B

EECS C106B / 206B Robotic Manipulation and Interaction **Spring 2022**

Lecture 1B: (Intro to Modeling, Dynamics and Feedback Control)

Scribes: Kevin Chen, Jonathan Li, Sagar Patel

1.1 General Announcements and Admin

1. HW1 released.
   (a) Due Wednesday, Feb 2nd.
2. Project 1A released.
   (a) Recommended to go through the spec before lab section this week.
3. Friday Lab 11-2 is oversubscribed, consider switching to Friday 2-5.
4. Discussions start this week.

1.2 Overview

This lecture provides an overview of dynamics and covers the topics of modeling, linearization, stability, and control. Throughout the lecture, these concepts will be tied to several real-world systems examples, including the quadrotor, spring mass system, and the pendulum.

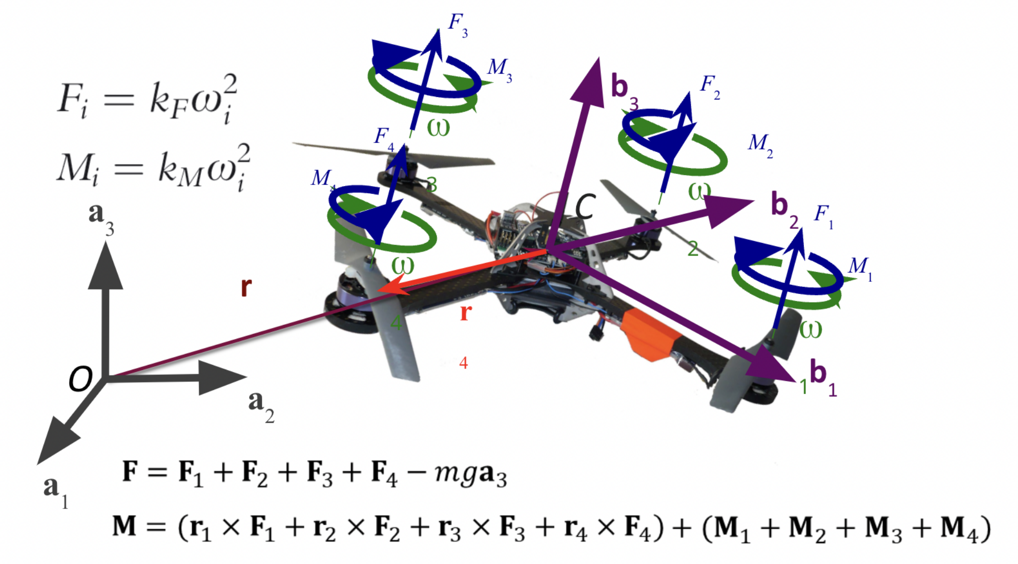
It starts with a review of controls using a slide deck from UPENN GRASP lab (see the slide links on the course website)

1.3 Dynamics and Modeling

1.3.1 Quadrotor Model

We begin by modelling the quadcopter. Quadcopters work with four rotors, two rotors spinning counter-clockwise and two clockwise. For each rotor, the spin is the opposite of the rotor on the opposite end.

The forces can be modelled as shown below, where each motor produces a force in proportional to the square of its rotational speed:


$F_i = k_F \omega_i^2$
$M_i = k_M \omega_i^2$
F_3
M_3
F_2
M_2
F_4
M_4
F_1
M_1
ω
$\mathbf{b}_3$
$\mathbf{b}_2$
$\mathbf{b}_1$
C
$\mathbf{r}$
O
$\mathbf{a}_3$
$\mathbf{a}_2$
$\mathbf{a}_1$
$\mathbf{F} = \mathbf{F}_1 + \mathbf{F}_2 + \mathbf{F}_3 + \mathbf{F}_4 - mg\mathbf{a}_3$
$\mathbf{M} = (\mathbf{r}_1 \times \mathbf{F}_1 + \mathbf{r}_2 \times \mathbf{F}_2 + \mathbf{r}_3 \times \mathbf{F}_3 + \mathbf{r}_4 \times \mathbf{F}_4) + (\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4)$

1.3.2 Newton-Euler Equations for Quadrotor

Writing this in Newton-Euler equations, with the standard rotation matrix, we have:

Newton-Euler Equations for a Quadrotor

$${}^{A}\boldsymbol{\omega}^{B} = p\,\mathbf{b}_1 + q\,\mathbf{b}_2 + r\,\mathbf{b}_3$$

$$m\ddot{\mathbf{r}} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + R \begin{bmatrix} 0 \\ 0 \\ \underbrace{\boxed{F_1 + F_2 + F_3 + F_4}}_{u_1} \end{bmatrix}$$

In inertial frame

$$I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \underbrace{\boxed{\begin{bmatrix} L(F_2 - F_4) \\ L(F_3 - F_1) \\ M_1 - M_2 + M_3 - M_4 \end{bmatrix}}}_{\mathbf{u}_2} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

In body frame

Note: there are notational conventions that are different what we are used to, where ω_1 is p, ω_2 is called q, and ω_3 is called r. Additionally, the reference frame designation is different, where ${}^{A}\omega^{B}$ gives how the B frame transforms to the A frame

There are two sets of inputs: u_1 which designates the forces and a vector of inputs u_2 which has three inputs for a total of 4 inputs. We can re-arrange the equations to matrix format:

Recall that $\mathbf{F}_i = k_F \omega_i^2$ and $\mathbf{M}_i = k_M \omega_i^2$

Let $\gamma = \frac{k_M}{k_F} = \frac{\mathbf{M}_i}{\mathbf{F}_i} \Leftrightarrow \mathbf{M}_i = \gamma \mathbf{F}_i$

$$I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} L(F_2 - F_4) \\ L(F_3 - F_1) \\ M_1 - M_2 + M_3 - M_4 \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

$$I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \underbrace{\boxed{\begin{bmatrix} 0 & L & 0 & -L \\ -L & 0 & L & 0 \\ \gamma & -\gamma & \gamma & -\gamma \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix}}}_{\mathbf{u}_2} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

And combining the input matrix, we have:

Putting everything together, we have inputs:

$$\mathbf{u} = \begin{bmatrix} \underline{u_1} \\ \\ \boldsymbol{u}_2 \\ \\ \end{bmatrix} = \begin{bmatrix} \underline{\text{thrust}} \\ \text{moment about } x \\ \text{moment about } y \\ \text{moment about } z \end{bmatrix}$$

Note: All quantities are in the body frame!

$$\mathbf{u} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & L & 0 & -L \\ -L & 0 & L & 0 \\ \gamma & -\gamma & \gamma & -\gamma \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} \qquad \mathrm{F}_i = k_F \omega_i^2$$

1.3.3 State Space

You can define state by:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \tag{1.1}$$

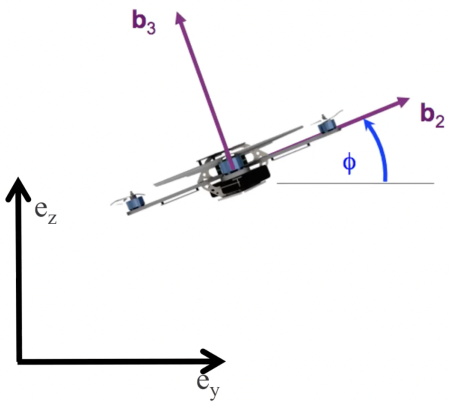
For robotic systems, q typically describes the position and $\dot{q}$ describes the velocity

The state is described by Ordinary Differential Equations (ODEs)

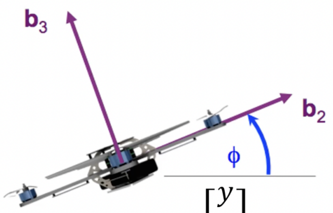
Example: Planar Quadrotor

For a planar quadrotor, the state variables can be chosen as follows:

$$\mathbf{q} = \begin{bmatrix} y \\ z \\ \phi \end{bmatrix}, \mathbf{x} = \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix}$$



For the quadrotor, the state matrix maps to the following:



$$\begin{bmatrix} \ddot{y} \\ \ddot{z} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} 0 \\ -g \\ 0 \end{bmatrix} + \begin{bmatrix} -\frac{1}{m}\sin\phi & 0 \\ \frac{1}{m}\cos\phi & 0 \\ 0 & \frac{1}{I_{xx}} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\mathbf{x} = \begin{bmatrix} \boldsymbol{q} \\ \dot{\boldsymbol{q}} \end{bmatrix} = \begin{bmatrix} y \\ z \\ \phi \\ \dot{y} \\ \dot{z} \\ \dot{\phi} \end{bmatrix} \qquad \dot{\mathbf{x}} = \begin{bmatrix} x_4 \\ x_5 \\ x_6 \\ 0 \\ -g \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -m^{-1}\sin x_3 & 0 \\ m^{-1}\cos x_3 & 0 \\ 0 & I_{xx}^{-1} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$$

Note that the state derivative $\dot{\mathbf{x}}$ is a function of state $\mathbf{x}$ and input $\mathbf{u}$

1.4 Building a state space model

Professor's note: the modelling of rotorcraft or physical systems is not trivial, it is a critical and difficult part of the process

In order to build the state space model, write the system dynamics in matrix form where $\dot{\mathbf{x}}$ is a function of state $\mathbf{x}$ and input $\mathbf{u}$

A system is Linear Time-invariant (LTI) if it can be written in the form $\dot{x} = A\mathbf{x} + B\mathbf{u}$

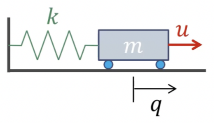
Most robotic systems are non-linear, however, they can be linearized for computational purposes.

1.4.1 LTI Example: Spring Mass System

The following system is Linear:

$$m\ddot{q}(t) + kq(t) = u(t)$$

Order of the system = 2



Rewrite $\ddot{q}(t) = \frac{u(t) - kq(t)}{m}$

State vector $\mathbf{x} = [x_1\ x_2]^{\mathrm{T}} = [q\ \ \dot{q}]^{\mathrm{T}}$

Coupled equations $\dot{x}_1 = x_2,\ \dot{x}_2 = \frac{u - kx_1}{m}$

Then $\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{u - kx_1}{m} \end{bmatrix}$, or $\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} u$

➢ Linear system

1.4.2 Non-linear Example: Pendulum

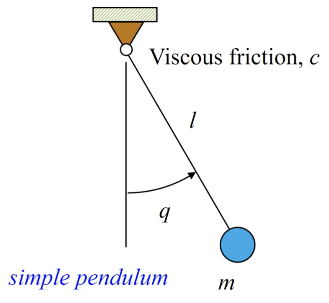
The following system is non-linear, however it can be linearized with the constraint $\dot{\mathbf{x}} = f(\mathbf{x}) = 0$

Equation of motion

$$\ddot{q} + \frac{c}{ml^2}\dot{q} + \frac{g}{l}\sin q = 0$$

State space representation

$$x = \begin{bmatrix} q \\ \dot{q} \end{bmatrix}, \dot{x} = \begin{bmatrix} x_2 \\ -\frac{g}{l}\sin x_1 - \frac{c}{ml^2}x_2 \end{bmatrix}$$



➢ **Nonlinear** $\Longrightarrow$ Linearize around equilibria $\quad \dot{\mathbf{x}} = f(\mathbf{x}) \equiv 0$

1.5 Stability

Stability is an important concept in control, as we often want to implement controllers for robotic systems such that they operate near an equilibrium state. In this section, we will learn how to analyze the stability of systems by examining their behavior around equilibrium points.

1.5.1 Equilibrium

A system is at static equilibrium if $\dot{x} = f(x) \equiv 0$. We say that a configuration q_e is at static equilibrium if $x(t_0) = q_e \Rightarrow x(t > t_0) = q_e$. In other words, at equilibrium, the state of the system is unchanging, as the derivative of its state is 0.

We are interested in analyzing the behavior of systems around these equilibrium points. However, this behavior is not always easy to analyze for nonlinear systems, so we can simplify our analysis by linearizing our system around equilibria.

1.5.2 Linearization

The goal of linearization is that given a nonlinear system $\dot{x} = f(x, u), x, f \in \Re^n, u \in \Re^m$ we want to derive an approximate linear system $\dot{x} = Ax + Bu$ about an equilibrium point (x_e, u_e).

We achieve this by taking the Taylor series expansion around the equilibrium point:

$$f(x_e + \Delta x, u_e + \Delta u) = f(x_e, u_e) + \left[\frac{\partial f}{\partial x}\right]_{x_e,u_e} + \left[\frac{\partial f}{\partial u}\right]_{x_e,u_e} \tag{1.2}$$

$$\dot{x} + \Delta\dot{x} \approx f(x_e, u_e) + \left[\frac{\partial f}{\partial x}\right]_{x_e,u_e} + \left[\frac{\partial f}{\partial u}\right]_{x_e,u_e} \tag{1.3}$$

$$\Delta\dot{x} = A\Delta x + B\Delta u \tag{1.4}$$

We redefine $\Delta x \triangleq x$ and $\Delta u \triangleq u$ to yield $\Delta\dot{x} = Ax + Bu$ where

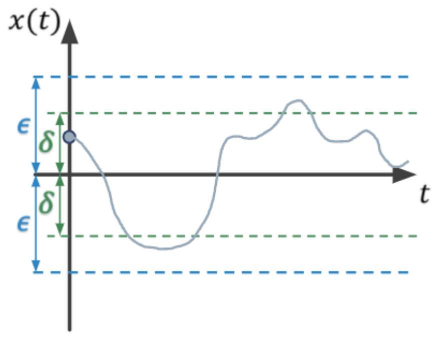
$$A_{nxn} = \left[\frac{\partial f}{\partial x}\right]_{x_e,u_e} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{(x_e,u_e)} \tag{1.5}$$

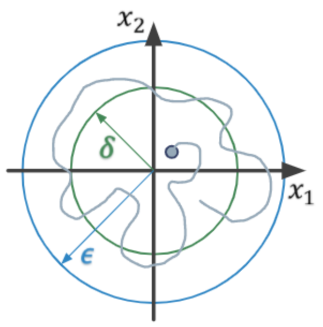
and

$$B_{nxm} = \left[\frac{\partial f}{\partial u}\right]_{x_e,u_e} = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial u_1} & \cdots & \frac{\partial f_n}{\partial u_n} \end{bmatrix}_{(x_e,u_e)} \tag{1.6}$$

1.5.3 Stability in the Sense of Lyapunov (i.s.L)

An equilibrium point x_e of the system $\dot{x} = f(x)$ is stable in the sense of Lyapunov if for any $\epsilon > 0$ there exists a value $\delta(t_0, \epsilon) > 0$ such that if $||x(t_0, x_0) - x_e|| < \delta$ then $||x(t; t_0, x_0) - x_e|| < \epsilon$ for all $t \geq t_0$. An interpretation of this statement is that configurations starting within a certain distance from the equilibrium will remain within a distance from it forever.



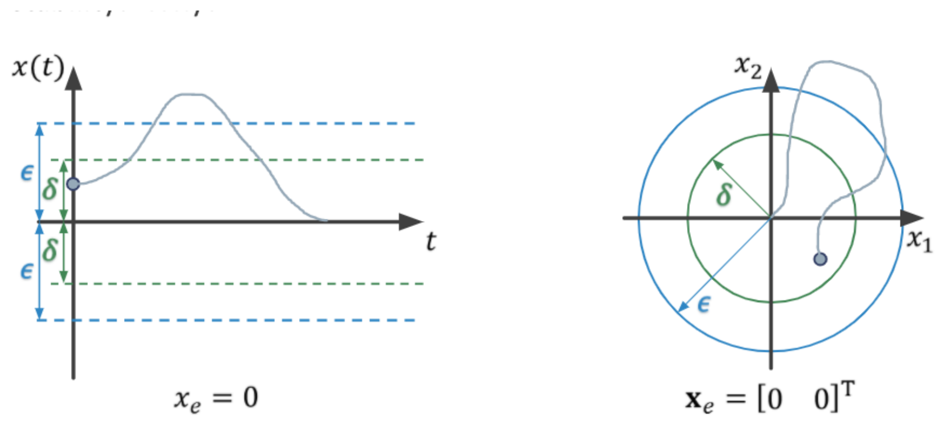


Some properties of Lyapunov Stability are the following:
An equilibrium point is unstable if it is not stable i.s.L
The equilibrium point is uniformly stable i.s.L if $\delta = \delta(\epsilon)$

While stability i.s.L provides bounds on the system near the equilibrium points, it does not answer the questions of whether the system will actually reach the equilibrium point, or will the system stay at exactly the equilibrium point for all future times. For this, we expand our analysis to a few other types of stability.

1.5.4 Stability Analysis

We now introduce some stronger statements of stability. The first is asymptotic stability. We say that an equilibrium point is **asymptotically stable** i.s.L if it is:
1. Stable (i.s.L)
2. Convergent: $x(t; t_0, x_0) \to x_e$ as $t \to \inf$
Note that convergence alone does not imply asymptotic stability, as a system can be convergent, but not



bounded around the equilibrium point. Also, asymptotic stability does not answer the question of how fast the system converges to equilibrium.
Another statement of stability is exponential stability. We say that an equilibrium point $x_e = 0$ is exponentially stable if there exists coefficient $m \geq 0$ and rate $a \geq 0$ such that $||x(t)|| \leq ||x_0||me^{-a(t-t_0)}$ for all x_0 in some ball around $x_e = 0$. This statement of stability provides an analysis of both how bounded the system is around the equilibrium point, as well as the rate at which it converges.

So far, these are all just local definitions of stability around an equilibrium point, as we were free to choose small δ in order to start our system at a state x_0 that is sufficiently close to x_e. A stronger statement of stability is that an equilibrium point x_e is **globally stable** if it is stable for all initial conditions x_0.

1.5.5 Stability of Linear-Time Invariant (LTI) Systems

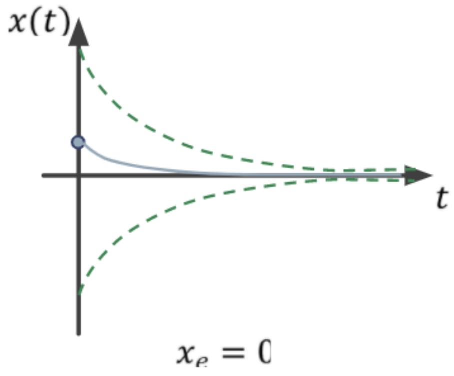
A Linear-Time Invariant system is defined as

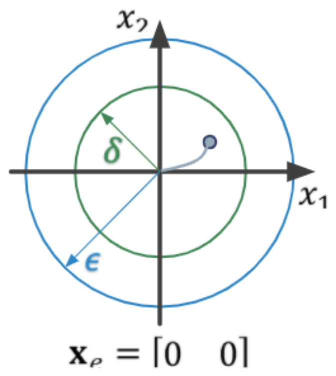
$$\dot{x} = Ax \tag{1.7}$$

$$\dot{x} = Ax \tag{1.8}$$

$$x \in \mathbf{R}^n \tag{1.9}$$

$$A \in \mathbf{R}^{nxn}, \text{ constant} \tag{1.10}$$





An LTI system is asymptotically stable if and only if all the eigenvalues of A have strictly negative real parts.

For LTI systems, asymptotic stability $\Leftrightarrow$ exponential stability.

The system is marginally stable if and only if all the eigenvalues of A have nonpositive real parts, at least one has zero real part, and every eigenvalue with zero real parts has its algebraic multiplicity equal to it's geometric multiplicity. Algebraic multiplicity refers to the number of times an eigenvalue is a root of the characteristic polynomial, while geometric multiplicity refers to the number of eigenvectors associated with an eigenvalue. The concept of marginally stable is used to define systems that will not give an unbounded output given an impulse of finite magnitude input.

1.6 Control

1.6.1 First Order System Control

Linear control problems can be framed in the context of a state x and an input u. x and u are related by some model–for example, the kinematic model

$$\dot{x} = u \tag{1.11}$$

where x is a position description and u is a velocity input.

Based on some input of u, we then wish to follow some trajectory $x^{des}(t)$. The general approach to solving a control problem like this is to define an error between our desired state $x^{des}(t)$ and our actual state $x(t)$. Mathematically, we can express this as

$$e(t) = x^{des}(t) - x(t) \tag{1.12}$$

which we then want to converge exponentially to 0.

A strategy we can use to solve this is to choose u such that

$$\dot{e} + K_p e = 0 \tag{1.13}$$

This is a differential equation, and if $K_p > 0$, then our solution becomes

$$e(t) = \exp\left(-K_p(t - t_0)\right)e(t_0). \tag{1.14}$$

Our desired input $u(t)$ is given by

$$u(t) = \dot{x}^{des}(t) + K_p e(t) \tag{1.15}$$

where the first term on the right is called the "feedforward" term and the second is called the "feedback" term.

This type of control is known as proportional (P) control.

1.6.2 Second Order System Control

If our input is acceleration instead of velocity, our kinematic model is second order ($\ddot{x} = u$). Our strategy then is to find an appropriate input u for some solution to the error equation:

$$e(t) = x^{des}(t) - x(t) \tag{1.16}$$

$$\ddot{e} + K_d\dot{e} + K_p e = 0 \tag{1.17}$$

We can pick some $K_p, K_d > 0$ such that

$$u(t) = \ddot{x}^{des}(t) + K_d\dot{e}(t) + K_p e(t) \tag{1.18}$$

This control strategy is known as proportional-derivative (PD) control. The $K_p e(t)$ term is the "proportional" term (which you may notice was present in the previous first-order proportional control strategy), and the $K_d\dot{e}(t)$ is the "derivative" term.

1.6.3 Control for Trajectory Tracking

PID control is a further variation on PD control, via addition of an "integral term". This term is added to make steady state error go to 0, and accounts for model error or disturbances:

$$u(t) = \ddot{x}^{des}(t) + K_d\dot{e}(t) + K_p e(t) + K_l \int_0^t e(\tau)\, d\tau \tag{1.19}$$

PID is a commonly used control strategy across many applications.

1.6.4 Control Gains

The values that we choose for K_d, K_p, K_l change our system response (Figure 1.1). These values are called the "gains" of the system.

In the second order system example on Figure 1.2, K_p and K_d can be chosen such that a desired damping ratio is achieved. The system can be modeled as

$$\ddot{e} + K_d\dot{e} + K_p e = 0 \tag{1.20}$$

and rewritten to express its eigenvalues as

$$\ddot{e} + 2\zeta\omega_n\dot{e} + {\omega_n}^2 e = 0 \tag{1.21}$$

$$\lambda = -\omega_n(\zeta \pm i\sqrt{1-\zeta^2}) \tag{1.22}$$

In this example, we see that choosing a ζ value of 0.7 is an optimal value for the system.

In general, the time it takes for a system to respond to disturbances and return to a steady state can be measured as one indicator of the quality of the system.

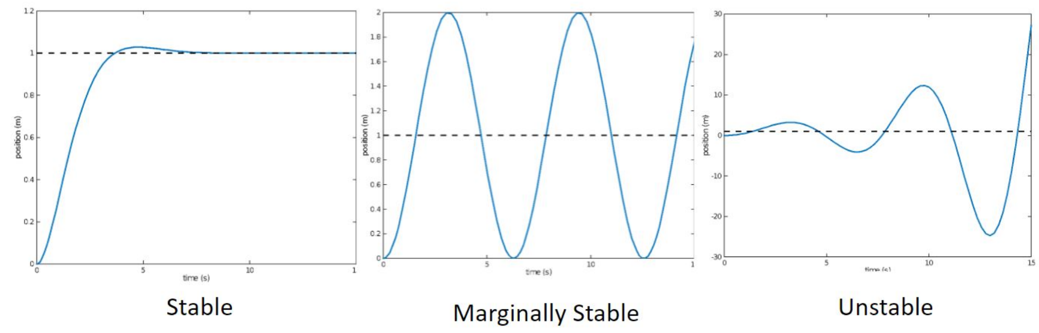


Figure 1.1: The position-time graphs of a stable system, a marginally stable system, and an unstable system.

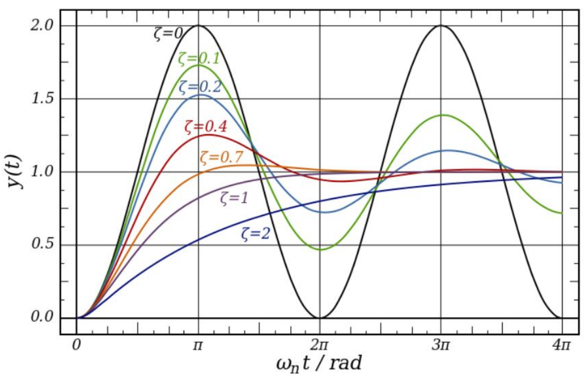


Figure 1.2: The response curves of a system with different ζ values.