Chapter 9: Introduction to Control

July 24, 2015

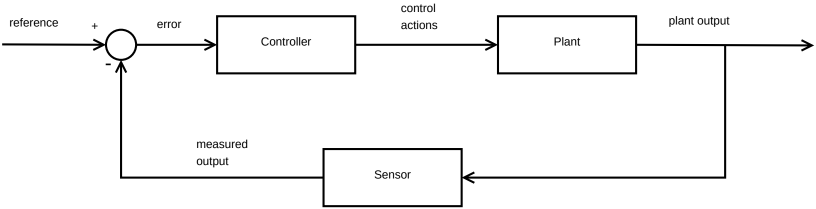
Outline

An introduction to control

Outline

What is control theory?

Control theory is an interdisciplinary branch of engineering and mathematics that deals with the behavior of dynamical systems with inputs, and how their behavior is modified by feedback. The usual objective of control theory is to design a controller to produces inputs to the plant so its output follows a desired reference signal which may be a fixed or changing value. - wikipedia



Controllers

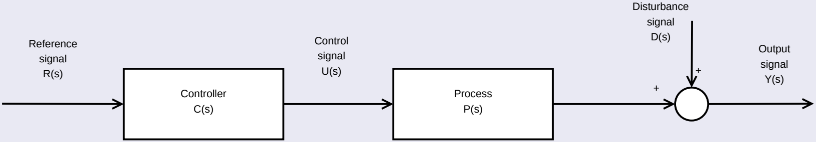
Types of controllers

- On-off controller
  - e.g., Thermostate at home
- **PID controllers, Lead and lag compensators (this course)**
  - Cruise-control in your car
  - Temperature, level, flow, presssure, pH, ... in chemical plants.
- More advanced controllers
  - State-space feedback controllers (e.g., LQR)
  - Model Predictive Controller (MPC)
  - Fuzzy Control
  - Neuro-fuzzy Control
  - ...

Open-loop System

Definition

In an open loop control system the actual output signal $Y(s)$ has no effect on the control action.



$$Y(s) = P(s)U(s) = P(s)C(s)R(s)$$

Open-loop System

Example

- You are pouring a glass of water, but you **cannot look at the glass**.
- The desired output is a full glass of water within a reasonable time.
- The input can have two values: on or off (assume a quite primitive tap).
- It will not be easy to do this successfully.

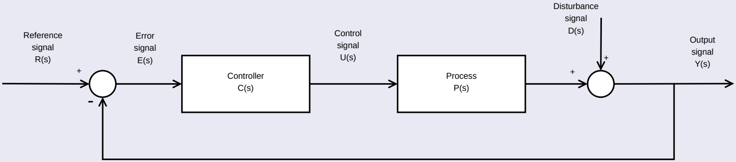
The solution is evident: look at the glass while pouring!

A general set-up of a closed loop system

Definition

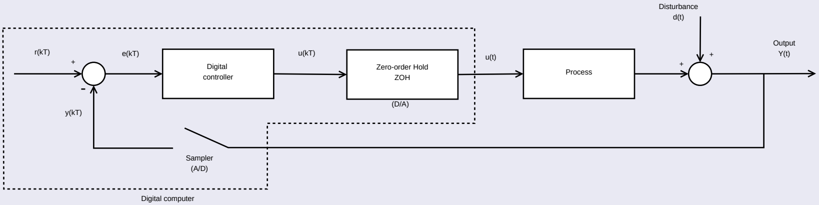
In a closed-loop system the output of the controller is influenced by the output of the system using a negative feedback loop.

Classical control-loop

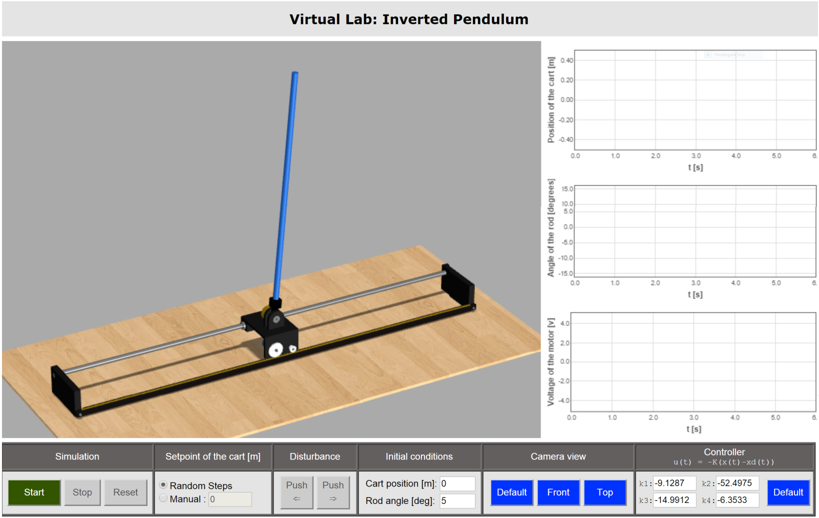


$$\frac{Y(s)}{R(s)} = \frac{P(s)C(s)}{1+P(s)C(s)}R(s) + \frac{1}{1+P(s)C(s)}D(s)$$

Digital Control Loop



Outline



KU Leuven/ESAT
Oscar Mauricio Agudelo
Bart De Moor

Outline

What is good control?

Depends on the application

- Stability
- Disturbance rejection
- Reference tracking
- Sensitivity to errors in the model
- ...

Outline

Examples: stability

Space shuttles are like inverted pendulums. Control systems make sure they do not flip over.

Outline

Examples: Disturbance rejection

- Your body will try to keep your internal temperature as constant as possible, no matter how hot/cold it is outside.

Flickr.com
tent86
Marathon Des Sables 046

Jack Zalium
Enduring
Creative Commons

Outline

Examples: Reference tracking

Audi has a system for automatic driving in traffic jams. The audi will follow the car in front of him at an appropriate distance.
https://www.youtube.com/watch?v=Qa_ZSRj0WM0

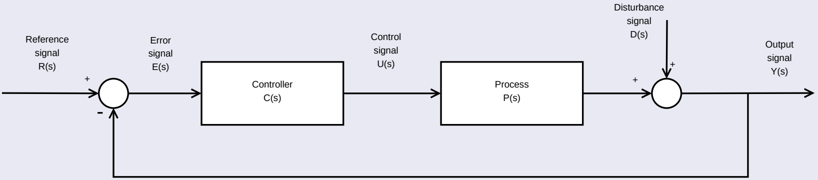
Outline

Exercise: name the correct property

Outline

Transfer function of a closed-loop system

Classical control loop



$$Y(s) - D(s) = P(s)U(s) \quad \text{with } U(s) = C(s)E(s)$$
$$Y(s) - D(s) = P(s)C(s)E(s) \quad \text{with } E(s) = R(s) - Y(s)$$

$$Y(s) - D(s) = P(s)C(s)(R(s) - Y(s))$$
$$Y(s) - D(s) = P(s)C(s)R(s) - P(s)C(s)Y(s)$$

$$Y(s) = \frac{P(s)C(s)}{1 + P(s)C(s)}R(s) + \frac{1}{1 + P(s)C(s)}D(s)$$

Transfer function from $R(s)$ to $Y(s)$

Definition

We define H as the transfer function from $R(s)$ to $Y(s)$

$$H(s) \triangleq \frac{Y(s)}{R(s)} = \frac{P(s)C(s)}{1 + P(s)C(s)} \quad \text{when } D(s) = 0$$

This transfer function $H(s)$ will help us to evaluate tracking
Almost perfect tracking: the output $Y(s)$ will follow $R(s)$ very closely $\Rightarrow H(s) \approx 1$

Transfer function from $D(s)$ to $Y(s)$

Definition

We define $M(s)$ as the transfer function from $D(s)$ to $Y(s)$.

$$M(s) \triangleq \frac{Y(s)}{D(s)} = \frac{1}{1 + P(s)C(s)} \quad \text{when} R(s) = 0$$

If the disturbance rejection of the control system is very good the disturbances will have almost no effect on the output $\Rightarrow M(s) \approx 0$

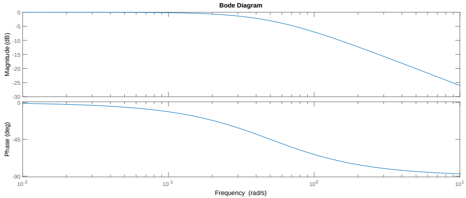
$$\begin{cases} |H(j\omega)| \cong 1 \\ |M(j\omega)| \cong 0 \end{cases}$$

$\Rightarrow |P(j\omega)C(j\omega)|$ **(open loop gain)** is very large.

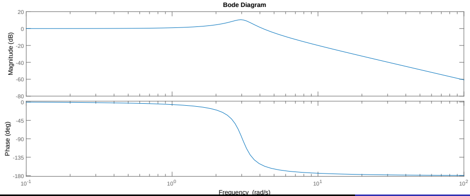
! a large open loop amplification might lead to an unstable system

Example: Which controller do you prefer

The transfer function from $R(s)$ to $Y(s)$ for two different controllers for high precision surgery



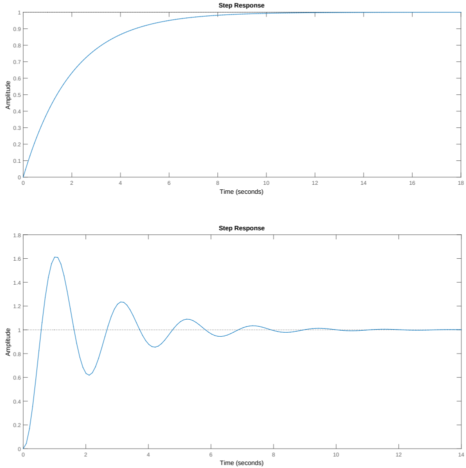
$$\frac{0.5}{s+0.5}$$



$$\frac{10}{1.09s^2+s+10}$$

Example: step response of both controllers

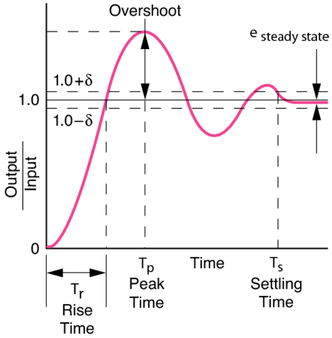
The ouput $Y(s)$ when $R(s)$ is a step function.



Quality of reference tracking

Look at the step response of the transfer function from $R(s)$ to $Y(s)$. The quality can be determined using these criteria:

- Rise-time
- Settling time
- Steady state error
- Overshoot
- ...



Model errors

- In practice, the transfer function $P(s)$ is often unknown. It is important to know how model errors affect the result. Sensitivity and robustness are key concepts to evaluate these effects.
- Sensitivity
  - Quantifies the effect of small model errors on the output.
- Robustness
  - Refers to bigger changes of the model. A controller is robust if it works properly over a given set of parameters.

Outline

Sensitivity

Sensitivity is a measure of the effect of a (small) disturbance on the model (e.g. variations in the process parameters)

$$Y(s) + \Delta Y(s) = \frac{(P(s) + \Delta P(s))C(s)}{1 + (P(s) + \Delta P(s))C(s)} R(s) + \frac{1}{1 + (P(s) + \Delta P(s))C(s)} D(s)$$

Look at the effect on the system without disturbances ($D(s){=}0$)

$$\Delta Y(s) = \frac{(P(s) + \Delta P(s))C(s)}{1 + (P(s) + \Delta P(s))C(s)} R(s) - \frac{P(s)C(s)}{1 + P(s)C(s)} R(s)$$

Sensitivity

$$
\begin{aligned}
&= \frac{(P(s)+\Delta P(s))C(s)(1+P(s)C(s)) - P(s)C(s) - P(s)C(s)(P(s)+\Delta P(s))C(s)}{(1+(P(s)+\Delta P(s))C(s)(1+P(s)C(s))} R(s) \\
&= \frac{\Delta P(s)C(s)}{(1+(P(s)+\Delta P(s))C(s))(1+P(s)C(s))} R(s) \\
&= Y(s)\frac{\Delta P(s)}{P(s)} \frac{1}{1+(P(s)+\Delta P(s))C(s)}
\end{aligned}
$$

Sensitivity

Now take the relative change due to this disturbance of the model and take the limit for $\partial x \to 0$; this gives the following (measure of the) sensitivity:

$$S_P^Y(s) = \frac{\frac{\partial Y}{Y}(s)}{\frac{\partial P}{P}(s)} = \frac{\partial Y}{\partial P}(s) \cdot \frac{P(s)}{Y(s)} = \frac{1}{1 + P(s)C(s)}$$

- Again, a very large $|P(s)C(s)|$ looks like a good choice, but again there is a risk for instability!
- Note that the sensitivity can be determined for any parameter

Outline

Example 1: robustness

Example

Suppose we want to stabilize the system $P(s) = \frac{1}{(s-a)}$. We only know that $a \in [0.20; 0.80]$. We propose to use a proportional controller $C(s) = K$. The closed loop transfer function becomes

$$S(s) = \frac{P(s)C(s)}{1 + P(s)C(s)} = \frac{\frac{1}{s-a}K}{1 + \frac{1}{s-a}K} = \frac{K}{s - a + K}$$

The controller stabilizes the system if $-a + K > 0$. If we choose K very large the system will be robust against changes in a.

Example 2: robustness

Example

Suppose we want to stabilize the system $P(s) = \frac{s}{(s-a)}$ with $a \in [0.20; 0.80]$. If we choose the control law $C(s) = \frac{K}{s}$ this results in the same transfer function.

$$S(s) = \frac{P(s)C(s)}{1 + P(s)C(s)} = \frac{\frac{s}{s-a}\frac{K}{s}}{1 + \frac{s}{s-a}\frac{K}{s}} = \frac{K}{s - a + K}$$

Again we choose $K > a$ to ensure stability. But how does the controller perform on the slightly perturbed system $\tilde{P}(s) = \frac{(s+\epsilon)}{(s-a)}$? The transfer function from $R(s)$ to $Y(s)$ becomes

$$S(s) = \frac{\frac{s+\epsilon}{s-a}\frac{K}{S}}{1 + \frac{s+\epsilon}{s-a}\frac{K}{s}} = \frac{(s+\epsilon)K}{s^2 + (K-a)s + \epsilon K}$$

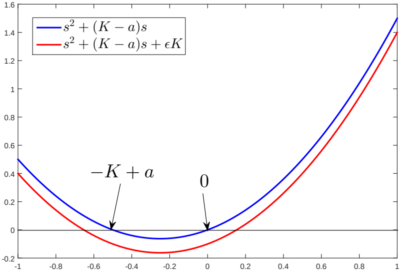
Example 2: robustness

$$\tilde{S}(s) = \frac{(s+\epsilon)K}{s^2 + (K-a)s + \epsilon K}$$

The figure on the right plots

$$s^2 + (K-a)s$$
$$s^2 + (K-a)s + \epsilon K$$

For ϵ negative the system becomes unstable. The control system is not robust and useless in practical situations. You shouldnt use control laws which rely on a pole-zero cancellation.



Robustness of steady-state error

Definition

The steady state error is defined as follows:

$$\begin{aligned}\lim_{t\to\infty} e(t) &= \lim_{t\to\infty} (r(t) - y(t)) \\ &= \lim_{s\to 0} s(R(s) - Y(s)) \qquad \text{(final value theorem)}\end{aligned}$$

Open-loop system

A very small steady state error (preferably zero) indicates that the controller tracks the reference very well

For the open loop system with a step reference

$$e_{ol}(\infty) = \lim_{s \to 0} s(1 - C(s)P(s))R(s) = 1 - C(0)P(0)$$

- So the open loop controller can be free of a steady state error (for a step reference $\epsilon(t)$) by calibrating the controller such that $C(0)P(0) = 1$
- $\Rightarrow$ a precise calibration of the DC gain

Steady state error

Closed-loop system:

$$
\begin{aligned}
e_{cl}(\infty) &= \lim_{s\to 0} s\left(1 - \frac{C(s)P(s)}{1 + C(s)P(s)}\right) R(s) \\
&= \lim_{s\to 0} \frac{s}{1 + C(s)P(s)} R(s) = \frac{1}{1 + C(0)P(0)}
\end{aligned}
$$

- The steady state error is small if $C(0)P(0)$ is very large
- Again, calibrating the DC gain
- The difference is that here we only need a large gain, which is far less demanding than having to make it equal to 1

An open loop controller is not robust

We can now show how this results in a great advantage of the closed loop strategy over the open loop strategy:

- If P changes slightly (for instance due to a factor that has not been taken up into the model) to $P + \Delta P$
- Then making $e_{ol}(\infty)$ small would require to calibrate anew
- Whereas $e_{cl}(\infty)$ would remain small, as long as $(P(0) + \Delta P(0))C(0)$ remains large

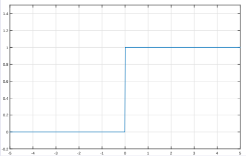
Hence an open loop controller can not control the output **robustly** against changes in P

Outline

Steady state error: step

Definition

$$\epsilon(t) = \begin{cases} 0, & t \leq 0 \\ \frac{1}{2}, & t = 0 \\ 1, & t \geqslant 0 \end{cases}$$



For a step reference $r(t) = A\epsilon(t) \Rightarrow R(s) = \frac{A}{s}$

$$e_{cl}(\infty) = \frac{A}{1 + C(0)P(0)}$$

So $e_{cl}(\infty) = 0$ if C(s)P(s) has a least one pole in zero $(C(0)P(0) \rightarrow \infty)$

Steady state error: ramp

For a ramp reference function $r(t) = At\epsilon(t) \Rightarrow R(s) = \frac{A}{s^2}$, we have

$$e_{cl}(\infty) = \lim_{s \to 0} \frac{s}{1 + C(s)P(s)} \frac{A}{s^2}$$

So $e_{cl}(\infty) = 0$ if $C(s)P(s)$ has at least two poles in zero.
$\left(\lim_{s \to 0} sC(s)P(s) = \infty \right)$

Type of a system

- The type of a system is determined by the number of poles $P(s)C(s)$ has in zero, and hence it is linked to what type of references it can track perfectly
- Write $P(s)C(s)$ as $\frac{K\prod_{k=1}^{m}(s-z_k)}{s^l\prod_{k=1}^{n-l}(s-p_k)}$ with l the amount of poles in zero
- We say this system is of type l
  - It will be able to track references of shape $At^0\epsilon(t)$ up to $At^{(l-1)}\epsilon(t)$ perfectly (for $t\rightarrow\infty$, $\lim\limits_{t\rightarrow\infty} e_{cl}(t)=0$)
  - A reference of shape $At^l\epsilon(t)$ will be missed by a finite factor (for $t\rightarrow\infty$, $\lim\limits_{t\rightarrow\infty} e_{cl}(t)=C$)
  - A reference of shape $At^{l+\cdots}\epsilon(t)$ will be missed by infinity (for $t\rightarrow\infty$, $\lim\limits_{t\rightarrow\infty} e_{cl}(t)=\infty$)

Example of a type 0 system

Example

Step reference $r(t) = A\epsilon(t) \Rightarrow R(s) = \frac{A}{s}$

$$
\begin{aligned}
e_{cl}(\infty) &= \lim_{s \to 0} \frac{s}{1 + C(s)P(s)} \frac{A}{s} \\
&= \lim_{s \to 0} \frac{A}{1 + \frac{K \prod_{k=1}^{m}(-z_k)}{\prod_{k=1}^{n-l}(-p_k)}} = \frac{A}{1 + K_p}
\end{aligned}
$$

where K_p is the static position constant

Example of a type 0 system

Example

Ramp reference $r(t) = At\epsilon(t) \Rightarrow R(s) = \frac{A}{s^2}$

$$e_{cl}(\infty) = \lim_{s \to 0} \frac{s}{1 + C(s)P(s)} \frac{A}{s^2}$$
$$= \lim_{s \to 0} \frac{1}{1 + \frac{K \prod_{k=1}^{m}(-z_k)}{\prod_{k=1}^{n-l}(-p_k)}} = \infty$$

Steady state errors - type of a system

- With:

$$K_p = \lim_{s \to 0} P(s)C(s) \qquad K_p = \text{Static position constant}$$
$$K_v = \lim_{s \to 0} sP(s)C(s) \qquad K_v = \text{Static velocity constant}$$
$$K_a = \lim_{s \to 0} s^2 P(s)C(s) \qquad K_a = \text{Static acceleration constant}$$

- And the respective steady state errors for different system types are:

Type I	**Step $\mathbf{A\epsilon(t)}$**	**Ramp $\mathbf{At\epsilon(t)}$**	**Parabola $\mathbf{\frac{At^2\epsilon(t)}{2}}$**
0	$\frac{A}{1+K_p}$	∞	∞
1	0	$\frac{A}{K_v}$	∞
2	0	0	$\frac{A}{K_a}$

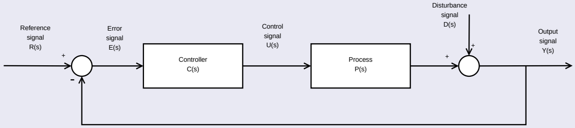
Outline

1. Basics
   - Control Theory
   - Demo: Inverted Pendulum
2. Control Goals
   - Stability
   - Disturbance rejection
   - Reference tracking
   - Exercise
3. Closed-loop systems
   - Sensitivity
   - Robustness
   - Types of systems and Steady State Error
   - Noise- and disturbance rejection

Noise rejection and disturbance rejection

Noise rejection and disturbance rejection

- Both refer to rejection of the input D but are not the same
- Measurement noise is often modelled by white noise. (High frequency signal)
- Disturbances are actual changes to the state of the system



Noise versus Disturbances

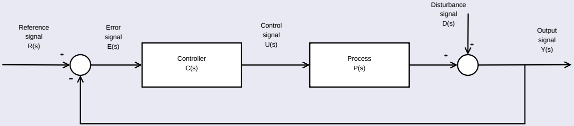
- Cruise control:
  Measurement errors on the speed are noise
  A change in slope is a disturbance

Noise versus Disturbances

- Remember from the start of the lecture

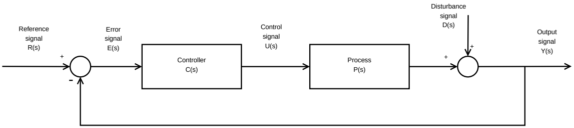
$$Y(s) = \frac{P(s)C(s)}{1+P(s)C(s)}R(s) + \frac{1}{1+P(s)C(s)}D(s)$$

- Choosing $T(s) = \frac{1}{1+P(s)C(s)}$ sufficiently small results in the rejection of D. This can be achieved by choosing $C(s)$ sufficiently high



Disturbance vs noise

- But what happens to measurement noise
  - The amplified measurement noise will be applied to the input of the plant.
  - In reality measurement noise requires no control actions. Good noise rejection means that the controller ignores this noise.



Disturbance versus noise rejection

- Good disturbance rejection requires fast action to bring the system back to the desired state
- Good noise rejection requires that measurement noise will be ignored
- Note that a controller can not see the difference between measurement noise and disturbances. Slow controllers will be less sensitive to measurement noise. Fast system will have better disturbance rejection

Slow Controllers
Not sensitive to noise
Small control inputs

↔

Fast Controllers
Good disturbance rejection
Fast tracking

Robust Controllers
Model errors will not affect the behaviour of the system strongly

↔

Aggresive Controllers:
Exchanges robustness for better performance