 10) represent equal distances on the log scale.

Decades and Octaves I

In logarithmic frequency scales, specific terms are used to describe frequency intervals:

Decade

A **decade** is an interval between two frequencies with a ratio of 10.

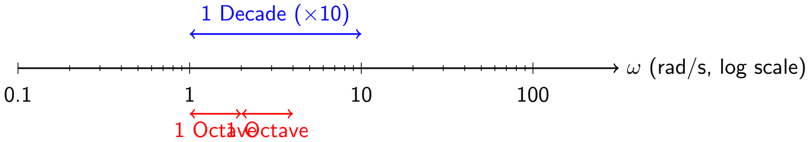
- Example: 1 rad/s to 10 rad/s is one decade.
- Mathematically: $\omega_2 = 10\omega_1 \implies \log_{10}(\omega_2/\omega_1) = 1$

Octave

An **octave** is an interval between two frequencies with a ratio of 2.

- Example: 2 rad/s to 4 rad/s is one octave.
- Mathematically: $\omega_2 = 2\omega_1 \implies \log_2(\omega_2/\omega_1) = 1$

Logarithmic Frequency Scale Visualization



- The physical distance between 1 and 10 is the same as between 10 and 100.
- The distance between 1 and 2 (one octave) is the same as the distance between 2 and 4.

The Decibel (dB) Scale for Magnitude

- For the magnitude of the transfer function $|G(j\omega)|$, we use a logarithmic unit called the **decibel (dB)**.
- By definition, the magnitude in dB is given by:

$$|G(j\omega)|_{\text{dB}} = 20 \log_{10} |G(j\omega)|$$

- This converts multiplicative gains into additive terms.
- For a cascaded system $G(s) = G_1(s)G_2(s)$:

$$\begin{aligned} |G(j\omega)| &= |G_1(j\omega)| \times |G_2(j\omega)| \\ 20 \log_{10} |G(j\omega)| &= 20 \log_{10} |G_1(j\omega)| + 20 \log_{10} |G_2(j\omega)| \\ |G(j\omega)|_{\text{dB}} &= |G_1(j\omega)|_{\text{dB}} + |G_2(j\omega)|_{\text{dB}} \end{aligned}$$

To plot the frequency response using asymptotic approximations (Bode plots), the open-loop transfer function $G(s)H(s)$ must first be written in **time-constant form** (or standard form).
Consider a general transfer function factored into poles and zeros:

$$G(s)H(s) = \frac{K(s+z_1)(s+z_2)}{s^N(s+p_1)(s+p_2)(s^2+2\zeta\omega_n s+\omega_n^2)}$$

We factor out the constants $z_1, z_2, p_1, p_2, \omega_n^2$ to get terms of the form $(1+s\tau)$:

$$G(s)H(s) = \frac{K \cdot z_1 z_2 \left(1+\frac{s}{z_1}\right)\left(1+\frac{s}{z_2}\right)}{s^N \cdot p_1 p_2 \omega_n^2 \left(1+\frac{s}{p_1}\right)\left(1+\frac{s}{p_2}\right)\left(1+2\zeta\left(\frac{s}{\omega_n}\right)+\left(\frac{s}{\omega_n}\right)^2\right)}$$

Standard Form of $G(j\omega)H(j\omega)$

Let the new constant be $K_B = \frac{Kz_1z_2}{p_1p_2\omega_n^2}$. This is referred to as the **Bode gain**.

Substituting $s = j\omega$, the standard form becomes:

Standard Time-Constant Form for Frequency Response

$$G(j\omega)H(j\omega) = \frac{K_B(1+j\omega\tau_{z1})(1+j\omega\tau_{z2})}{(j\omega)^N(1+j\omega\tau_{p1})(1+j\omega\tau_{p2})\left[1-\left(\frac{\omega}{\omega_n}\right)^2+j2\zeta\left(\frac{\omega}{\omega_n}\right)\right]}$$

where:

- $\tau_{z1} = 1/z_1$, $\tau_{z2} = 1/z_2$ are zero time constants.
- $\tau_{p1} = 1/p_1$, $\tau_{p2} = 1/p_2$ are pole time constants.

Basic Factors in the Standard Form

Any linear time-invariant system's transfer function in standard form is composed of four basic types of factors. The total Bode plot is the superposition of the plots of these individual factors:

1. **Constant gain:** K_B
2. **Roots at the origin:** $(j\omega)^{\pm N}$ (Poles or zeros at the origin)
3. **Simple roots:** $(1 + j\omega\tau)^{\pm 1}$ (First-order poles or zeros on the real axis)
4. **Quadratic roots:** $\left[1 - \left(\frac{\omega}{\omega_n}\right)^2 + j2\zeta\left(\frac{\omega}{\omega_n}\right)\right]^{\pm 1}$ (Complex conjugate poles or zeros)

For each factor, we will separately determine its magnitude (in dB) and its phase angle, and then add them together to obtain the overall frequency response.

Superposition of Factors

For a system composed of various factors $G(j\omega) = \frac{K_B \cdot \text{Zero Factors}}{\text{Pole Factors}}$, the magnitude and phase are calculated as:

Logarithmic Magnitude (in dB):

$$\begin{aligned}
20\log_{10}|G(j\omega)| = {} & 20\log_{10}|K_B| \\
& + \sum \left(20\log_{10}|\text{Zero Factors}|\right) \\
& - \sum \left(20\log_{10}|\text{Pole Factors}|\right)
\end{aligned}$$

Phase Angle:

$$\begin{aligned}
\angle G(j\omega) = {} & \angle K_B \\
& + \sum (\angle \text{Zero Factors}) \\
& - \sum (\angle \text{Pole Factors})
\end{aligned}$$

Transfer Function in Bode Form

A general open-loop transfer function $G(s)H(s)$ can be expressed in terms of basic standard factors by substituting $s = j\omega$:

$$G(j\omega)H(j\omega) = \frac{K(1 + j\omega T_1)(1 + j\omega T_2)\cdots}{(j\omega)^N(1 + j\omega T_a)\left[1 + 2\zeta\left(\frac{j\omega}{\omega_n}\right) + \left(\frac{j\omega}{\omega_n}\right)^2\right]\cdots}$$

The four basic standard factors in frequency response analysis are:

- **Constant gain:** K
- **Integral and derivative factors:** $(j\omega)^{\pm N}$ (poles and zeros at the origin)
- **First-order factors:** $(1 + j\omega T)^{\pm 1}$ (simple poles and zeros)

Standard Factors of a Transfer Function II

- **Quadratic factors:** $\left[1 + 2\zeta\left(\frac{j\omega}{\omega_n}\right) + \left(\frac{j\omega}{\omega_n}\right)^2\right]^{\pm 1}$ (complex conjugate poles and zeros)

Gain Factor Characteristics

The constant gain factor is K. Its magnitude in decibels is given by:

$$M_{\text{dB}} = 20\log_{10} K$$

The phase angle is:

$$\phi = \angle K = \begin{cases} 0^\circ & \text{if } K > 0 \\ -180^\circ & \text{if } K < 0 \end{cases}$$

Impact on the Bode Plot:

- The **magnitude plot** is a horizontal straight line at $20\log_{10} K$ dB across all frequencies.

Constant Gain K II

- The **phase plot** is a horizontal straight line at 0° (assuming positive K).
- Changing the value of K shifts the entire magnitude plot vertically up or down, but it does *not* affect the slope or shape of the magnitude curve, nor does it affect the phase plot.

Integral and Derivative Factors $(j\omega)^{\pm 1}$

Consider a **pole at the origin** (integral factor):
$G(j\omega) = \frac{1}{j\omega} = (j\omega)^{-1}$

- **Magnitude:** $M = \left|\frac{1}{j\omega}\right| = \frac{1}{\omega}$
- **Magnitude (dB):** $M_{\text{dB}} = 20\log_{10}\left(\frac{1}{\omega}\right) = -20\log_{10}\omega$
- **Phase:** $\phi = \angle\left(\frac{1}{j\omega}\right) = -90^\circ$

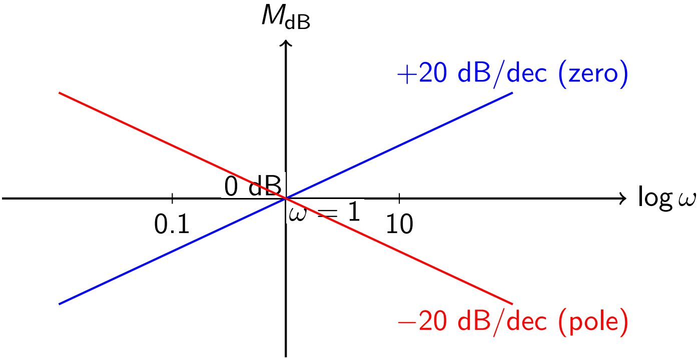
Consider a **zero at the origin** (derivative factor): $G(j\omega) = j\omega$

- **Magnitude (dB):** $M_{\text{dB}} = 20\log_{10}\omega$
- **Phase:** $\phi = \angle(j\omega) = +90^\circ$

Bode Plot: $(j\omega)^{\pm 1}$ I

- For $\frac{1}{j\omega}$ (pole at origin), the magnitude plot is a straight line passing through 0 dB at $\omega = 1$ rad/sec with a slope of **-20 dB/decade**. The phase is constantly -90°.
- For $j\omega$ (zero at origin), the magnitude plot passes through 0 dB at $\omega = 1$ rad/sec with a slope of **+20 dB/decade**. The phase is constantly $+90^\circ$.
- Generalizing for $(j\omega)^{\pm N}$, the magnitude slope is $\pm 20N$ dB/decade, and the constant phase is $\pm 90N^\circ$.

Bode Plot: $(j\omega)^{\pm 1}$ II



First-Order Factors $(1 + j\omega T)^{\pm 1}$

Consider a **simple pole**: $G(j\omega) = \frac{1}{1+j\omega T}$

- **Magnitude (dB):**
  $M_{\text{dB}} = -20 \log_{10} |1 + j\omega T| = -20 \log_{10} \sqrt{1 + (\omega T)^2}$
- **Phase:** $\phi = -\tan^{-1}(\omega T)$

Corner Frequency

The frequency at which the real and imaginary parts of the factor are equal, i.e., $\omega T = 1$ or $\omega = \frac{1}{T}$, is called the **corner frequency** (ω_c).
It divides the frequency response into two distinct asymptotic regions: low frequencies $(\omega \ll \omega_c)$ and high frequencies $(\omega \gg \omega_c)$.

Asymptotic Approximations for $\frac{1}{1+j\omega T}$

1. **Low-frequency asymptote ($\omega \ll \frac{1}{T}$):**
   - $\omega T \ll 1 \implies \sqrt{1+(\omega T)^2} \approx 1$
   - $M_{\text{dB}} \approx -20\log_{10} 1 = 0$ dB
   - The low-frequency asymptote is a horizontal line at 0 dB.
2. **High-frequency asymptote ($\omega \gg \frac{1}{T}$):**
   - $\omega T \gg 1 \implies \sqrt{1+(\omega T)^2} \approx \omega T$
   - $M_{\text{dB}} \approx -20\log_{10}(\omega T)$
   - This asymptote is a straight line with a slope of **-20 dB/decade**, which intersects the 0 dB line exactly at the corner frequency $\omega = \frac{1}{T}$.

Magnitude Error and Phase Characteristics

Exact Magnitude at the Corner Frequency: At $\omega = \frac{1}{T}$, the asymptotic approximation gives 0 dB. However, the exact magnitude is:

$$M_{\text{dB}} = -20\log_{10}\sqrt{1+1^2} = -20\log_{10}\sqrt{2} \approx -3 \text{ dB}$$

The maximum error between the exact curve and the asymptotes occurs at the corner frequency and is 3 dB.

Phase Angle Transitions:

- At $\omega = 0.1/T$: $\phi = -\tan^{-1}(0.1) \approx -5.7^\circ \approx 0^\circ$
- At $\omega = 1/T$: $\phi = -\tan^{-1}(1) = -45^\circ$
- At $\omega = 10/T$: $\phi = -\tan^{-1}(10) \approx -84.3^\circ \approx -90^\circ$

Note: For a simple zero $(1 + j\omega T)$, the high-frequency magnitude slope is $+20$ dB/dec, and the phase transitions from 0° to $+90^\circ$.

Quadratic Factors

Consider a **complex conjugate pole pair**:

$$G(j\omega) = \frac{1}{1 + 2\zeta\left(\frac{j\omega}{\omega_n}\right) + \left(\frac{j\omega}{\omega_n}\right)^2} = \frac{1}{\left(1 - \left(\frac{\omega}{\omega_n}\right)^2\right) + j\left(2\zeta\frac{\omega}{\omega_n}\right)}$$

where ζ is the damping ratio and ω_n is the undamped natural frequency.

- **Magnitude (dB):**

$$M_{\text{dB}} = -20\log_{10}\sqrt{\left(1 - \frac{\omega^2}{\omega_n^2}\right)^2 + \left(2\zeta\frac{\omega}{\omega_n}\right)^2}$$

- **Phase:**

$$\phi = -\tan^{-1}\left(\frac{2\zeta\omega/\omega_n}{1 - \omega^2/\omega_n^2}\right)$$

Asymptotes of Quadratic Factors

Low-frequency asymptote ($\omega \ll \omega_n$):

- $\frac{\omega}{\omega_n} \ll 1 \implies M_{\text{dB}} \approx -20\log_{10}\sqrt{1^2+0} = 0$ dB
- The asymptote is a horizontal line at 0 dB.

High-frequency asymptote ($\omega \gg \omega_n$):

- $\frac{\omega}{\omega_n} \gg 1 \implies M_{\text{dB}} \approx -20\log_{10}\sqrt{\left(\frac{\omega^2}{\omega_n^2}\right)^2} = -40\log_{10}\left(\frac{\omega}{\omega_n}\right)$
- This asymptote is a straight line with a slope of **-40 dB/decade**, intersecting the 0 dB line at the corner frequency $\omega = \omega_n$.

Resonant Peak and Phase Variation

Resonance in Underdamped Systems

For underdamped systems ($0 < \zeta < 0.707$), the exact magnitude curve exhibits a resonant peak in the vicinity of ω_n.

- **Resonant frequency:** $\omega_r = \omega_n\sqrt{1-2\zeta^2}$
- **Resonant peak magnitude:** $M_r = \frac{1}{2\zeta\sqrt{1-\zeta^2}}$

Phase Angle Variation:

- For $\omega \ll \omega_n$, $\phi \approx 0^\circ$.
- At the corner frequency $\omega = \omega_n$, $\phi = -90^\circ$ (regardless of ζ).
- For $\omega \gg \omega_n$, $\phi \approx -180^\circ$.

The transition of the phase curve from 0° to -180° becomes much steeper for smaller values of the damping ratio ζ.

Introduction to Frequency Domain Specifications I

- The performance of a control system in the frequency domain is evaluated using various specifications.
- For a standard second-order system, time domain performance specifications like maximum overshoot, peak time, and settling time are closely correlated with frequency domain specifications.
- Frequency domain specifications are usually defined for closed-loop systems, but open-loop specifications like Gain Margin and Phase Margin are crucial for evaluating relative stability.

Introduction to Frequency Domain Specifications II

Key Significance

Frequency domain specifications give a clear picture of the system's stability, speed of response, and its ability to track varying inputs and reject high-frequency noise.

Key Frequency Domain Specifications

The following are the major specifications used to describe the frequency response of a system:

- **Resonance Peak (M_r)**
- **Resonant Frequency (ω_r)**
- **Bandwidth (ω_b)**
- **Cut-off Frequency (ω_c)** and **Cut-off Rate**
- **Gain Crossover Frequency (ω_{gc})**
- **Phase Crossover Frequency (ω_{pc})**
- **Phase Margin (PM)**
- **Gain Margin (GM)**

Resonance Peak (M_r)

The maximum value of the magnitude of the closed-loop frequency response is called the resonance peak, denoted by M_r. It occurs at the resonant frequency ω_r.

- A large resonance peak implies a large maximum overshoot in the transient step response.
- For a standard second-order system, M_r depends only on the damping ratio ζ:

$$M_r = \frac{1}{2\zeta\sqrt{1-\zeta^2}} \quad \text{for } 0 \leq \zeta \leq 0.707$$

- **Resonant Frequency (ω_r):** The frequency at which the maximum magnitude (M_r) occurs.

$$\omega_r = \omega_n \sqrt{1 - 2\zeta^2}$$

Bandwidth (ω_b)

The range of frequencies for which the system magnitude $M(\omega)$ does not fall below -3 dB (or $1/\sqrt{2}$ times) of its zero-frequency or low-frequency value.

- **Cut-off Frequency (ω_c):** The frequency at which the magnitude drops to -3 dB. Thus, the bandwidth spans from 0 to ω_c.
- Bandwidth is an indicator of the speed of response. A larger bandwidth corresponds to a faster rise time and a quicker transient response.

Bandwidth, Cut-off Frequency, and Cut-off Rate II

- **Cut-off Rate:** The slope of the magnitude curve beyond the cut-off frequency. It indicates how effectively the system attenuates high-frequency noise. A steeper cut-off rate implies better noise rejection.

Gain and Phase Crossover Frequencies I

These frequencies are defined from the *open-loop* frequency response $G(j\omega)H(j\omega)$ to determine the relative stability of the closed-loop system.

Gain Crossover Frequency (ω_{gc})

The frequency at which the magnitude of the open-loop transfer function is unity (or 0 dB).

$$|G(j\omega_{gc})H(j\omega_{gc})| = 1 \text{ or } 0 \text{ dB}$$

Phase Crossover Frequency (ω_{pc})

The frequency at which the phase angle of the open-loop transfer function is exactly $-180°$.

$$\angle G(j\omega_{pc})H(j\omega_{pc}) = -180^\circ$$

Phase Margin (PM)

Phase Margin (PM)

The additional phase lag that can be introduced into the system at the gain crossover frequency (ω_{gc}) without causing instability.

- It is calculated at $\omega = \omega_{gc}$.
- Formula:

$$\text{PM} = 180^\circ + \angle G(j\omega_{gc})H(j\omega_{gc}) = 180^\circ + \phi_{gc}$$

- **Physical Meaning:** If the phase plot drops further down and touches -180° precisely at ω_{gc}, the system becomes marginally stable. PM measures the "safety margin" in terms of phase.
- For a stable system, PM must be positive.

Gain Margin (GM)

Gain Margin (GM)

The factor by which the system gain can be increased before the system becomes unstable. It is evaluated at the phase crossover frequency (ω_{pc}).

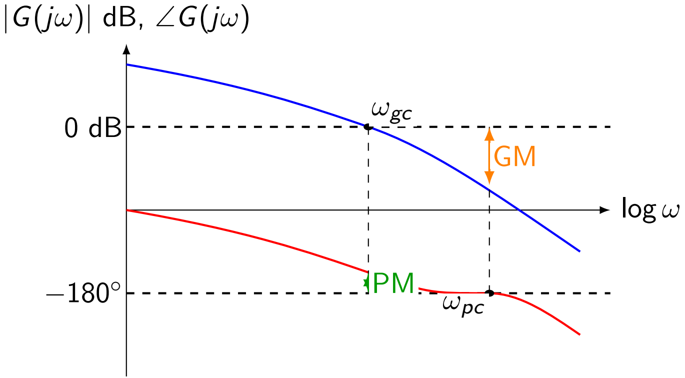
- It is usually expressed in decibels (dB).
- Formula:

$$\text{GM (linear)} = \frac{1}{|G(j\omega_{pc})H(j\omega_{pc})|}$$

$$\text{GM (dB)} = 20\log_{10}\left(\frac{1}{|G(j\omega_{pc})H(j\omega_{pc})|}\right) = -20\log_{10}|G(j\omega_{pc})H(j\omega_{pc})|$$

- For a stable system, GM (in dB) must be positive.

Graphical Representation (Bode Plot)



- **Stable System:** Both GM and PM are positive ($\omega_{gc} < \omega_{pc}$).

Stability Criteria via GM and PM I

The relative stability of a system can be directly assessed using the crossover frequencies, Gain Margin, and Phase Margin:

System Status	Crossover Freq.	GM (dB)	PM
Stable	$\omega_{gc} < \omega_{pc}$	Positive	Positive
Marginally Stable	$\omega_{gc} = \omega_{pc}$	Zero	Zero
Unstable	$\omega_{gc} > \omega_{pc}$	Negative	Negative

Design Trade-offs

A very large PM and GM implies a highly stable system but usually results in a sluggish time response (poor transient performance). A well-designed control system typically targets PM between 30° and 60° and $GM > 6$ dB.

Introduction to Bode Plots I

Bode Plot

A Bode plot is a graph of the frequency response of a system. It is a combination of a Bode magnitude plot, representing the magnitude (usually in decibels) of the frequency response, and a Bode phase plot, representing the phase shift.

- Introduced by Hendrik W. Bode in the 1930s.
- Used extensively for analyzing the stability and frequency response characteristics of control systems.
- Represents the transfer function $G(s)$ in the frequency domain as $G(j\omega)$ by substituting $s = j\omega$.
- Plotted on semi-logarithmic graph paper where frequency (ω) is on a logarithmic scale.

Standard Form of Transfer Function I

- Before sketching a Bode plot, express the open-loop transfer function $G(s)H(s)$ in the time-constant form (Bode form).
- For a general transfer function:

$$G(s)H(s) = \frac{K(s+z_1)(s+z_2)\dots}{s^N(s+p_1)(s+p_2)\dots}$$

- Substitute $s = j\omega$ and rewrite in the standard normalized form:

$$G(j\omega)H(j\omega) = \frac{K'(1+j\omega\tau_{z1})(1+j\omega\tau_{z2})\dots}{(j\omega)^N(1+j\omega\tau_{p1})(1+j\omega\tau_{p2})\dots}$$

where:

  - K' is the overall Bode gain.
  - τ_{zi}, τ_{pi} are the time constants of the zeros and poles respectively.
  - N is the type number of the system (number of poles at the origin).

Steps for Constructing Bode Magnitude Plot I

1. **Identify the Basic Factors**: Recognize the independent components in $G(j\omega)H(j\omega)$ such as constant gain, poles/zeros at origin, and first-order poles/zeros.
2. **Determine Corner Frequencies**: Calculate corner (or break) frequencies for all first-order factors using $\omega_c = \frac{1}{\tau}$, and arrange them in ascending order.
3. **Determine Initial Slope**:
   - The plot starts with a line of slope $-20N$ dB/decade, where N is the number of poles at the origin (or $+20N$ for zeros).
   - The starting magnitude at an initial low frequency ω_L (chosen smaller than the lowest corner frequency) is $20 \log K' - 20N \log \omega_L$.
4. **Draw the Asymptotic Plot**:
   - Proceed along the frequency axis and change the slope at each corner frequency.
   - Add -20 dB/decade for each simple pole, and $+20$ dB/decade for each simple zero encountered.

Steps for Constructing Bode Phase Plot I

- The phase angle equation is given by the angle of the complex function:

$$\phi(\omega) = \angle G(j\omega)H(j\omega)$$

- For the standard time-constant form, the phase is the sum of angles of zeros minus the sum of angles of poles:

$$\phi(\omega) = \sum \angle(zeros) - \sum \angle(poles)$$

$$\phi(\omega) = \sum \tan^{-1}(\omega\tau_z) - 90^\circ N - \sum \tan^{-1}(\omega\tau_p)$$

- **Steps**:
  1. Select a range of frequencies ω, typically covering a decade below the lowest corner frequency to a decade above the highest corner frequency.
  2. Calculate the total phase angle ϕ at each selected ω.
  3. Plot ϕ versus ω on the linear y-axis of a semi-log graph and connect the points smoothly.

Example 1: Pure Gain (Constant Factor) I

- Transfer Function:
  $G(s) = K$
- Frequency Domain:
  $G(j\omega) = K$
- **Magnitude**:

$$|G(j\omega)|_{dB} = 20 \log K$$

- **Phase**:
  $\angle G(j\omega) = 0^\circ$ for $K > 0$
  $\angle G(j\omega) = -180^\circ$ for $K < 0$

Key Observation

For a constant gain factor, the magnitude plot is a horizontal straight line across all frequencies. The phase plot is also a horizontal straight line, indicating no phase shift variations with frequency.

Example 2: Integrator (Pole at Origin) I

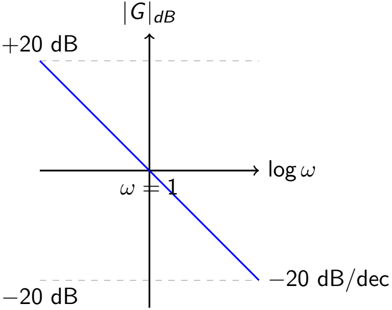
- Transfer Function: $G(s) = \frac{1}{s}$
- $G(j\omega) = \frac{1}{j\omega} = -j\frac{1}{\omega}$
- **Magnitude**:

$$|G(j\omega)|_{dB} = -20\log\omega$$

- **Phase**:

$$\angle G(j\omega) = -90^\circ$$

- The magnitude plot is a straight line with a constant slope of -20 dB/decade.
- It crosses the 0 dB axis at $\omega = 1$ rad/sec.



Example 3: First-Order Pole (Simple Pole) I

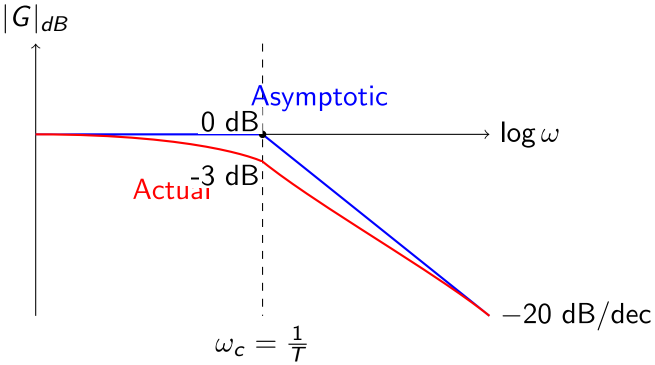
- Transfer Function: $G(s) = \frac{1}{1+sT}$
- Frequency Domain: $G(j\omega) = \frac{1}{1+j\omega T}$
- **Corner Frequency**: $\omega_c = \frac{1}{T}$
- **Magnitude**:

$$|G(j\omega)|_{dB} = -20 \log \sqrt{1 + (\omega T)^2}$$

  - For low frequencies ($\omega \ll \omega_c$): $|G(j\omega)|_{dB} \approx -20 \log(1) = 0$ dB.
  - For high frequencies ($\omega \gg \omega_c$): $|G(j\omega)|_{dB} \approx -20 \log(\omega T)$ dB.
- **Phase**:

$$\angle G(j\omega) = -\tan^{-1}(\omega T)$$

Bode Plot for First-Order Pole I



Bode Plot for First-Order Pole II

Error at Corner Frequency

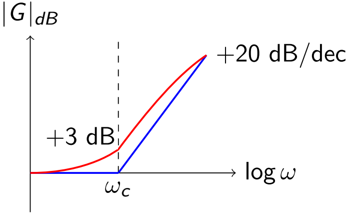
The maximum error between the asymptotic approximation and the exact magnitude plot occurs exactly at the corner frequency ω_c, and is approximately -3 dB.

Example 4: First-Order Zero (Simple Zero) I

- Transfer Function: $G(s) = 1 + sT$
- Frequency Domain: $G(j\omega) = 1 + j\omega T$
- **Corner Frequency**: $\omega_c = \frac{1}{T}$
- **Magnitude**:

$$|G(j\omega)|_{dB} = 20 \log \sqrt{1 + (\omega T)^2}$$

  - $\omega \ll \omega_c$: Asymptote at 0 dB.
  - $\omega \gg \omega_c$: Asymptote slope +20 dB/dec.



- **Phase**:

$$\angle G(j\omega) = \tan^{-1}(\omega T)$$

  - Approaches $+90^\circ$ at high frequencies.

Summary of Elementary Factors I

Factor	$G(j\omega)$	**Mag. Slope (dB/dec)**	**Phase Angle**
Constant Gain	K	0	0° (if $K > 0$)
Integrator (Pole at origin)	$\frac{1}{j\omega}$	-20	-90°
Derivative (Zero at origin)	$j\omega$	$+20$	$+90^\circ$
Simple Pole	$\frac{1}{1+j\omega T}$	0 then -20 (at ω_c)	$0^\circ \to -90^\circ$
Simple Zero	$1 + j\omega T$	0 then $+20$ (at ω_c)	$0^\circ \to +90^\circ$

Additive Property

Because Bode plots use a logarithmic magnitude scale, the total magnitude curve is simply the linear addition of the asymptotic magnitude curves of all individual factors.