Week 5 Slides

Recap of some really important control stuff …

- A **transfer function** is the ratio "output over input" using the Laplace form of those signals – zero initial conditions are assumed
$$G(s) = \frac{Y(s)}{U(s)}$$
- The steady-state gain (settling output signal from a unit step in) is found by setting $s = 0$. e.g., if
$$G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

  the steady-state gain=1.

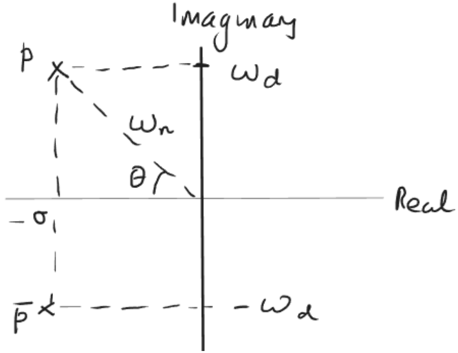
- The **poles** of a transfer function are found by setting the denominator polynomial equal to zero. This tells us a about the time response for various inputs, especially to check **stability**.

- For example: we obtain the following transfer function from the equations of motion

$$G(s) = \frac{s + 12}{s^2 + 8s + 12}$$

The pole is found from $s^2 + 8s + 12 = 0$. This gives $s = -2, -6$ and (also knowing the steady state gain) we can easily sketch the step response.

- The real part of the pole tells us about settling time ($t_s = 3/\sigma$)
- The real part **must be negative** for a stable system
- The imaginary part tells us about oscillation frequency

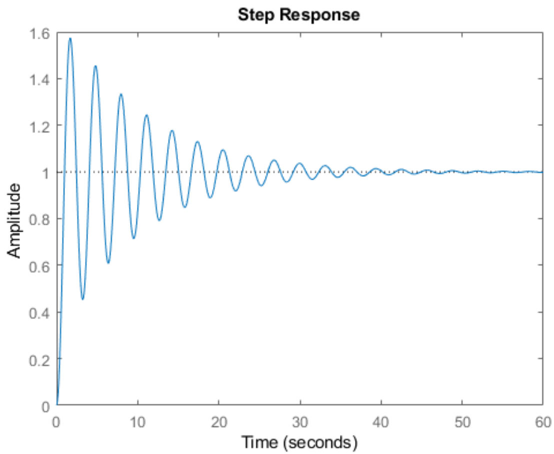


$$\boxed{\begin{aligned} \omega_d &= \omega_n\sqrt{1-\zeta^2} \\ \sigma &= \zeta\omega_n \\ \omega_n &= |p| \\ \cos\theta &= \zeta \end{aligned}}$$

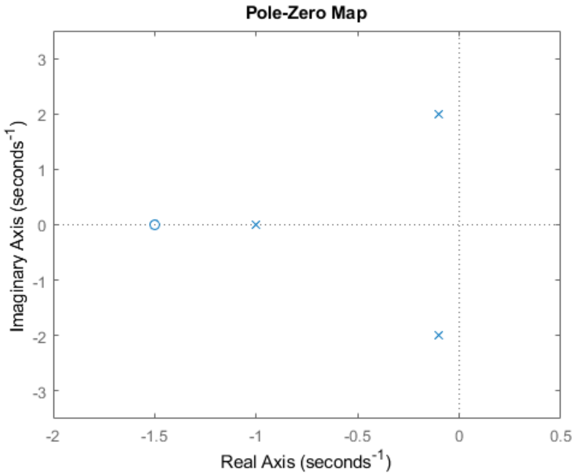
This week the main topic is to see how **feedback gains change the closed-loop poles**. This can be used to design a well-behaved control system based on theory. Also (see below) **feedback does not affect zeros**.

Relative stability is another term for "well-behaved", or "good performance". A system can be stable but still have poor behaviour.

Suppose a closed-loop control system has this response:



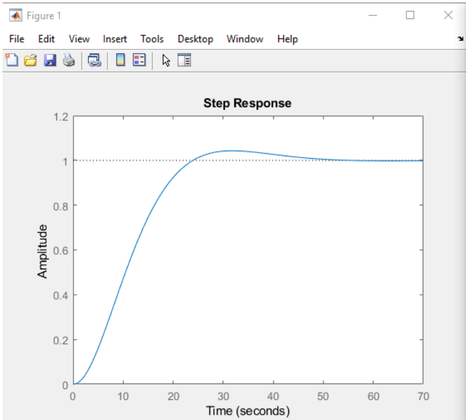
It’s stable and has zero steady-state error, but the result is terrible!

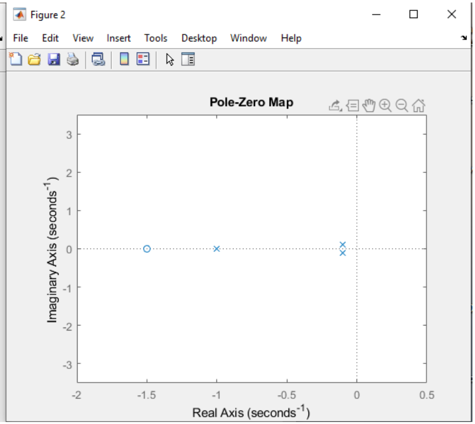


The poles seem to be too close to the imaginary axis (makes for slow settling) but also the **angle** from the origin to the dominant poles is a problem ($\cos\theta = \zeta$).

Here the dominant pole is at -0.1+2j , so $\theta = 87°$ $(\tan\theta = 20)$
The damping ratio ζ is 0.05.

Suppose we can move the poles towards the real axis to –0.1+0.1j. The real part is unchanged but we reduce the frequency.





This reduces angle θ to $= 45^{\circ}$ and hence increases the damping ratio from 0.05 to 0.707 – this is a nice damping ratio with only around 5% overshoot.

There may be a problem with slow settling, it depends on the application.

The settling time is now $3/\sigma$ = 30 seconds. It could be OK for steering a supertanker for example.

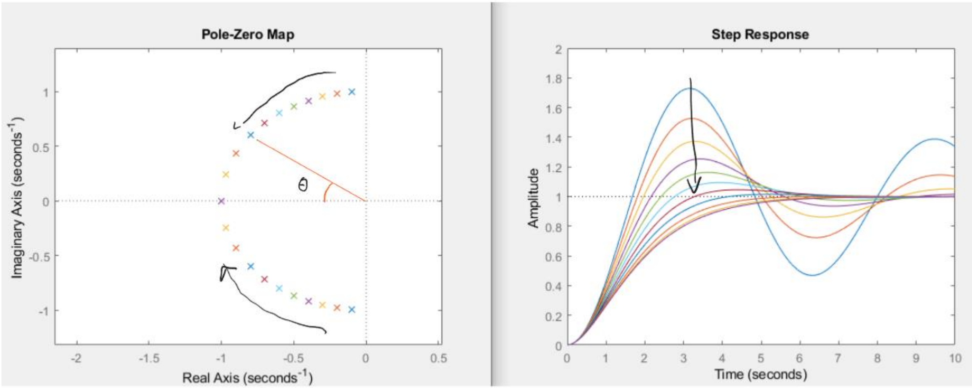
We see that as the angle θ decreases the step response becomes less oscillatory. Again the ideal value for ζ depends on the application.

For a surgical robot it may be better to have very little overshoot, e.g. ζ=0.8.

How theta (or zeta) affects the step response

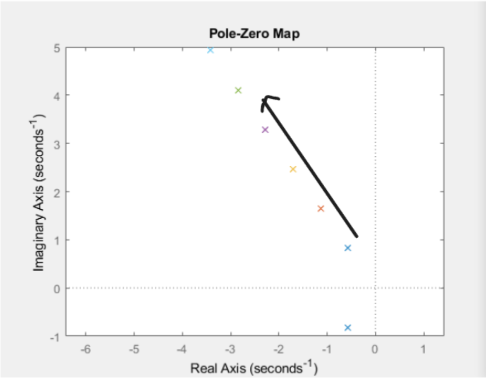
Here we change zeta using the following values in Matlab

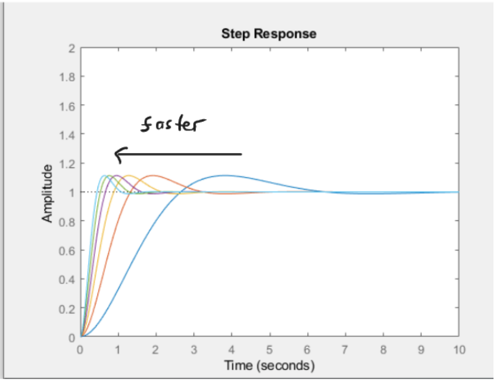
`[0.1:0.1:0.9,0.97,1]`



E.g., at $\theta = 55°$ the overshoot is a little over 10%, and the damping ratio is a little over 0.55. In most cases we don't want to exceed this.

Or we can fix the damping ratio and increase the distance from the origin





Here zeta=0.55, theta=56.6 degrees, overshoot =13%.

As ω_n increases the response becomes quicker but the shape is completely unchanged. The real and imaginary parts change together and the settling time reduces.

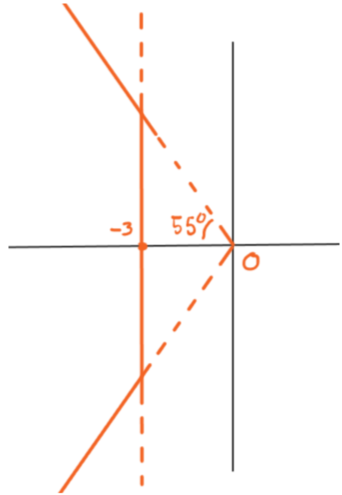
Recall: with $p = -\sigma \pm j\omega$ the response follows the exponential

$$e^{pt} = e^{-\sigma t} e^{j\omega t}$$

and the settling time is $t_s = 3/\sigma$ (to within 5% of the final value, since $e^{-3} = 0.0498$)

To find a good pole location we can choose a maximum value for the settling time t_s and then set the real part based on $\sigma \geq 3/t_s$.

We can use this, together with $\theta \leq 55°$ (say) to define **a region in the s-plane** where we want the poles to sit.



e.g., if the system is to settle in less than 1 s **we want all our poles to be to the left of the solid red lines**.

Depending on the system and the control purpose the angle and the σ value may change.

E.g., we may require 1% for maximum overshoot and settling in less than 100ms in the autofocus controller for a high-precision camera lens.

In some cases, actuator or power limits mean we can't achieve this, so we look for some compromise (faster but more overshoot etc.)

Think of a control application where physical constraints will limit the speed of response.

To summarize when looking at poles in the complex plane we plan something like this …

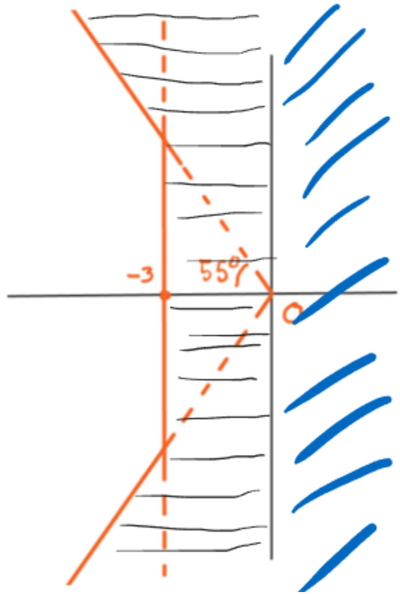
the blue shaded region is unstable

the **black** shaded region is stable but is too slow of too oscillatory or both.

the unshaded region is stable and hase good relative stability

Care - this is only a guideline!

Especially, overshoot can be higher **when the transfer function has one or more zeros**. In which case the angle shown might need to be decreased further.



Summary of step-response characteristics

- settling time: (5%) $t_s = 3/\sigma$ (2%) $t_s = 4/\sigma$
- peak time = one half the cycle time, based on the damped natural frequency, $t_p = \pi/\omega_d$
- overshoot ratio

$$e^{-\zeta\pi/\sqrt{1-\zeta^2}}$$

- rise time = time from 5% to 95% of final value (approximation)

$$t_r \approx \frac{0.60 + 2.16\zeta}{\omega_d}$$

Tutorial Questions

1. Find the transfer function of a system with a zero at $s = -2$, poles at $s = -1, -4$ and a steady state gain equal to 1.
2. Find the transfer function of a system with a zero at $s = -1.5$, poles at $s = -1, -0.1 \pm 2j$ and a steady-state gain equal to 10.
3. Find the transfer function of a system with damping ratio equal to 0.7 and a damped natural frequency ω_d corresponding to a 10 Hz oscillation; the steady-state gain is 100.
4. Use Matlab to plot the step-responses for the transfer functions in the above questions. From your plot, estimate the settling time and overshoot ratio in each case.
5. Compare your results with the equations above.
6. Find the transfer function of a system which has
   - unit steady-state gain
   - no zeros
   - two poles
   - 2 second settling time
   - 5% overshoot (zeta=0.7 is a good approximation)

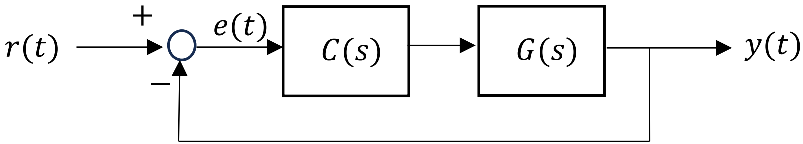
Closed-loop Poles

All the above applies to any transfer function with one or two (dominant) poles.

The poles were fixed because the transfer function is fixed.

For control systems the designer can **select controller gains** and therefore 'move poles around' to get the desirable response.

Here $G(s)$ is the plant to be controlled and $C(s)$ is the controller to be chosen.



Example 1

G(s) might be a second order system with poor relative stability

$$G(s) = \frac{1}{s^2 + 0.2s + 1}$$

while $C(s)$ is a PD controller

$$G(s) = K_p + K_d s$$

The closed loop transfer funtion (CLTF) is forward_path/(1+loop) …

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

and

$$CG = \frac{K_p + K_d s}{s^2 + 0.2s + 1} = \frac{N}{D}$$

(write N and D for numerator and denominator)

$$T(s) = \frac{N/D}{1 + N/D} = \frac{N}{D + N}$$

so

$$T(s) = \frac{K_p + K_d s}{s^2 + 0.2s + 1 + K_p + K_d s}$$

or

$$T(s) = \frac{K_p + K_d s}{s^2 + (K_d + 0.2)s + 1 + K_p}$$

The characteristic equation is

$$s^2 + (K_d + 0.2)s + 1 + K_p = 0$$

and the solutions are the **closed-loop poles**.

Now we can ‘place’ the poles where we want by choosing suitable values for the controller gains. For example let’s aim for a **damping ratio of 0.7 and settling time 0.5s**

Using $t_s = 3/\sigma$ we find $\sigma = \zeta\omega_n = 6$,

Hence $\omega_n = 6/0.7$ = 8.57.

The desired poles are

$$p = -\sigma \pm j\omega_d = -6 \pm 6.12j$$

Or we can just look at the desired closed-loop characteristic equation

$$s^2 + 2\zeta\omega_n s + \omega_n^2 = 0$$

Substituting the choices of ζ and ω_n

$$s^2 + 12s + 73.47 = 0$$

which should match

$$s^2 + (K_d + 0.2)s + 1 + K_p = 0$$

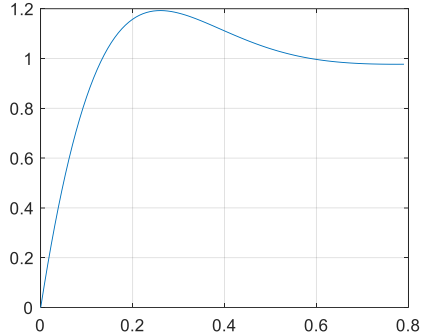
Hence

$$K_d = 11.8, K_p = 72.47$$

We have designed a PD controller to meet specified criteria!

Continuing with the same example, let's check the design

```
Kd=11.8;Kp=72.47;
s=tf('s');
D=s^2+(Kd+0.2)*s+1+Kp;
N=Kp+Kd*s;
T=N/D;
[y,t]=step(T);
plot(t,y),grid
```

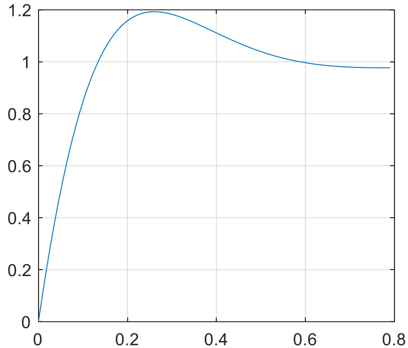


For damping ratio 0.7 the overshoot should be around 5%. Here it is quite a bit higher, due to the zero in the closed-loop transfer function. The zero comes from the PD controller (in the forward path). The zero did not change when we closed the loop

$$T(s) = \frac{N/D}{1 + N/D} = \frac{N}{D + N}$$

We can get Matlab to do a bit more of the calculation as follows

```
Kd=11.8;Kp=72.47;
s=tf('s');
G=1/(s^2+0.2*s+1);
C=Kp+Kd*s;
T=feedback(C*G,1);
[y,t]=step(T);
plot(t,y),grid
```



A subtle point here is that we don't directly use the Matlab expression for "forward_path/(1+loop)", we prefer to use **feedback**.

If we use this code instead, the plot is the same as before, but Matlab doesn't do the algebra properly!

```
T1=C*G/(1+C*G);
step(T1)
```

```
T =

    11.8 s + 72.47
  ------------------
  s^2 + 12 s + 73.47

Continuous-time transfer function.

>> T1

T1 =

     11.8 s^3 + 74.83 s^2 + 26.29 s + 72.47
  --------------------------------------------
  s^4 + 12.2 s^3 + 76.87 s^2 + 26.69 s + 73.47

Continuous-time transfer function.
```

As we know, the correct result is a second order transfer function (highest power in the denominator is s^2) but the T1 version has fourth order.
The reason is that T1 has “cancelling poles and zeros”

```
>> pole(T1)

ans =

  -6.0000 + 6.1213i
  -6.0000 - 6.1213i
  -0.1000 + 0.9950i
  -0.1000 - 0.9950i

>> zero(T1)

ans =

  -6.1415 + 0.0000i
  -0.1000 + 0.9950i
  -0.1000 - 0.9950i
```

Quick Summary

- Using feedback changes the pole locations
- Feedback does not change the positions of the zeros
- In some cases we may ‘place the poles’ where we want – to guarantee stability and also relative stability
- In most cases we won’t be able to do this.
- Instead we can use Matlab tools to monitor the influence of controller gains on closed-loop poles
- Poles just tell us about settling (real part) and oscillation (imaginary part). They provide an analytical tool to test stability and relative stability.

Example 2 (Design of a PD controller using Matlab).

Given the plant transfer function

$$G(s) = \frac{s + 8}{s^3 + 7s^2 + 15s + 25}$$

a) Find the open-loop poles and zeros and decide if the plant is stable
b) Plot the open-loop step response to confirm
c) Apply PD control in the form $C(s) = K(1 + 0.1s)$ and use Matlab to find the poles as K is increased.
d) Aiming for theta $\theta < 45°$ and $t_s < 1$ sec, find a suitable value for K and plot the closed-loop step response.

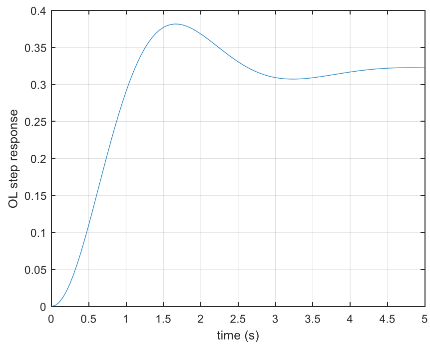
```
% parts (a) and (b)
p=pole(G); %open-loop poles
[y,t]=step(G,5);
figure(1)
plot(t,y)
```

```
p =

  -5.0000 + 0.0000i
  -1.0000 + 2.0000i
  -1.0000 - 2.0000i
```

clearly stable, with slowest pole $-1 \pm 2j$

For part (c) there is a Matlab function called **rlocus** that can help, but we leave that until another day!

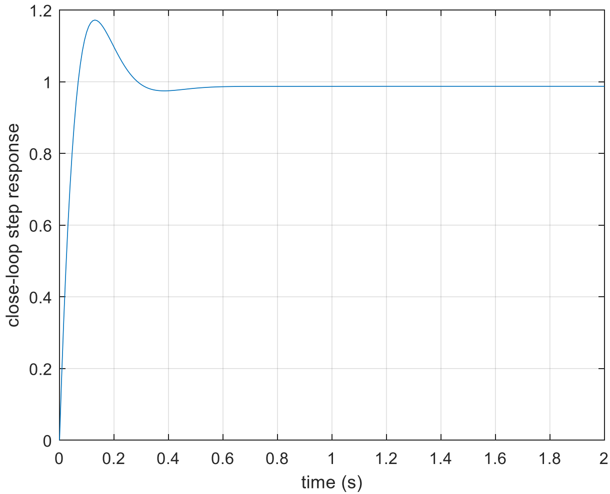


Instead we choose a set of increasing values of K and check whether all the poles are in a region to the left of –3 and theta less than 45 degrees ….

```
% part (c)
KK=(1:10:500); %column of trial values
n=length(KK);
for i=1:n
    K=KK(i);
    C=K*(1+0.1*s);
    T=feedback(C*G,1);
    p=pole(T);
    ang=angle(p);
    ang=ang(ang>0); %eliminate negative angles
    theta=max(pi-ang);theta=theta*180/pi;
    sigma=-real(p);sigma=min(sigma); %worst sigma
    if sigma>3 && theta<45
        break
    end
end
disp([i,K,p,theta,sigma])
```

The basic idea is to keep increasig the value of K until both criteria is met.

The result is K=241, theta = 43.9148 sigma = 6.9521



The result looks fine, but perhaps the gain is unnecessarily high?

Again, we have used pole locations to design a controller.

Tutorial Questions (continued)

7. Adapt the Matlab script for Example 2 in a way that reduces the value of K but still meets the design criteria.
8. Use the overshoot equation on Slide 13 to calculate the value of zeta which gives 20% overshoot. [hint: take the natural logarithm of the equation given]
9. Find the poles associated with ζ=0.5 and ω_n=50. Find the associated second-order transfer function(s) given the steady-state gain is 100 and (i) there are no zeros (ii) there is a zero at $s = -2$.
10. Plot the step response of the transfer functions from question 9. Use the plots to find (a) settling time, (b) percent overshoot, (c) rise time (from 5% to 95% of the final value). Compare with the predictions of the formulae of Slide 13 and comment.