Linear Control Systems
25411

Instructor. Behzad Ahi | **Assignment 7**

Fall 1402 | **Due Date: 1402/10/25**

Note 1: ***purple*** *problems are bonus ones.*

1 Phase-Lead and Phase-Lag Compensators

Consider the first-order compensator with the transfer function

$$G_c(s) = \frac{K(s+z)}{s+p}$$

When $|z| < |p|$, the compensator is called a phase-lead compensator and when $|z| > |p|$, the compensator is called a phase-lag compensator. The phase-lead and the phase-lag compensators transfer functions can be written as

Phase-Lead	Phase-Lag
$G_{lead}(s) = \dfrac{K(1+\alpha\tau s)}{\alpha(1+\tau s)}$	$G_{lag}(s) = \dfrac{K\alpha(1+\tau s)}{1+\alpha\tau s}$
$\tau = \dfrac{1}{p}$	$\tau = \dfrac{1}{z}$
$\alpha = \dfrac{p}{z}$	$\alpha = \dfrac{z}{p}$

Now, for each of the phase-lead and the phase-lag compensators proof the following equations

Description	Phase-Lead	Phase-Lag
The extremum value of the phase lead or phase lag occurs at a frequency ω_m	$\omega_m = \dfrac{1}{\tau\sqrt{\alpha}}$	$\omega_m = \dfrac{1}{\tau\sqrt{\alpha}}$
Extremum phase angle $= \phi_m$	$\alpha = \dfrac{1+\sin(\phi_m)}{1-\sin(\phi_m)}$	$\alpha = \dfrac{1-\sin(\phi_m)}{1+\sin(\phi_m)}$

2 Phase-Lag Compensator Design

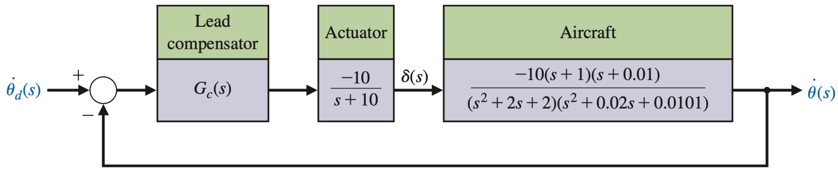
An adaptive suspension vehicle uses a legged locomotion principle. The control of the leg can be represented by a unity feedback system with

$$G(s) = \frac{K}{s(s+10)(s+14)}$$

We desire to achieve a steady-state error for a ramp input of 10% and a damping ratio of the dominant roots of $\zeta = 0.707$. Determine a suitable lag compensator, and determine the actual overshoot and the time to settle (to within 2% of the final value).

3 Phase-Lead Compensator Design

Consider the aircraft unity feedback control system shown below



where $\dot{\theta}(t)$ is is the pitch rate (rad/s) and $\delta(t)$ is the elevator deflection (rad). The four poles represent the phugoid and short-period modes. The phugoid mode has a natural frequency of 0.1 rad/s, and the short period mode is 1.4 rad/s.

a) Using Bode plot methods, design a phase-lead compensator to meet the following specifications: (1) settling time (with a 2% criterion) to a unit step of $T_s \leq 2$ seconds, and (2) percent overshoot of P.O. $\leq 10\%$.

b) Simulate the closed-loop system with a step input of $10°$/s, and show the time history of $\dot{\theta}(t)$.

4 Design for Deadbeat Response

Consider the optimized coefficients of the given closed-loop transfer function $T(s)$

$$T(s) = \frac{a_0}{a_n s^n + a_{n-1}s^{n-1} + \cdots + a_2 s^2 + a_1 s + a_0}$$

System Order n	a_6	a_5	a_4	a_3	a_2	a_1	a_0	$\rightarrow$	Settling Time T_s
$n = 2$	0	0	0	0	1	$1.82\omega_0$	ω_0^2	$\rightarrow$	$T_s = \frac{4.82}{\omega_0}$
$n = 3$	0	0	0	1	$1.90\omega_0$	$2.20\omega_0^2$	ω_0^3	$\rightarrow$	$T_s = \frac{4.04}{\omega_0}$
$n = 4$	0	0	1	$2.20\omega_0$	$3.50\omega_0^2$	$2.80\omega_0^3$	ω_0^4	$\rightarrow$	$T_s = \frac{4.81}{\omega_0}$
$n = 5$	0	1	$2.70\omega_0$	$4.90\omega_0^2$	$5.40\omega_0^3$	$3.40\omega_0^4$	ω_0^5	$\rightarrow$	$T_s = \frac{5.43}{\omega_0}$
$n = 6$	1	$3.15\omega_0$	$6.50\omega_0^2$	$8.70\omega_0^3$	$7.55\omega_0^4$	$4.05\omega_0^5$	ω_0^6	$\rightarrow$	$T_s = \frac{6.04}{\omega_0}$

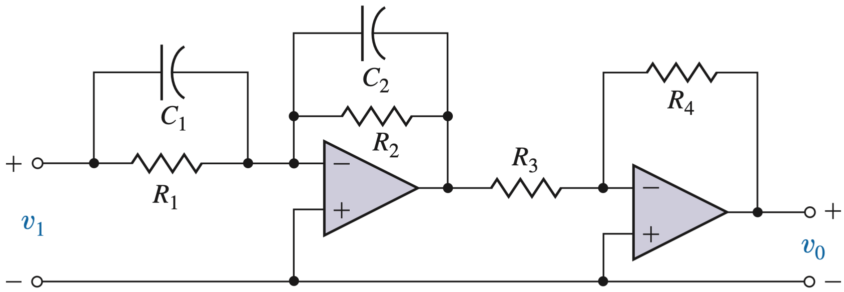
a) Determine appropriate ω_0 for each transfer function to equalize their settling time $T_s = 1$ and then plot the step response of all the optimized transfer functions above on the same axes and compare the results.

b) plot the Bode diagram of all the derived transfer functions in part b on the same axes and compare the results.

Note 2: *Use a legend to show which response is for which system.*

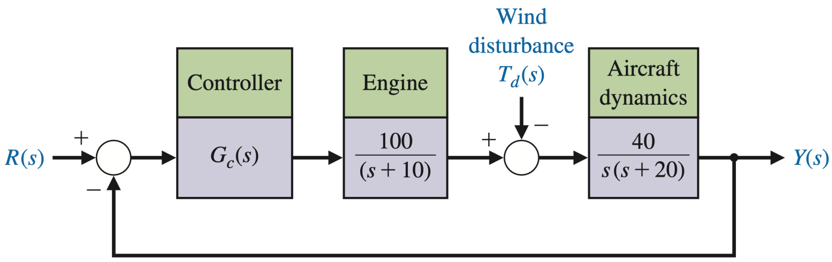
5 Operational Amplifier Circuit as Compensator

Before 1970, controllers were implemented using analog components. Considering the following circuit, derive the transfer function $G(s) = V_0(s)/V_1(s)$ and derive corresponding conditions to implement a lead compensator.



6 Bi-wing Aircraft Control

The heading control of the traditional bi-wing aircraft shown below, is represented by the following block diagram:



a) Determine the minimum value of the gain K when $G_c(s) = K$, so that the steady-state effect of a unit step disturbance $T_d(s) = 1/s$ is less than or equal to 5% of the unit step ($y(\infty) = 0.05$).

b) Determine whether the system using the gain of part a is stable.

c) Design a compensator using one stage of lead compensation, so that the phase margin is P.M. $= 30°$.

d) Design a two-stage lead compensator so that the phase margin is P.M. $= 55°$.

e) Compare the bandwidth of the systems of parts c and d.

f) Suggest a control scheme to completely suppress effect of unit step disturbance at steady-state.

g) Plot the step response $y(t)$ for the systems of parts c, d and f. Compare percent overshoot and settling time (with a 2% criterion).

MATLAB Assignments

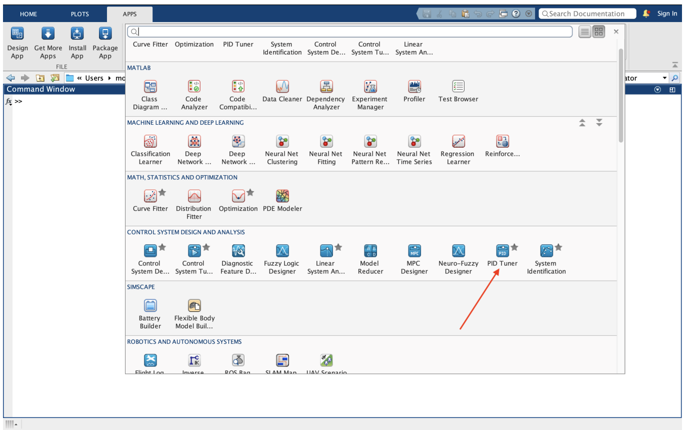
7 PID Tuner Toolbox

The PID Tuner in MATLAB is a graphical user interface (GUI) for automatically tuning PID controllers. Here's a brief outline of using the PID Tuner toolbox in MATLAB:

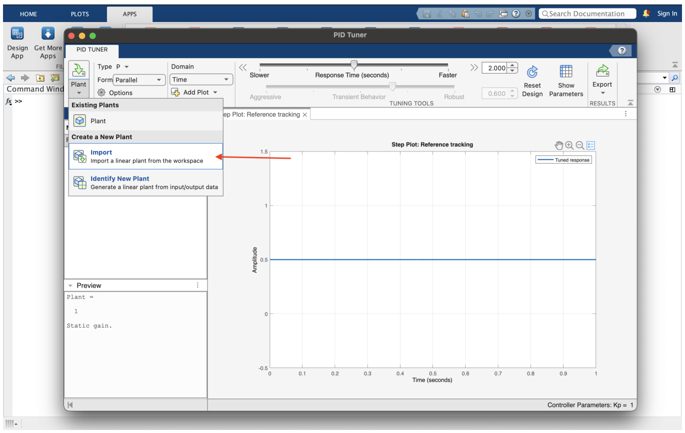
I) Create a Plant Model. You will need to have a plant model (transfer function or state-space) in MATLAB workspace. This model represents the system you want to control using a PID controller.

```
sys = tf([1], [1, 0.8, 1]);
```

II) Open the PID Tuner. You can open the PID Tuner by either typing pidTuner in the MATLAB command window or by clicking on the PID Tuner icon in the APPS tab in the MATLAB tool-strip.



III) Import the Plant. Once you have your model, you can import it to the PID Tuner by clicking on the Import Plant button in the PID Tuner window.



IV) Tune the PID Controller. After importing the plant model, PID Tuner will display the step response of the plant. You can then use PID Tuner to interactively adjust the PID controller gains to get the desired response. You can also press the Reset Design button and let MATLAB to automatically design the controller.

V) Analyze the Response. PID Tuner allows you to analyze the response of the closed-loop system to make sure it meets the desired specifications such as settling time, overshoot, etc.

VI) Export the Tuned PID Controller. Once you are satisfied with the performance, you can export the tuned PID gains to the MATLAB workspace for use in your MATLAB code.

Assignment: Design all of the P, PI, PD, and PID controllers for each of the given transfer functions. Aim to minimize settling time while ensuring that the overshoot remains below 5%. Finally, demonstrate the impact of K_p, K_i, and K_d on settling time, overshoot, and steady-state error. Don't forget to plot and compare the results. Is it even possible to stabilize the system in part d using a PID controller (use Root Locus to answer)?

a) $G_1(s) = \dfrac{1}{10s + 1}$

b) $G_2(s) = \dfrac{1}{s^2 + 0.1s + 1}$

c) $G_3(s) = \dfrac{1}{s(s^2 + 0.1s + 1)}$

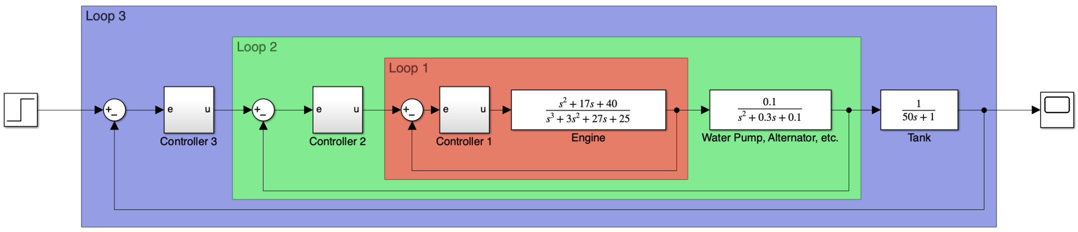
d) $G_4(s) = \dfrac{1}{s(s^4 - 1)}$

8 Cascade Control System

Cascade control is a method where the output of one (master) controller adjusts the set point of another (slave) controller, creating a system with at least two nested feedback loops. Cascade control typically is used when you have some processes with relatively slow dynamics (outer loops) and some other relatively with fast dynamic (inner loops). Cascade control should generally not be used if the inner loop is not at least three times faster than the outer loop, because the improved performance may not justify the added complexity (you should tune more controllers). In addition to the diminished benefits of cascade control when the inner loop is not significantly faster than the outer loop, there is also a risk of interaction between the two loops that could result in instability, especially if the inner loop is tuned very aggressively.

In this problem we aim to regulate the water level in a tank using an engine motor. The issue involves modifying the throttle position of the engine, resulting in a change in the engine's output power. Subsequently, we convert this output power into electrical power through an alternator and other necessary equipment to operate a water pump for filling the tank. Additionally, the tank has output flow for the automated irrigation system of a farm.

The default measurement of system is output of `loop 3`. Now, let suppose that we can measure both the engine speed (output of `loop 1`) and the water flow rate of the pump (output of `loop 2`). In this case, we can incorporate two inner loops to establish the cascade control configuration as specified. The reference signal, $r(t)$, is a unit step signal. It's important to keep in mind that the input of the engine must fall within the range $0 \leq u_1 \leq 1$. While it may be challenging to meet this constraint completely, you should satisfy this constraint as much as possible.



a) Assuming that the only thing to control is the engine's speed. Applying a unit step, design `Controller 1` to minimize the settling time of `Loop 1` while ensuring that $0 \leq u_1 \leq 1$. Choose an appropriate controller structure such as PID.

b) Now, assume that `Loop 1` is transparent to `Loop 2` due to its relatively fast dynamics (i.e., it can be completely ignored), repeat the previous step for `Loop 2`

c) Repeat the process of part b for `Loop 3`.

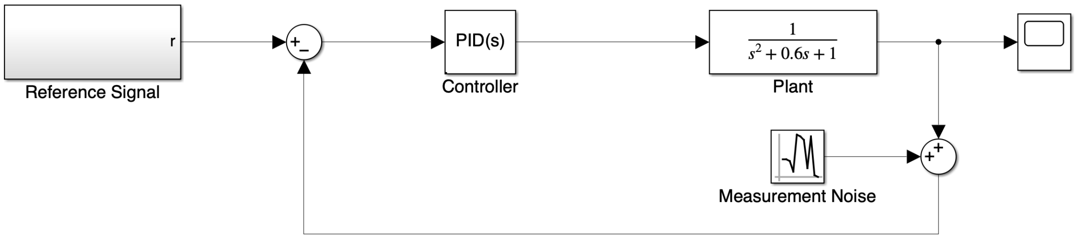
d) Implement the full cascade control structure and simulate the control system. Plot every control signal u_1, u_2 and u_3 on the same axes to demonstrate the impact of each loop on the control signal u_1. Note that to be able to implement inner loops, we just need to add two sensors, and no extra actuators is required.

e) Let's say the engine speed and the water flow rate of the pump cannot be measured. Therefore, the cascade control structure is not an option, and only one controller needs to be designed (since feedback loops 1 and 2 are no longer implementable). Your task now is to design a new one-stage controller to reduce the settling time of the closed-loop system while making sure that the control input, u_1, stays within the range of $0 \leq u_1 \leq 1$. You have the option to utilize the `pidtuner` toolbox for designing the controller.

f) Compare the effectiveness of the cascade control structure and the conventional control system. Plot the control signals on the same axes for comparison.

9 Reducing The Effect of Measurement Noise

Measurement noise in control systems refers to unpredictable and random changes in the measurements taken by sensors. The effect of measurement noise can be detrimental to the performance of a control system. It can lead to inaccurate feedback being provided to the controller, causing instability, oscillations, or poor control performance. Specifically, it limits the achievable closed-loop bandwidth. This causes the controlled system not to behave as desired and can even lead to damage or safety hazards in certain applications. We model the effect of measurement noise as follows:



a) Create two PID controllers using the `pidtuner` toolbox. One of them should have an aggressive setting (transient behavior = 0), while the other one should be set to be conservative or robust (transient behavior = 0.8). Aim for a settling time of $T_s = 1$ for your controllers. Implement the controllers in Simulink and compare their closed-loop performance while varying the measurement noise variance. Test with at least 5 different measurement noise variances in range of $0 \leq \sigma^2 \leq 1$. The reference signal $r(t)$ is a unit step signal.

We are aware that the system under our control exhibits a significantly reduced frequency response at high frequencies, and it is only at low frequencies that its frequency characteristics are clearly observable and measurable (the system's cutoff frequency is approximately 1.45 rad/sec). Conversely, we are familiar with the frequency content of measurement noise, which typically presents as white noise. Our objective now is to attenuate the high frequency components of the closed-loop system without compromising its useful frequency features, by employing a low-pass filter to mitigate the impact of noise. Suppose we should place the designed low-pass filter in the feed-forward path of the control loop and before the controller, i.e. on the error signal $e(t)$. Note that in part a, you tuned PID coefficients without considering existence of measurement noise.

b) Implement a first-order low-pass system and adjust it for a desirable closed-loop performance. Recommend one tuned system for the aggressive controller and another for the robust controller. Then, compare the closed-loop performance and the control signals $u(t)$.

As we are aware, the main system is a second-order system, and its frequency response decreases much more rapidly after the cutoff frequency compared to first-order systems. This means that we can effectively remove the frequency content beyond the cutoff frequency of the filter.

c) This time, attempt to configure two low-pass filters with duplicate poles to optimize the performance of the closed-loop control system. Is the cutoff frequency you achieved for the 2nd order low-pass filter higher than the cutoff frequency for the 1st order low-pass filter? Why? Test with at least 5 different measurement noise variances.