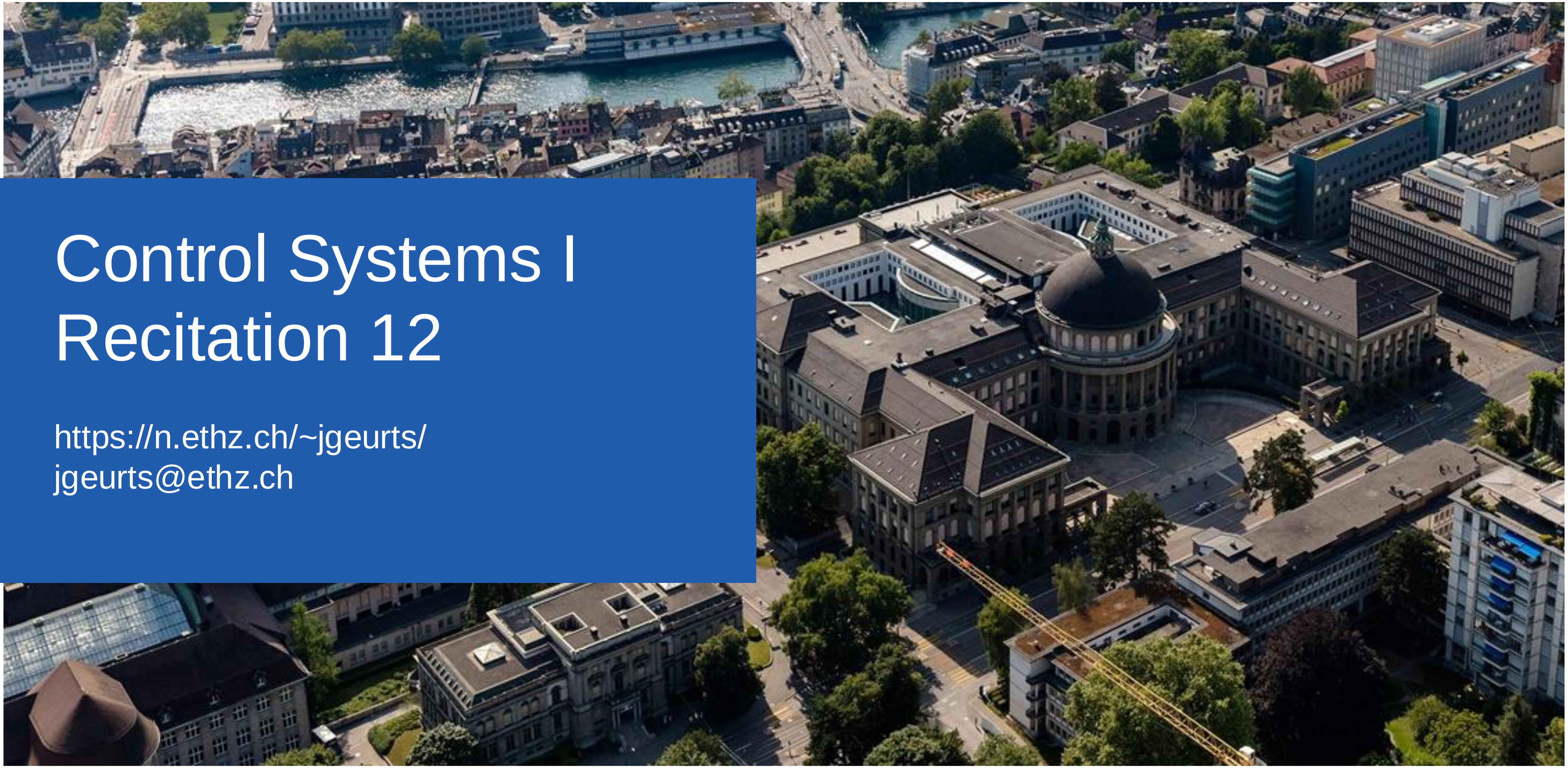


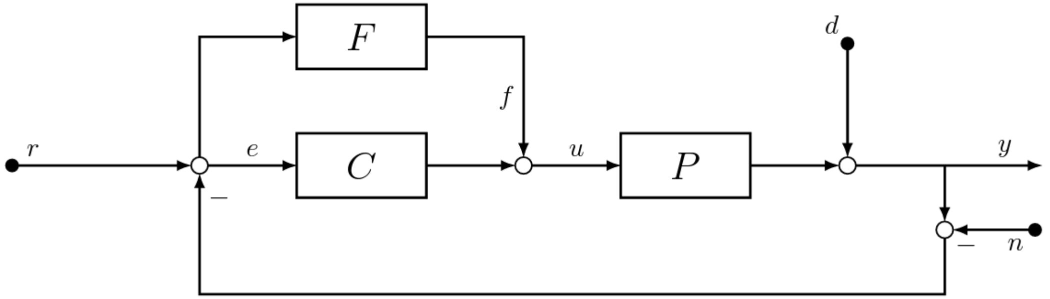
Control Systems I
Recitation 12

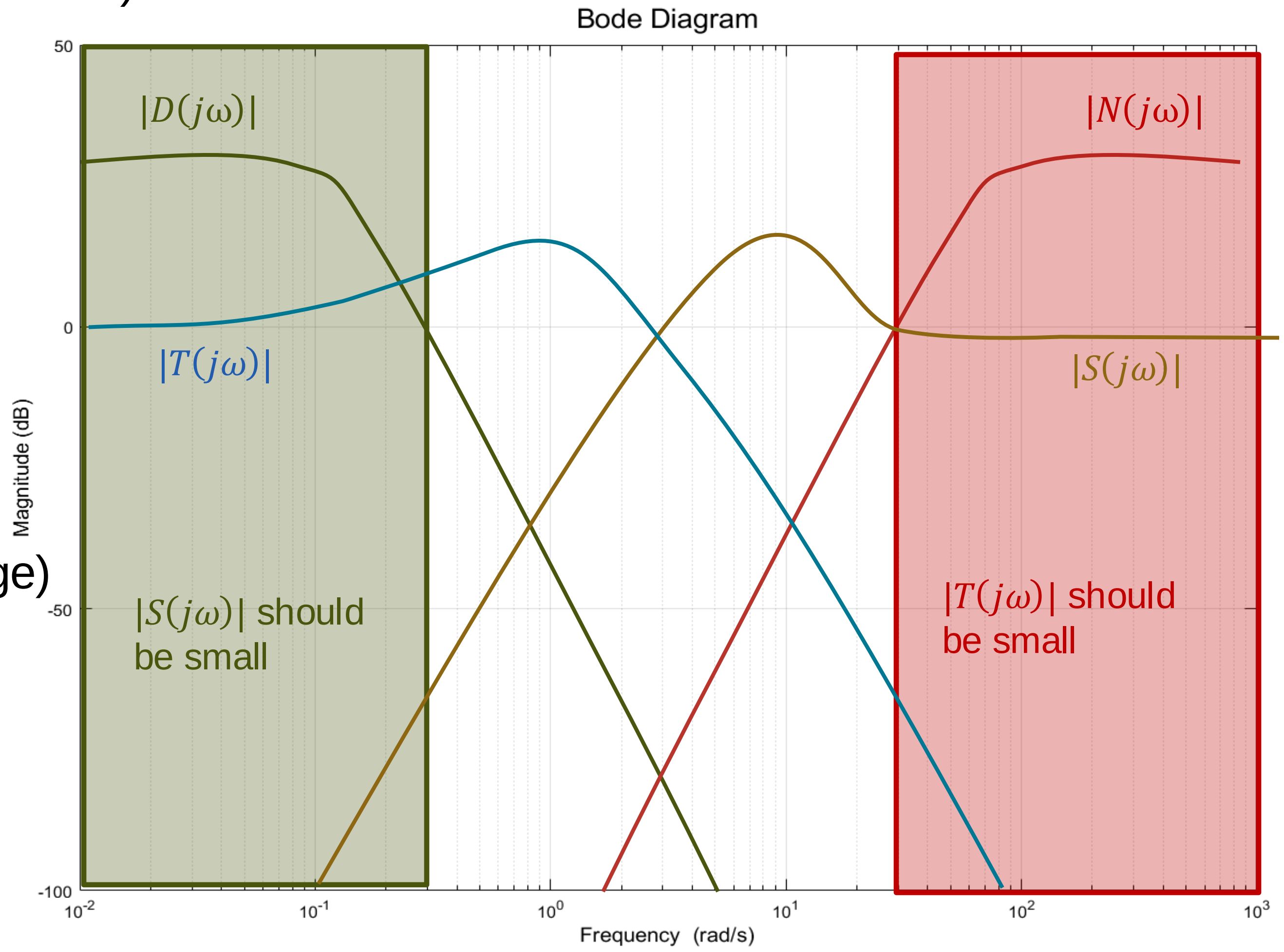
https://n.ethz.ch/~jgeurts/
jgeurts@ethz.ch

Last Week

Frequency-Domain Specifications

- **Disturbance Rejection** ($Y(s) = S(s)D(s)$, $S(s)$ small)
  - $S(s)$ only needs to be small up to ω_d
  - $T(s)$ must then be 1 in that region
- **Noise Rejection** ($Y(s) = S(s)N(s)$, $T(s)$ small)
  - $T(s)$ only needs to be small after ω_n
  - $S(s)$ must then be 1 in that region
- Reference Tracking ($Y(s) = T(s)R(s)$, $T(s)$ large)
  - For free from disturbance rejection

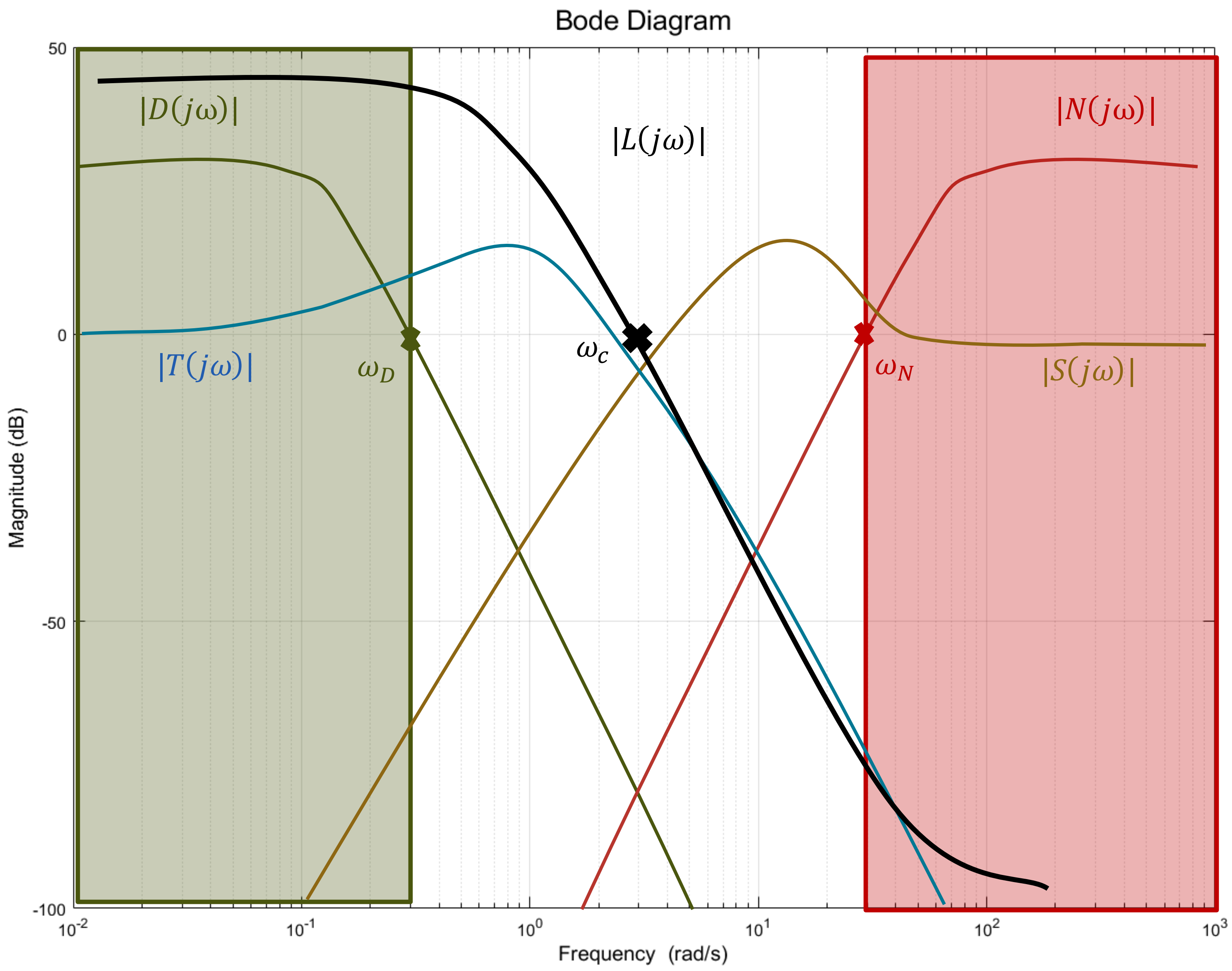




Last Week

Frequency-Domain Specifications

- For $|S(j\omega)| \ll 1$
  - $\Rightarrow \left|\frac{1}{1+L(j\omega)}\right| \ll 1 \Rightarrow |L(j\omega)| \gg 1$
- For $|T(j\omega)| \ll 1$
  - $\Rightarrow \left|\frac{L(j\omega)}{1+L(j\omega)}\right| \ll 1 \Rightarrow |L(j\omega)| \ll 1$



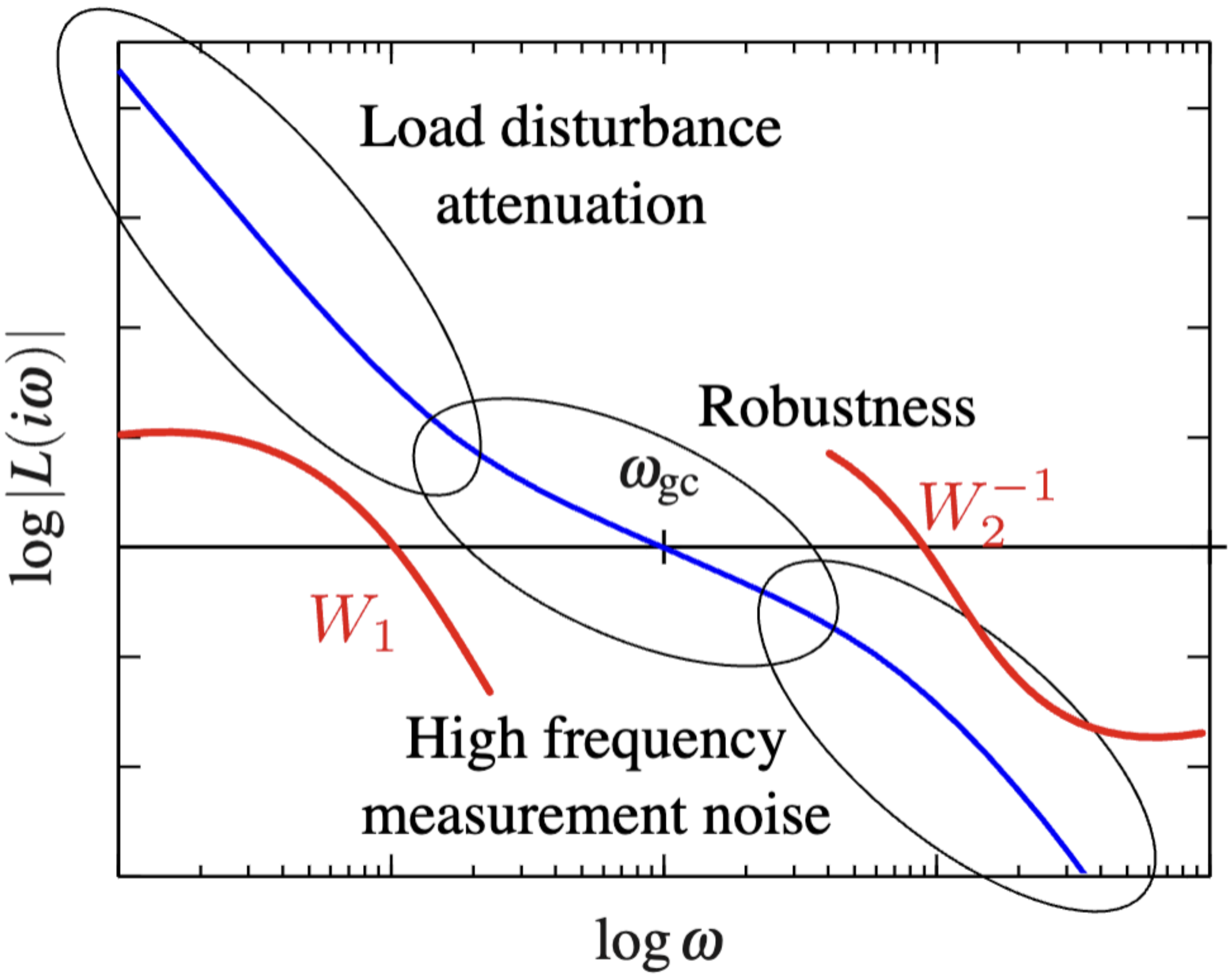
ETH zürich

Last Week

Frequency-Domain Specifications

- For $|S(j\omega)| \ll 1$
  - $\Rightarrow \left|\frac{1}{1+L(j\omega)}\right| \ll 1 \Rightarrow |L(j\omega)| \gg 1$
  - $|L(j\omega)| > |W_1(j\omega)|$

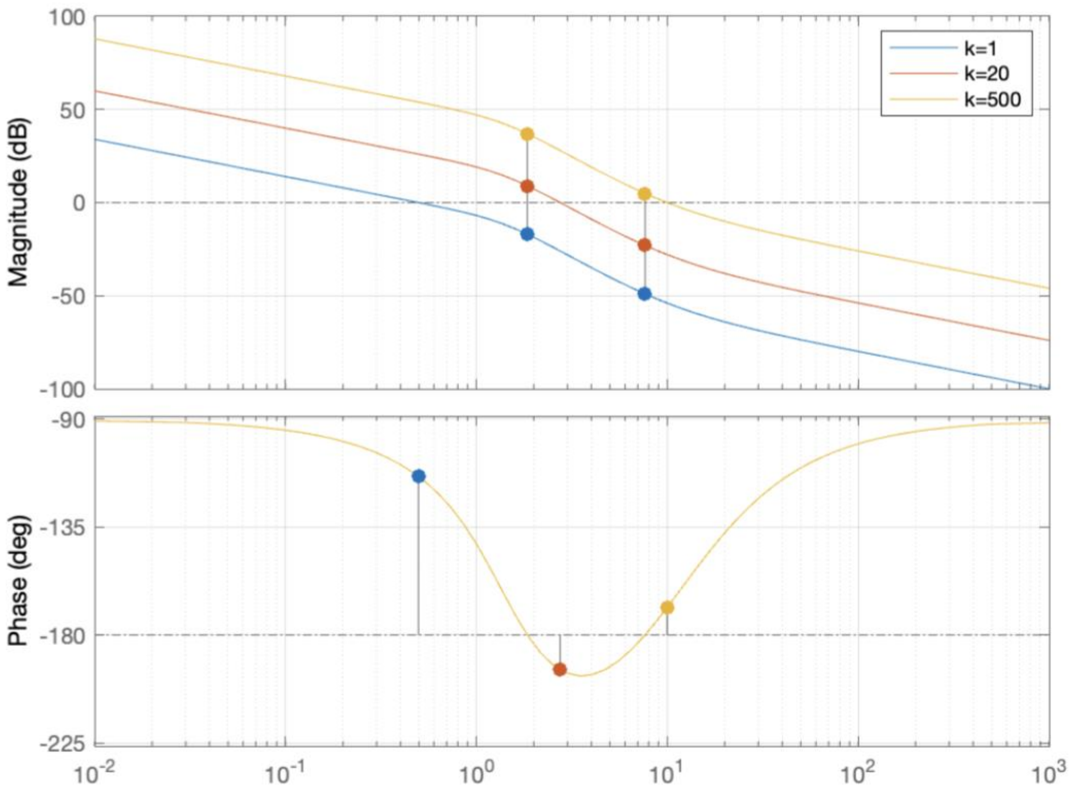
- For $|T(j\omega)| \ll 1$
  - $\Rightarrow \left|\frac{L(j\omega)}{1+L(j\omega)}\right| \ll 1 \Rightarrow |L(j\omega)| \ll 1$
  - $|L(j\omega)| < \left|W_2^{-1}(j\omega)\right|$



Last Week

Loop Shaping

- What can we do better?
  - Use our knowledge and choose a combination of the following elements
  - Gain: k, Integrator: $1/s$, Lead: $\frac{s/a+1}{s/b+1}$ $(0 < a < b)$, Lag: $\frac{s/a+1}{s/b+1}$ $(0 < b < a)$
- **Gain: k**
  - Shifts the Magnitude Diagram up or down
  - Phase stays constant
- User Guide:
  - Move system up or down as desired
- **Integrator: $1/s$**
  - Gets rid of steady state error
  - $|L(0)| \to \infty$
- User Guide:
  - Add as many integrators as needed to get rid of the steady state error (beware of phase)

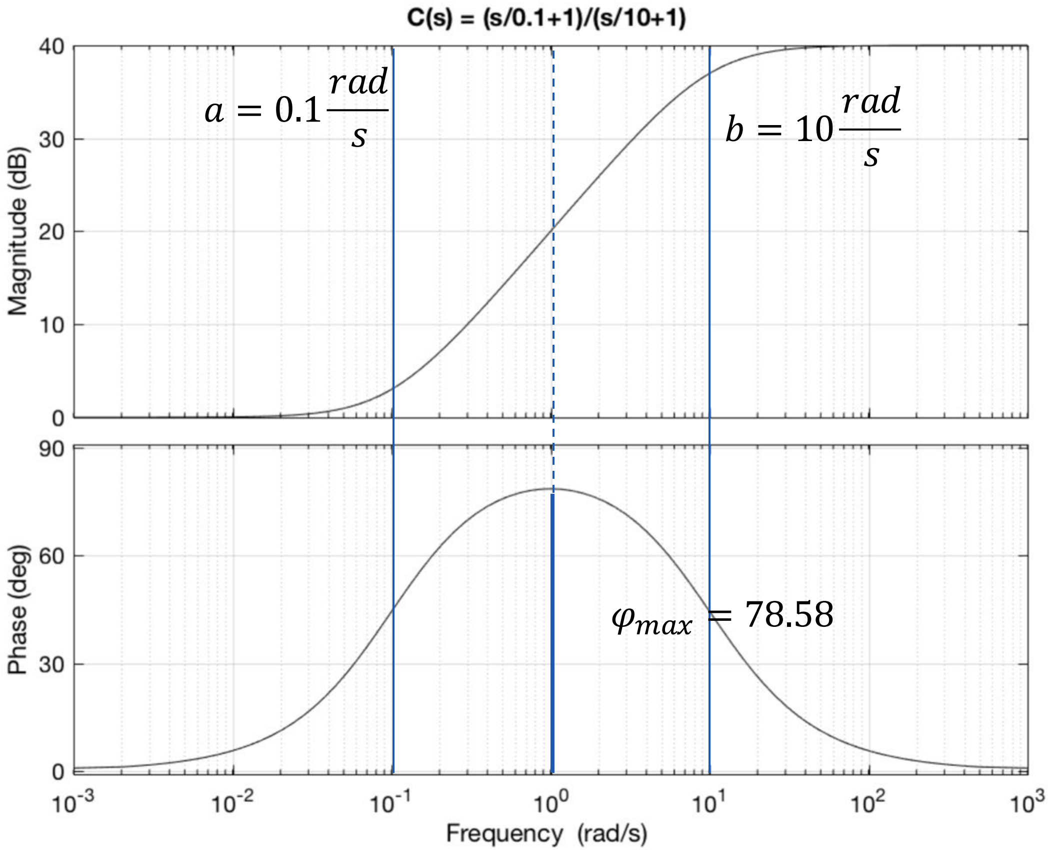


ETH zürich

Last Week

Loop Shaping

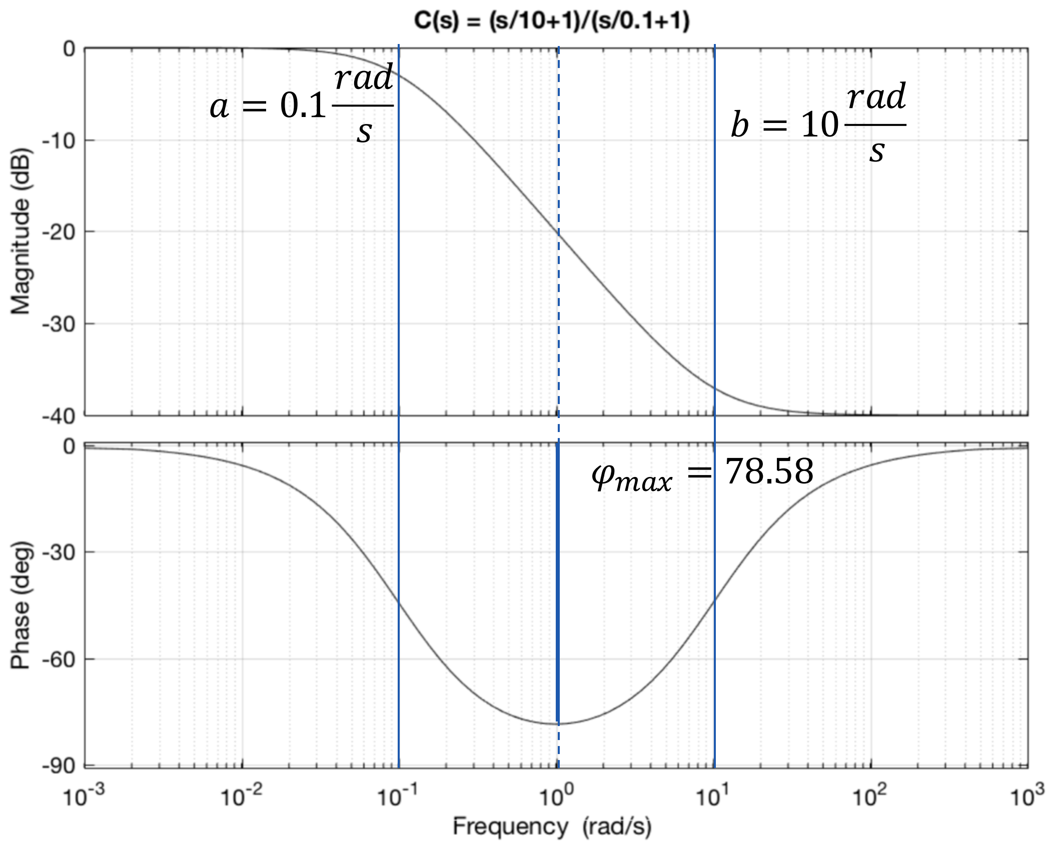
- **Lead:** $\frac{s/a+1}{s/b+1} = \frac{\mathbf{b}}{\mathbf{a}}\frac{s+a}{s+b}\ (\mathbf{0 < a < b})$, PD-Controller for $b \to \infty$
  - Increases magnitude by b/a at high frequencies
  - Creates a slope of $20\frac{dB}{dec}$ between a and b
  - Increases the phase at $\sqrt{ab}$ (midpoint of $[a, b]$) by:
    - $\varphi_{max} = 2\arctan\left(\sqrt{b/a}\right) - 90°$
- User guide:
  - Used to increase the phase margin
  - Pick desired crossover frequency $\sqrt{ab}$
  - Pick b/a for desired phase shift ($b/a \uparrow \to \varphi \uparrow$)
  - Use a gain k to shift crossover frequency
- Danger:
  - Increases magnitude at high frequencies -> Sensitive to noise



Last Week

Loop Shaping

- **Lag:** $\frac{s/a+1}{s/b+1} = \frac{\mathbf{b}}{\mathbf{a}}\frac{s+a}{s+b}\,(\mathbf{0} < \mathbf{b} < \mathbf{a})$, PI-Controller for $b \to 0$
  - Decreases magnitude by b/a at high frequencies
  - Creates a slope of $-20\frac{dB}{dec}$ between a and b
  - Decreases the phase at $\sqrt{ab}$ (midpoint of $[a, b]$) by:
    - $\varphi_{max} = 2\arctan\left(\sqrt{b/a}\right) - 90°$
- User guide:
  - Used to improve disturbance rejection/ref tracking
  - Pick a small enough to not affect ω_c and φ_{margin}
    - For this $a \ll \omega_c$
- Danger:
  - Phase lag at small frequencies -> reduction of phase margin



Last Week

Loop Shaping

- Split System: $P(s) \;=\; P_{mp}(s)\, D(s)$
  - $P(s) = \frac{s-z}{s-p} = \frac{s+z}{s+p} \cdot \frac{s+p}{s-p} \cdot \frac{s-z}{s+z}$
- Non-minimumphase zero: $D(s) = -\frac{s-z}{s+z} \rightarrow \angle D(j\omega) = -2\arctan\left(\frac{\omega}{z}\right)$
  - At ω_c the system lags $-2\arctan\left(\frac{\omega_c}{z}\right)$
  - Nmp zeros force the system to be slow as max gain and crossover frequency are reduced
    - Slow (small z) nmp zeros are worse than fast (large z) ones
- Unstable pole: $D(s) = \frac{s+zp}{s-p} \rightarrow \angle D(j\omega) = -2\arctan\left(\frac{p}{\omega}\right)$
  - At ω_c the system lags $-2\arctan\left(\frac{p}{\omega_c}\right)$
  - Unstable poles force the system to be faster as min gain and crossover frequency are increase
    - Slow (small p) poles are better than fast (large p) ones
    - Fast system requires strong and fast controllers/ actuators

Last Week

Performance Limits

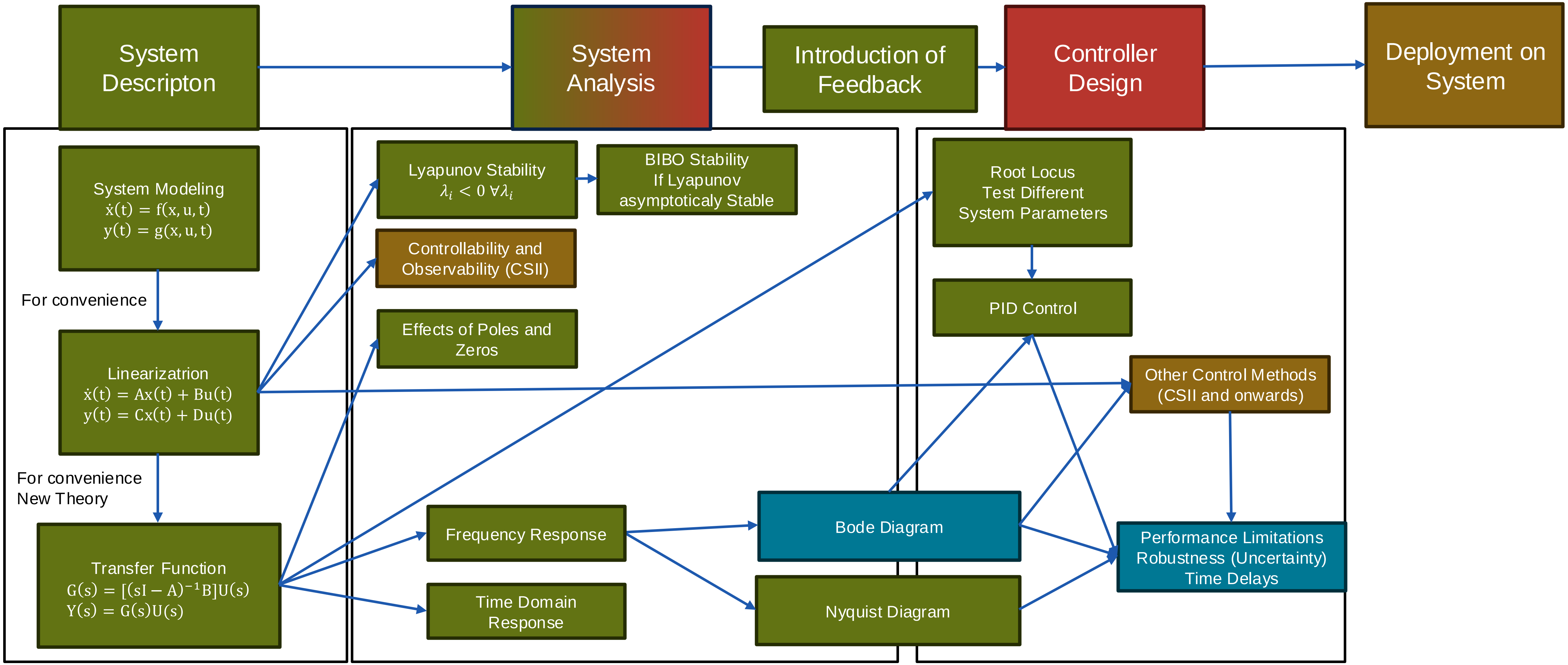
- Sometimes it is just not possible to get all requirements due to nmp zeros or unstable poles
- Rule of thumb on the crossover frequency limits
  - Nominal: $\max\{10 \cdot \omega_d, 2 \cdot \omega_{p^+}\} < \omega_c < \min\left\{\frac{1}{10} \cdot \omega_n, \frac{1}{2} \cdot \omega_\tau, \frac{1}{2} \cdot \omega_{\zeta^+}\right\}$
  - Conservative: $\max\{10 \cdot \omega_d, 5 \cdot \omega_{p^+}\} < \omega_c < \min\left\{\frac{1}{10} \cdot \omega_n, \frac{1}{5} \cdot \omega_\tau, \frac{1}{5} \cdot \omega_{\zeta^+}\right\}$
  - ω_d and ω_n are the crossover frequencies of the disturbance and noise
  - ω_{p^+} and ω_{z^+} are the unstable poles and nmp zeros
  - $\omega_\tau = \frac{1}{\tau}$ effect of the time delay (next week)
- If ω_c the system can be controlled reasonably (no design specification)

Outline

- Tine Delay
  - What?
  - Effects
  - Example
  - What to do?
  - Approximations
- Controller Design
  - How to implement a controller
- Non-Linearities
- Cascaded Control
  - What?
  - Example

Conceptual Recap

Classical Control Approach



Time Delays

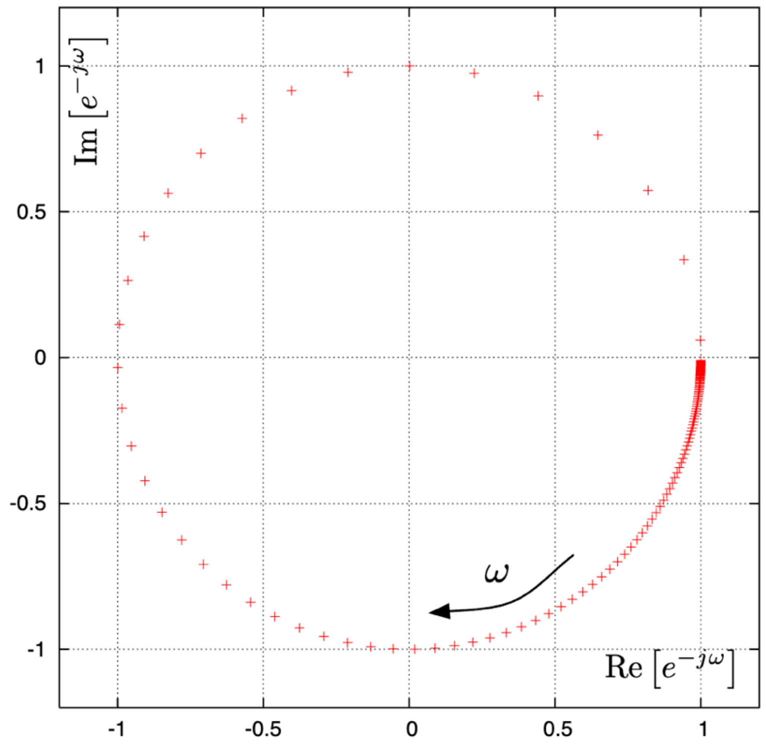
What?

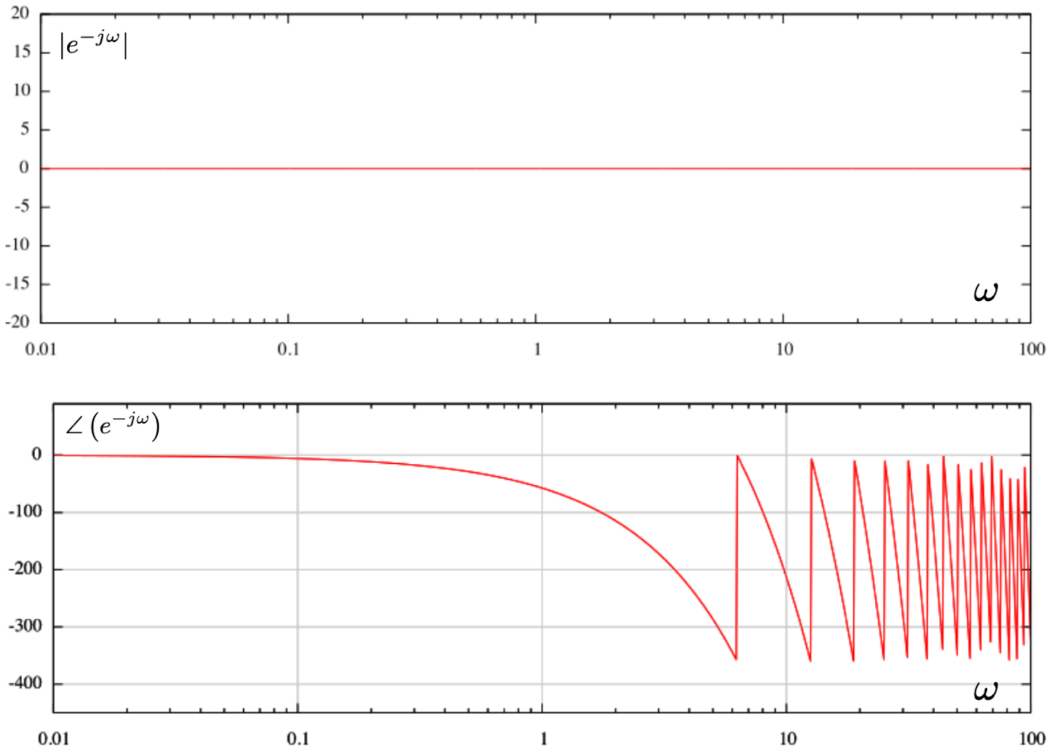
- Whenever an event/transition takes time:
  - Computing a control output using a computer
  - Goods on a conveyor belt with a sensor on the end
  - Long range control (e.g space crafts)
- Definition:
  - A time delay is a **linear** operator that transforms an input signal $t \to u(t)$ into a delayed output signal $y(t) = u(t - T)$, where $T \geq 0$ is the delay.
- Transfer Function:
  - $Y(s) = e^{-sT}U(s) \to G(s) = e^{-sT}$
- Not a polynomial!

Time Delays

What?

- Transfer Function:
  - $Y(s) = e^{-sT}U(s) \rightarrow G(s) = e^{-sT}$
- Bode Plot and Nyquist Plot:

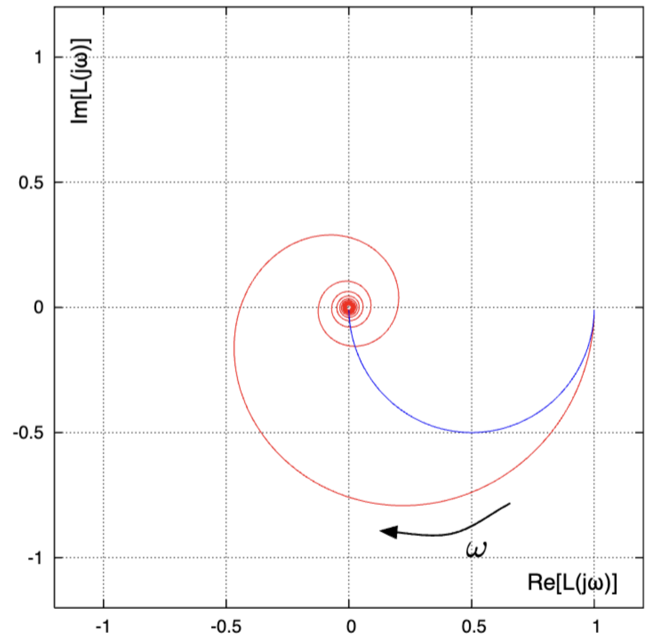


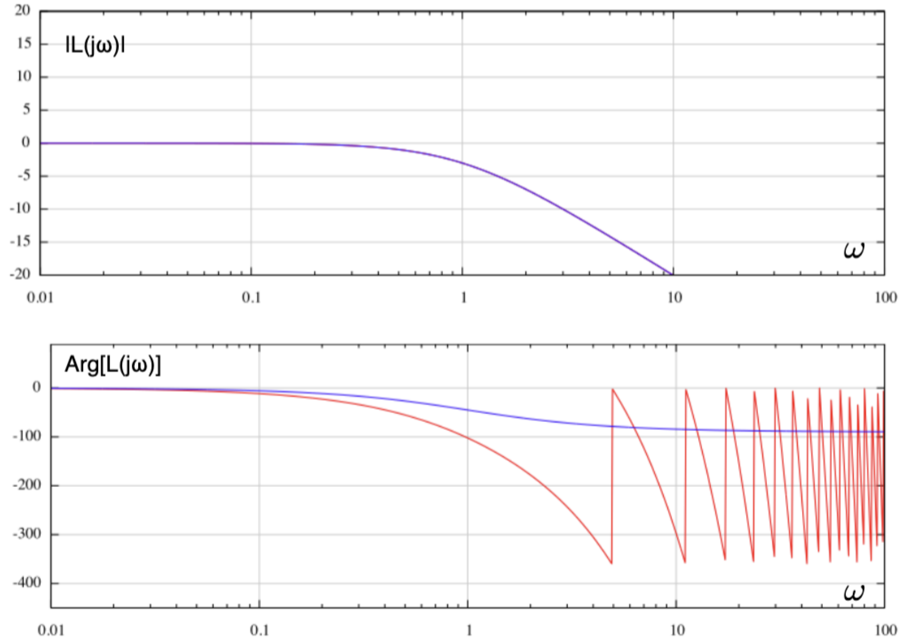


Time Delays

Effects

- System with a Time Delay:
  - $L'(s) = e^{-sT}L(s)$
  - $|L'(j\omega)| = \left|e^{-j\omega T}L(j\omega)\right| = |L(j\omega)|$
  - $\angle L'(j\omega) = \angle\left[e^{-j\omega T}L(j\omega)\right] = \angle L(j\omega) - \omega T,\ \omega > 0$
- **We lose phase margin!**
  - $\phi_{m,T} = \phi_{m,0} - \omega_c T$
  - Phase margin decrease
  - Decrease is dependent on the cross-over frequency

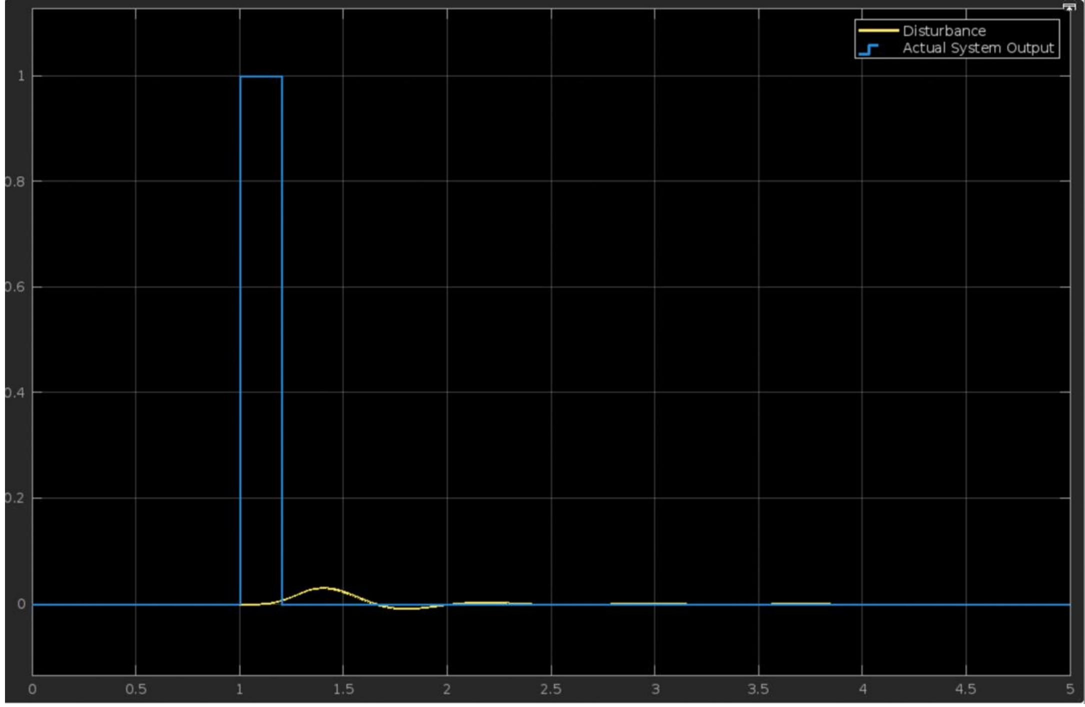


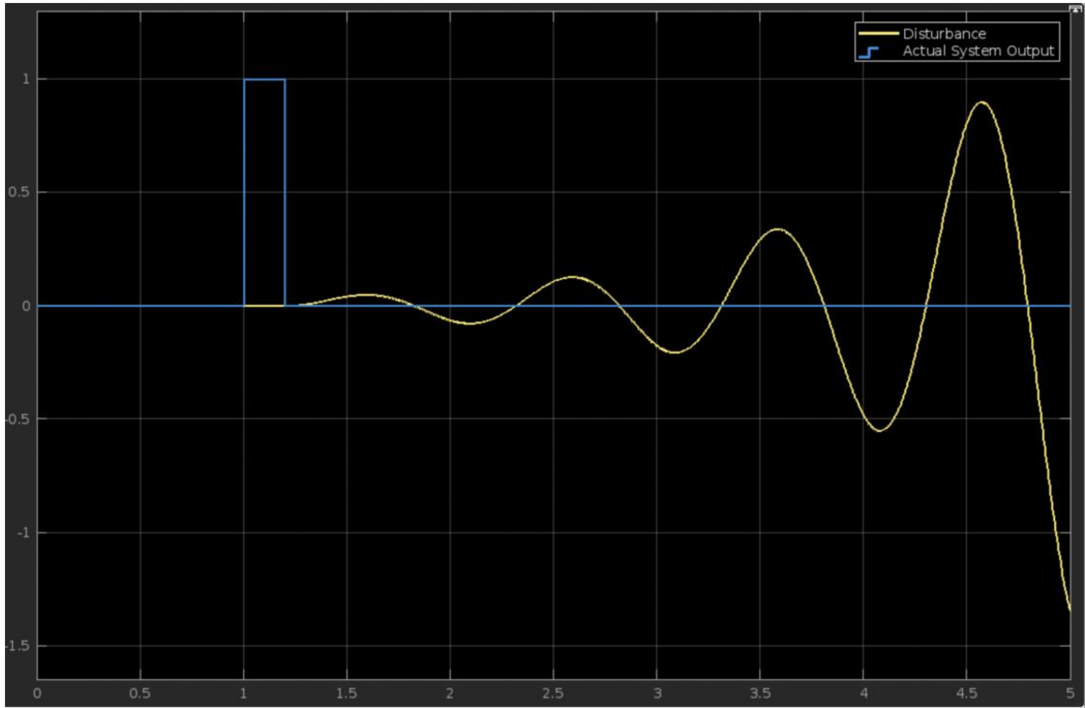


Time Delays

Effects - Example

- Consider the inverted pendulum (upright position):
- $\dot{x} = \begin{bmatrix} 0 & 1 \\ \frac{3g}{2L} & -\frac{3c_f}{mL^2} \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{1}{mL^2} \end{bmatrix} u, \ y = \begin{bmatrix} 1 & 0 \end{bmatrix} x$
- $C(s) = k_p + \frac{k_d s}{T_f s+1}: \ k_p = 35, k_d = 5, T_f = 0.001$
- Plot 1: $T = 0.1$
- Plot 2: $T = 0.2$
- To large of a time delay can make the system unstable
  - Sometimes nothing can help, and we need more sophisticated controllers





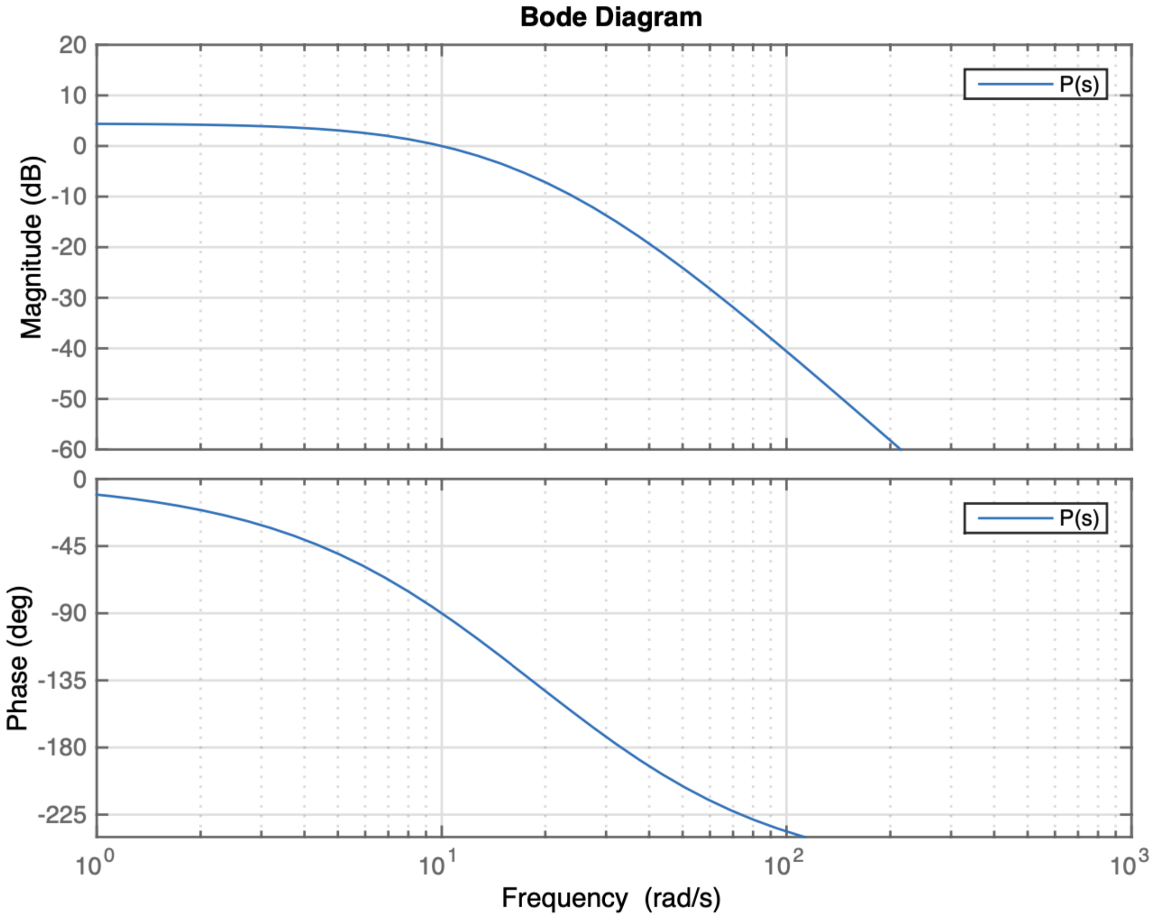
Time Delays

Example

- Phase margin reduction:
  - $\phi_{m,T} = \phi_{m,0} - \omega_c T$

Question 45 *Choose the correct answer. (1 Point)*

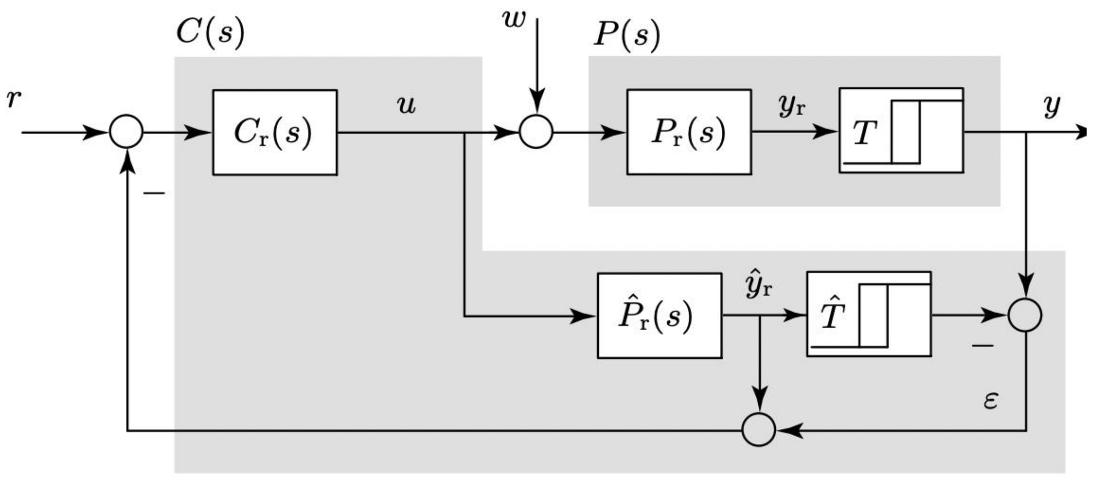
You are given the Bode plot of a plant however the time delay has not been included in the model. You have a time delay of $T_d = \frac{\pi}{40}\ s$. What is the new phase margin/ phase reserve of your system?



Time Delays

What to do?

- Ignore them and hope for the best
  - Create a controller ignoring the time delay
  - Check the closed loop system:
    - If instable redesign with higher phase margin or lower crossover frequency
  - Repeat until success
- Smith Predictor (not in lecture)
  - Main Idea incorporate time delay knowledge into controller
  - “Simulate” Time Delay in the controller and shit output accordingly



Time Delays

Approximations

- Transfer Function is not a Rational Function:
  - $G(s) = e^{-sT}$
  - We can't use root-locus, python or other tools that require rational functions
- Taylor Approximation:
  - $e^{-sT} = 1 - sT + 1/2(sT)^2 - 1/3(sT)^3 + \cdots$
  - $e^{-sT} \approx 1 - sT + 1/2(sT)^2$
  - Non-causal with two non-minimumphase zeros -> Only good for $T \ll 1$
  - Magnitude diverges which is not the case
- Padé:
  - First order: $e^{-sT} \approx \frac{2/T-s}{2/T+s}$
  - We see the non-minumumphase zeros -> Can't increase gain to much
  - Allows us to do Root-Locus
- For both: Always check with Nyquist for actual stability

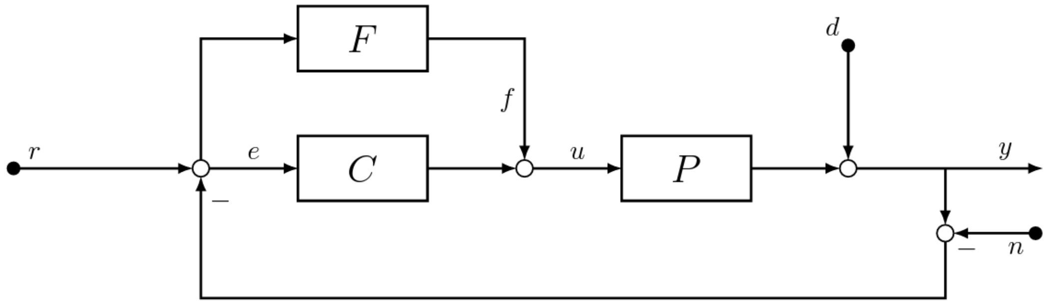
Time Delays

Approximations - Example

- Consider the Following System:
  - $P(s) = 2\frac{s+1}{s^2+3s+2}e^{-2s} + \frac{1}{s+2}e^{-s}, e^{-sT} \approx \frac{2/T-s}{2/T+s}$
- What would be the first-order Padé approximation?
  -
  -
  -
  -

Controller Design

How to Implement a Controller



- We saw how to design a controller TF:
  - $C(s) = k + \frac{c_m s^m + \cdots + c_0}{s^n + a_{n-1} s^{n-1} + \cdots + a_0}$
- We can transfer that to a state space system:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ & & \ddots & & & \\ 0 & 0 & 0 & 0 & \dots & 1 \\ -a_0 & -a_1 & \dots - a_{n-1} & & & \end{bmatrix}, \qquad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix},$$

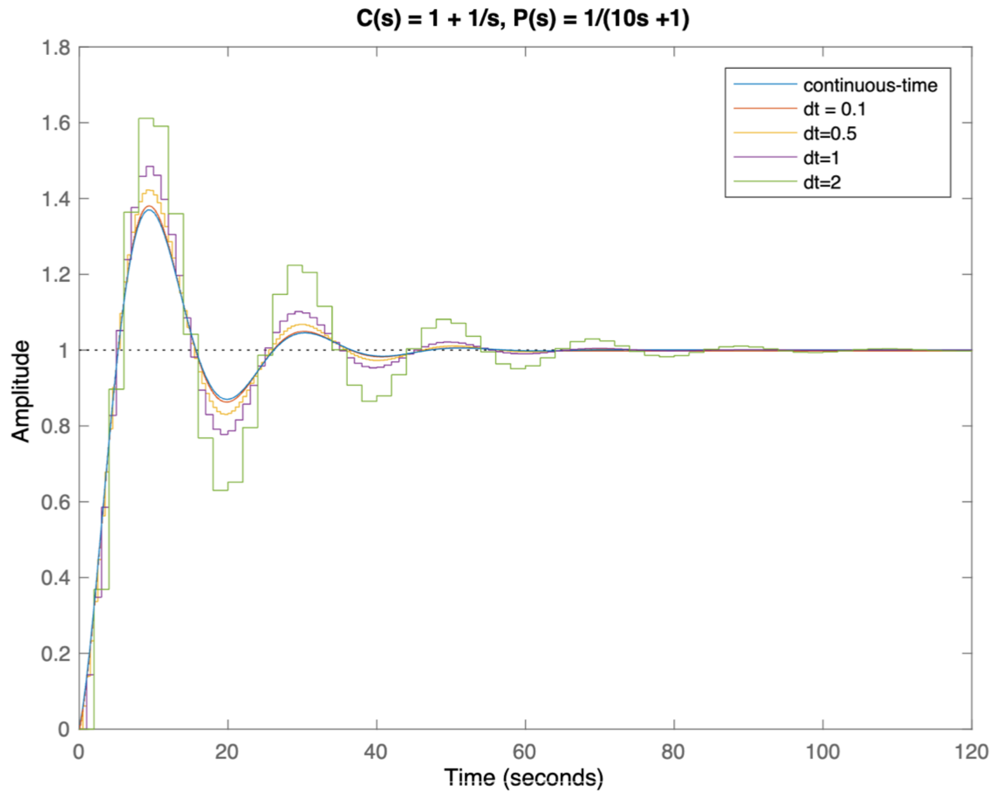
$$C = \begin{bmatrix} c_0 & c_1 & \dots & c_{n-1} \end{bmatrix}, \qquad D = [d]$$

- $\tilde{x}(t) = A\tilde{x}(t) + Be(t), e(t) = r(t) - y(t)$
- $u(t) = C\tilde{x}(t) + De(t)$

Controller Design

How to Implement a Controller

- Physical System:
  - Create an Analog Electronics circuit
    - Use Resistors, Capacitors and Inductors
- Software: Euler Approximation (“simulate the controller”)
  - Init $\tilde{x}[k] = 0, \mathrm{e}[\mathrm{k}-1] = 0$
  - For every k:
    - $e[k] = r[k] - y[k]$
    - Update system: $x[k] = x[k] + (Ax[k] + Be[k])dt$
    - Compute output: $u[k] = Cx[k] + De[k] + K_D(e[k] - e[k-1])/dt$
    - Send $u[k]$ to actuators
    - Update: $e[k-1] = e[k]$
  - We keep the output between to step constant -> zero order hold
    - Introduces a time delay of $dt/2$



Non-Linearities

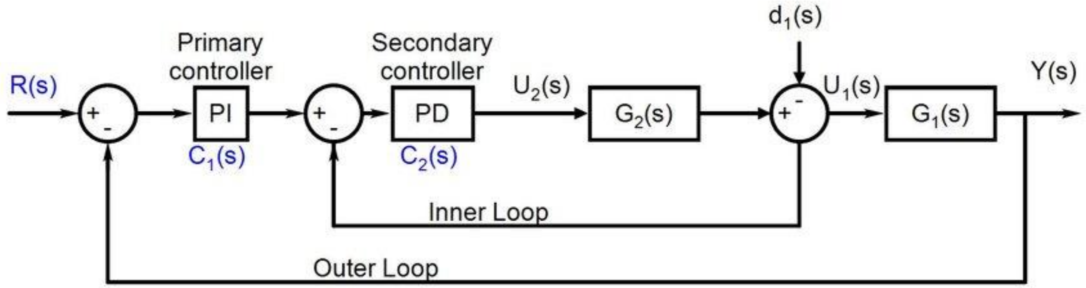
Linear Systems Theory for Non-Linear Systems

- Most real systems are **non-linear,** but non-linear control theory is hard
- What are the ways to use linear systems theory?
  - Linearization around equilibrium
    - Design a Linear Controller and apply to Non-linear system
      - PID, LQR, Linear MPC
    - Often works well!
    - Stability guarantees only around small deviation of equilibrium!!!
    - Check stability for “expected” deviation or using non-linear systems theory
  - Describing Functions
    - Extend Frequency Domain Methods to non-linear systems
  - Robust Control
    - Create a controller to handle modelling errors and noise

Cascaded Control

What?

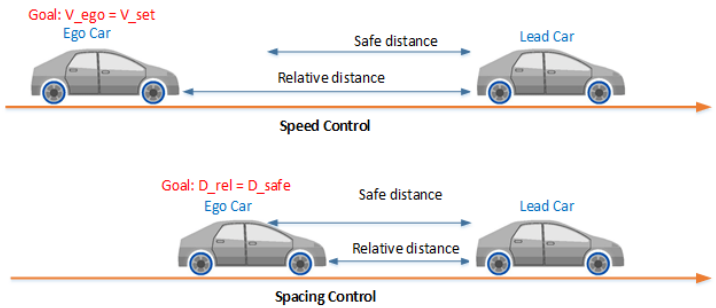
- Split Control Problem into two parts -> Often easier!
  - Low-Level/Inner-Loop(fast) and High-Level/Outer-Loop(slow)
- Low-Level/Inner-Loop : often output of High-Level controller
  - Mostly SISO
  - Often PID/PD
  - Joint Controller, Velocity Controller
- High-Level/Outer-Loop : Used to achieve main goal
  - Often MIMO
  - Usually more complex than a PID/PD
    - MPC, Neural Network, LQR
  - Kinematic Control, Trajectory Following
- Low Level needs to be much faster than High Level!
  - Rule of thumb $\omega_{LL} \approx 10 * \omega_{HL}$ or $\omega_{LL} \approx 5 * \omega_{HL}$
    - E.g inner loop runs at 300Hz and outer loop at 30Hz

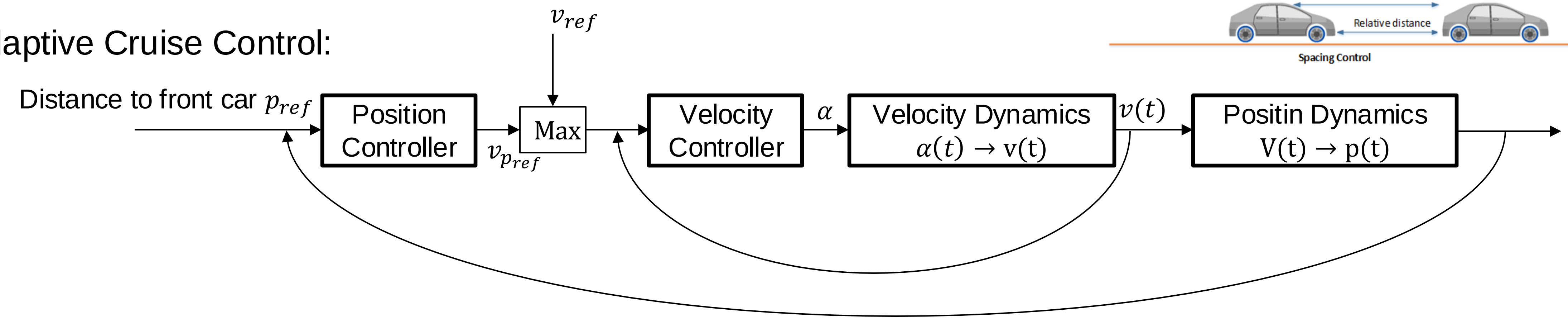


Cascaded Control

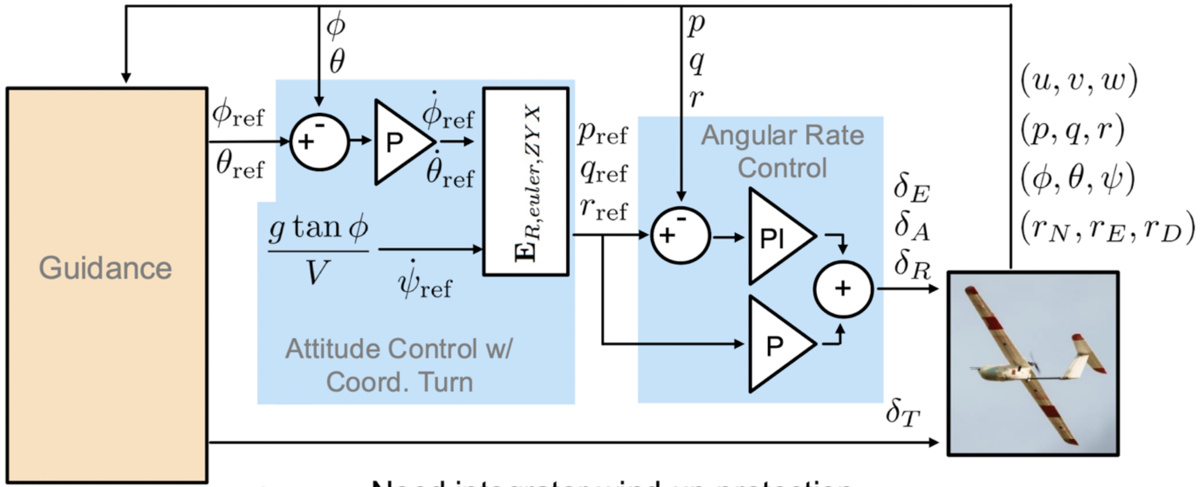
Examples

- Adaptive Cruise Control:





- Airplane:



Path or trajectory reference

- Need integrator wind-up protection
- Should scale actuator output with airspeed: $1/V^2$
- Bandwidth of inner (low-level) loop should be sufficiently higher than the outer (guidance)

Exercise 10

What to do?

- 1:
  - 1.1 Do
  - 1.2 Do
  - 1.3 Skip
- 2:
  - a) do
  - b) do
  - c) skip
  - d) skip