Introduction to Open-Loop Controller Design

Contents map

developed content units	**taxonomy levels**
closed-loop control	u1, e1
open-loop control	u1, e1

prerequisite content units	**taxonomy levels**
LTI system	u1, e1

Main ILO of sub-module "Introduction to Open-Loop Controller Design"

Explain the difference between open-loop and closed-loop control architectures, using the car speed control example as context.

Formulate the mathematical model of a physical system (car dynamics) and linearize it around a desired equilibrium point.

Design a basic open-loop controller by inverting the system model or its DC gain, and analyze its limitations in terms of disturbance rejection and parameter sensitivity.

Evaluate the trade-offs between response speed and actuator effort when implementing pole cancellation in open-loop control design.

Assess the practical challenges of implementing open-loop control in real-world systems, using the automatic gate example as a case study

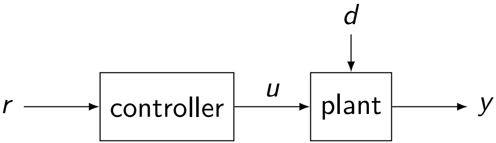
Roadmap

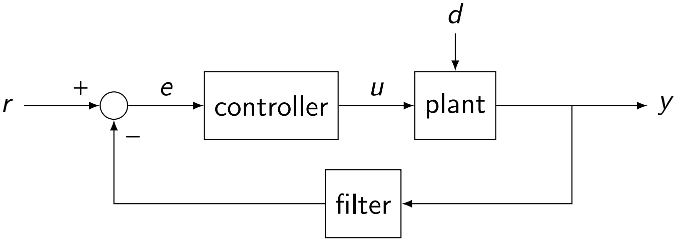
- overview of the structure
- building intuition through an example

What do we learn now?

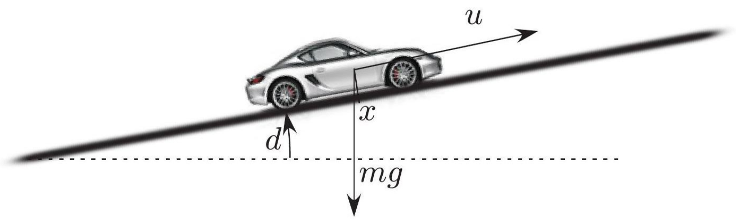
introduction to how to design an open-loop controller

Basic Control Architectures: open-loop vs. closed-loop (or feedback)





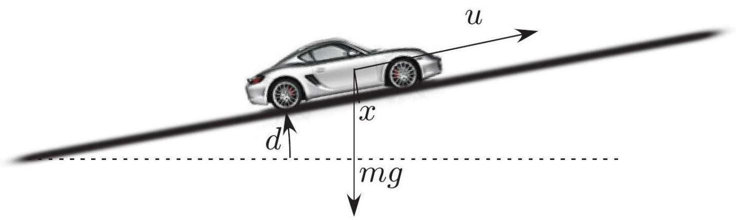
Example: car speed control



Signals:

- output: $y(t)$ car speed
- input: $u(t)$ force generated by an engine
- disturbance: $d(t)$ road slope

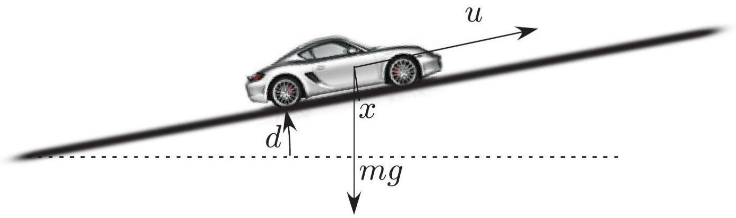
Example: car speed control



Goal:

$$\text{find } u(t) = \phi(\star) \text{ such that } y(t) \simeq \alpha r(t) \text{ regardless of } d$$

Example: car speed control



Model parameters:

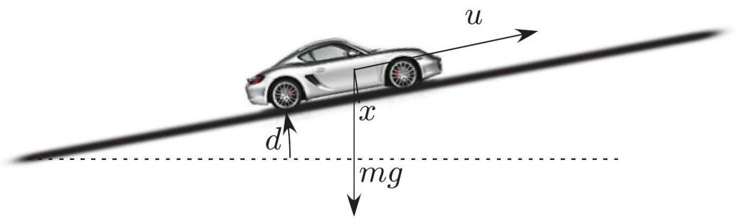
- m: car mass
- b: friction coefficient
- g: gravitational acceleration

how to do it?

Strategy:

- find the system model
- linearize it around the desired equilibrium
- find the transfer function of the system
- design the controller so that the closed-loop system transfer function is the one we want

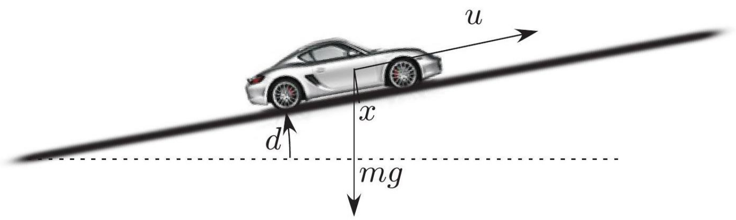
Example: car speed control



Model:

$$m\dot{y}(t) = -by(t) + u(t) - mg\sin\left(d(\int ydt)\right)$$

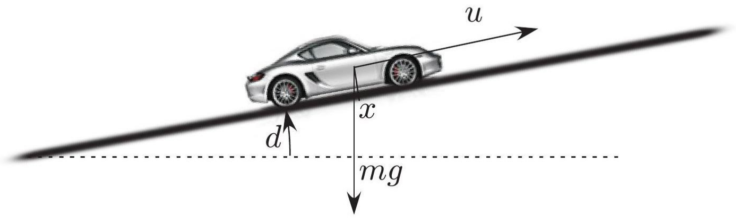
Example: car speed control



Linearized model:

$$m\dot{y}(t) = -by(t) + u(t) - mgd(t)$$

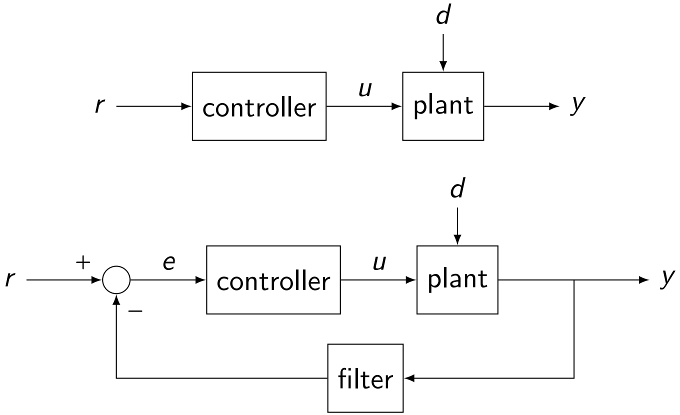
Example: car speed control



Laplace (model):

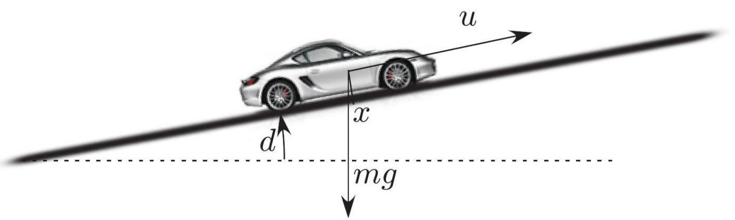
$$Y(s) = \frac{1}{ms + b}\big(U(s) - mgD(s)\big)$$

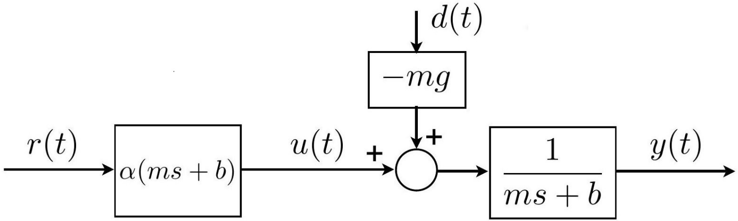
Fundamental architectures for car speed control: open-loop vs. closed-loop (or feedback)



Car speed control in open loop

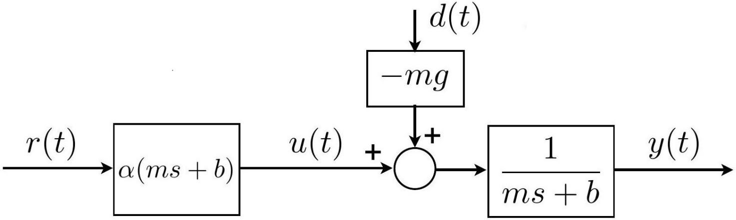
= inversion of the system model



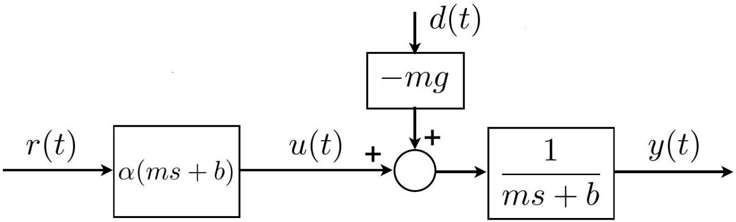


goal: $y(t) \simeq \alpha r(t)$ i.e. $W_{ry}(s) = \alpha$

What are the problems?



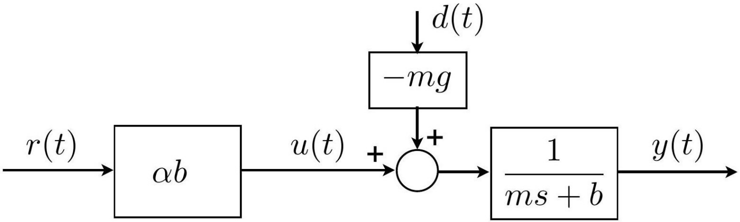
What are the problems?



How to solve this? We settle for the output to follow the reference only at steady state $\implies$ we only invert the DC gain

Car speed control in open loop

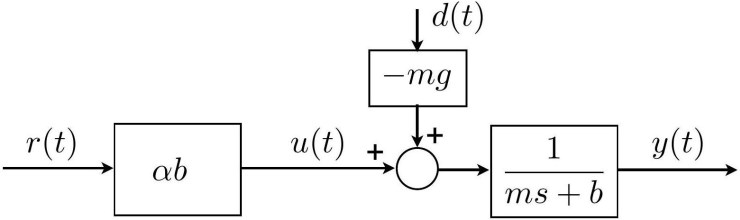
= inversion of the system model gain



$$\Longrightarrow \quad W_{ry}(s) = \frac{\alpha b}{ms + b} \quad \Longrightarrow \quad W_{ry}(0) = \alpha \quad \Longrightarrow \quad y(\infty) = \alpha r(\infty)$$

Car speed control in open loop

= inversion of the system model gain

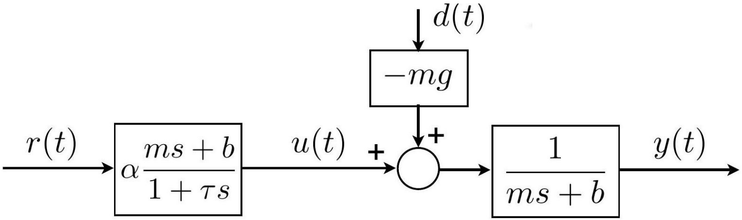


$$\Longrightarrow \quad W_{ry}(s) = \frac{\alpha b}{ms+b} \quad \Longrightarrow \quad W_{ry}(0) = \alpha \quad \Longrightarrow \quad y(\infty) = \alpha r(\infty)$$

but the rise time depends on m and b!

Car speed control in open loop

= inversion of the system model gain and cancellation of the stable pole

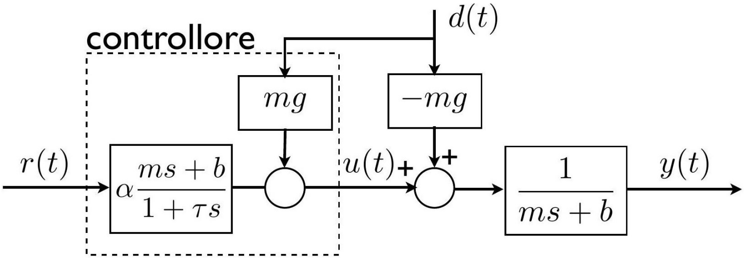


$$\Longrightarrow \; W_{ry}(s) = \frac{\alpha}{\tau s + 1} \; \Longrightarrow \; W_{ry}(0) = \alpha \; \Longrightarrow \; y(\infty) = \alpha r(\infty)$$

and the rise time depends on the controller parameters

What if we could measure or predict d?

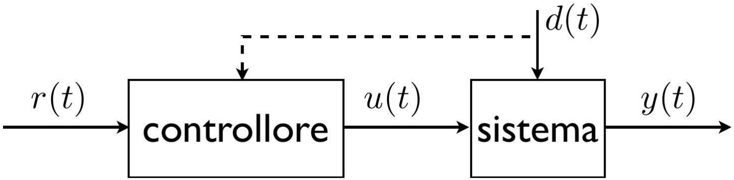
Open-loop control of a car's speed



$$Y(s) = W_{ry}(s)R(s) + W_{dy}(s)D(s)$$

$$W_{dy}(s) = (mg - mg)\frac{1}{ms+b} = 0$$

Generalization of the example



- Objectives of open-loop control: $W_{ry}(s) \simeq \alpha \quad W_{dy}(s) \simeq 0$
- Problems of open-loop control:
  - need to know exactly the model and its parameters
  - need to measure disturbances
  - impossibility to control unstable systems

Practical example: DIY automatic gate in open loop

Recipe:

- buy and install the gate
- do some experiments to understand the linear regime
- create a model from the data
- invert it
- build an RCL circuit that implements that controller

Summarizing

Explain the difference between open-loop and closed-loop control architectures, using the car speed control example as context.

Formulate the mathematical model of a physical system (car dynamics) and linearize it around a desired equilibrium point.

Design a basic open-loop controller by inverting the system model or its DC gain, and analyze its limitations in terms of disturbance rejection and parameter sensitivity.

Evaluate the trade-offs between response speed and actuator effort when implementing pole cancellation in open-loop control design.

Assess the practical challenges of implementing open-loop control in real-world systems, using the automatic gate example as a case study

Most important python code for this sub-module

The python.control library

... as virtually in all the modules of this part of the course

Self-assessment material

Question 1

What is the fundamental limitation of open-loop control compared to closed-loop control?

Potential answers:

I: It requires more computational power
II: It cannot compensate for unmeasured disturbances or model inaccuracies
III: It only works for nonlinear systems
IV: It requires more sensors than closed-loop control
V: I do not know

Question 2

Why do we typically linearize nonlinear system models before designing controllers?

Potential answers:

I: Because all physical systems are fundamentally linear

II: Because nonlinear controllers cannot be implemented in practice

III: Because most controller design tools and analysis methods are developed for linear systems

IV: Because linearization increases system stability

V: I do not know

Question 3

In the car speed control example, why can't perfect disturbance rejection be achieved in practice through open-loop control?

Potential answers:

I: Because disturbances cannot be measured under any circumstances

II: Because it requires perfect knowledge of both the system model and disturbance characteristics

III: Because open-loop controllers are inherently unstable

IV: Because the car's mass changes during operation

V: I do not know

Question 4

What is the main practical issue with designing an open-loop controller by perfectly inverting the system model?

Potential answers:

I: It makes the system too fast
II: It often results in a non-causal controller that requires knowledge of future inputs
III: It requires solving differential equations in real-time
IV: It makes the control signal too smooth
V: I do not know

Question 5

In the car speed control example, why might choosing a very small time constant in the controller be problematic?

Potential answers:

I: It would make the controller too simple
II: It would require unrealistically large control forces from the engine
III: It would make the car accelerate too slowly
IV: It would prevent the car from reaching the desired speed
V: I do not know

Recap of module “Introduction to Open-Loop Controller Design”

- open-loop control is structurally simple but not very robust

?

Introduction to closed-loop controller design

Contents map

developed content units	taxonomy levels
closed-loop control	u1, e1

prerequisite content units	taxonomy levels
LTI system	u1, e1

Main ILO of sub-module "Introduction to closed-loop controller design"

Explain the fundamental differences between open-loop and closed-loop control systems in terms of error correction and disturbance rejection

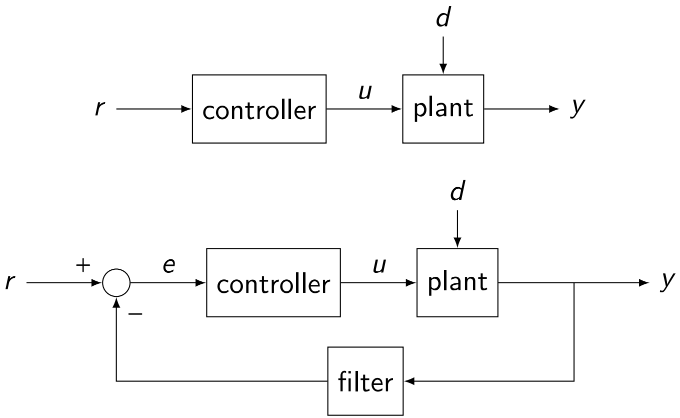
Roadmap

- what it is
- examples

What do we learn now?

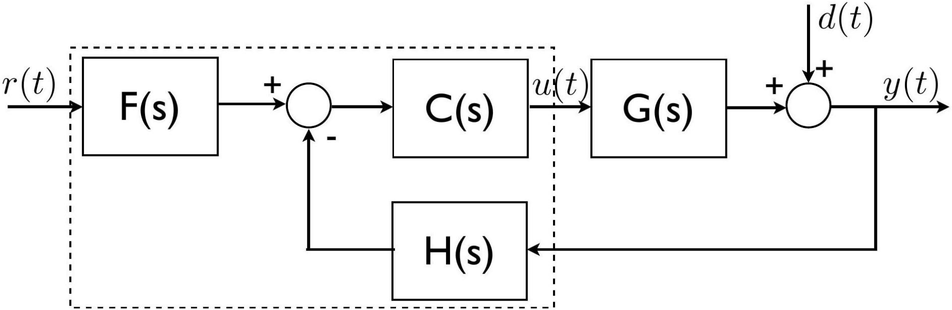
introduction to how to design a closed-loop controller

Fundamental control architectures: open-loop vs. closed-loop (or feedback)

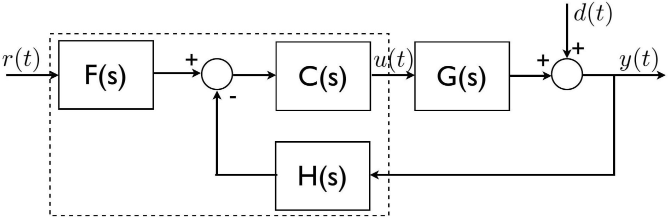


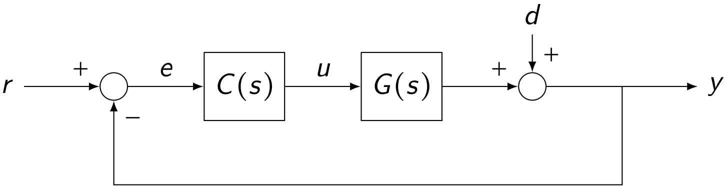
Notation

- $G(s)$: Transfer function of the system to be controlled
- $H(s)$: Transfer function of the sensor
- $C(s)$: Transfer function of the controller
- $F(s)$: Transfer function of the shaping filter

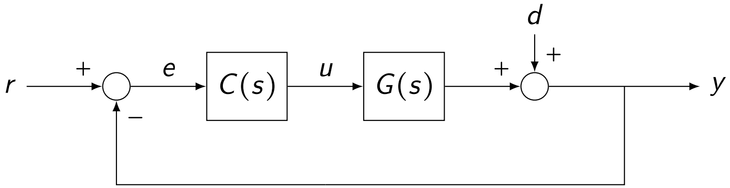


By moving blocks we can switch to standard notation





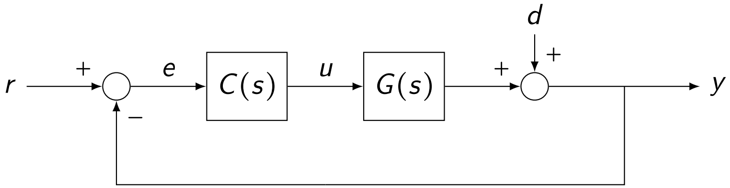
But what would we ideally want?



$$W_{ry}(s) = \frac{C(s)G(s)}{1 + C(s)G(s)} \approx 1 \quad \Longrightarrow \quad y(t) \approx r(t)$$

$$W_{dy}(s) = \frac{1}{1 + C(s)G(s)} \approx 0 \quad \Longrightarrow \quad y(t) \approx \text{ not affected by } d(t)$$

And how is it done 90% of the time?



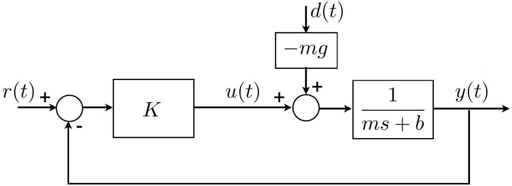
With

- $C(s) = K_p$, proportional controller
- $C(s) = K_p + \dfrac{K_i}{s}$, proportional-integral controller
- $C(s) = K_p + K_d s$, proportional-derivative controller
- $C(s) = K_p + \dfrac{K_i}{s} + K_d s$, proportional-integral-derivative controller

And how to build a PID?

- analog: `https://www.youtube.com/watch?v=Sw3NEA3GEnI`
- digital: in a few modules

Example: car speed control with a P controller



with $C(s) = K$ and $G(s) = \dfrac{1}{ms + b}$, implying

$$W_{ry}(s) = \frac{C(s)G(s)}{1 + C(s)G(s)} = \frac{K}{ms + b + K} \approx 1 \qquad \Longrightarrow \qquad y(t) \approx r(t)$$

$$W_{dy}(s) = \frac{-mgG(s)}{1 + C(s)G(s)} = \frac{-mg}{ms + b + K} \approx 0 \qquad \Longrightarrow \qquad y(t) \approx \text{ not affected by } d(t)$$

Summarizing

Explain the fundamental differences between open-loop and closed-loop control systems in terms of error correction and disturbance rejection

Most important python code for this sub-module

The python.control library

. . . as virtually in all the modules of this part of the course

Self-assessment material

Question 6

What is the primary conceptual advantage of closed-loop control over open-loop control?

Potential answers:

I: Ability to automatically correct errors using feedback
II: Higher computational efficiency in implementation
III: Elimination of all system disturbances
IV: Reduced need for sensors in the system
V: I do not know

Question 7

Why are PID controllers so widely used in practice despite their simplicity?

Potential answers:

I: They can perfectly eliminate all system nonlinearities

II: They provide effective performance across many applications with relatively simple implementation

III: They require no tuning parameters for optimal performance

IV: They eliminate the need for system modeling entirely

V: I do not know

Question 8

In the standard closed-loop control notation, what does the transfer function $W_{dy}(s) \approx 0$ imply about the system?

Potential answers:

I: The system cannot track reference signals
II: The controller has become unstable
III: The system effectively rejects disturbances
IV: The sensor measurements are inaccurate
V: I do not know

Question 9

What fundamental limitation prevents a real control system from achieving perfect tracking ($W_{ry}(s) = 1$) and perfect disturbance rejection ($W_{dy}(s) = 0$) simultaneously?

Potential answers:

I: The need for digital implementation
II: Sensor accuracy limitations
III: Fundamental trade-offs between performance, robustness, and stability
IV: The cost of high-quality actuators
V: I do not know

Question 10

In the car speed control example with a P controller, what happens to both $W_{ry}(s)$ and $W_{dy}(s)$ as the proportional gain K is increased?

Potential answers:

I: Both transfer functions approach infinity
II: $W_{ry}(s)$ approaches 0 while $W_{dy}(s)$ approaches 1
III: $W_{ry}(s)$ approaches 1 while $W_{dy}(s)$ approaches 0
IV: Both transfer functions become oscillatory
V: I do not know

Recap of module "Introduction to closed-loop controller design"

- feedback control is more promising, but requires designing more things compared to open loop