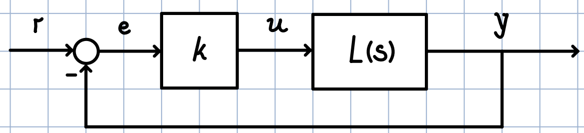
8. Recital 08.11.24

Recap

We started to look at how a system reacts when introducing feedback:

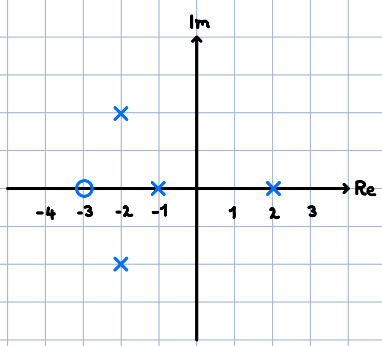


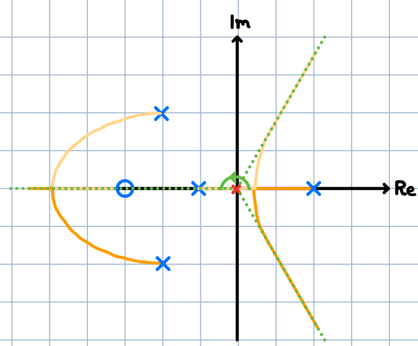
Specifically we looked at how changing the gain k affected the closed loop poles. We noticed that we can determine the location of the closed loop poles based on the open loop poles. This allows us to quickly see if some controler is feasible or not. The results can be summarized in the root locus.

To sketch the root locus we need to consider the following rules:

- The closed loop poles are symmetric about the real axis.
- The number of closed loop poles is equal to the number of open loop poles.
- The closed loop poles approach the open loop poles as $k \to 0$
- The closed loop poles approach the open loop zeros as $k \to \infty$. If theres more poles then zeros they go "to infinity".

Reviewing examples:





#Poles: 4 #Zeros: 1

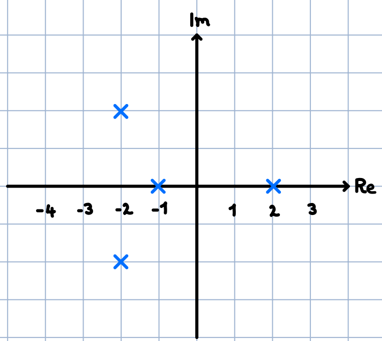
Asymptote centroid: $\dfrac{-1-2-2+2+3}{4-1} = 0$

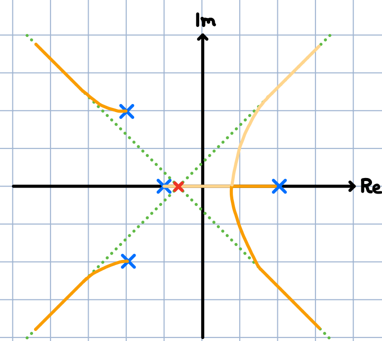
Asymptote angles: $n = \{0;1;2\}$

$\dfrac{2\cdot 0+1}{4-1}\cdot 180° = 60°$

$\dfrac{2\cdot 1+1}{4-1}\cdot 180° = 180°$

$\dfrac{2\cdot 2+1}{4-1}\cdot 180° = 300°$





#Poles: 4 #Zeros: 0

Asymptote centroid: $\dfrac{-1-2-2+2}{4} = -\dfrac{3}{4}$

Asymptote angles: $n = \{0;1;2;3\}$

$\dfrac{2\cdot 0+1}{4}\cdot 180° = 45°$

$\dfrac{2\cdot 1+1}{4}\cdot 180° = 135°$

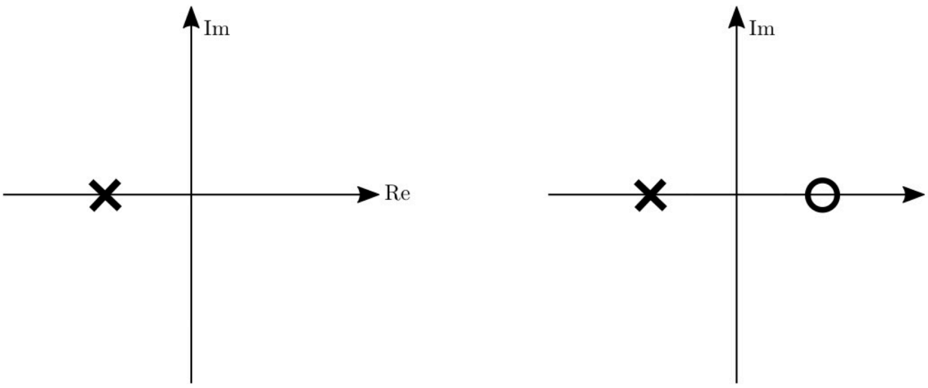
$\dfrac{2\cdot 2+1}{4}\cdot 180° = 225°$

$\dfrac{2\cdot 3+1}{4}\cdot 180° = 315°$

Exam question: W19

Question 24 *Mark all correct statements. (2 Points)*

Consider the following pole/zero-diagrams of two plants $P(s)$:



Open-Loop System 1 | Open-Loop System 2

Evaluate which of the following statements are correct:

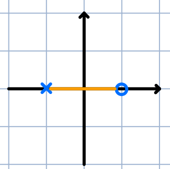
[A] Any controller $C(s) = k$ with $k > 0$ will stabilize open-loop system *2*.

[B] A controller $C(s) = \frac{k}{s}$ with $k > 0$ will destabilize open-loop system *2* already for very small gains k.

[C] Any controller $C(s) = k$ with $k > 0$ will stabilize open-loop system *1*.

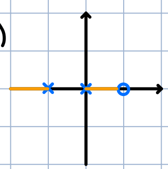
[D] A controller $C(s) = \frac{k}{s}$ with $k > 0$ will destabilize open-loop system *1* already for very small gains k.

a.)



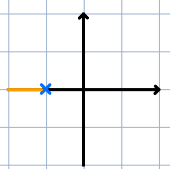
If k is large pole is in RHP, i.e. destabilize X

b.)



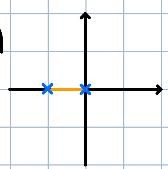
for every k>0 the pole at the origin will be in the RHP. ✓

c.)



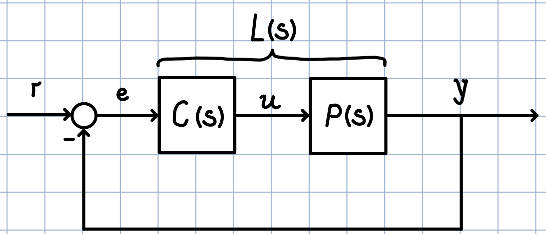
System is always stable. ✓

d.)



Poles are always in LHP for k>0 X

We also introduced some new notation for TFs of a closed loop system. In a system with some controler $C(s)$ and a plant $P(s)$ we can define some important TFs.



→ Open loop TF:

$$L(s) = C(s)\,P(s)$$

→ Complementary Sensitivity:

maps $r \rightarrow y$

$$T(s) = \frac{L(s)}{1 + L(s)}$$

→ Sensitivity:

maps $r \rightarrow e$

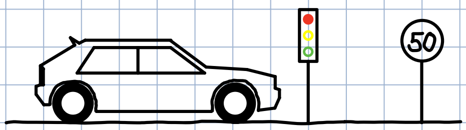
$$S(s) = \frac{1}{1 + L(s)}$$

Time-Domain Specifications

So far, we assessed wether we can stabilize our system with feedback, based on the open loop TF $L(s)$.

Next to stability, there are many different requirements that are of interest when designing a controler.

Consider a car driving with cruise control:



Assume we can modell the car with a stable first-order system.

$$G(s) = \frac{1}{\tau s + 1} \iff \begin{cases} \dot{x} = -\frac{1}{\tau}x + \frac{1}{\tau}u \\ y = x \end{cases}$$

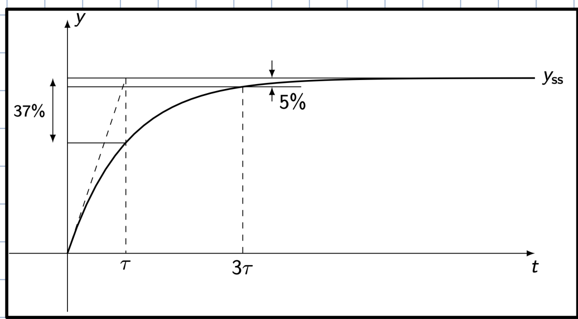
pole at $s = -\frac{1}{\tau}$

We are standing at a red traffic light and as soon as it turns green we want to accelerate to the max. allowed velocity of 50 km/h. So the reference changes from 0 to 50 in the instant that the light turns green. This corresponds to a scaled step-response to our system:

$$y(t) = Ce^{At}\cancelto{0}{x_0} + C\int_0^t e^{A(t-p)}B\cancelto{50}{u(p)}\,dp + \cancelto{0}{D}u(t)$$

$$= 50\int_0^t e^{-\frac{1}{\tau}(t-p)}\frac{1}{\tau}\,dp$$

$$= 50\left(1 - e^{-\frac{t}{\tau}}\right)$$



We can see that the behavior is dependend on τ. We can define a settling time T_d, i.e. the time it takes for the system to get within d% of the steady-state.

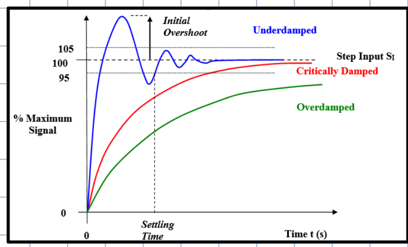
$$T_d = \tau \ln(100/d)$$

This measure of T_d can help us adjust the behavior of the system, such that it doesn't accelerate too fast or too slow. Since T_d is directly proportional to τ, and τ influences the location of the pole of $G(s)$, this time-domain specification translates to a constraint on the location of the poles of the system.

This means that we can choose our controles in such a way that the time domain specification is met, by controling where our poles go.

Usually the first order aproximation from above is not enough to describe a system. So let's approximate with a second order system. We can define our TF to be:

$$G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \iff \begin{cases} \dot{x} = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix} x + \begin{bmatrix} 0 \\ \omega_n^2 \end{bmatrix} u \\ y = x \end{cases}$$



for an underdamped system ($\zeta<1$) with zero initial condition we get:

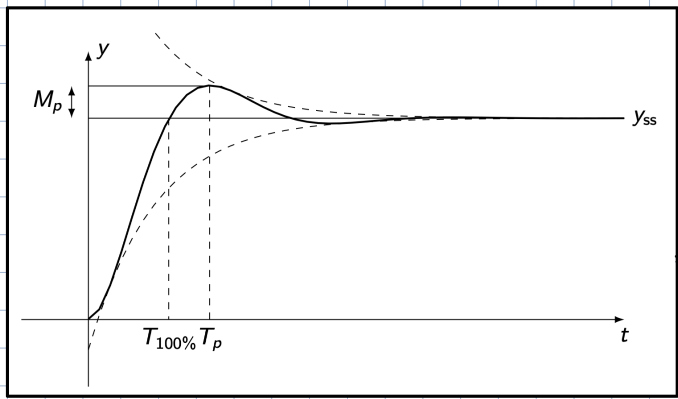
$$y(t) = 1 - \frac{1}{\cos\varphi} e^{\sigma t} \cos(\omega t + \varphi)$$

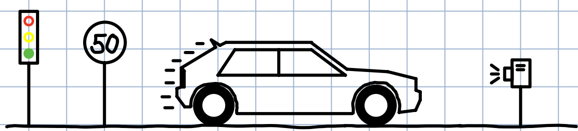
Again we can define important characteristics of the step response:

→ Time to peak: $T_p = \frac{\pi}{\omega}$

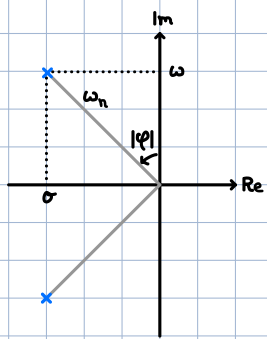
→ Peak overshoot: $M_p = e^{\frac{\sigma\pi}{\omega}}$

→ Rise time: $T_{100\%} = \frac{\frac{\pi}{2} - \varphi}{\omega} \approx \frac{\pi}{2\omega_n}$

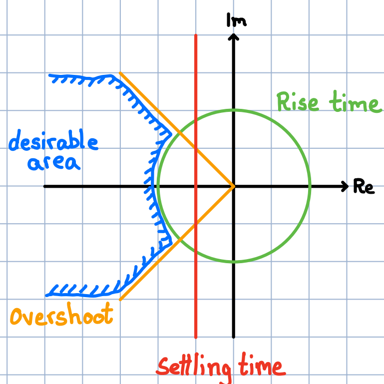




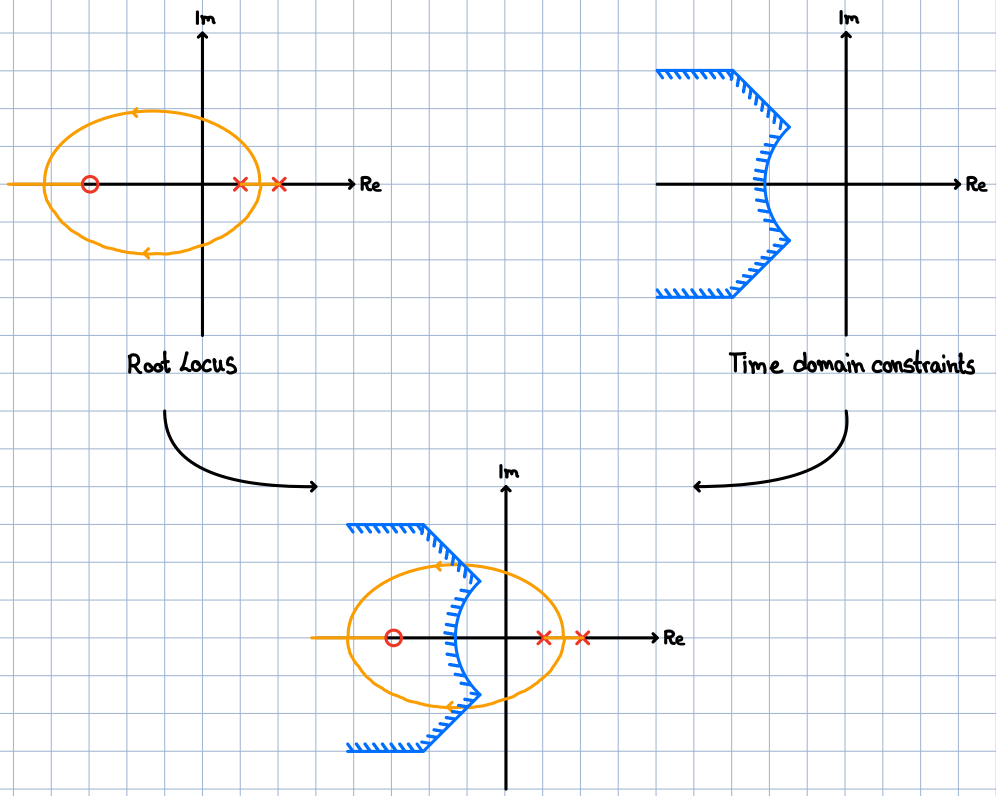
We can map out the poles in the complex plane and observe that each component of the time response affects the poles differently.



That means that we can again, take requirements in the time domain and transform them into constraints on the location of the poles of the system. When designing our closed loop system we can take these restrictions into account.



A way to relate these restrictions to the root locus is overlaying both

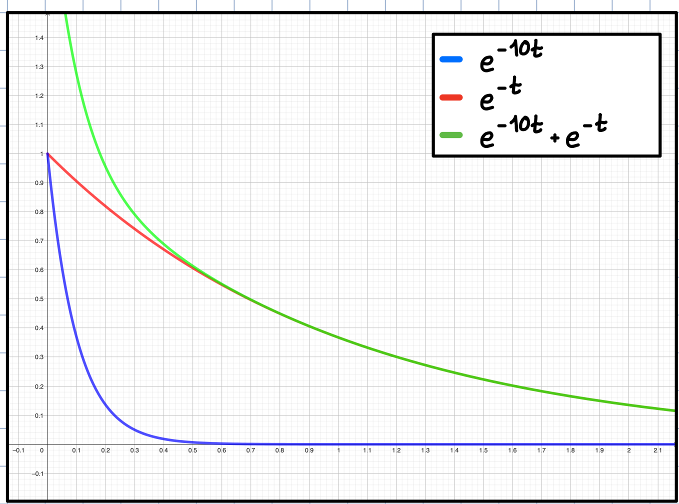


we can now choose k s.t. the poles fall in the restricted area.

Dominant Pole Approximation:

What if our system has more than two poles?

Often we can approximate the higher order system. Remember that every pole corresponds to an exponential. The real part of the pole indicates how fast the exponential grows or decays. For poles in the LHP we can say that poles further away from the imaginary axis are "faster" since they decay at a higher rate.



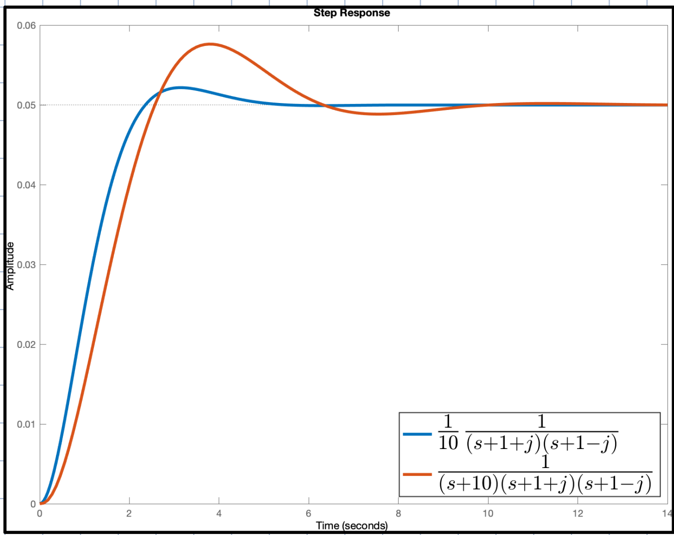
We observe that the be combined behavior can be well approximated by the slower pole, i.e. the pole closer to the imaginary axis.

Example:

Given $G(s) = \dfrac{1}{(s+10)(s+1+j)(s+1-j)}$, we can neglect the pole at -10 since it is faster than the other two.

We can thus approximate $G(s)_{dom} = \dfrac{1}{10} \dfrac{1}{(s+1+j)(s+1-j)}$.

The factor $\frac{1}{10}$ ensures that the steady-state value ramains the same



Exam question : HS 17

Box 7: Questions 34, 35

You are considering the following transfer function:

$$L(s) = \underbrace{e^{-0.001 \cdot s}}_{\text{ignore this}} \cdot \frac{(s+1)\left(s+\frac{1}{100}\right)}{\left(s+\frac{1}{2}\right)(s+3)(s+4)(s+30)}$$

Question 34 *Choose the correct answer. (1 Point)*

To simplify the calculations, you approximate the transfer function with a simpler model. Select the approximation which can reasonably be assumed to model the behavior of $L(s)$ for low frequencies ω with negligible error.

- ☐ $L(s) \approx \frac{1}{3000} \cdot \frac{(s+1)}{(s+\frac{1}{2})(s+3)(s+4)}$
- ☐ $L(s) \approx \frac{1}{30} \cdot \frac{(s+1)}{(s+\frac{1}{2})(s+3)(s+4)}$
- ☒ $L(s) \approx \frac{1}{30} \cdot \frac{(s+1)(s+\frac{1}{100})}{(s+\frac{1}{2})(s+3)(s+4)}$
- ☐ $L(s) \approx \frac{(s+1)(s+\frac{1}{100})}{(s+\frac{1}{2})(s+3)(s+4)}$

i. Fastest pole at -30, to be eliminated.

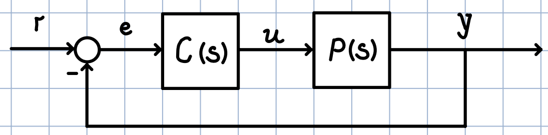
ii. correction factor needed, such that steady state remains the same. $\left(\frac{1}{30}\right)$

Steady-state error:

From a few weeks ago, we know the final value theorem,

$$\lim_{t\to\infty} y(t) = \lim_{s\to 0} s\,Y(s) = \lim_{s\to 0} s\,G(s)\,U(s)$$

to determine the steady state output of a system. Can we do something similar with the closed loop system?



Let's try to compute the steady-state error. We already defined the TF from $r \to e$ as $S(s) = \dfrac{1}{1+L(s)}$.

We can now write:

$$\lim_{t\to\infty} e(t) = \lim_{s\to 0} s\,S(s)R(s) = \lim_{s\to 0} s\,\frac{1}{1+L(s)}R(s) = e_{ss}$$

Looking at $L(s)$ in Bode-Form we see that:

$$L(s) = \frac{k_{Bode}}{s^{\delta}} \frac{\left(\frac{s}{-z_1}+1\right)\left(\frac{s}{-z_2}+1\right)\ldots\left(\frac{s}{-z_m}+1\right)}{\left(\frac{s}{-p_1}+1\right)\left(\frac{s}{-p_2}+1\right)\ldots\left(\frac{s}{-p_{n-q}}+1\right)} \xrightarrow{s\to 0} L(0) = \frac{k_{Bode}}{s^{\delta}} \text{ will go} \to \infty$$

$\delta \triangleq$ number of poles at the origin, type of system.

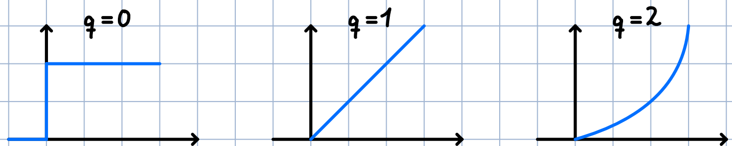
if $L(0) \to \infty$ then $\lim_{s\to 0} s\,\dfrac{1}{1+L(s)}R(s) = e_{ss} \to 0$

That means that in order to obtain a steady-state error of 0, we need a pole at the origin, also known as integrator.

Depending on the input, we potentially need more than one integrator. When considering the steady-state error we usually look at ramp inputs. These are given by:

$$r(t) = \frac{1}{q!}t^{q} \quad \text{or in the s-domain} \quad R(s) = \frac{1}{s^{q}}$$

q → order of ramp.



Depending on the order of the ramp and the type of the system we get different steady-state errors

e_{ss}	$q=0$	$q=1$	$q=2$
Type 0	$\dfrac{1}{1+k_{\text{Bode}}}$	∞	∞
Type 1	0	$\dfrac{1}{k_{\text{Bode}}}$	∞
Type 2	0	0	$\dfrac{1}{k_{\text{Bode}}}$

Example:

Given an open loop TF $L(s) = \frac{s+5}{s^2+3s}$, calculate the steady-state error to a 1^{st} order ramp.

Let's bring L into Bode form: $L(s) = \frac{5}{3}\,\frac{1}{s}\,\frac{\frac{s}{5}+1}{\frac{s}{3}+1}$, where $k_{Bode} = \frac{5}{3}$

We can see that this a system of type 1, since $\delta = 1$. We also know that the input ramp is of order 1.

Using the table from above we get:

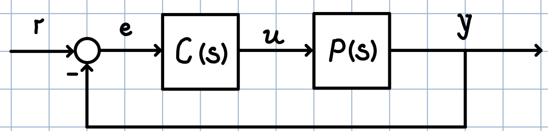
$$e_{ss} = \frac{1}{k_{Bode}} = \frac{3}{5}$$

The final value theorem would yield the same result.

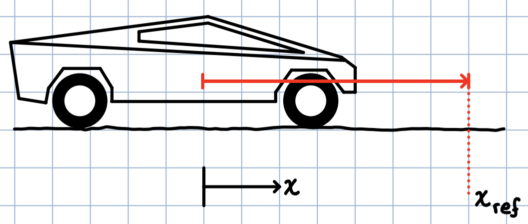
PID Control

a brief introduction.

Reall what our controler is designed to do:



It takes the error, i.e. the difference between the current output and the reference we want to achieve, and transforms it into an input to the plant. Consider an eletric car and it's closed loop representation



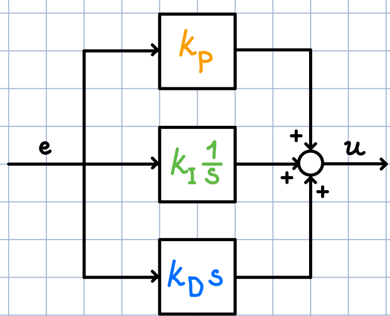
In this context the controler will take the desired position x_{ref} and susbstract the actual position x, generating the error, i.e. how far away we are from x_{ref}. The controler must now interpret this error and transform it to an input to the plant (the car). This could be a voltage to the motors. So the controler transforms the error to an input. The question is, how do we convert an error into a command?

One way to do it is with PID control.

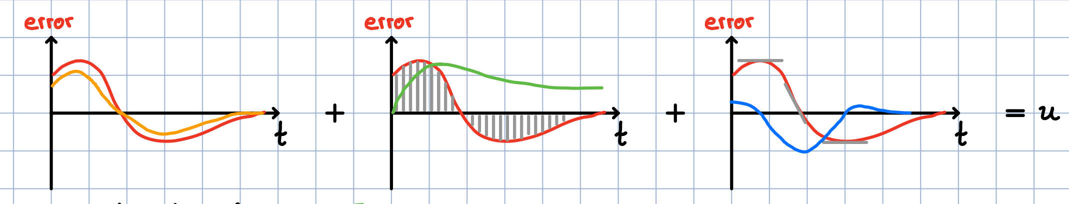
PID control is everywhere! What does PID stand for?

P → Proportional
I → Integral
D → Derivative

each term handles the error differently



How do we interpret all of these terms?



P: takes the value of the error and scales it with k_P

I: takes the integral of the error and scales it with k_I

D: takes the derivative of the error and scales it with k_D

The TF of such a controller is then given by:

$$C(s) = k_P + \frac{k_I}{s} + k_D s = \frac{k_D s^2 + k_P s + k_I}{s}$$

or in the time domain:

$$u(t) = k_P e(t) + k_I \int_0^t e(\tau)\, d\tau + k_D \dot{e}(t)$$

The challange is then to find the right gains that fullfill the requirements. This is called PID tuning.

Prof. Frazzoli's PID design recipie:

1. Assume proportional control **P**.
2. Draw the root locus.
3. If the root locus does NOT go through the "good" region ⇒ Need to add a **D** term. Go back to step 2.
4. Choose a gain that places the dominant poles in the "good" region.
5. If the steady-state error is too large ⇒ Need to add a **I** term. Go back to step 2.
6. If **PID** is not enough, stay tuned for more advanced methods.

The effects of each Term can be summarized.

→ Proportional

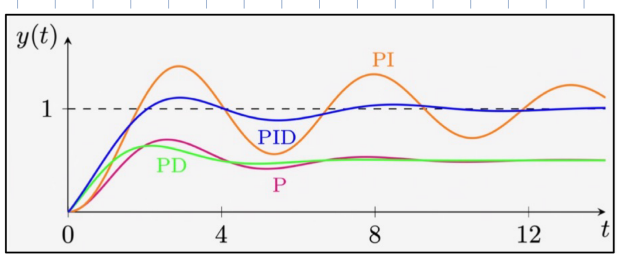
Decrease steady state error
Increase close-loop bandwidth
Increase sensitivity to noise
Can reduce stability margin for high order systems.

→ Integral

Eliminates the steady state error to a step (if CL is stable)
Reduces stability margins.

→ Derivative

Reduces overshoot, increases damping
Improves stability margins
Increase sensitivity to noise



Exam Problem: S19

Question 29 *Mark all correct statements. (2 Points)*
Mark all the correct statements about the behavior of PID controllers.

- [x] The derivative action can be highly sensitive to measurement noise, thus it is often fed with a filtered error signal.
- [ ] Inserting an integral action always helps to achieve better tracking error.
- [x] As the derivative gain increases, the steady-state error is not affected.
- [ ] ~~As the proportional gain increases, the phase margin increases.~~
- [ ] The derivative action amplifies overshoots in the system.
- [ ] High proportional gain does not affect the sensitivity to noise

Check summary above