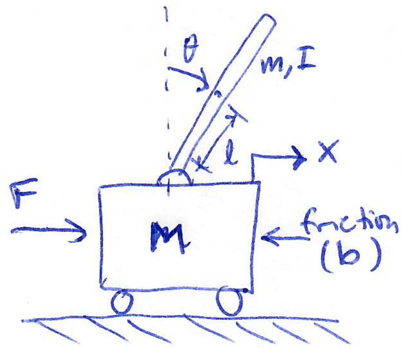
ME 4555 – Lecture 26 – Root locus design II

Inverted pendulum on a cart. Approximate transfer function for small θ is given by:



$$\frac{\theta}{F} = \frac{ml\,s}{\left((M+m)(I+ml^2)-m^2l^2\right)s^3 + b(I+ml^2)s^2 - (M+m)mgl\,s - bmgl}$$

using values:

$M = 0.5\,kg$

$m = 0.2\,kg$

$b = 0.1\,N/m/s$

$l = 0.3\,m$

$I = 0.006\,kg\cdot m^2$

we obtain:

$$G = \frac{\theta}{F} = \frac{4.5455\,s}{s^3 + 0.1818\,s^2 - 31.18\,s - 4.455}$$

Using zpk command in Matlab:

$$G = \frac{4.5455\,s}{(s-5.565)(s+5.604)(s+0.1428)}$$

Objective: design a feedback controller that achieves a settling time of < 5 seconds and max θ of no more than 0.05 radians ($\sim 2.9°$) if we give the cart an impulsive jolt of 1 N·sec.

We will do the design entirely in Matlab. Guided by our knowledge of PID control and root locus.

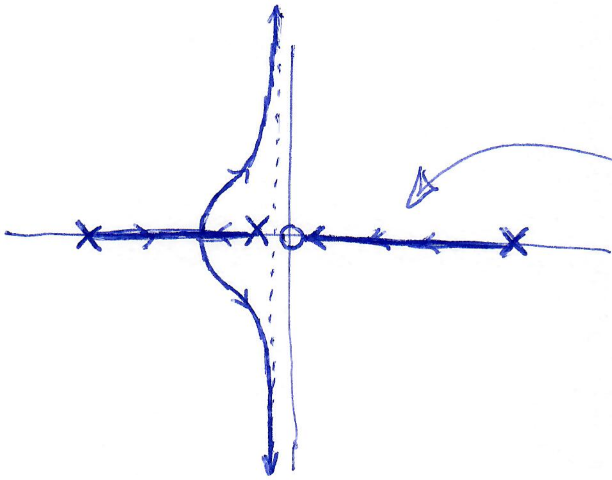
Root locus for $C(s) = K$ (P-control)

poles: $\{-5.60, -0.14, 5.56\}$

zeros: $\{0\}$.

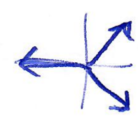
$$\sigma_A = \frac{\Sigma p_i - \Sigma z_i}{3-1} = -0.09.$$

two asymptotes ($\pm \frac{\pi}{2}$).

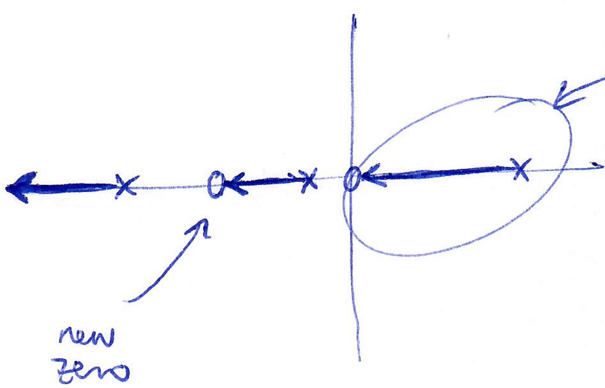


will always be unstable because of the unstable pole. Here

Can we solve this by adding a pole or a zero?

- if we add a pole, we will have 3 asymptotes, so it will make stability worse, i.e. 

- if we add a zero (i.e. PD control) this won't fix the problem of the unstable pole. For example:



still unstable!

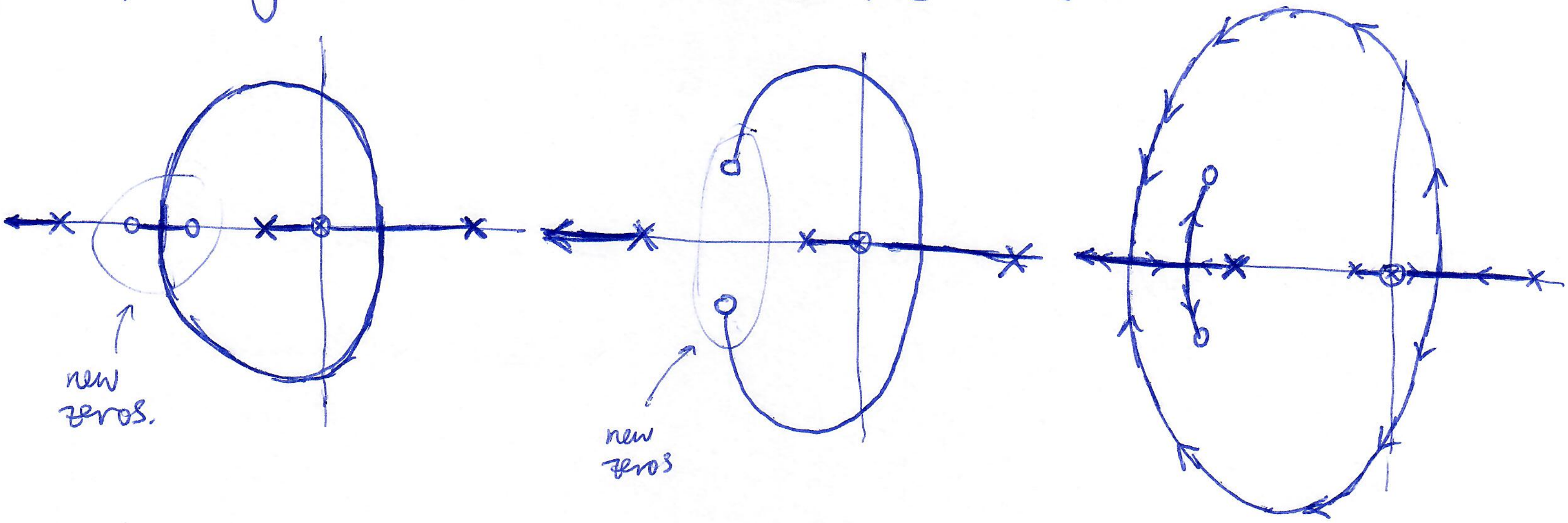
- what if we did PID control?

$$C(s) = K_p + K_d s + K_i \frac{1}{s}$$

$$= \frac{K_d s^2 + K_p s + K_i}{s} = \frac{K(s-z_1)(s-z_2)}{s}$$

adds an integrator and two zeros. The integrator cancels the zero at the origin!

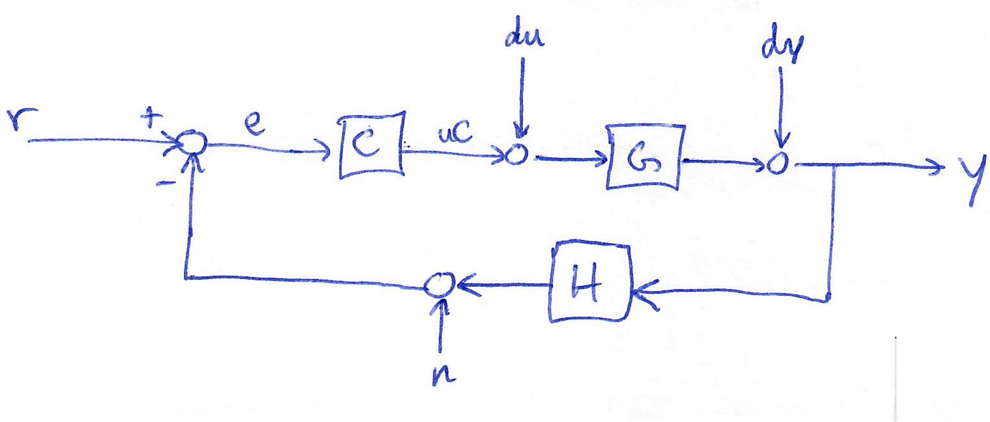
resulting root locus with some options/scenarios depending on how we place the zeros:



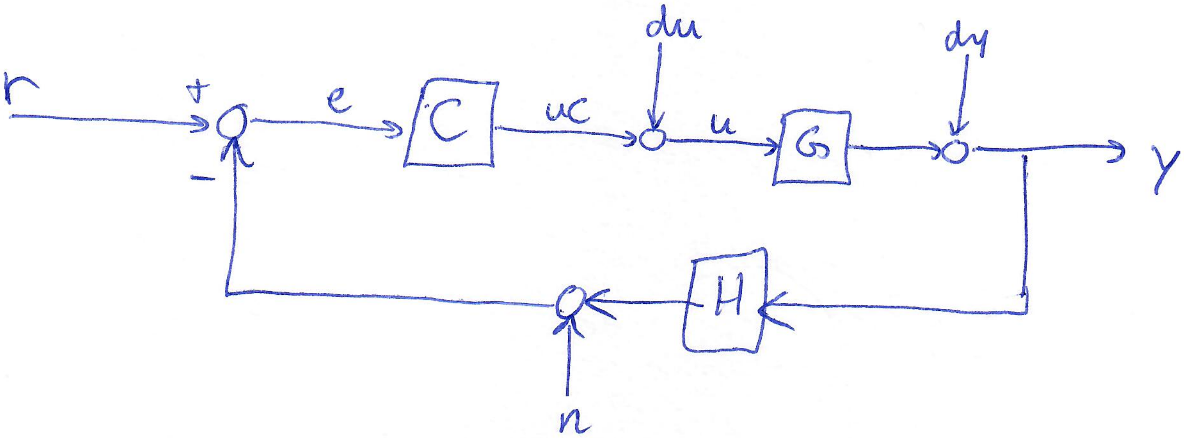
Each of these designs can bring all poles to the left-half plane, it's a matter of seeing which design works best.

- ☆ use rltool in Matlab
- ☆ create different compensators.
- ☆ view C(S), extract K_p, K_i, K_d

To look at impulse response, we want impulse on the cart (F), not an impulse in our reference (r).



we want impulse at du and to see effect on y.
Matlab calls this "du2y"



r : <u>reference input</u>. Used for tracking. Typically we care about things like step response. The transfer function "r2y" is $\dfrac{CG}{1+GCH}$.

du : <u>disturbance input</u>: For example, knocking the cart with your hand, or wind blowing on it exerting a force. Transfer function "du2y" is $\dfrac{G}{1+GCH}$.

dy: <u>disturbance output</u>: For example directly moving the pendulum to a new angle rather than exerting force on the cart. Transfer function "dy2y" is $\dfrac{1}{1+GCH}$. Also called the "sensitivity function".

n: <u>measurement noise</u>: For example noise in the sensor that measures the angle. TF "n2y" is $\dfrac{-1}{1+GCH}$. (negative of)

→ in all cases, poles are roots of $1+GCH$. only zeros are different.

→ sometimes we want e.g. "r2u" to know how large the input gets when our controller is running.