Control Theory Bootcamp: Basics

Electronics and Robotics Club

June 09, 2025

Curated by Radhika Agarwal
Convener of Electronics and Robotics Club

Introduction

Have you ever tried balancing a pencil on the end of your finger? Maybe for the first few times it falls off. But with some practice, you can balance it like a pro. It definitely takes a lot of muscle control to balance a pencil. Even the slightest vibration could tip it over.

This seemingly mundane example tells us several key ideas about control that extend to almost every engineering discipline. We seek to maintain the state of a particular system by controlling it while there are opposing factors that seek to destabilize the system. Many devices that seem to work like a charm require a good amount of control in the background for them to function properly. Control systems ensure that unpredictable disturbances or noise do not jeopardize their functioning, while also maintaining optimal resource usage and stability.

If you don't want your systems to go out of control, this is the place to understand the key ideas behind controlling!

Control systems - the basic definitions

A **Control System** is a collection of components that are collectively responsible for bringing the output of the system to the one desired by the input. (Note that a control system contains a controller, and is not the same thing.)

The **input** consists of instructions and measurements of parameters that dictate what the desired output should be.

The **output** consists of a set of variables that describe the features of the system, collectively referred to as the **state of the system**. The state of a system is generally represented by x.

For example - In a car, the input could be the instructions to a driver, while the output could be the position and velocity of the car.

A control system typically consists of the following components:

- **Plant** - The part of the control system that is being controlled. It consists of the **actuator**, which executes the control command, and the **process** which responds to the actuation and undergoes a time evolution. In many cases, the plant could be a dynamic system like a car or a pendulum.
- **Controller** - The part of the control system that provides control commands to the plant so that the state gets driven to the desired one. The control command is generally represented by u.
- **Sensors** - The part of the control system that takes an observation/measurement of the state completely or partially. The sensor observation is generally represented by y.

Looking at a car in more detail, we might realise that the driver acts as a controller and the car as a plant. The speedometer serves as a sensor that guides the driver to press the accelerator more or less to achieve a particular speed.

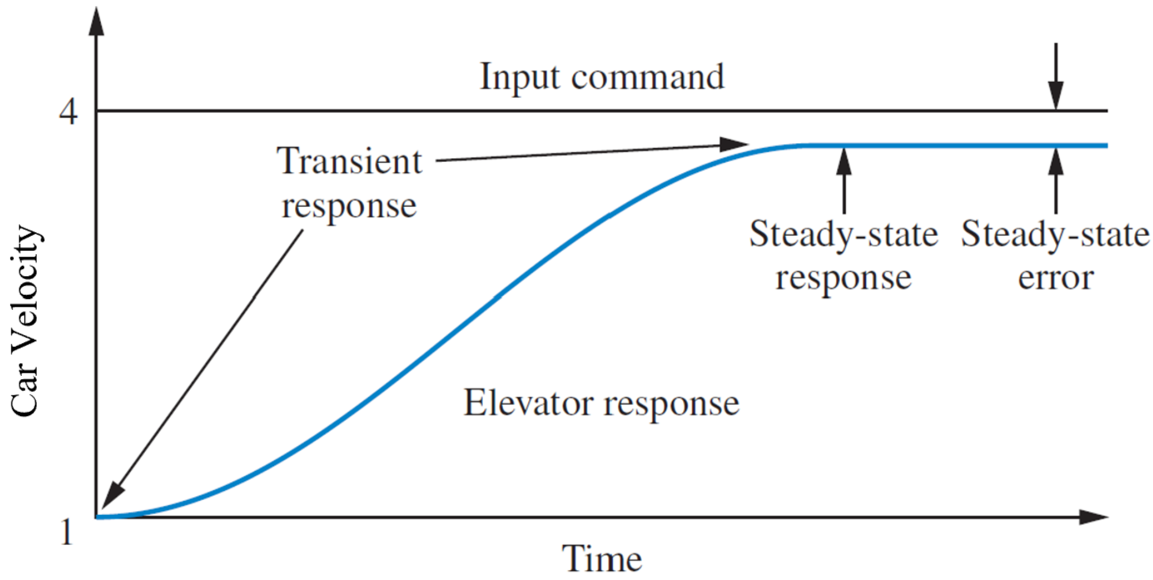

Figure 1: Car response

Two key performance measures in the car system are the transient response and the steady-state error. The transient response affects driver comfort and control-if it's too fast, the ride feels jerky; if too slow, the car feels unresponsive. The steady-state error reflects how accurately the car reaches the desired speed. A large error can reduce the effectiveness of systems like cruise control, impacting fuel efficiency and driving experience.

The interesting thing we notice here is that the sensor gives feedback to the driver about what action to perform. This brings us to the next topic: **open and closed loop control**.

Types of Control Systems

1) Based on Feedback

Open Loop Control Systems

Applying control commands without taking any measurements of the output.

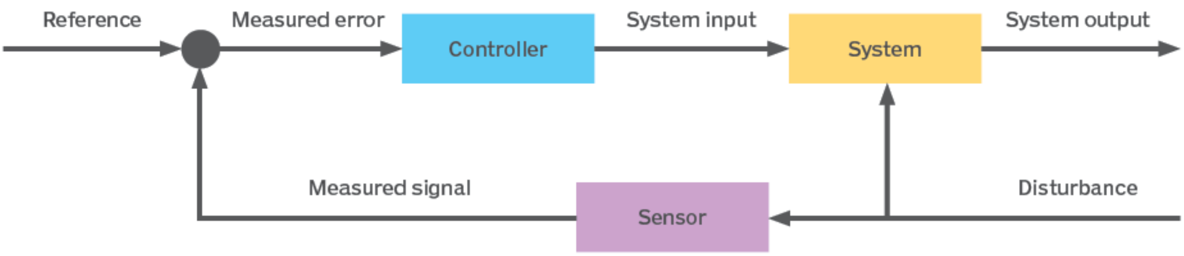

Figure 2: Structure of an open loop control system.

Example

In the example of the car, if the driver does not pay attention to the speedometer or traffic and presses the accelerator regardless of it, the system exhibits **open loop control**. The output - speed - does not affect the control command - pressing the accelerator. The inputs to the controller - the driver - are from their mind and are unaffected by sensory information from the surroundings.
It becomes quite obvious why such a system is problematic. It cannot handle uncertainties like changes in the speed limit, or the presence of vehicles or pedestrians nearby, and will give an undesirable output in such situations. However, if the traffic conditions are exactly known beforehand, the driver could simply memorise a predefined sequence of accelerator presses and still manage to drive the car safely.
We can immediately see that this is a rather artificial example, and for any car to function correctly, the controller needs feedback from its environment.

Closed Loop Control Systems / Feedback Control Systems

Applying control commands based on the measurements of the output with the help of a controller.

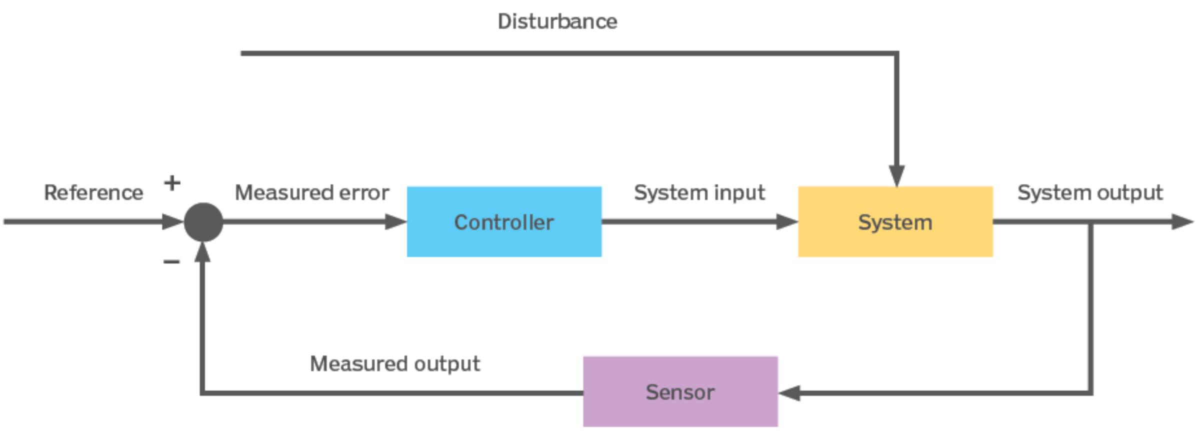

Figure 3: Structure of a closed loop control system.

Example

By taking note of the speedometer and the surrounding traffic, the driver can now correct their actions so that the car continues to perform desirably. The exact nature of this corrective response is called the **control law**, which will be a key topic of discussion in upcoming sections.

Advantages

Closed loop control offers distinct advantages over open loop control, such as:

- Greater resistance to noise, bias, and disturbances in the system
- More efficient energy consumption
- Ability to alter the dynamic properties of the overall system

2) Based on Energy Expenditure

Passive Control

Passive control involves minimal to no energy expenditure to impose the desired behaviour.
Example: Streamlined shapes of vehicles.

Active Control

Active control involves greater energy expenditure to achieve the desired behaviour.
Example: Externally oscillating the base of an inverted pendulum to keep it stable.

Linear and Nonlinear systems

Linear Systems

Linear systems are systems where the evolution of the state of the system is related linearly with the state. An example is radioactive decay.

$$\dot{x} = Ax$$

In a more general context, systems in which the output depends linearly on the input (satisfying the properties of scaling and linear combination) are also described as linear systems. These systems can be represented using linear differential equations.

Nonlinear Systems

Nonlinear systems are systems where the evolution of the state is not related linearly with the state. An example is a simple pendulum in a gravitational field.

$$\dot{x} = f(x)$$

Many nonlinear systems can be treated approximately as linear systems around certain states (called **fixed points**, where $\dot{x} = 0$), through **local linearization**.

Time-Invariant Systems

Time-invariant systems are systems where the coefficients in the equations governing the state are not dependent on time.
Linear and Time-Invariant (LTI) systems are important because they are easier to analyze and a broad class of problems can be approximated using such systems.

Mathematical Modeling of Systems

Transfer Function

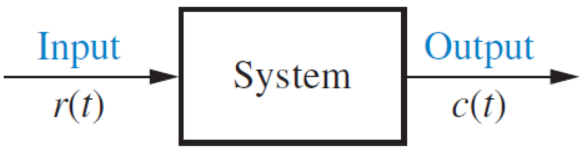

Figure 4: Block diagram representation of a system

We will now try to formulate the system representation shown in Figure 4 by establishing a viable definition for a function that algebraically relates a system's output to its input. Let us begin by writing a general nth-order, linear, time-invariant differential equation,

$$a_n \frac{d^n c(t)}{dt^n} + a_{n-1} \frac{d^{n-1} c(t)}{dt^{n-1}} + \cdots + a_0 c(t) = b_m \frac{d^m r(t)}{dt^m} + b_{m-1} \frac{d^{m-1} r(t)}{dt^{m-1}} + \cdots + b_0 r(t)$$

where $c(t)$ is the output, $r(t)$ is the input, and the a_i's, b_i's, and the form of the differential equation represent the system. Take Laplace on both sides and assume all initial conditions to be zero, reducing the equaiton to:

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

Now form the ration of the output transform $C(s)$ divided by the input transform $R(s)$:

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

This ration $G(s)$ is called the **Transfer Function** and it is evaluated with zero initial condtions.

Second-Order Systems

A general **second-order linear system** has the following transfer function:

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

Where:

- ω_n is the **natural frequency** of the system.
- ζ is the **damping ratio**.

The characteristic equation (denominator) determines the poles of the system:

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

Solving this gives the poles:

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

Depending on the value of ζ, the nature of the system response varies:

- $\zeta = 0$: Undamped (purely oscillatory)
- $0 < \zeta < 1$: Underdamped (oscillatory but decaying)
- $\zeta = 1$: Critically damped (fastest non-oscillatory response)
- $\zeta > 1$: Overdamped (slow non-oscillatory response)

When $0 < \zeta < 1$, the system has complex-conjugate poles and oscillates with a **damped natural frequency** ω_d given by:

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

Relevance of Constants in Physical Scenarios

The constants ζ, ω_n, and ω_d significantly affect the physical behavior of control systems. They define how fast and how smoothly a system responds:

- **Overshoot**: How far the response exceeds the desired value — increases as ζ decreases.
- **Settling Time**: Time to reach and stay within a range of the desired value — depends on ζ and ω_n.
- **Rise Time**: Time to go from 10% to 90% of the final value — inversely related to ω_n.
- **Oscillations**: Occur when poles have imaginary parts — present for $\zeta < 1$.

> **Note**
>
> Pole locations in the complex plane directly dictate system response. Closer poles to the origin mean faster response. Poles farther to the left mean greater stability. Poles near the imaginary axis mean oscillatory behavior.

Zeroes and Poles

Lets write the transfer function $G(s)$ as a fraction with its numerator $N(s)$ and denominator functions $D(s)$:

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

- Roots of the equation $N(s) = 0$ are defined as the **System Zeroes**. The transfer function tends to 0.
- Roots of the eqaution $D(s) = 0$ are defined as the **System Poles**. The transfer function tends to ∞.

> **Note**
>
> The poles and zeros are properties of the transfer function, and therefore of the differential equation describing the input-output system dynamics. Together with the gain constant K they completely characterize the differential equation, and provide a complete description of the system.

If instead you chose to express your system in terms of matrices, in space state representation that is:

$$\dot{x} = Ax$$

Then the **eigenvalues** (equivalent to the poles of the transfer functon) help determine the system's natural behaviour - how it resopnds over time.

Controllability of Systems

Dynamics of a Linear System in the Presence of a Controller

$$\dot{x} = Ax + Bu$$

where

- x is the **state vector** - it describes the state of the system (position, velocity, temperature, etc.)
- A is the **system matrix** - it tells how the state naturally evolves on its own

- B is the **input matrix** - it tells how external input affects the state
- u is the **control input** -this is the signal you provide to the system through the controller to drive the system towards the desired behavior.

Example

Example: In an inverted pendulum: x could be angle and angular velocity and u could be the force applied to the cart.

Stability

In order to explain stability, we start from the fact that the total response of a system is the sum of the natural response and the forced response. When you studied linear differential equations, you probably referred to these responses as the homogeneous and the particular solutions, respectively. **Natural Response** describes the way the system dissipates or acquires energy. The form or nature of this response is dependent only on the system, not the input. On the other hand, the form or nature of the **Forces Response** is dependent on the input. Thus, for a linear system, we can write:

Total Response = Natural Response + Forced Response

In very simple terms, a linear, time-invariant system is **stable** if the natural response approaches zero as time approaches infinity.
A linear, time-invariant system is **unstable** if the natural response grows without bound as time approaches infinity.

In $\dot{x} = Ax$ the eigenvalues (or called **Poles**) determine the system's natural behaviour - like how it responds over time.

- **Negative real parts** - stable (response dies off)
- **Positive real parts** - unstable system
- **Imaginary parts** - oscillations

Checking the eigenvalues of the final matrix helps us decide if the system is stabilizable (and whether it can be stabilized through control).

Controllability

Controllability is the property of a system where it is possible to access any state in the state space with a suitable controller. In other words, the reachable set $R_t = \mathbb{R}^n$ from any given initial state in finite time.

Now in the above equation, if you design a controller where $u = Kx$, where K is the **gain matrix**, it is called a **Full State feedback**.
You are essentially saying that at every time instant, I will choose the control input u based on the current state x multiplied by some gain K.
Substituting $u = Kx$ in the equation you get:

$$\dot{x} = (A + BK)x$$

And thus the new dynamics of the system depend on the matrix $A + BK$.

Thus if the system is controllable, you can find a suitable K to place poles anywhere you want, i.e. you can choose $A + BK$ to be stable.

In many cases, this implies that the eigenvalues of the system can be set to any arbitrary value (**arbitrary pole placement** in the context of transfer functions).

Test for Controllability of a System

Controllability matrix:

$$C = \begin{bmatrix} B & AB & A^2B & A^3B & \dots & A^{n-1}B \end{bmatrix}$$

where n is the state space dimension.

The system is controllable $\iff C$ is full rank.

Example

$$A = \begin{bmatrix} 1 & 0 \\ 1 & 2 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$C = \begin{bmatrix} B & AB \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 2 \end{bmatrix}$$

$$\text{rank}(C) = 2, \text{ so the system is controllable.}$$

Further Reading

To know more about stability, controllability, reachability, and related concepts, refer to the following:

- Difference between controllability and reachability
- Reachability and controllability

Since these topics are quite math-intensive, you might Watch Steve Brunton videos from 4 to 11 (the previous ones have already been covered above).

PID Control

PID stands for **Proportional-Integral-Derivative**. It is a special type of closed-loop control law that is widely used.

Control Law

$$u(t) = K_p e(t) + K_i \int e(t)\,dt + K_d \frac{de(t)}{dt} = K_p \left(e(t) + \frac{1}{\tau_i} \int e(t)\,dt + \tau_d \frac{de(t)}{dt} \right)$$

Where:

- K_p is the proportional weight
- K_i is the integral weight
- K_d is the derivative weight
- τ_i is the integral time constant
- τ_d is the derivative time constant
- $e(t)$ is the measured error between desired state and actual state
- $u(t)$ is the plant input and also the controller output

Intuition behind the role of different components

- **P (Proportional)** - In many simple situations, proportional control ensures that the desired state (set point) is asymptotically reached from the initial state. It deals with the knowledge of the present error.
- **I (Integral)** - Integral control takes into account the past errors and their duration of persistence, thus it plays an important role in reducing steady-state error in many situations. It deals with the knowledge of past errors.
- **D (Derivative)** - Derivative control takes into account the rate at which the error is decreasing, thus it plays an important role in preventing possible overshoot due to the integrator. It deals with the knowledge of future error change.

Drawbacks of P

- In the discrete-time version, the state can oscillate around a certain mean state which isn't the set point.
- Not suitable in cases where a certain state needs to be maintained and external forces like gravity are present irrespective of the error.

Drawbacks of I

- The possibility of overshooting the desired state is high.
- In the case of actuator saturation, **integral wind-up** can occur.

Drawbacks of D

- High-frequency noise can make the derivative contribution higher than required.

Continuous and Discrete-time Systems

Continuous-time systems - Systems where the evolution of the state is considered for any time. Example:

$$\dot{x} = Ax$$

Discrete-time systems - Systems where the evolution of the state is considered only at discrete time steps. Example (discrete-time, linear system):

$$x_{k+1} = \tilde{A}x_k$$

Each time step k is separated by $\Delta t \geq \epsilon > 0$.

Discrete-time PID Control

$$u(k) = K_p \left(e(k) + \frac{1}{\tau_i} \sum_{i=0}^{k} e(i)\Delta t + \tau_d \frac{e(k) - e(k-1)}{\Delta t} \right)$$

The discrete form of the PID controller is used when the sampling frequency $\left(\frac{1}{\Delta t}\right)$ is much lower compared to the speed of the dynamics of the system. This is also one of the forms of PID Control used in most practical situations, especially in programs.