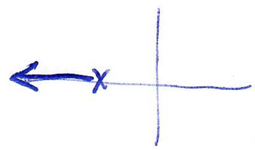
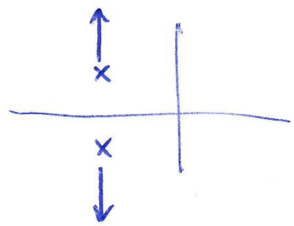
ME 4555 - Lecture 23 - PID control

★ We saw that proportional control (P) can be used to make response faster by moving poles.

1^{st} order :  pole moves left (τ gets smaller)

2^{nd} order  poles move up/down (ω_d gets larger)

But unfortunately, we always get step response steady-state value < 1 (there is always an error).

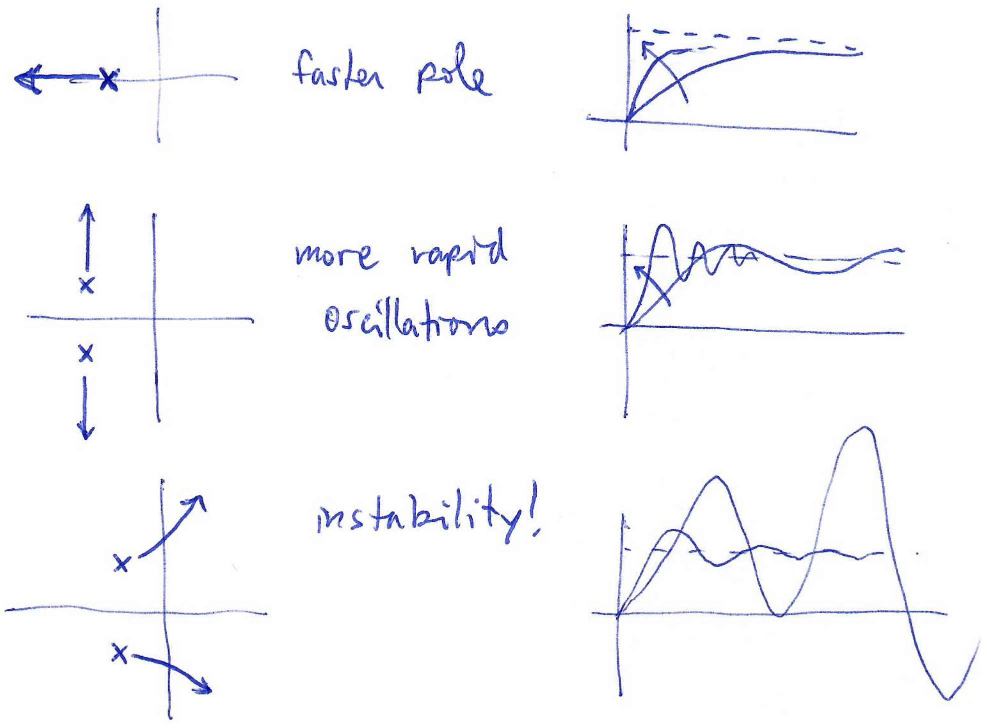
★ We saw that introducing an integrator into our compensator (I) and combining with proportional gain (PI), we eliminate the steady-state error, but we introduce another pole, so this may introduce overshoot.

★ We can reduce overshoot by adjusting the compensator based on the derivative (D) of the error.

If we combine all three, it's called a PID controller.
We can also use a subset (P, PI, PD, etc.).

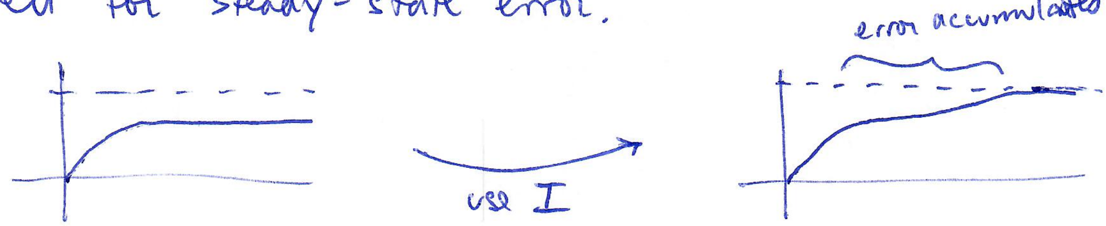
The PID controller is by far the most common type of controller used in industrial/commercial applications.

"P" control tends to speed up response. Possibilities include:



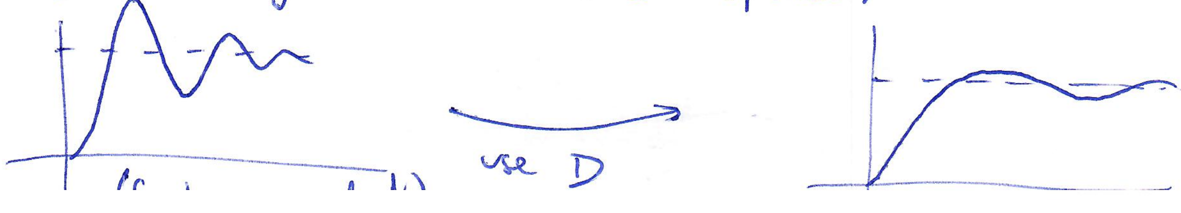
Note: P control only cares about the instantaneous value of the error (where is it now?)

"I" control operates on the integral of the error, so it looks into the past (where has the error been?). This can help correct for steady-state error.

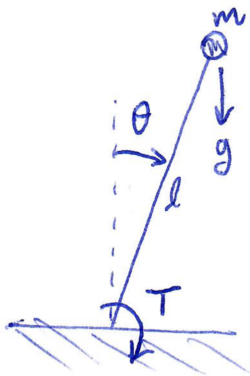


Sometimes, if a lot of error has accumulated, even if we are close to our target, we can overshoot it.

"D" control looks at $\frac{de}{dt}$ and adjusts based on this. If we are approaching our target too quickly, it slows us down. This can mitigate overshoot, but may slow down the response.



Ex inverted pendulum.



Actual dynamics:

$$mgl\sin\theta + T = ml^2\ddot{\theta}$$

$$\Rightarrow \quad \ddot{\theta} - \frac{g}{l}\sin\theta = \frac{T}{ml^2}$$

For simplicity, let's say $\frac{g}{l} = 1$, $\sin\theta \approx \theta$ (small angle) and normalize the torque input. So our transfer function is

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

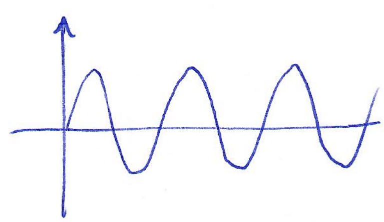
Poles are at $s = \pm 1$ (unstable).

Using proportional control: $\dfrac{K G(s)}{1 + K G(s)} = \dfrac{K}{s^2 + K - 1}$

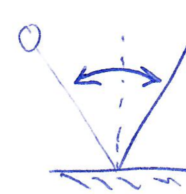
If $0 < K \le 1$, we have real poles $s = \pm\sqrt{1-K}$ (unstable).

If $K > 1$, we have imaginary poles $s = \pm i\sqrt{K-1}$ (undamped)

Impulse response:



i.e.



oscillation.

Let's see what happens if we use derivative action ("D").

Let $C(s) = K_p + K_d s$ (PD controller).

we now have damping!

$$\frac{GC}{1+GC} = \frac{\frac{K_p + K_d s}{s^2 - 1}}{1 + \frac{K_p + K_d s}{s^2 - 1}} = \frac{K_p + K_d s}{s^2 + K_d s + K_p - 1}$$

Note: this closed loop map has nonzero steady-state to a step input (use FVT: $\lim_{t\to\infty} y(t) = \frac{K_p}{K_p - 1}$)

However, for an inverted pendulum, the goal is to keep it balanced ($r = 0$) so we won't even use a step input.

When $r = 0$, this is what we call a regulator problem.

☆ play with interactive pendulum PID visualization

→ try adding K_p, watch poles come together and break apart.

→ add damping by increasing K_d.

→ what happens when we increase K_i?

→ can the system be made unstable?