Control strategies

We need to be in control of things

TECHNICAL UNIVERSITY
OF CLUJ-NAPOCA
ROMANIA

January 17, 2022

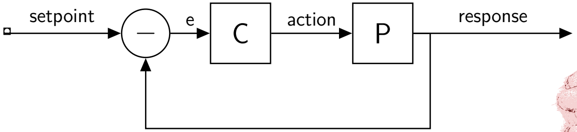
Agenda

- Control theory background
- Actuator dynamics
- Independent joint control
- Computed torque control
- Force control

Control theory

Feedback loops

The robot is a process, and if we want to accomplish some tasks, we need to be able to control its various aspects.

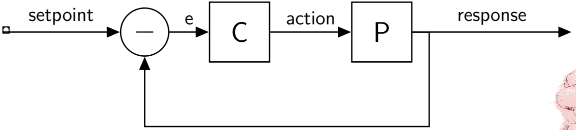


A simple process with feedback

Control theory

Feedback loops

The robot is a process, and if we want to accomplish some tasks, we need to be able to control its various aspects.



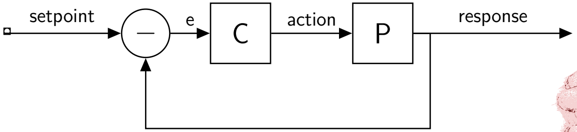
A simple process with feedback

What is the process?

Control theory

Feedback loops

The robot is a process, and if we want to accomplish some tasks, we need to be able to control its various aspects.



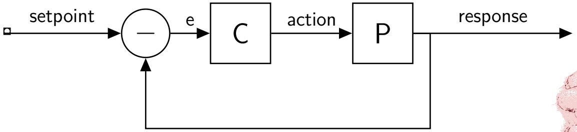
A simple process with feedback

What is the process? What is the setpoint?

Control theory

Feedback loops

The robot is a process, and if we want to accomplish some tasks, we need to be able to control its various aspects.



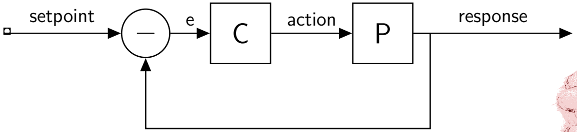
A simple process with feedback

What is the process? What is the setpoint? What is the response?

Control theory

Feedback loops

The robot is a process, and if we want to accomplish some tasks, we need to be able to control its various aspects.



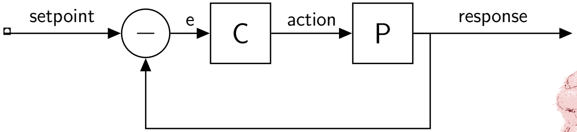
A simple process with feedback

What is the process? What is the setpoint? What is the response? What is the action?

Control theory

Feedback loops

The robot is a process, and if we want to accomplish some tasks, we need to be able to control its various aspects.

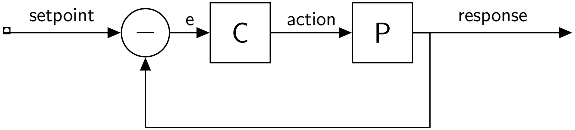


A simple process with feedback

What is the process? What is the setpoint? What is the response? What is the action? What is the controller?

Control theory

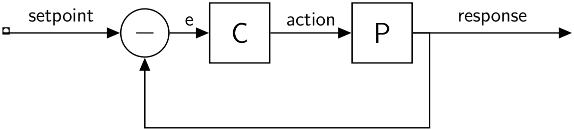
Process



A simple process with feedback

Control theory

Process

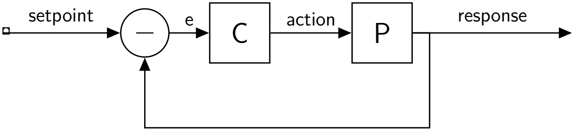


A simple process with feedback

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

Control theory

Process



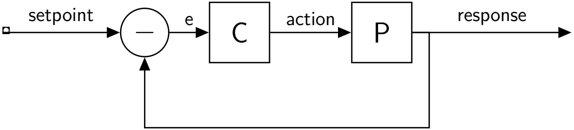
A simple process with feedback

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

Is this the **real** process?

Control theory

Setpoint, response, and action

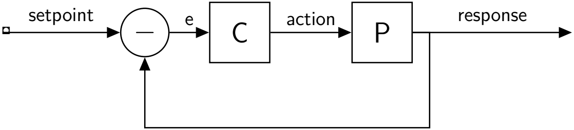


A simple process with feedback

What can our setpoint and response be?

Control theory

Setpoint, response, and action



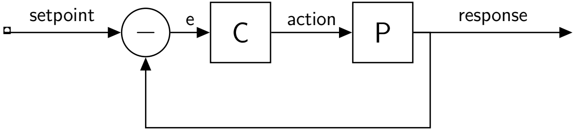
A simple process with feedback

What can our setpoint and response be?

- End-effector pose

Control theory

Setpoint, response, and action



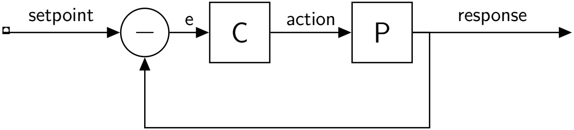
A simple process with feedback

What can our setpoint and response be?

- End-effector pose
- Joint coordinates

Control theory

Setpoint, response, and action



A simple process with feedback

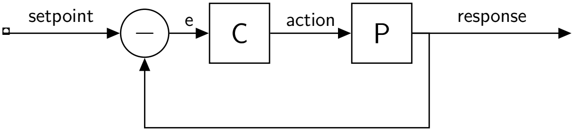
What can our setpoint and response be?

- End-effector pose
- Joint coordinates

What are the pros/cons of each?

Control theory

Setpoint, response, and action



A simple process with feedback

What can our setpoint and response be?

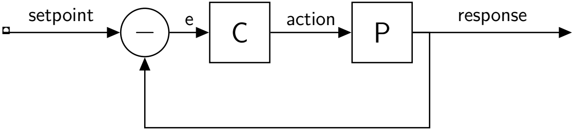
- End-effector pose
- Joint coordinates

What are the pros/cons of each?

What should the action be?

Control theory

Setpoint, response, and action



A simple process with feedback

What can our setpoint and response be?

- End-effector pose
- Joint coordinates

What are the pros/cons of each?
What should the action be?

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

Control theory

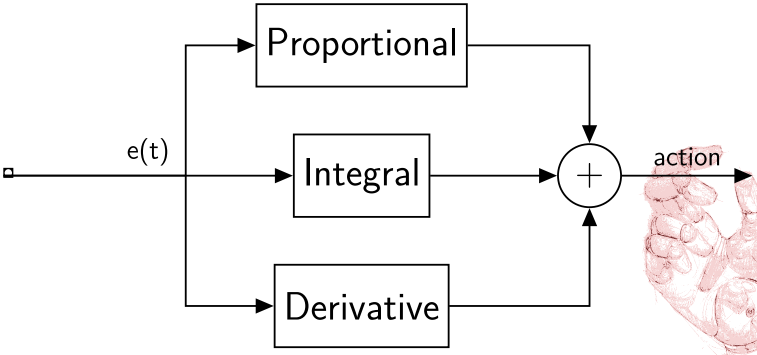
Controller

One of the most basic, robust and used controller of all times!

Control theory

Controller

One of the most basic, robust and used controller of all times!

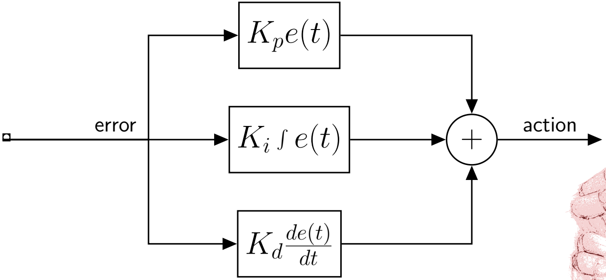


The PID controller

Control theory

PID controller

One of the most basic, robust and used controller of all times!

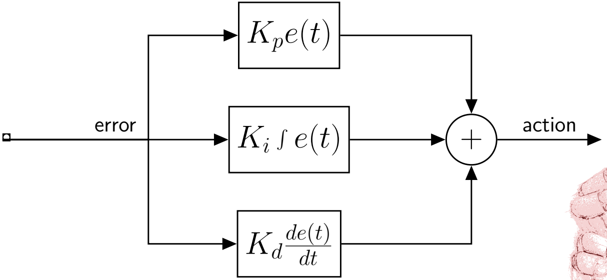


The PID controller

Control theory

PID controller

One of the most basic, robust and used controller of all times!



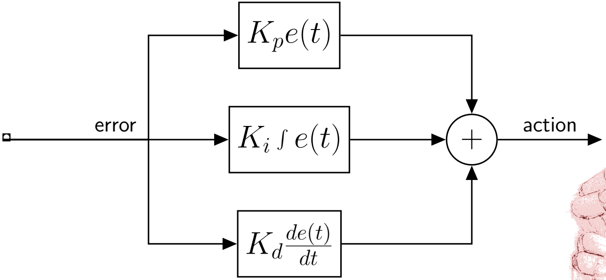
The PID controller

Setpoint - response: Joint coordinates

Control theory

PID controller

One of the most basic, robust and used controller of all times!



The PID controller

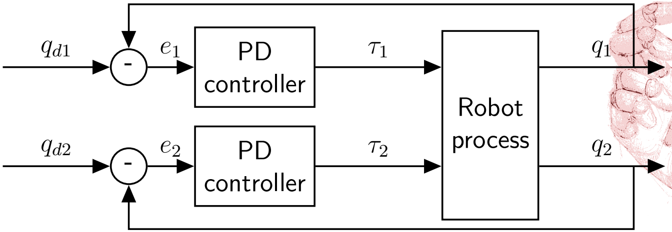
Setpoint - response: Joint coordinates

Can I have multiple inputs $(q_1, q_2, \ldots, q_n)$ to a single controller?

Robotic controllers

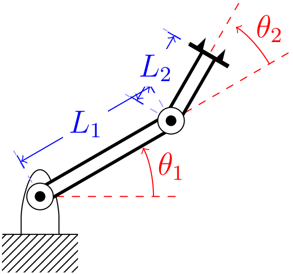
Independent joint control

With this control strategy, we control each joint individually. If we are controlling e.g. position, then we need to solve the inverse kinematics to define the joint coordinates. These are then used as our setpoints.



Robotic controllers

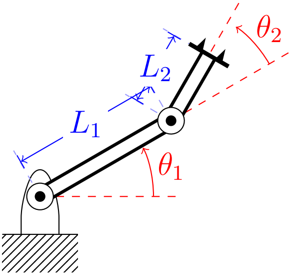
Independent joint control



The independent joint control considers that each joint moves independently and can therefore be controlled independently. Is this true?

Robotic controllers

Independent joint control

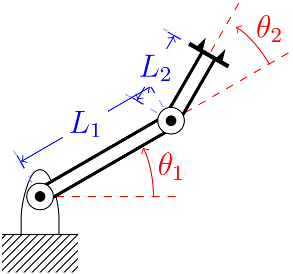


The independent joint control considers that each joint moves independently and can therefore be controlled independently. Is this true?

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

Robotic controllers

Independent joint control



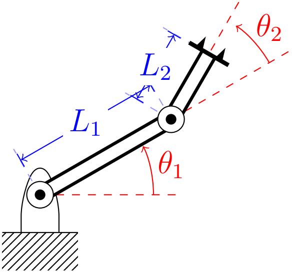
The independent joint control considers that each joint moves independently and can therefore be controlled independently. Is this true?

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

$$d_{1,1}\ddot{q}_1 + d_{1,2}\ddot{q}_2 + \cdots + d_{1,n}\ddot{q}_n + c_{1,1}\dot{q}_1 + c_{1,2}\dot{q}_2 + \cdots + c_{1,n}\dot{q}_n + g_1(q) = \tau_1$$

Robotic controllers

Independent joint control



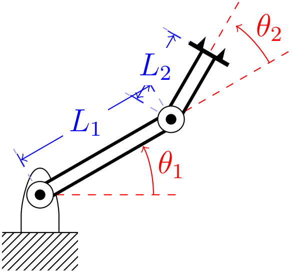
The independent joint control considers that each joint moves independently and can therefore be controlled independently. Is this true?

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

$$d_{1,1}\ddot{q}_1 + d_{1,2}\ddot{q}_2 + \cdots + d_{1,n}\ddot{q}_n + c_{1,1}\dot{q}_1 + c_{1,2}\dot{q}_2 + \cdots + c_{1,n}\dot{q}_n + g_1(q) = \tau_1$$
$$d_{2,1}\ddot{q}_1 + d_{2,2}\ddot{q}_2 + \cdots + d_{2,n}\ddot{q}_n + c_{2,1}\dot{q}_1 + c_{2,2}\dot{q}_2 + \cdots + c_{2,n}\dot{q}_n + g_2(q) = \tau_2$$

Robotic controllers

Independent joint control



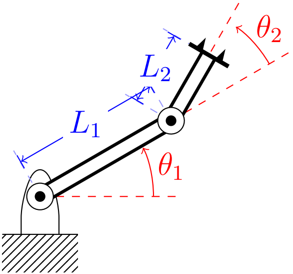
The independent joint control considers that each joint moves independently and can therefore be controlled independently. Is this true?

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

$$\begin{gathered}
d_{1,1}\ddot{q}_1 + d_{1,2}\ddot{q}_2 + \cdots + d_{1,n}\ddot{q}_n + c_{1,1}\dot{q}_1 + c_{1,2}\dot{q}_2 + \cdots + c_{1,n}\dot{q}_n + g_1(q) = \tau_1 \\
d_{2,1}\ddot{q}_1 + d_{2,2}\ddot{q}_2 + \cdots + d_{2,n}\ddot{q}_n + c_{2,1}\dot{q}_1 + c_{2,2}\dot{q}_2 + \cdots + c_{2,n}\dot{q}_n + g_2(q) = \tau_2 \\
\vdots
\end{gathered}$$

Robotic controllers

Independent joint control



The independent joint control considers that each joint moves independently and can therefore be controlled independently. Is this true?

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

$$\begin{gathered}
d_{1,1}\ddot{q}_1 + d_{1,2}\ddot{q}_2 + \cdots + d_{1,n}\ddot{q}_n + c_{1,1}\dot{q}_1 + c_{1,2}\dot{q}_2 + \cdots + c_{1,n}\dot{q}_n + g_1(q) = \tau_1 \\
d_{2,1}\ddot{q}_1 + d_{2,2}\ddot{q}_2 + \cdots + d_{2,n}\ddot{q}_n + c_{2,1}\dot{q}_1 + c_{2,2}\dot{q}_2 + \cdots + c_{2,n}\dot{q}_n + g_2(q) = \tau_2 \\
\vdots
\end{gathered}$$

Joints do not move independently!

Robotic controllers

Independent joint control

The independent joint control can take us rather far as long as:

- The motions performed are slow.
- If this is the case, then each controller can deal with the other joints motion as disturbances.
- We tune each controller diligently.

Independent joint control

Tuning the parameters

To tune the PID parameters analytically, we need to write a transfer function for each joint coordinate. To do that, we need the equation of motion for each joint.

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau$$

Independent joint control

Tuning the parameters

To tune the PID parameters analytically, we need to write a transfer function for each joint coordinate. To do that, we need the equation of motion for each joint.

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

Can we 'decouple' the joint coordinates?

Independent joint control

Tuning the parameters

If we want to decouple the joint coordinates, we need for each joint to consider the effects from the motion of the other joints as disturbances.

$$d_{ii}\ddot{q}_i + c_{ii}\dot{q}_i = \tau - w$$

Independent joint control

Tuning the parameters

If we want to decouple the joint coordinates, we need for each joint to consider the effects from the motion of the other joints as disturbances.

$$d_{ii}\ddot{q}_i + c_{ii}\dot{q}_i = \tau - w$$

Where the term w contains all the off diagonal elements from matrices D and C, and the gravity terms G.

Independent joint control

Tuning the parameters

If we design a PD controller, then the input signal becomes:

$$\tau_i = K_{Di}\dot{e}_i + K_{Pi}e_i$$

Independent joint control

Tuning the parameters

If we design a PD controller, then the input signal becomes:

$$\tau_i = K_{Di}\dot{e}_i + K_{Pi}e_i$$

And the equation of motion becomes:

$$d_{ii}\ddot{q}_i + (c_{ii} + K_{Di})\dot{q}_i + K_{Pi}q_i = K_{Pi}q_{di} - w$$

Independent joint control

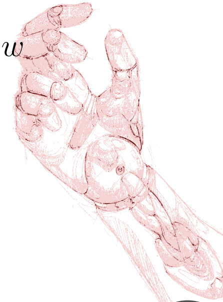
Tuning the parameters

If we design a PD controller, then the input signal becomes:

$$\tau_i = K_{Di}\dot{e}_i + K_{Pi}e_i$$

And the equation of motion becomes:

$$d_{ii}\ddot{q}_i + (c_{ii} + K_{Di})\dot{q}_i + K_{Pi}q_i = K_{Pi}q_{di} - w$$



Considering:

$$e_i = q_{di} - q_i, \dot{e}_i = \dot{q}_{di} - \dot{q}_i, \dot{q}_{di} = 0$$

Independent joint control

Tuning the parameters

$$d_{ii}\ddot{q}_i + (c_{ii} + K_{Di})\dot{q}_i + K_{Pi}q_i = K_{Pi}q_{di} - w$$

This represents a second-order system, which you should know how to calculate PD parameters for a stable fast response.

Independent joint control

Tuning the parameters

$$d_{ii}\ddot{q}_i + (c_{ii} + K_{Di})\dot{q}_i + K_{Pi}q_i = K_{Pi}q_{di} - w$$

This represents a second-order system, which you should know how to calculate PD parameters for a stable fast response.

What happens when our assumptions are not met?

Control theory

System linearization

Starting from the dynamic model of the robot:

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

We are seeking an input function that can convert this model into a linear closed loop system.

Control theory

System linearization

Starting from the dynamic model of the robot:

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

We are seeking an input function that can convert this model into a linear closed loop system.

What about this one:

$$\tau = D(q)a + V(q,\dot{q}), \text{ where } V(q,\dot{q}) = C(q,\dot{q})\dot{q} + G(q)$$

Control theory

System linearization

Starting from the dynamic model of the robot:

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

We are seeking an input function that can convert this model into a linear closed loop system.

What about this one:

$$\tau = D(q)a + V(q,\dot{q}), \text{ where } V(q,\dot{q}) = C(q,\dot{q})\dot{q} + G(q)$$

Resulting in: $\ddot{q} = a$

WOT???

Control theory

System linearization

WOT???

Remember that:

$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$

is just a *model* of the robot process, not the **actual** process.

WOT???

Remember that:

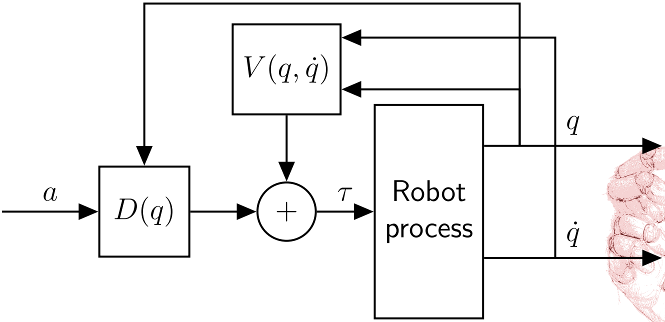
$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau$

is just a *model* of the robot process, not the **actual** process.

We are using our prior knowledge from the model to linearize our real process.

Control theory

System linearization



We can therefore 'linearize' our system, and then control it using state-space feedback for a linear system!

Control theory

System linearization

We can therefore 'linearize' our system, and then control it using state-space feedback for a linear system!

What should our input for the new system be? (i.e. a)

$$\ddot{q} = a$$

An *obvious* input would be:

$$a = -K_0 q - K_1 \dot{q} + r$$

And the closed loop form of our system becomes:

$$\ddot{q} + K_1 \dot{q} + K_0 q = r$$

Where r is our reference.

Control theory

Computed torque control

$$\ddot{q} = a$$

An *obvious* input would be:

$$a = -K_0 q - K_1 \dot{q} + r$$

And the closed loop form of our system becomes:

$$\ddot{q} + K_1 \dot{q} + K_0 q = r$$

Where r is our reference. Choosing an r to follow the desired trajectories of $q, \dot{q}, \ddot{q}$ like this:

$$r(t) = \ddot{q}_d(t) + K_0 q_d(t) + K_1 \dot{q}_d(t)$$

We end up with zero tracking error.

Control theory

Joint position control?

Could we control something else, more meaningful than joint positions?

Control theory

Joint position control?

Could we control something else, more meaningful than joint positions?

- End effector pose

Control theory

Joint position control?

Could we control something else, more meaningful than joint positions?

- End effector pose
- Joint velocities

Control theory

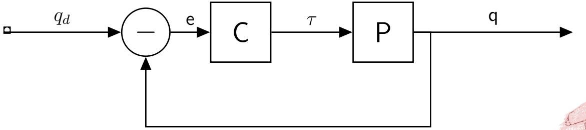
Joint position control?

Could we control something else, more meaningful than joint positions?

- End effector pose
- Joint velocities
- End effector velocities

Control theory

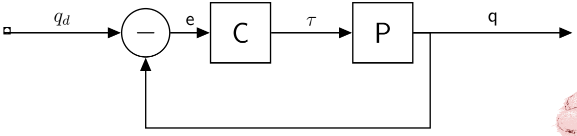
Joint position control



Controlling joint position

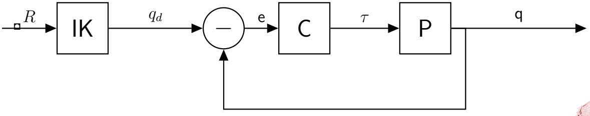
Control theory

End effector pose control



Control theory

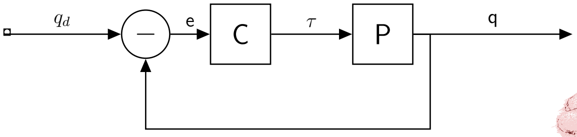
End-effector pose control



Controlling end-effector position

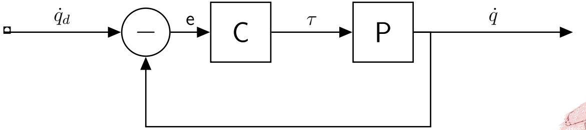
Control theory

Joint velocity control



Control theory

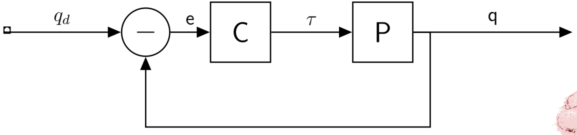
Joint velocity control



Controlling joint velocity

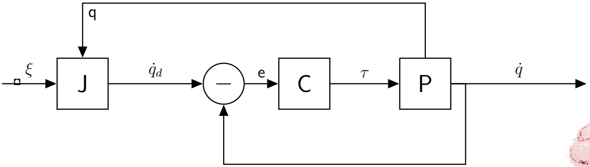
Control theory

End-effector velocity control



Control theory

End-effector velocity control



Controlling end-effector velocity

Questions?