Feedback Control

- Open-Loop Control
- Feedback Control
- Proportional Control

Contents

Open-loop Control

Linear control system model

$$\dot{x}(t) = ax(t) + bu(t) \quad \text{and} \quad x(0) = x_0$$

Open-loop control

- First choose $u(t)$ on the control horizon $[0, T]$ (offline)
- Send the input u to the system and use it "blindly".

Open-loop Control

Example

$$\dot{x}(t) = x(t) + u(t) \quad \text{and} \quad x(0) = 1$$

Open-loop control

- If we choose $u(t) = -1$ the system state satisfies

$$\dot{x}(t) = x(t) - 1 \quad \text{with} \quad x(0) = 1 \;,$$

  i.e., we have $x(t) = 1$ for all $t \geq 0$.

Question: would this work in practice ?

Open-loop Control

Problem:

$$\dot{x}(t) = x(t) - 1 \quad \text{and} \quad x(0) = 1 + \epsilon$$

- If there is a small error $\epsilon \neq 0$ in the initial value, we have

  $$x(t) = 1 + \epsilon e^t \ ,$$

  i.e., the error grows exponentially in t.

- Similar problem occur if the dynamic model is not accurate.

Open-loop Control

Problem:

$$\dot{x}(t) = x(t) - 1 \quad \text{and} \quad x(0) = 1 + \epsilon$$

- If there is a small error $\epsilon \neq 0$ in the initial value, we have

$$x(t) = 1 + \epsilon e^t \ ,$$

  i.e., the error grows exponentially in t.

- Similar problem occur if the dynamic model is not accurate.

Summary: disadvantages of open-loop control

- If we have small errors in the model (model-plant mismatch) or if we work with inaccurate initial values, the online implementation can be unstable / inaccurate.
- There may be external "disturbances", the control design might have to be adapted online rather than pre-computed offline.
- If we have a very long control horizon, we might not want to plan all our actions "in advance", but rather in dependence on the current "situation".

Summary: disadvantages of open-loop control

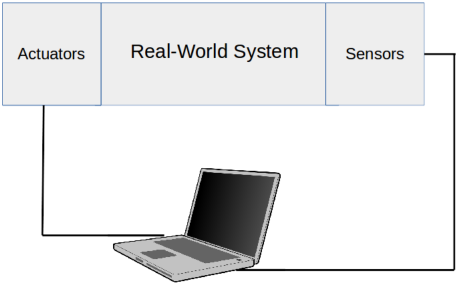
- If we have small errors in the model (model-plant mismatch) or if we work with inaccurate initial values, the online implementation can be unstable / inaccurate.
- There may be external "disturbances", the control design might have to be adapted online rather than pre-computed offline.
- If we have a very long control horizon, we might not want to plan all our actions "in advance", but rather in dependence on the current "situation".

Summary: disadvantages of open-loop control

- If we have small errors in the model (model-plant mismatch) or if we work with inaccurate initial values, the online implementation can be unstable / inaccurate.
- There may be external "disturbances", the control design might have to be adapted online rather than pre-computed offline.
- If we have a very long control horizon, we might not want to plan all our actions "in advance", but rather in dependence on the current "situation".

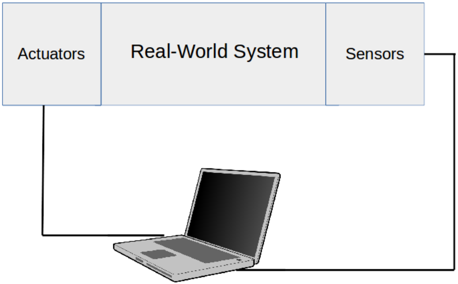
Contents

Main idea of feedback control



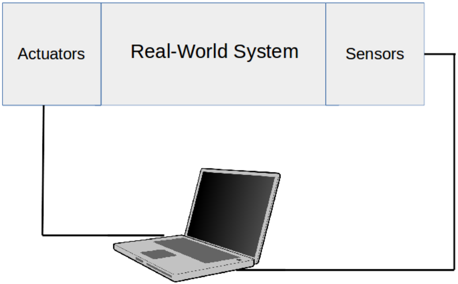
1. Receive measurement of $x(t)$ from sensor
2. Compute $u(t)$ in dependence on $x(t)$
3. Send $u(t)$ to the actuator

Main idea of feedback control



1. Receive measurement of $x(t)$ from sensor
2. Compute $u(t)$ in dependence on $x(t)$
3. Send $u(t)$ to the actuator

Main idea of feedback control

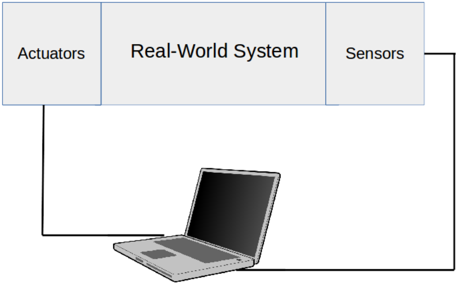


1. Receive measurement of $x(t)$ from sensor

2. Compute $u(t)$ in dependence on $x(t)$

3. Send $u(t)$ to the actuator

Main idea of feedback control



Feedback law

- The map, μ, from the measurement $x(t)$ to a control decision $u(t)$, is called a control law:

$$u(t) = \mu(x(t)) \ .$$

Closed-loop system

If we substitute $u(t) = \mu(x(t))$, we obtain the closed-loop system dynamics:

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

- If μ is an affine function, this is a linear system.
- If μ is non-affine, we have to solve a so-called nonlinear differential equation for finding $x(t)$.

Closed-loop system

If we substitute $u(t) = \mu(x(t))$, we obtain the closed-loop system dynamics:

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

- If μ is an affine function, this is a linear system.
- If μ is non-affine, we have to solve a so-called nonlinear differential equation for finding $x(t)$.

Closed-loop system

If we substitute $u(t) = \mu(x(t))$, we obtain the closed-loop system dynamics:

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

- If μ is an affine function, this is a linear system.
- If μ is non-affine, we have to solve a so-called nonlinear differential equation for finding $x(t)$.

Contents

- Open-Loop Control
- Feedback Control
- Proportional Control

Affine feedback (P-control)

- Proportional control is based on affine feedback laws,

$$\mu(y(t)) \;=\; u_{\mathrm{s}} + k(y(t) - x_{\mathrm{s}}) \;=\; u_{\mathrm{s}} + k * e(t)$$

  Here, $y(t) = x(t)$ is called the sensor output.

- Constant $u_{\mathrm{s}} \in \mathbb{R}$ is a control offset
- Constant $x_{\mathrm{s}} \in \mathbb{R}$ is called the set-point (sometimes denoted by y_{ref})
- The function $e(t) = x(t) - x_{\mathrm{s}}$ is called error signal.
- The constant $k \in \mathbb{R}$ is called the *proportional gain*

Affine feedback (P-control)

- Proportional control is based on affine feedback laws,

$$\mu(y(t)) \;=\; u_{\mathrm{s}} + k(y(t) - x_{\mathrm{s}}) \;=\; u_{\mathrm{s}} + k * e(t)$$

  Here, $y(t) = x(t)$ is called the sensor output.

- Constant $u_{\mathrm{s}} \in \mathbb{R}$ is a control offset
- Constant $x_{\mathsf{s}} \in \mathbb{R}$ is called the set-point (sometimes denoted by y_{ref})
- The function $e(t) = x(t) - x_{\mathrm{s}}$ is called error signal.
- The constant $k \in \mathbb{R}$ is called the *proportional gain*

Affine feedback (P-control)

- Proportional control is based on affine feedback laws,

$$\mu(y(t)) \;=\; u_{\mathrm{s}} + k(y(t) - x_{\mathrm{s}}) \;=\; u_{\mathrm{s}} + k * e(t)$$

  Here, $y(t) = x(t)$ is called the sensor output.

- Constant $u_{\mathrm{s}} \in \mathbb{R}$ is a control offset
- Constant $x_{\mathsf{s}} \in \mathbb{R}$ is called the set-point (sometimes denoted by y_{ref})
- The function $e(t) = x(t) - x_{\mathrm{s}}$ is called error signal.
- The constant $k \in \mathbb{R}$ is called the *proportional gain*

Affine feedback (P-control)

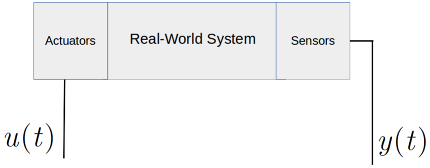
- Proportional control is based on affine feedback laws,

$$\mu(y(t)) \;=\; u_{\mathrm{s}} + k(y(t) - x_{\mathrm{s}}) \;=\; u_{\mathrm{s}} + k * e(t)$$

  Here, $y(t) = x(t)$ is called the sensor output.

- Constant $u_{\mathrm{s}} \in \mathbb{R}$ is a control offset
- Constant $x_{\mathsf{s}} \in \mathbb{R}$ is called the set-point (sometimes denoted by y_{ref})
- The function $e(t) = x(t) - x_{\mathrm{s}}$ is called error signal.
- The constant $k \in \mathbb{R}$ is called the *proportional gain*

Model-free P-control tuning: Step 1

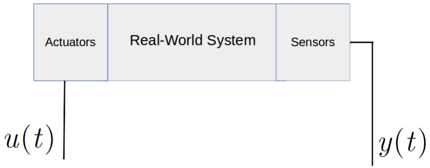


If the system is stable:

- Play around with the open loop system.
- Adjust $u(t) = u_{\mathsf{s}}$ such that open-loop system satisfies

$$x(T) \approx x_{\mathsf{s}} \qquad \text{after (a possibly long) time } T\,.$$

Model-free P-control tuning: Step 1

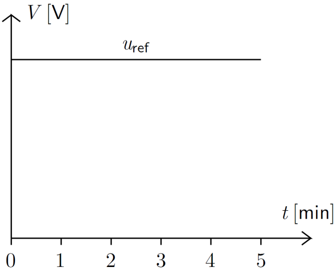


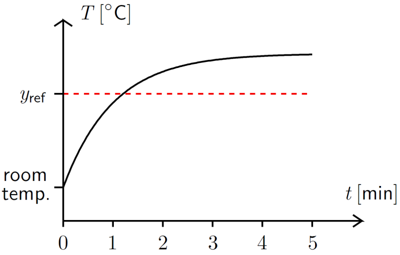
If the system is stable:

- Play around with the open loop system.
- Adjust $u(t) = u_\mathsf{s}$ such that open-loop system satisfies

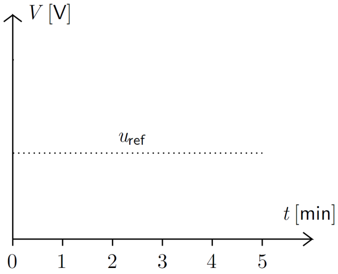
$$x(T) \approx x_\mathsf{s} \qquad \text{after (a possibly long) time } T\,.$$

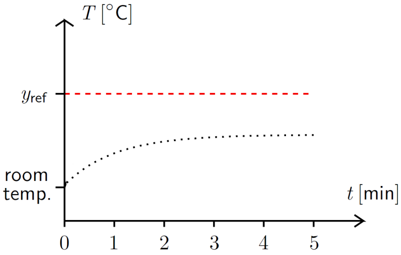
Tuning Step 1



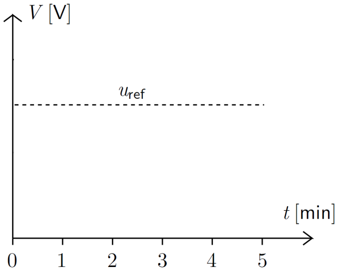


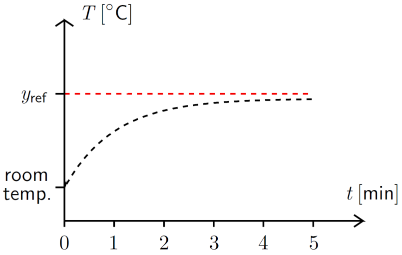
Tuning Step 1



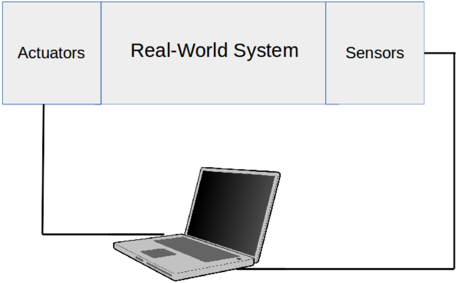


Tuning Step 1



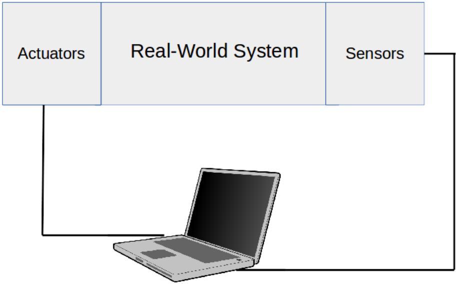


Model-free P-control tuning: Step 2



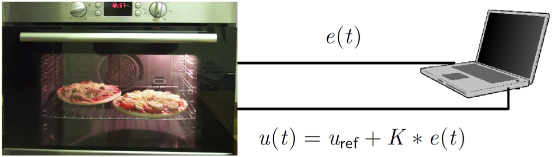
- Increase / decrease the control gain k and test the closed-loop behavior of the system
- Fine-tuning of u_s and k if needed.

Model-free P-control tuning: Step 2

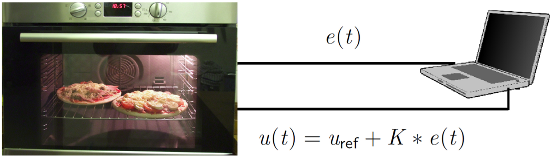


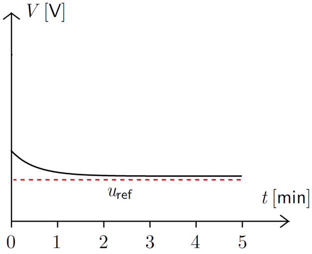
- Increase / decrease the control gain k and test the closed-loop behavior of the system
- Fine-tuning of u_s and k if needed.

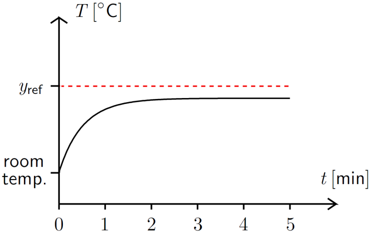
Tuning Step 2



Tuning Step 2

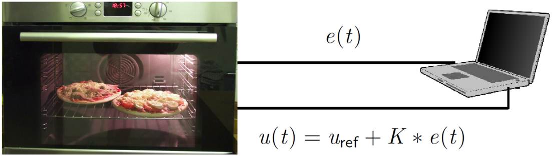


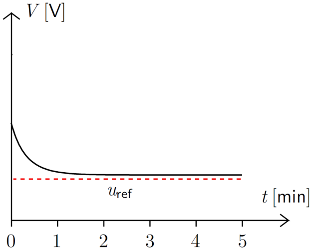


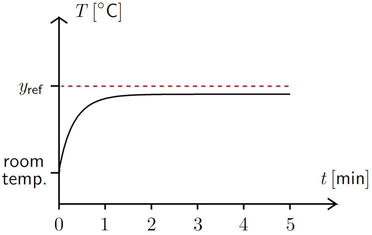


$$k = -0.25 \left[\frac{\mathrm{V}}{^\circ\mathrm{C}}\right]$$

Tuning Step 2

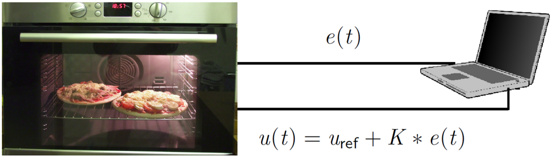


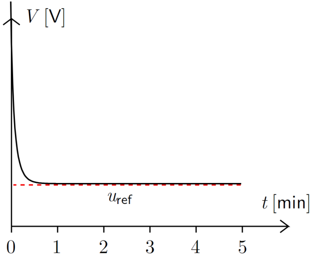


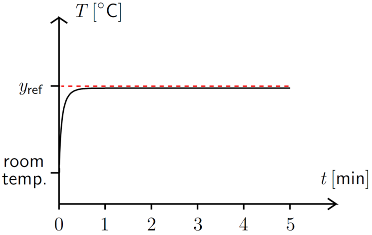


$$k = -0.50 \left[\frac{\text{V}}{^\circ\text{C}}\right]$$

Tuning Step 2

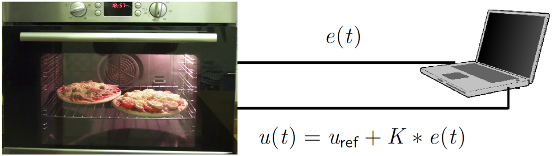




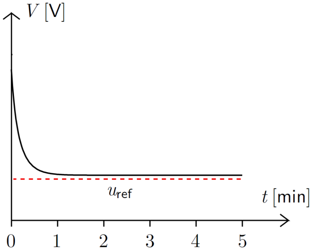


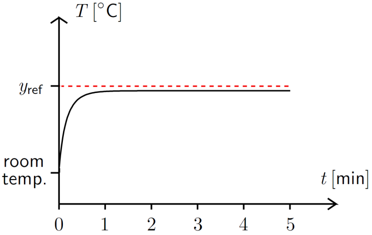
$$k = -2.00 \left[\frac{\mathrm{V}}{{}^\circ\mathrm{C}}\right]$$

Tuning Step 2



$$u(t) = u_{\text{ref}} + K * e(t)$$





$$k = -1.00 \left[\frac{\text{V}}{^\circ\text{C}}\right]$$

Model-based P-control tuning

- If we have a system model of the form

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

the control design can be done more systematically, i.e., we can design u_{s} and k without "trial-and-error" real-world experiments.

Model-based P-control tuning: Step 1

Model

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

Set-point

- Let us assume that the set point x_{s} is given.
- If $b \neq 0$, the associated steady-state control input can be found by solving

$$0 = ax_{\mathrm{s}} + bu_{\mathrm{s}} \qquad \Leftrightarrow \qquad u_{\mathrm{s}} = -\frac{ax_{\mathrm{s}}}{b} \ .$$

Model-based P-control tuning: Step 2

Closed-loop model

$$
\begin{aligned}
\dot{x}(t) &= ax(t) + b(k(x(t) - x_\mathrm{s}) + u_\mathrm{s}) \\
&= a(x(t) - x_\mathrm{s}) + ax_\mathrm{s} + b(k(x(t) - x_\mathrm{s}) + u_\mathrm{s}) \\
&= (a + bk)(x(t) - x_\mathrm{s}) + \underbrace{ax_\mathrm{s} + bu_\mathrm{s}}_{=0}
\end{aligned}
$$

Explicit solution

- The explicit solution for the closed-loop system state is given by

$$
x(t) = x_\mathrm{s} + (x(0) - x_\mathrm{s})e^{(a+bk)t} \ .
$$

Model-based P-control tuning: Step 2

Closed-loop system state:

$$x(t) = x_{\mathrm{s}} + (x(0) - x_{\mathrm{s}})e^{(a+bk)t} \ .$$

Choice of the proportional gain:

- Choose k such that $a + bk < 0$, i.e., such that

$$\lim_{t \to \infty} x(t) = x_{\mathrm{s}} \ .$$

- Choose $|k|$ not too large, as the actual control input,

$$u(t) = k(x(t) - x_{\mathrm{s}}) + u_{\mathrm{s}} \ ,$$

  may be large if $x(t) - x_{\mathrm{s}}$ is large.

Example: temperature control

Example: The temperature $T(t)$ inside an oven can be modelled as

$$\dot{T}(t) = \frac{u(t)}{C} - \frac{k_{\mathsf{Wall}}}{C}(T(t) - T_{\mathrm{room}}) \,.$$

Notation:

- Supplied power at the heating coil: $u(t) = P_{\mathrm{coil}}(t)$
- room temperature: T_{room}
- heat transition coefficient of the wall: k_{Wall}
- heat capacity of the oven: C

Example: temperature control

Example: The temperature $T(t)$ inside an oven can be modelled as

$$\dot{T}(t) = \frac{u(t)}{C} - \frac{k_{\mathsf{Wall}}}{C}(T(t) - T_{\mathrm{room}}) \; .$$

Notation:

- Supplied power at the heating coil: $u(t) = P_{\mathrm{coil}}(t)$
- room temperature: T_{room}
- heat transition coefficient of the wall: k_{Wall}
- heat capacity of the oven: C

Example: temperature control

Set point: We want to bring the temperature to

$$T_{\mathrm{s}} = 200^{\circ}\mathsf{C}\ .$$

Steady-state control input:

- Solve the steady equation

$$0 = \frac{u_{\mathrm{s}}}{C} - \frac{k_{\mathsf{Wall}}}{C}(T_{\mathrm{s}} - T_{\mathrm{room}})\ ,$$

- We find the explicit expression

$$u_{\mathrm{s}} = k_{\mathsf{Wall}}(T_{\mathrm{s}} - T_{\mathrm{room}})\ .$$

Example: temperature control

Set point: We want to bring the temperature to

$$T_{\rm s} = 200^\circ \mathsf{C} \; .$$

Steady-state control input:

- Solve the steady equation

$$0 = \frac{u_{\rm s}}{C} - \frac{k_{\mathsf{Wall}}}{C}(T_{\rm s} - T_{\rm room}) \; ,$$

- We find the explicit expression

$$u_{\rm s} = k_{\mathsf{Wall}}(T_{\rm s} - T_{\rm room}) \; .$$

Example: temperature control

Set point: We want to bring the temperature to

$$T_{\mathrm{s}} = 200^{\circ}\mathsf{C} \; .$$

Steady-state control input:

- Solve the steady equation

$$0 = \frac{u_{\mathrm{s}}}{C} - \frac{k_{\mathsf{Wall}}}{C}(T_{\mathrm{s}} - T_{\mathrm{room}}) \; ,$$

- We find the explicit expression

$$u_{\mathrm{s}} = k_{\mathsf{Wall}}(T_{\mathrm{s}} - T_{\mathrm{room}}) \; .$$

Example: temperature control

Proportional controller:

$$u(t) = u_{\mathrm{s}} + k(T(t) - T_{\mathrm{s}}) \ .$$

Closed-loop system:

- Closed-loop system has the form

$$\dot{T}(t) = \frac{k - k_{\mathrm{Wall}}}{C} \left[T(t) - T_{\mathrm{s}} \right] \ .$$

- Temperature converges to the desired set-point T_{s}, if $k < k_{\mathrm{Wall}}$,

$$T(t) = T_{\mathrm{s}} + (T(0) - T_{\mathrm{s}}) e^{\frac{k - k_{\mathrm{Wall}}}{C} t} \ .$$

Example: temperature control

Proportional controller:

$$u(t) = u_{\rm s} + k(T(t) - T_{\rm s}) \ .$$

Closed-loop system:

- Closed-loop system has the form

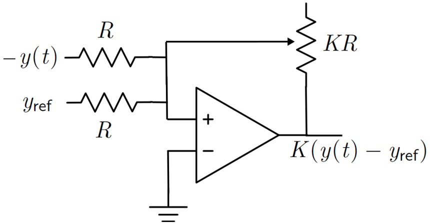
$$\dot{T}(t) = \frac{k - k_{\rm Wall}}{C} \left[T(t) - T_{\rm s} \right] \ .$$

- Temperature converges to the desired set-point $T_{\rm s}$, if $k < k_{\rm Wall}$,

$$T(t) = T_{\rm s} + (T(0) - T_{\rm s}) e^{\frac{k - k_{\rm Wall}}{C} t} \ .$$

Proportional Control Circuit

- Replace "computer" by electrical circuit (e.g. for our oven):

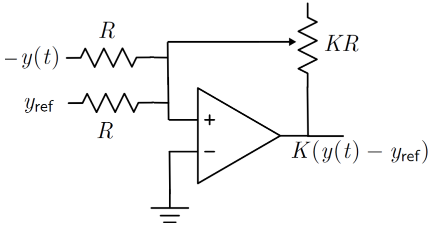


(here $y(t)$, y_{ref} and $K(y(t) - y_{\text{ref}})$ are voltages)

- Use summing amplifier (or modify above circuit) to add offset

Proportional Control Circuit

- Replace "computer" by electrical circuit (e.g. for our oven):



(here $y(t)$, y_{ref} and $K(y(t) - y_{\mathsf{ref}})$ are voltages)

- Use summing amplifier (or modify above circuit) to add offset