# Introduction to Control Theory

The material is absed on the Control Theory course, Innopolis, Spring 2024

May 15, 2024

## Contents

# 1 Introduction

**Dynamical systems**

Let's consider the $n$-th order ordinary differential equation (ODE):

$$\mathbf{x}^{(n)} = \mathbf{f}(\mathbf{x}^{(n-1)}, \mathbf{x}^{(n-2)}, ..., \ddot{\mathbf{x}}, \dot{\mathbf{x}}, \mathbf{x}, t),$$

where $x(t)$ is a solution for the system, and $t$ is an independent variable (usually time). This equation represents the dynamics of the system and it is called a **dynamical system**.
x is called the **state** of the dynamical system.

In canonical form, linear ODE is represented in the following way:

$$a_n z^{(n)} + a_{n-1} z^{(n-1)} + ... + a_2 \ddot{z} + a_1 \dot{z} + a_0 z = b_0$$

The set $\{\mathbf{x},\ \dot{\mathbf{x}}\ ...,\ \mathbf{x}^{(n-1)}\}$ is called the **state of the system**.

**State** of the system is a minimal set of variables that describe the system. Based on the current state and future inputs, we can predict the behaviour of the system.

## 1.1 Introducing input

General form of an n-th order linear ODE with an input can be presented as follows:

$$a_n y^{(n)} + ... + a_2 \ddot{y} + a_1 \dot{y} + a_0 y = u(t) \tag{1}$$

State-space representation of a linear system with an input is:

$$\dot{x} = Ax + Bu \tag{2}$$

$A$ is called a **state matrix** and x is a **state vector**,
$B$ is called a **control matrix** and $u$ is a **control vector**.
$u$ might be either a scalar or a vector.

## 1.2 Introducing output

Equations might also have an output, which can have plenty of physical meanings and interpretations. Let's list some of them: what we measure (position and orientation of a motor), what we want to control (the height of the quadrotor).

Output is usually defined as $y$.

Example of system wth input and output:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\ \mathbf{y} = \mathbf{C}\mathbf{x} \end{cases} \tag{3}$$

If $u$ and $y$ are scalars, the system is called *single-input single-output (SISO)*, if they are vectors - *multi-input multi-output (MIMO)*.

### Linear systems

In case if relationships between state, output and control are linear, we can model system in following form:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\ \mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \end{cases} \tag{4}$$

Where

- $\mathbf{x} \in \mathbb{R}^n$: states of the system
- $\mathbf{y} \in \mathbb{R}^l$: output vector

- $\mathbf{u} \in \mathbb{R}^m$: control inputs
- $\mathbf{A} \in \mathbb{R}^{n \times n}$: state matrix
- $\mathbf{B} \in \mathbb{R}^{n \times m}$: input matrix
- $\mathbf{C} \in \mathbb{R}^{l \times n}$: output matrix
- $\mathbf{D} \in \mathbb{R}^{l \times m}$: feedforward matrix

If matrices A, B, C, D are time-independent, then we call such systems **time-invariant.** More frequently we work with systems when the output does not depend on control.

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\ \mathbf{y} = \mathbf{C}\mathbf{x} \end{cases} \tag{5}$$

## 1.3 ODE to State-Space conversion

$$\dddot{y} + a_2\ddot{y} + a_0 y = u$$

$$\begin{bmatrix} \dot{y} \\ \ddot{y} \\ \dddot{y} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \\ \ddot{y} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ u \end{bmatrix}$$

## 1.4 State-Space to ODE conversion

$$\begin{cases} \dot{x} = Ax \\ y = Cx \end{cases}$$

We need to represent this system as an ODE in the form:

$$y^{(n)} = d_{n-1}y^{(n-1)} + d_{n-2}y^{(n-2)} + \cdots + d_1\dot{y} + d_0$$

Let's take the derivative of $y$:

$$\dot{y} = C\dot{x} = CAx$$

...

$$y^{(n)} = CA^{(n)}x$$

$$y = \begin{bmatrix} C \\ CA \\ \dots \\ CA^{(n-1)} \end{bmatrix} x = Ox,$$

$O$ is called the **observability matrix**.

$$x = O^{-1}y$$

Then,

$$y^{(n)} = CA^{(n)}x = CA^{(n)}O^{-1}y = CA^{(n)}O^{-1} \begin{bmatrix} y \\ \dot{y} \\ \vdots \\ y^{(n-1)} \end{bmatrix}$$

# 2 Stability

## 2.1 Critical Point (Node)

**Critical Point (Node)**

Consider the following LTI:

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

$x_0$ is called a **Node**, or **Critical Point**, if $f(x_0) = 0$.

## 2.2 Stability

A system is **stable** if:

$$\|x(0) - x_0\| < \delta \quad \|x(t) - x_0\| < \epsilon$$

We can think of it as: if the starting point is in the $\delta$-neighborhood of the node $x_0$, the rest of the trajectory $x(t)$ is in the $\epsilon$-neighborhood of the node.

Or, the solutions starting from the $\delta$-sized ball do not diverge.

### Asymptotic Stability

A system is **asymptotically stable** if:

$$\|x(0) - x_0\| < \delta \rightarrow \lim_{t \to \infty} x(t) = x_0$$

For any initial point that lies in the $\delta$-sized ball, the trajectory will asymptotically approach the node (point $x_0$). Or, the solutions starting from the $\delta$-sized ball, converge to the node.

## 2.3 Stability of autonomous LTI

### Autonomous Systems

A system is considered **autonomous** if its evolution depends only on time.

Example

$$\dot{x} = Ax$$

#### 2.3.1 Diagonal matrices

Let's introduce a trick for autonomous Linear Time-Invariant (LTI) systems.

First of all, recall the properties of a diagonal matrix and eigen-decomposition.

- **Diagonal matrix**

$$\dot{z} = Dz$$

$$\begin{cases} \dot{x}_1 = d_1 x_1 \\ \dots \\ \dot{x}_n = d_n x_n \end{cases}$$

  The solution of each of the equations is: $x_i = C_i e^{d_i t}$. So, the system is asymptotically stable when for all $i$, $d_i < 0$. The system is stable when $d_i \leq 0$.

- **Eigen-decomposition**
  We can represent the matrix as $A = VDV^{-1}$, where $D$ is a diagonal matrix.

Given an autonomous Linear Time-Invariant (LTI) system, let's switch to the system with a diagonal matrix:

$$\dot{x} = Ax$$

$$\dot{x} = VDV^{-1}x$$

$$V^{-1}\dot{x} = V^{-1}VDV^{-1}x = DV^{-1}x$$

Change of variables: $z = V^{-1}x$, $\dot{z} = Dz$,

$$V^{-1}\dot{x} = Dz$$

The system is asymptotically stable when for all the elements of $D$ are $< 0$. The system is stable when the elements of $D$ are $\leq 0$.

### 2.3.2 Upper triangular matrices

Eigenvalues of upper triangular matrices are the diagonal elements.

### 2.3.3 General case

Consider the LTI:

$$\dot{x} = Ax$$

The system is called **stable** iff real parts of eigenvalues of A are non-positive.

The system is called **asymptotically stable** iff real parts of eigenvalues of A are strictly negative.

# 3 Control

The main tasks of control theory include **control design**, **trajectory tracking**, and **point-to-point control**.

## 3.1 Control Design

**Stabilizing control**

The task of **stabilizing control** is defining the control law that makes a certain solution of a dynamical system stable.

This is true for both linear and nonlinear systems.
Consider a Linear Time-Invariant (LTI) system:

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

and choose the control as a linear function of the state:

$$u = -Kx,$$

where $K$ is the **control gain**.

***Stability Condition***

Then, the closed-loop system can be represented as:

$$\dot{x} = (A - BK)x.$$

The system is asymptotically stable if the eigenvalues of the matrix $A - BK$ have strictly negative real parts.
Or, matrix $(A - BK) \in \mathcal{H}$ should be Hurwitz.

## 3.2 Trajectory Tracking

The task is to stabilize the system around a reference trajectory.
Let $x^*(t)$ and $u^*(t)$ be solutions for the system $\dot{x} = Ax + Bu$. This means that:

$$\dot{x}^* = Ax^* + Bu^*.$$

Define the error as $e = x^* - x$ and $v = u^* - u$.
Then, the error dynamics become:

$$\dot{e} = Ae + Bv.$$

To stabilize the system, suggest $v = -Ke$, then:

$$\dot{e} = (A - BK)e.$$

Therefore, the control law becomes:

$$u = u^* + K(x^* - x).$$

## 3.3 Point-to-Point Control

Point-to-point control differs from trajectory tracking in that the reference input is constant, $x^* = \text{const}$, and feed-forward control is also constant, $u^* = \text{const}$.
Since the error dynamics and the stabilizing control are the same as in trajectory tracking, the control law becomes:

$$u = K(x^* - x) + u^*.$$

The dynamics of the system become:

$$\dot{x} = (A - BK)x + BKx^* + Bu^*.$$

# 4 Laplace Transform

## 4.1 Laplace Transform

The Laplace transform is an integral transformation that converts a function of a real variable $t$ (time domain) to a function of a complex variable $s$ (frequency domain).

We can think of Laplace transform ass as general case of Fourier transform (Steve Brunton).

The Laplace transform is a tool for solving differential equations by transforming them into algebraic equations.

**The Laplace transform** of a function $f(t)$ is given as:

$$F(s) = \mathcal{L}\{f(t)\} = \int_0^\infty f(t)e^{-st}dt \tag{6}$$

where $F(s)$ is called an **image** of the function and $s = \alpha + \beta i$ is a complex frequency.

### 4.1.1 Some Useful Properties

Linear properties:

$$\mathcal{L}\{f(t) + g(t)\} = \mathcal{L}\{f(t)\} + \mathcal{L}\{g(t)\} \tag{7}$$

$$\mathcal{L}\{af(t)\} = a\mathcal{L}\{f(t)\} \tag{8}$$

Final value theorem:

$$f(\infty) = \lim_{s\to 0} sF(s) \tag{9}$$

The final value theorem is useful because it gives the long-term behavior for a particular function.

### 4.1.2 Inverse Laplace Transform

The inverse Laplace transform transforms the image of your function $F(s)$ from the frequency domain to the time domain $x(t)$:

$$f(t) = \mathcal{L}^{-1}\{F\}(t) = \frac{1}{2\pi i}\lim_{T\to\infty}\int_{\gamma-iT}^{\gamma+iT} e^{st}F(s)\,ds \tag{10}$$

However, in practice, we mostly use precalculated Laplace transforms and then try to decompose the image $X(s)$ into known transforms of functions obtained from a table, and construct the inverse by inspection, or just use some symbolic routines.

## 4.2 Laplace Transform of a Function's Derivative

For us, one of the most useful properties of Laplace transform is that if we apply it to the derivative of a given variable, it will result in the following:

$$\mathcal{L}\left\{\frac{dx(t)}{dt}\right\} = s\mathcal{L}\left(x\right) = sX(s) \tag{11}$$

which is true for $x(0) = 0$.

Thus, we can define a **derivative operator**:

$$\frac{dx}{dt} \xrightarrow{\mathcal{L}} sX(s) \tag{12}$$

The proof is as follows, using the definition of Laplace transform:

$$\mathcal{L}\left\{\frac{dx}{dt}\right\} = \int_0^\infty \frac{dx}{dt}e^{-st}dt \tag{13}$$

Then using integration by parts:

$$\int_0^\infty \frac{dx}{dt}e^{-st}dt = \left[xe^{-st}\right]_0^\infty - \int_0^\infty -se^{-st}xdt \tag{14}$$

which yields:

$$\left[xe^{-st}\right]_0^\infty + s\int_0^\infty e^{-st}xdt = x(0) + s\mathcal{L}\{x(t)\} = x(0) + sX(s) \tag{15}$$

By induction, it can be shown that:

$$\mathcal{L}\left\{\frac{d^n x}{dt^n}(t)\right\} = s^n \cdot \mathcal{L}\{x(t)\} + s^{n-1}x(0) + \cdots + x^{(n-1)}(0) \tag{16}$$

### 4.2.1 Applications to Linear ODEs

Let us consider the following ODE:

$$a_n x^{(n)} + a_{n-1}x^{(n-1)} + ... + a_2\ddot{x} + a_1\dot{x} + a_0 x = u_m b^{(m)} + b_{m-1}u^{(m-1)} + ... + b_2\ddot{u} + b_1\dot{u} + b_0 u \tag{17}$$

Notice that we introduce a new variable that we call the input $u$ (control).

Applying the inverse Laplace transform with zero initial conditions yields:

$$\begin{aligned} &a_n s^{(n)}X(s) + a_{n-1}s^{(n-1)}X(s) + \cdots + a_2 s^2 X(s) + a_1 sX(s) + a_0 X(s) \\ &= b_m s^{(m)}U(s) + b_{m-1}s^{(m-1)}U(s) + \cdots + b_2 s^2 U(s) + b_1 sU(s) + b_0 U(s) \end{aligned} \tag{18}$$

**Example 1**

$$\dddot{y} + a\dot{y} + by = u \tag{19}$$

$$S^2Y(S) + ASY(S) + BY(S) = U(S) \tag{20}$$

$$Y(S) = \frac{1}{(S^2 + AS + B)}U(S) \tag{21}$$

This form is called a **transfer function**.

**Example 2**

$$2\dddot{y} + 5\dot{y} - 40y = 10u \tag{22}$$

$$2SY(S) - 4Y(S) = U(S) \tag{23}$$

$$Y(S)(2S - 4) = U(S) \tag{24}$$

$$Y(S) = \frac{1}{(2S - 4)}U(S) \tag{25}$$

Consider the ODE with the input $u$.

## 4.3 State-Space to Transfer Function

Let's study the relation between the input and the output of the dynamical system.

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases}$$

Let's consider the 1st equation and rewrite it in the form:

$$SIX(S) = AX(S) + BU(S)$$

$$(SI - A)X(S) = BU(S)$$

$$X(S) = (SI - A)^{-1}BU$$

Then the initial system can be rewritten in the following form:

$$\begin{cases} X = (SI - A)^{-1}BU \\ Y = C((SI - A)^{-1}B + D)U \end{cases}$$

## 5 Bode

By steady-state we mean that initial conditoins have stopped playing a role. After some time passed we expect the output of the system to not depend on the initial conditions, but on the input.

Consider a system:

$$Y(s) = G(s)U(s),$$

$G(s)$ is a transfer function, $U(s)$ is a Laplace space input.

$u(t) = sin(\omega t)$ in time domain takes form $\frac{\omega}{\omega^2+s^2}$ in Laplace domain.

$$Y(s) = G(s)\frac{\omega}{\omega^2 + s^2}$$

,

If a transfer function is a rational fraction, it can be reprresented in the following way:

$$G(s) = \frac{n(s)}{(s+p_1)(s+p_2)+\cdots+(s+p_n)} = \frac{r_1}{s+p_1} + \cdots + \frac{r_n}{s+p_n}$$

$p_i$ in this equations are the **poles**.

Previously, we introduced the stability analysis based on eigenvalues, but there is an equvivalent analysis based on poles.

A Bode plot usually consists of magnitude and phase response of a transfer function.

Transfer functions in s-domain quickly become cumbersome to analyse as the control system gets complicated. It's easy to understand the critical properties of the system by looking at the Bode plot.

# 6 Discrete dynamics

Discrete case for dynamics:

$$x_{i+1} = Ax_i + Bu_i$$

Such an equation is easy to work with: no derivatives are used here, it is easy to simulate.
For this system, we can propose the control:

$$u_i = -Kx_i + u_i*$$

## 6.1 Stability

Let's consider the system:

$$x_{i+1} = Ax_i$$

$$x_{i+1} = V^{-1}DVx_i$$

$$Vx_{i+1} = VV^{-1}DVx_i$$

The change of variables: $z_i = Vx_i$, $z_{i+1} = Vx_{i+1}$

$$z_{i+1} = DVz_i$$

Let's consider the following example:

$$\begin{bmatrix} \vec{x}_{1,i+1} \\ x_{2,i+1} \end{bmatrix} = \begin{bmatrix} \alpha & -\beta \\ \beta & \alpha \end{bmatrix} \begin{bmatrix} x_{1,i} \\ x_{2,i} \end{bmatrix}$$

Let's find the norms of $x_{i+1}$ and $x_i$.

$$\| \begin{bmatrix} x_{1,i} \\ x_{2,i} \end{bmatrix} \|^2 = x_{1,i}^2 + x_{2,i}^2$$

$$\| \begin{bmatrix} x_{1,i+1} \\ x_{2,i+1} \end{bmatrix} \|^2 = \left| \begin{bmatrix} \alpha x_{1,i} - \beta x_{2,i} \\ \beta x_{1,i} + \alpha x_{2,i} \end{bmatrix} \right|^2 = (\alpha x_{1,i} - \beta x_{2,i})^2 + (\beta x_{1,i} + \alpha x_{2,i})^2 = \beta^2(x_{2,i}^2 + x_{1,i}^2) + \alpha^2(x_{2,i}^2 + x_{1,i}^2) = (\alpha^2+\beta^2)(x_{1,i}^2 + x_{2,i}^2)$$

For stability we introduce:
**Continuous case:** Hurwitz matrix. **Discrete case:** Schur matrix.

**Dynamical Systems**

- Discrete system $x_{i+1} = Ax_i$ is **stable** if and only if the absolute values of $A$'s eigenvalues are less than or equal to 1: $|\lambda_i(A)| \le 1$.
- Discrete system $x_{i+1} = Ax_i$ is **asymptotically stable** if and only if the absolute values of $A$'s eigenvalues are less than 1: $|\lambda_i(A)| < 1$.

## 6.2 Analytical solution of Continuous LTI

**Case:** $\dot{x} = ax(t)$

Recall that we can describe exponent as a series:

$$e^a = 1 + a + \frac{a^2}{2} + \frac{a^3}{6} + \dots$$

$$e^A = I + A + \frac{AA}{2} + \frac{AAA}{6} + \dots$$

Suppose that the solution of the dynamical system is:

$$x(t) = e^{at}x(0)$$

$$x(t) = e^{At}x(0)$$

Let's rewrite the exponent:

$$x(t) = (I + At + \frac{AAt^2}{2} + \frac{AAAt^3}{6} + \dots)x(0)$$

$$\dot{x}(t) = (A + AAt + \frac{AAAt^2}{2} + \dots)x(0)$$

$$\dot{x}(t) = A(I + At + \frac{AAt^2}{2} + \dots)x(0)$$

$$\dot{x}(t) = Ae^{At}x(0)$$

$$\dot{x}(t) = Ax(t)$$

**Case:** $\dot{x} = ax(t) + bu(t)$

$$\dot{x} = ax(t) + bu(t)$$

$$\dot{x}e^{-at} - ae^{-at}x(t) = be^{-at}u(t)$$

$$\frac{d}{dt}(xe^{-at}) = be^{-at}u(t)$$

$$\int_0^t \frac{d}{d\tau}(e^{-a(t-\tau)}x(\tau))d\tau = \int_0^t \frac{d}{d\tau}(e^{-a(t-\tau)}bu(\tau))d\tau$$

$$e^{-at}x(t) - x(0) = \int_0^t e^{-a\tau}bu(\tau)d\tau$$

$$x(t) = e^{at}x(0) + e^{at}\int_0^t e^{-a\tau}bu(\tau)d\tau$$

$$x(t) = e^{at}x(0) + \int_0^t e^{a(t-\tau)}bu(\tau)d\tau$$

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

## 6.3 Discretization

$$\begin{cases} x_0 = x(0) \\ x_1 = x(\Delta t) \\ x_2 = x(2\Delta t) \\ \dots \\ x_n = x(n\Delta t) \end{cases}$$

$$\dot{x} = \frac{1}{\Delta t}(x_{i+1} - x_i)$$

$$Ax_i = \frac{1}{\Delta t}(x_{i+1} - x_i)$$

$$x_{i+1} = Ax_i\Delta t + x_i$$

$$\boxed{x_{i+1} = (A\Delta t + I)x_i}$$

or

$$\boxed{x_{i+1} = (I - A\Delta t)^{-1}x_i}$$

Thus, the discrete state matrix is: $A_d = A\Delta t + I$ The control matrix is: $B\Delta t$.

# 7 LQR

## 7.1 Intuition behind poles

In control theory, the system state-space equation

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

$$y = Cx + Du$$

has the transfer function

$$G(s) = C(sI - A)^{-1}B + D.$$

Since $(sI - A)^{-1} = \operatorname{adj}(sI - A)\det(sI - A)$, where $\operatorname{adj}(sI - A)$ is the adjugate of $sI - A$, the poles of $G(s)$ are the numbers that satisfy $\det(sI - A) = 0$. This is exactly the characteristic equation of matrix $A$, whose solutions are the eigenvalues of $A$.

## 7.2 LQR

In pole-placement method, we want to place the poles in the specific spots (or, we choose specific eigenvalues). But it is not intuitive where to place them, especially for complex systems, systems with numerous actuators. So, the new method is proposed. The key concept of the method lies in optimization of choosing $K$.

In LQR we find an optimal $K$ by choosing parameters that are important to us, specifically how well the system performs and how much effort it takes to reach this performance.

If $Q >> R$, then we are turning the problem of Let $J$ be an additive cost function:

$$J(x_0, p(x,t)) = \int_0^{\infty} g(x,u)$$

$Q$ - how bad if $x$ is not where it is supposed to be. $Q$ - nonnegative, positive semidefinite.

if the system is a positions, velocity, and $Q = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}$ we penalize for

Suppose there is the best control law:

$$u = -kx$$

that minimizes the quadratic cost function.

$$J = \int x^T Q x + u^T R u$$

**Hamiltonian-Jacobi-Bellman (HJB)**

$$\min_u [g(x,u) + \frac{dJ}{dx f(x,u)}] = 0$$

Cost on effectiveness and energy to reach this effectiveness.

## 7.3 Subtopic 2.1

## 7.4 Subtopic 2.2

# 8 Observer

When the full state feedback is unavailable, we introduce an observer to estimate the state $x$.

We already know how to create a system with a controller, but how to check the current state of the system?

$$\begin{cases} \dot{x} = Ax + Bu \\ u = -Kx \end{cases}$$

We can try to estimate it with measurements, for example with sensors. But in real life, the task is not that trivial due to some problems:

1. Lack of sensors. For a quadrotor, we cannot measure the height straightforwardly.
2. Measurements can be imprecise or biased.
3. Measurements can be only made in discrete time.

The key problem arises when the output of the system $y$ is not the whole state $x$, but $y = Cx$, which means that we are able to get the state partially.

These are just several problems that create difficulties for us to measure the state of the system.

## 8.1 Measurement and estimation

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \\ \hat{x}(t) = \text{estimate}(y(t)) \\ u = -K\hat{x} \end{cases}$$

$x$ and $y$ are the state and output (actual or true). When we do not know the exact state $x$, we can only estimate it. We estimate $x$ ($\hat{x}$) based on the history of $y$ values. The control law is now governed by the estimated state $\hat{x}$.

***Estimation error***

State estimation error is the following:

$$\epsilon = \hat{x} - x$$

## 8.2 Dynamics estimation

We can always find $y = C\hat{x} - y$.

$$y = C\hat{x} - y = C\hat{x} - Cx = C\epsilon$$

Let's suggest that the dynamics should also hold for the observed state:

$$\dot{\hat{x}} = A\hat{x} + Bu$$

Let's introduce in our equation a linear correction law $-L\,y$. Since $y = C\hat{x} - y$, we get:

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - C\hat{x}) \tag{26}$$

This is called the Luenberger observer.

But how to find a suitable observer gain $L$?

Let's subtract $\dot{x} = Ax + Bu$ from Equation (26) . The equation we got is the observer error dynamics:

$$\dot{\hat{x}} - \dot{x} = A\hat{x} - Ax + L(y - C\hat{x}) \tag{27}$$

$$\dot{\hat{x}} - \dot{x} = A\hat{x} - Ax + L(Cx - C\hat{x}) = A\hat{x} - Ax + LC(x - \hat{x}) \tag{28}$$

$$\dot{\epsilon} = A\epsilon - LC\epsilon = (A - LC)\epsilon \tag{29}$$

With no knowledge of $x$ nad $\hat{x}$, we can define the stability of the system.

What we want is the error converging to 0. To obtain this, the observer $\dot{\epsilon} = (A - LC)\epsilon$ needs to be stable. $A - LC \in \mathcal{H}$.

Recall:

- Controller design: find such $K$ that $A - BK \in \mathcal{H}$.
- Observer design: find such $L$ that: $A - LC \in \mathcal{H}$

But now the gain $L$ is in the left side for the observer (unlike $K$ for the controller), so we cannot use any stabilization methods (LQR, pole placement) right away.

We need to introduce the following change (or, we can solve the *dual problem*): find such L that:

$$A^T - C^T L^T \in \mathcal{H}$$

ANd now, for this equation we can use LQR or poole-placement.

## 8.3 Observer + Controller

$$\begin{cases} \dot{x} = Ax + BK\hat{x} \\ \dot{\hat{x}} = A\hat{x} - BK\hat{x} + LC(x - \hat{x}) \end{cases}$$

$$\begin{cases} \dot{x} = Ax + BK\hat{x} \\ \dot{\hat{x}} = A\hat{x} - BK\hat{x} - LC(\hat{x} - x) \end{cases} \tag{30}$$

Let's do a change of variables so that it would be easier to analyze the eigenvalues: $e = x - \hat{x}$, $\dot{e} = \dot{x} - \dot{\hat{x}}$. Let's subtract (1) of 30 from (2) of 30:

$$\dot{e} = A\hat{x} - BK\hat{x} - LC(\hat{x} - x) - Ax + BK\hat{x}$$
$$\dot{e} = Ae - LCe$$

$$\dot{e} = (A - LC)e \tag{31}$$

Now let's turn back to the equation 1 of the system and rewrite it.

$$\dot{x} = Ax - BK\hat{x} = Ax - BK(x - e)$$

$$\dot{x} = (A - BK)x + BKe \tag{32}$$

$$\begin{cases} \dot{e} = (A - LC)e \\ \dot{x} = (A - BK)x + BKe \end{cases}$$

$$\begin{bmatrix} \dot{x} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix} \begin{bmatrix} x \\ e \end{bmatrix}$$

The eigenvalues of the system are the union of eigenvalues of $(A - BK)$ and $(A - LC)$.

# 9 Filters

In literature, the system is frequently called a Plant, which in our case is actually a robot.

**Static system**

Linear systems for which the relation between input and output is constant are called **static systems**.

Example
(Laplace domain) $Y(s) = 10X(s)$.
Note that when we are talking about independency on time not the signal, but the relation has to be time-independent.

**Dynamic system**

Dynamic systemns are linear systems for whhich the relation between input and output depends on time.

Dynamic system Example 1 State-space plant:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases}$$

Apparently, the output $y$ does not depend on the input $u$.
Example 2
Laplace plant: $Y(S) = \frac{1}{s^2+2s+7}X(s)$ Time does not appear in this equaiton, but if we rewrite it into ode form:

$$\ddot{y} + 2\dot{y} + 7y = x$$

Example 3 ODE plant:

$$\ddot{y} + 5\dot{y} + y = u$$

For these systems, the output depends not only on the current (time-wise) value of input, but on the entire history of input values.
Static systems form algebraic linear equaitons. Dynamical systems create linear differential equaitons
Dynamic systems have **state**, which changes over the time.
For the system we can create the controller and Luenbberger observer:

# 10 Controllability. Observability

Not always can we control the system. The same applies to an observer. Sometimes, we cannot obtain the full information that we desire from the system.

Let's suggest the tools which can inform us whether we can succeed in designing a controller or observer or not.

**Controllability**

A system is controllable on $t_0 \leq t \leq t_f$ if it is possible to find a control input $u(t)$ that would drive the system to a desired state $x(t_f)$ from any initial state $x_0$.

**Observability**

A system is observable on $t_0 \leq t \leq t_f$ if it is possible to exactly estimate the state of the system $x(t_f)$, given any initial estimation error.

**Observability (Alternative)**

A system is observable on $t_0 \leq t \leq t_f$ if any initial state $x_0$ is uniquely determined by the output $y(t)$ on that interval.

**Cayley-Hamilton Theorem**: A matrix $M$ satisfies its own characteristic equation.

A characteristic equation can be written like:

$$\lambda^n + a_{n-1}\lambda^{n-1} + a_{n-2}\lambda^{n-2} + \ldots + a_0 = 0$$

So,

$$M^n + a_{n-1}M^{n-1} + a_{n-2}M^{n-2} + \ldots + a_0 I = 0$$

Meaning that $M^n$ is a linear combination of $M^{n-1} + M^{n-2} + \ldots + I$. Remember that if the system is controllable, we would be able to find such controller to the system, whcih would make the system stable, which means that the error goes to zero.

## 10.1 Controllability of Discrete LTI

Consider discrete LTI.

$$x_{i+1} = Ax_i + Bu_i$$

Let's expand it numerically:

$$x_2 = Ax_1 + Bu_1$$

$$x_3 = Ax_2 + Bu_2 = A(Ax_1 + Bu_1) + Bu_2 = A^2x_1 + ABu_1 + Bu_2$$

$$\ldots$$

$$x_{n+1} = A^n x_1 + A^{n-1}Bu_1 + ... + A^{n-k}Bu_k + ... + Bu_n$$

$$x_{n+1} - A^n x_1 = \begin{bmatrix} B & AB & A^2B \end{bmatrix} \begin{bmatrix} u_n \\ u_{n-1} \\ \vdots \\ u_1 \end{bmatrix}$$

Thus, we showed that $x_{n+1} - A^n x_1$ must be in the column space of $C = \begin{bmatrix} B & AB & A^2B \end{bmatrix}$

Now, note that since $x_{n+1}$ can be any real number and $x_1$ can take a zero value, $\mathbb{R}$ must be in the column space of C. So, C must be a full-rank matrix.

**Controllability**

A system $x_{i+1} = Ax_i + Bu_i$ is controllable (or any state can be reached) if $C = \begin{bmatrix} B & AB & A^2B \end{bmatrix}$ is full row rank.

## 10.2 Observability of Discrete LTI

Observability criterion shows whether it is in principle possible to estimate the system or not.

Consider a discrete LTI:

$$\begin{cases} x_{i+1} = Ax_i + Bu_i \\ y_i = Cx_i \end{cases}$$

And a Luenberger observer:

$$\hat{x}_{i+1} = A\hat{x}_i + Bu_i + L(y_i - C\hat{x}_i)$$

Change of variables: $e_i = \hat{x}_i - x_i$ will provide us with the following equation:

$$e_{i+1} = Ae_i + L(y_i - C\hat{x}_i)$$
$$e_{i+1} = Ae_i - LCe_i$$

Introduce the dual system (which is stable iff the original system is stable):

$$\epsilon_{i+1} = A^T\epsilon_i - C^TL^T\epsilon_i$$

$$\begin{cases} \epsilon_{i+1} = A^T\epsilon_i - C^Tv_i \\ v_i = -L^T\epsilon_i \end{cases}$$

Controllability matrix for the system:

$$O^T = \begin{bmatrix} C^T & A^TC^T \dots & {A^T}^{n-1}C^T \end{bmatrix}$$
$$O = \begin{bmatrix} C \\ CA \\ \dots \\ CA^{n-1} \end{bmatrix}$$

**Observability**

A system $x_{i+1} = Ax_i + Bu_i$ is **observable** (or observation error can go to 0 from any initial position) if $O = \begin{bmatrix} C \\ \dots \\ CA \\ CA^{n-1} \end{bmatrix}$ is full column rank.

# 11 Kalman Filter

## 12 Reference materials

(Controllability. Observability)

# 13 Linearization

While linear models are present in the real world (examples include DC motors, 3D printers, and mass-spring-damper systems), the majority of real-world systems are nonlinear (such as cars, airplanes, robot arms, underwater robots, and almost any mechanical system with linkages).

However, there is a method to describe these systems using our toolkits - through linearization.

**Linearization** is the technique that enables us to create a linear model for a nonlinear system.

So, we can introduce linear models for nonlinear systems that could describe the system pretty well locally. We can do control for this local approximation.

## 13.1 Taylor expansion

***Taylor expansion around node.***

Let's consider a non-linear dynamical system $\dot{x} = f(x, u)$ and a Taylor expansion around the node $(x_0, u_0)$, i.e., $f(x_0, u_0) = 0$, $x_0 = \text{const}$, $u_0 = \text{const}$.

Recall the physical meaning of a **node**: it is any position at which the robot remains static.

$$f(x, u) \approx f(x_0, u_0) + \frac{\partial f}{\partial x}(x - x_0) + \frac{\partial f}{\partial u}(u - u_0) + \text{HOT}$$

HOT = high-order terms.

$(x_0, u_0)$ are an expansion point.

Now let's introduce new variables $e$ and $v$ that represent the distances from the expansion point:

$$e = x - x_0$$

$$\dot{e} = \dot{x}$$

$$v = u - u_0$$

Then we can rewrite the Taylor expansion:

$$\dot{e} \approx \frac{\partial f}{\partial x} e + \frac{\partial f}{\partial u} v + \text{HOT}$$

$$\dot{e} \approx Ae + Bv + \text{HOT}$$

Let's drop the high-order term and get an approximation. And this approximation is called linearization.

$$\dot{e} = Ae + Bv$$

In this context, $(x_0, u_0)$ is a linearization point.

This type of linearization is proved to provide the best possible linear model for the given system.

***Taylor expansion along a trajectory.***

Now let's consider a non-linear dynamical system:

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

and a trajectory $x_0 = f(x_0, u_0)$. Note that now $x_0$ is not a point - it is a whole trajectory

$$f(x, u) \approx f(x_0, u_0) + \frac{\partial f}{\partial x}(x - x_0) + \frac{\partial f}{\partial u}(u - u_0) + \text{HOT}$$

We can denote $\dot{e} = \dot{x} - \dot{x}_0 = f(x, u) - f(x_0, u_0)$.

The rest stays the same: $e = x - x_0, v = u - u_0$

$$\dot{e} \approx Ae + Bv + \text{HOT}$$

Let's drop HOT and obtain linearization:

$$\dot{e} = Ae + Bv$$

Basically, nothing changes between expansion around a node and expansion along a trajectory as long as we introduce the change of variables described above. The change of variables is slightly different, but original function and the local approximation behave in the same way in the region of the linearization point.

So, can we linearize around each and any point?

1) If it is a node - yes.

2) If we want to linearize around some other point - yes, but the chnage of variables would entail slightly different results.

While we had:

$$e = x - x_0, \dot{e} = \dot{x},$$

now $\dot{e}$ is the differrence between the derivative of the state of our non-linear system and derivative of the state of our trajectory distance from the trajectory linearization.

The meaning of the variables changes slightly, but the mathematical expression is the exact same expression.

***Affine expansion***

The other way to obtain linearization, without change of varibales is as follows.

$$f(x, u) \approx f(x_0, u_0) + A(x - x_0) + B(u - u_0)$$

Denoting

$$f(x_0, u_0) - Ax_0 - Bu_0 = c$$

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

c makes it not a linear model, but affine, which means it has a constant term.

It can be a constant in the case of a node, and a function of time in the case of expansion along the trajectory.

Often we choose u to compensate c, so u will also be affine in this case.

## 13.2 Manipulator equations

Frequently, in robotics we deal with the following equation when describing a system:

$$H\ddot{q} + C\dot{q} + g = \tau$$

which is called a **manipulator equation**

However, cars, underwater robots are usually described by the other equations - Euler or Lagrangian equaitons.

Let's derive linearization for the manipulator equation.

Begin by proposing new variables:

$$x = q \vec{-} q_0, \dot{q} \vec{-} \dot{q}_0$$

$$u = \tau - \tau_0$$

Points around which we linearize are denoted from equation:

Let's introduce

$$\phi(\dot{q}, q, t) = H^{-1}(\tau - C\dot{q} - g)$$

# 14 Lyapunov theory

Stability is not defined only for linear systems, but for nonlinear systems as well.

There exist a small neighbourhood. Every solution that starts from this neighbourhood will asymptotically approach the same solution.

**Asymptotic stability criteria:**
Autonomous dynamic system $\dot{x} = f(x)$ is asymptotically stable if there exists a scalar function $V = V(x) > 0$.

Note that computing $\dot{V}(x)$ comes along with computing $\dot{x}$. We can think of $V(x)$ as energy (pseudo-energy).

Asymptotic stability means that the system converges, and marginal stability means that the system does not diverge.

**Asymptotic stability**

The system is called asymptotically stable if there exists a function $V > 0$ such that $\dot{V} < 0$, except for $V = 0, \dot{V} = 0$.

**Marginal stability**

The system $\dot{x} = f(x)$ is stable in the sense of Lyapunov if there exists $V > 0$ such that $\dot{V} \leq 0$.

**Lyapunov function**

A function $V > 0$ in this case is called a Lyapunov function.

**Example 1** Consider the following system:

$$\dot{x} = -x$$

Let's introduce a function $V = x^2 > 0$ for all $x \neq 0$. $\dot{V} = 2x \cdot \dot{x} = 2x(-x) = -2x^2 < 0$. $V$ satisfies the Lyapunov criteria, so the system is (asymptotically) stable.

**Example 2** Consider the equation of the pendulum:

$$\ddot{q} = -\dot{q} - \sin(q)$$

## 14.1 LaSalle's Principle

The system is called stable if

LaSalle's principle allows us to prove asymptotic stability even for $\dot{V}(x) \leq 0$ only for the trivial solution.

## 14.2 Linear case

Lyapunov theory applies for both nonlinear and linear systems. For linear systems the following feature exist:

For a system $\dot{x} = Ax$, we can always choose a Lyapunov candidate function in the form:

$$V = x^T S x,$$

where S is positive-definite

$$\dot{V} = \dot{x}^T S x + x^T S \dot{x} = (Ax)^T S x + x^T S A x = x^T A^T S x + x^T S A x = x^T (A^T S + S A) x$$