MECE 3350U
Control Systems

Lecture 14
Implementing PID Controllers

Videos in this lecture

Lecture: https://youtu.be/gHn6pdcAzvc

Exercise 68: https://youtu.be/Rd2bjM9QhIU

Exercise 69: https://youtu.be/ATmoujzc6-w

Exercise 70: https://youtu.be/fBDHh3DTI1o

Exercise 71: Not in the lecture.

Exercise 72: https://youtu.be/GJQket0JRT8

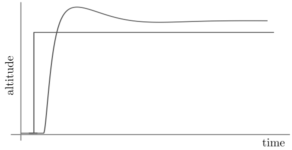
Outline of Lecture 14

By the end of today's lecture you should be able to

- Tune a PID controller
- Understand the limitations of PID controllers
- Understand how to implement PID controllers

Applications

The open-loop step response of the Osprey Tiltrotor aircraft to a step-input is shown in the graph.



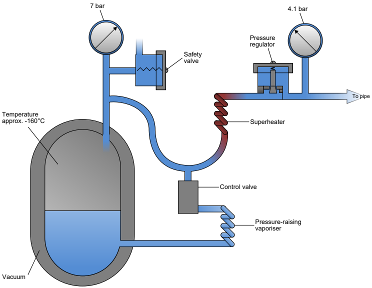
Implement a PID controller to eliminate the steady-state error. How do we select the PID gains?

Does the saturation of the propeller angle affect the performance of the controller?

Applications

A vacuum insulated evaporator allows the bulk storage of cryogenic liquids for industrial and medical applications.

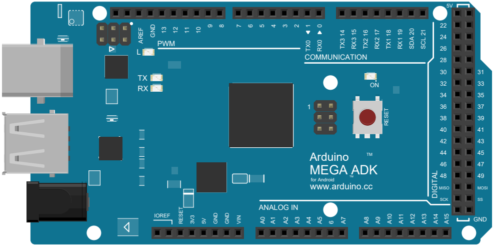
Without functioning insulation, the stored liquid will rapidly warm and undergo a phase transition to gas, increasing in volume and potentially causing a catastrophic failure.



Develop a PID controller to maintain a constant pressure in the vessel without knowing it transfer function.

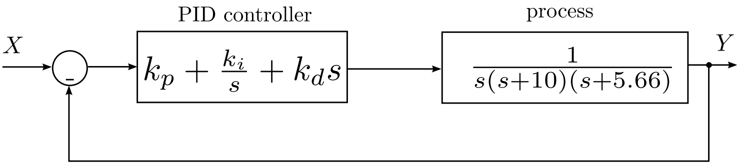
Applications

Can we implement a PID controller using a microcontroller?



PID tuning

Consider the following control scheme. Our object is to find suitable gains for the PID controller.



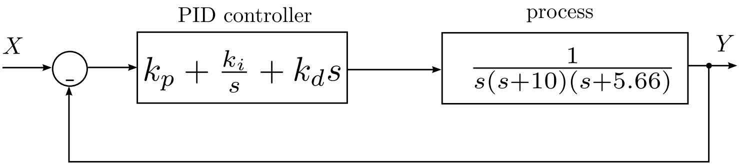
Requirements:

→ Small overshoot (less than 15%)

→ Settling time is less than 3 sec.

→ Zero steady state error.

Manual PID tuning

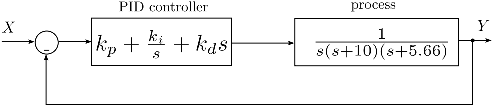


Step 1: Find the critical proportional gain k_p before instability.

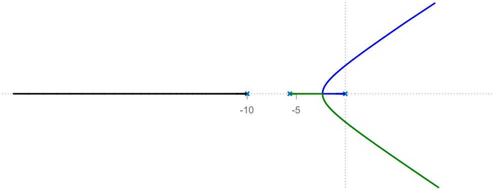
→ Set $k_i = k_d = 0$.

→ Slowly increase k_d to the edge of stability

Manual PID tuning



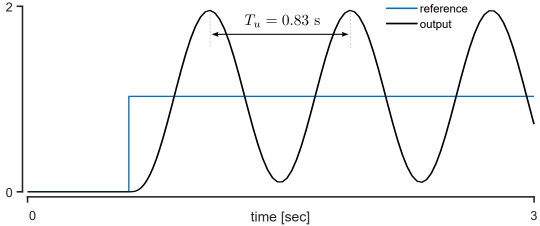
$$1 + k_p \frac{1}{s(s+10)(s+5.66)} = 0$$



How can we calculate the **ultimate gain** k_u?

Manual PID tuning

Ultimate gain: $k_u = 886.356$



Step 2: Reduce k_p to achieve a step response with approximately a quarter amplitude decay.

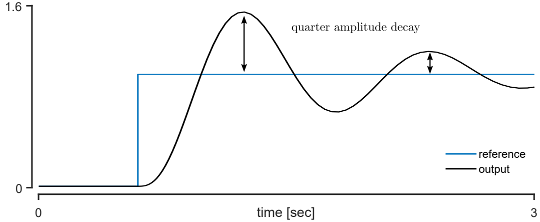
$\rightarrow$ I.e,: the overshoot drops to 25% of the initial value after one period.

$\rightarrow$ As an initial approximation set $k_p = 0.5k_u$.

$\rightarrow$ Period of oscillations $T_u = 0.83$ sec $\Rightarrow$ **Ultimate period**

Manual PID tuning

Quarter amplitude decay gain: $k_p \approx 0.5k_u = 886.356 \times 0.5$



Step 2: Set the proportional gain and manually analyse the derivative gain.

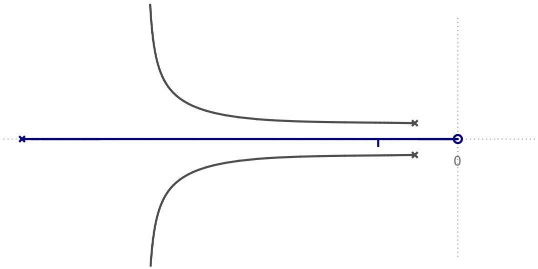
$\rightarrow$ For $k_d > 0$, we have

$$1 + k_d \frac{s}{s(s+10)(s+5.66)+k_p} = 0$$

Manual PID tuning

$k_p = 370, \; k_d > 0, \; k_i = 0$

$$1 + k_d \frac{s}{s(s+10)(s+5.66)+k_p} = 0$$



As k_d increases:

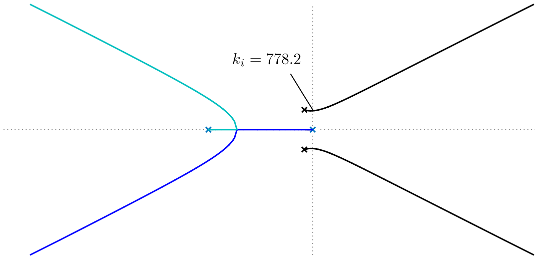
$\rightarrow$ Imaginary poles move to the left: Damping ratio increases

$\rightarrow$ For large values of k_d, the real pole dominates the response

Manual PID tuning

Step 3: $k_p = 370,\ k_d = 0,\ k_i > 0$

$$1 + k_i \frac{1}{s[s(s+10)(s+5.66) + k_p]} = 0$$



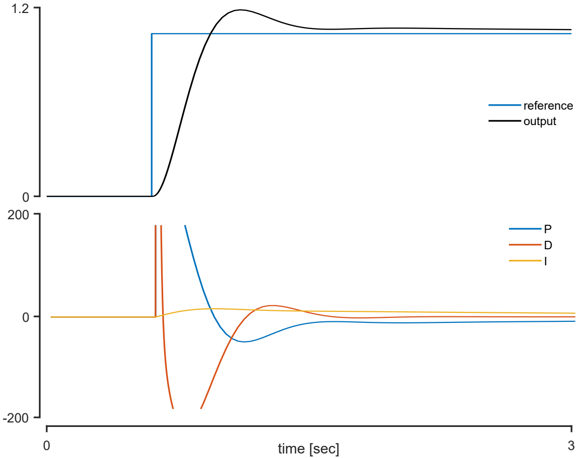
As k_i increases:

→ Complex poles move to the right: Higher overshoot, higher settling time

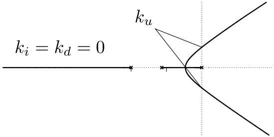
→ What should we do now? Open question since 1936!

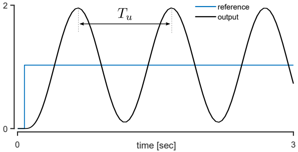
Manual PID tuning

Step response for $k_d = 370,\ k_i = 100,\ k_d = 60.$



Ziegler-Nichols PID tuning - Method 1



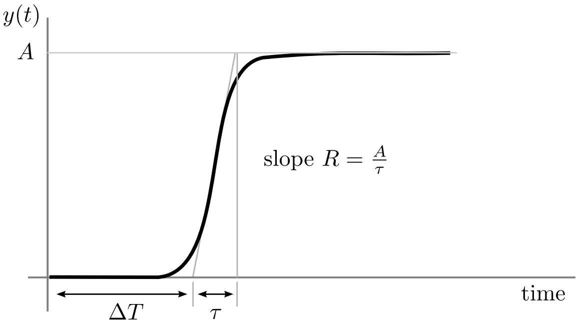


Controller type	k_p	k_i	k_d
Proportional $C(s) = k_p$	$0.5k_u$	0	0
Proportional-integral $C(s) = k_p + k_i s^{-1}$	$0.45k_u$	$\frac{0.54k_u}{T_u}$	0
PID $C(s) = k_p + k_i s^{-1} + k_d s$	$0.6k_u$	$\frac{1.2k_u}{T_u}$	$\frac{0.6K_u T_u}{8}$

Ziegler-Nichols PID tuning - Method 2

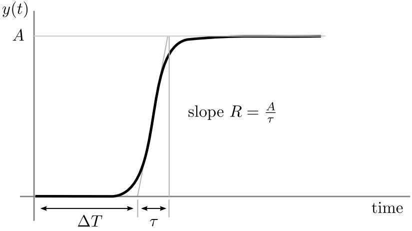
Many systems cane be approximated by the step response of

$$\frac{Y(s)}{U(s)} = \frac{Ae^{-\Delta ts}}{\tau s + 1} \tag{1}$$



This is a first order system with a time delay of Δt sec.

Ziegler-Nichols PID tuning - Method 2



Controller type	k_p	k_i	k_d
Proportional $C(s) = k_p$	$\frac{1}{R\Delta T}$	0	0
Proportional-integral $C(s) = k_p + k_i s^{-1}$	$\frac{0.9}{R\Delta T}$	$\frac{0.27}{R\Delta T^2}$	0
PID $C(s) = k_p + k_i s^{-1} + k_d s$	$\frac{1.2}{R\Delta T}$	$\frac{0.6}{R\Delta T^2}$	$\frac{0.6}{R}$

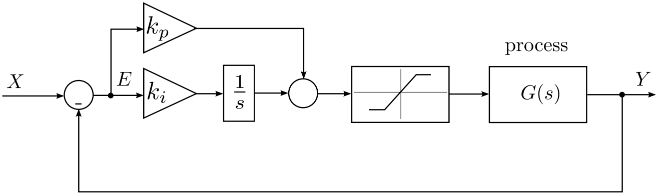
Integrator anti-windup

In many control systems, the output of the actuator can saturate.

→ A valve saturates when it is fully open

→ Control surfaces of an aircraft cannot bend beyond certain angles

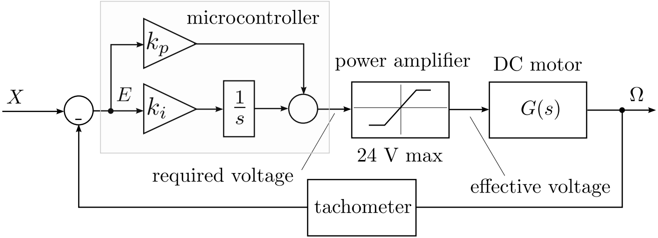
→ The output voltage of a motor speed controller is limited

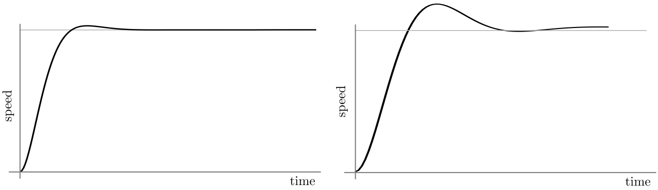


What are the effects of saturation in the controller?

Integrator anti-windup

Example: Consider a PI speed controller for a DC motor.



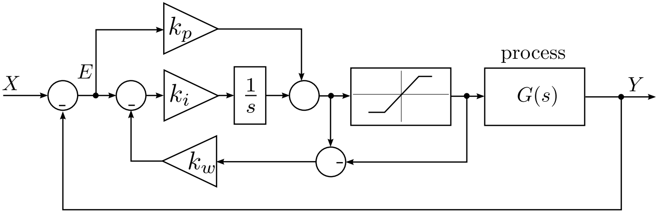


Integrator anti-windup

Solution 1: If the controller is implemented digitally:

$$\text{if } |u| \geq u_{max}, \text{ set } k_i = 0$$

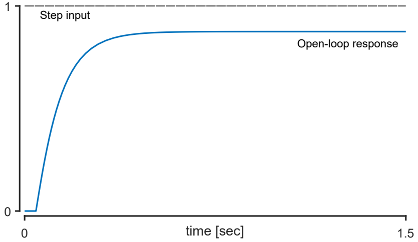
Solution 2: Anti windup loop



k_w can be determined experimentally.

Exercise 68 - Using Matlab

As a control engineer, you were required to design a cruise speed controller for a supersonic aircraft. The dynamic model of the system is unknown. The open-loop response of the aircraft to a step-input signal was measured and is shown in the graphic.



Implement a PID controller and find the optimal gains based on the Ziegler-Nichols tuning method.

Exercise 68 - continued

⇒ Open the Matlab file "PID-tuning-ZNI.m"

⇒ Open the Simulink file "PID-tuning-ZN.slx"

⇒ Run the Matlab script

⇒ Based on the open-loop response, determine the PID gains

⇒ Tune the controller

⇒ Add a small disturbance to the system and compare the open and closed loop response ($D = 0.5$)

Exercise 69 - Using Matlab

Consider a plant in an unit feedback configuration with the following transfer function for small signals

$$G(s) = \frac{1}{s}$$

and PI controller

$$D(s) = 2 + \frac{4}{s}.$$

Study the effect of windup and antiwindup on the response of the system.

Exercise 69 - continued

⇒ Open the Matlab file "PID-windup-1.m"

⇒ Open the Simulink file "PID-windup.slx"

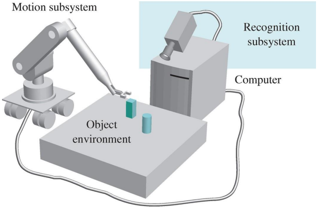
⇒ Run the Matlab script

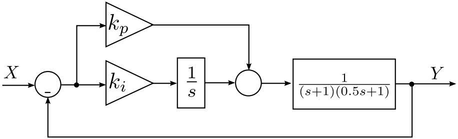
⇒ For kw = 0, there is no antiwindup

⇒ Increase kw and observe the effect on the overshoot and control effort

Exercise 70 - Using Matlab

A mobile robot using a vision system as the measurement device is shown. Design the controller so that the percent overshoot for a step input is less than 5% and the settling time is less than 6 seconds.



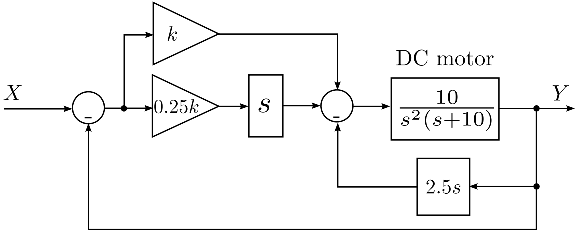


Note: there are several possible solutions to this problem.

Exercise 71 - Using Matlab

Exercise 72

A welding torch is remotely controlled to achieve high accuracy while operating in hazardous environments. A model of the arm control is shown.



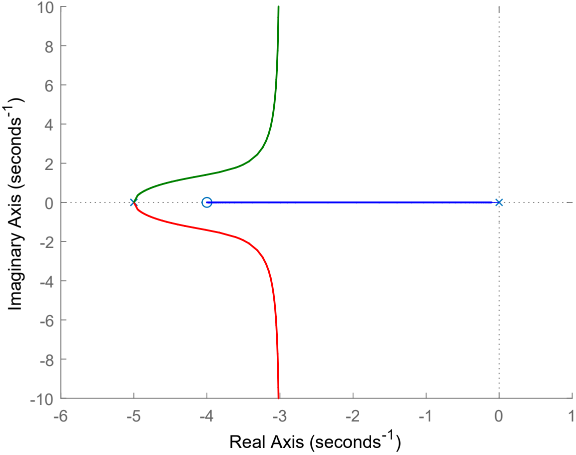
Select k to provide a satisfactory step response with P.O. $< 5\%$.

Exercise 72 - continued

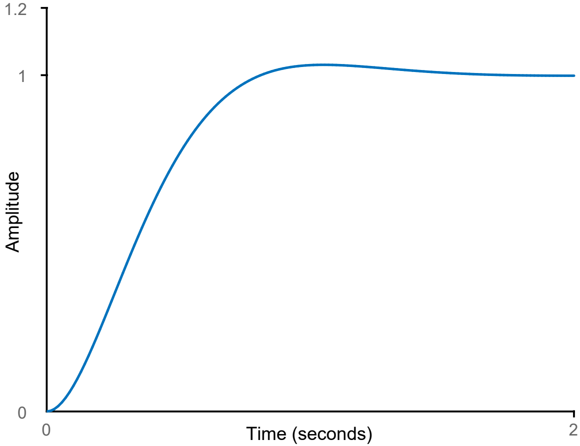
Recall that P.O. $= 100e^{-\zeta\pi/\sqrt{1-\zeta^2}}$

Exercise 72 - continued

Exercise 72 - continued



Exercise 72 - continued



Next class...

- Midterm examination
- More practice exercises can be found in Lecture 15