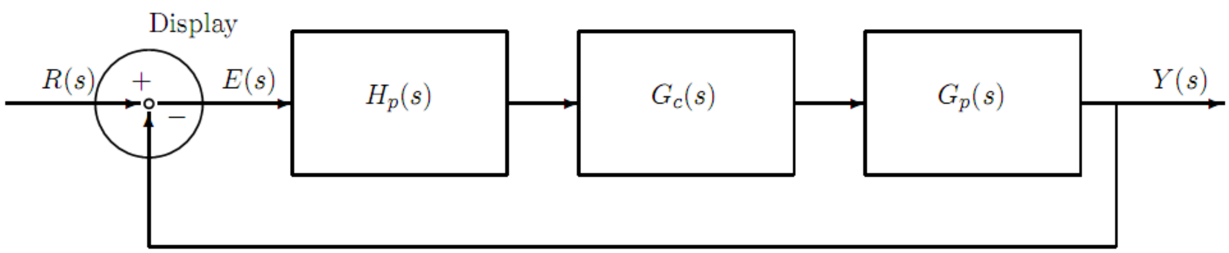
SYDE 352: Introduction to Control Systems

Design Project 2

Al Amir-khalili, 20198393
Dan Hendry, 20207096

Introduction

The purpose of the design project is to design a control $G_c(s)$ element that will yield satisfactory closed loop system performance is needed between the aircraft dynamics and the pilot of an airplane during the night provided that the activity can be modeled by the closed loop transfer function:



where,

$$G_p(s) = \frac{-10.45(s + 0.9871)}{(s + 1.204 + j1.492)(s + 1.204 - j1.492)} \quad H_p(s) = \frac{K_p \exp(-sT)}{T_N s + 1}.$$

Requirements and Objectives

- The closed loop system is table and has a gain margin of 10 dB ± 3 dB and a phase margin of 45° ± 5°.
- The tracking error is minimized for all input signals that have frequencies less than the crossover frequency.
- The sensitivity of the dominant closed loop poles to variations in TN is minimized.

Assumptions

- The exp(-sT) is approximated using eqn. 5-112 in the text book (Franklin et al. "Feedback Control of Dynamic Systems".)
- T = 0.2 s
- K_p = 20
- T_N lies within the range of 0.1 – 0.2, initially assume T_N = 0.1

Iterative Design Process: Derivation and Discussion

To satisfy the scope of this design it is clear that a controller is needed to be designed such that the open loop Nyquist plot will cross the unit circle as per the requirements. Refer to the Nyquist plot below as a reference to such desired behaviour.

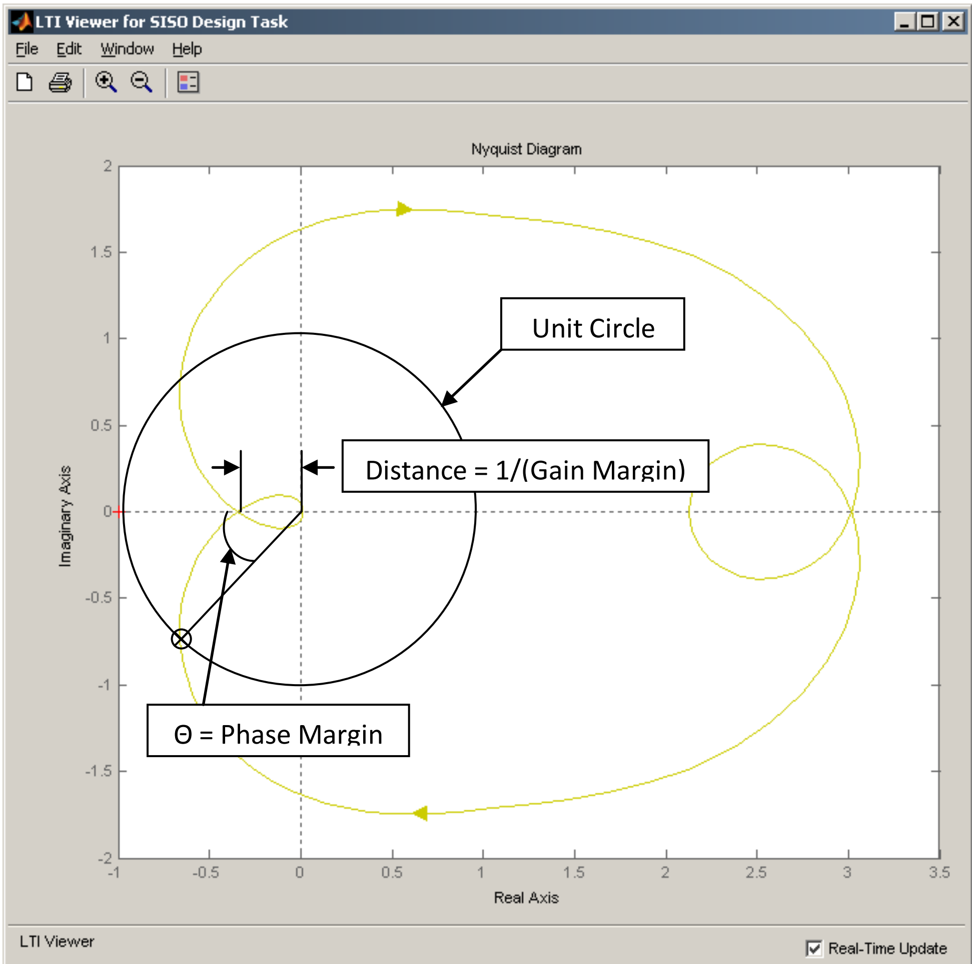


Figure 1: Annotated Nyquist Plot

In addition, it is required for the system to be optimized in such a way so that the tracking error is minimized over all input frequencies below the gain crossover frequency. This problem will be addressed later in the final design section of the report. The initial stage is to enter the transfer functions into Maple in order to first observe the characteristics of the plant. Luckily, MATLAB's sistool can easily interface with Maple using the function on the following page:

```
> RunSiso := proc(tf)

   #Matlab[evalM]("clear;");
   Matlab[setvar]("numer", ListTools[Reverse]([coeffs(sort(expand(numer(P_2))),s)]));
   Matlab[setvar]("denom", ListTools[Reverse]([coeffs(sort(expand(denom(P_2))),s)]));

   Matlab[evalM]("sys = tf(numer', denom')");
   Matlab[evalM]("sisotool(sys)");

end proc:
```

With this function call, the transfer functions derived in maple are passed into MATLAB's sisotool. This facilitates the design process by giving access to a wealth of analysis tools and allowing the design to be changed on the fly with visual feedback. After passing the transfer function into sisotool the following Nyquist plot is obtained:

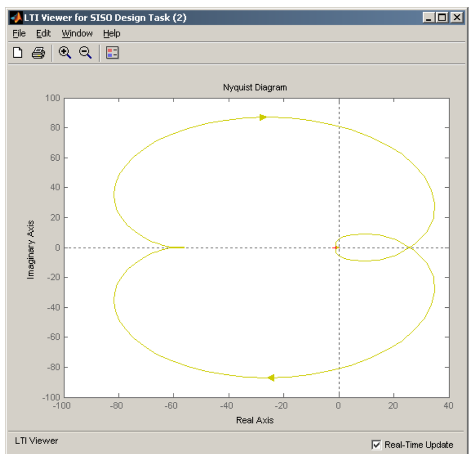


Figure 2: Original Nyquist Plot

The plant is clearly unstable since the Nyquist plot loops once around the real value of (-1). Also, the Gain Margin is too high therefore the plan needs to be scaled down. Since it is allowed for the inputs to the system to be inverted, the plan is iteratively refined by multiplying it by a gain of (-0.03) in Maple. The previous steps are repeated to produce the following Nyquist plot:

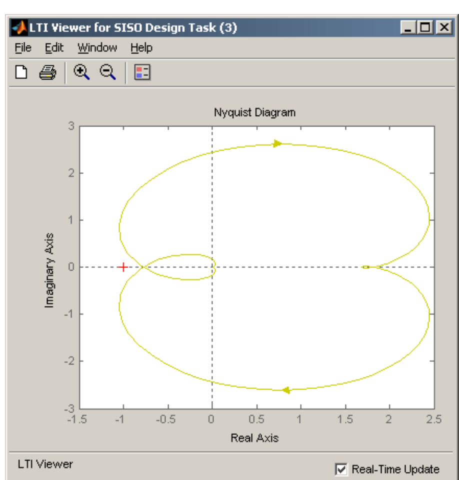


Figure 3: Nyquist Plot of the Plant with a Gain of -0.03

This is a much better result since the plant is actually stable and possesses the following characteristics as calculated by Maple:

```
Gain Margin: 2.16177         at: 6.80975
Phase Margin: 17.505         at: 5.64409
```

By looking at the Design Plot under the Graphical Tuning section of the sisotool it is observed that MATLAB gives roughly the same values:

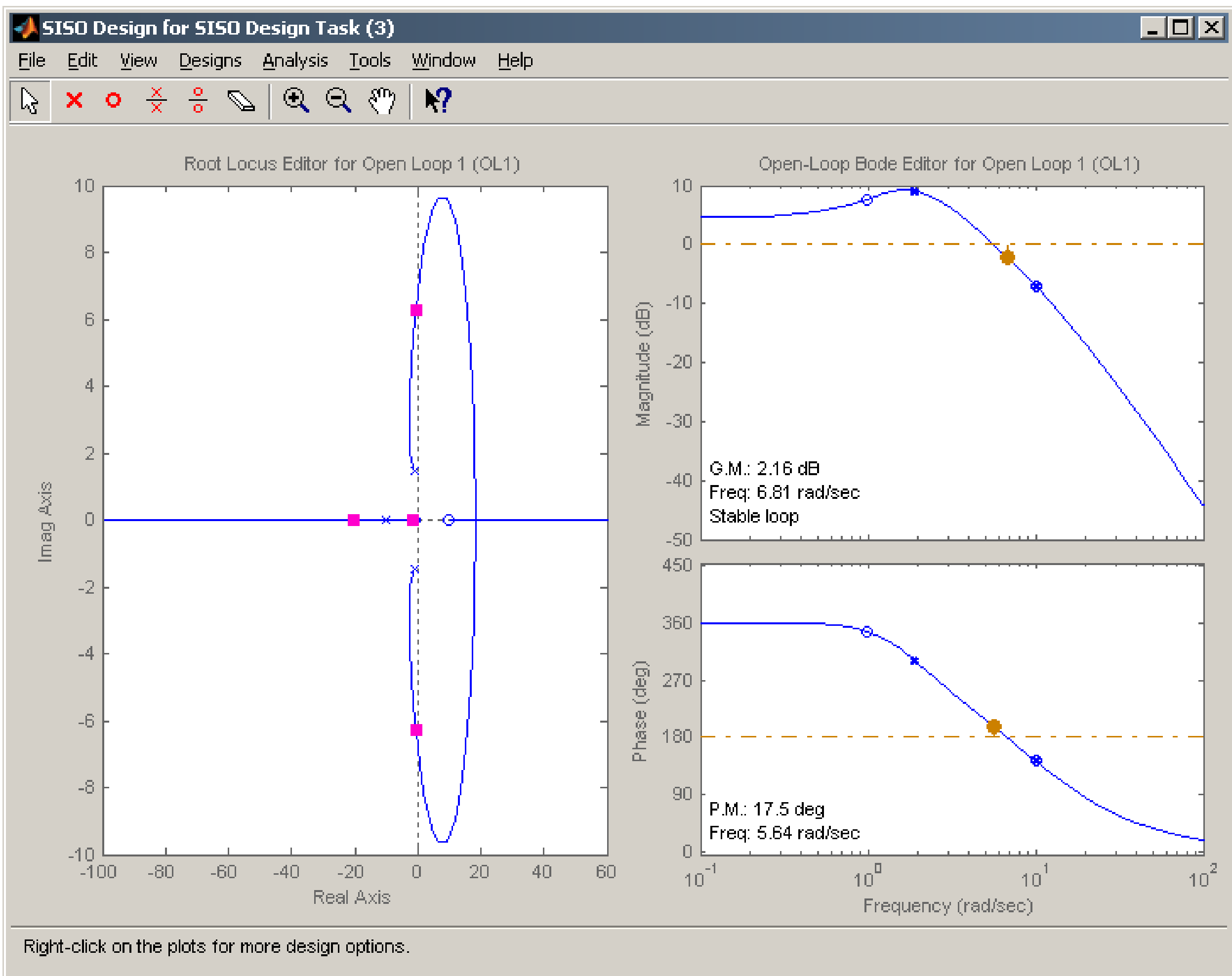


Figure 4: Sisotool Design Plot (Reference Iteration)

Using this lead or lag compensators can easily be added in order to adjust the system to meet the requirements. On the following page is an example of how a lead compensator affects the behaviour of the transfer function.

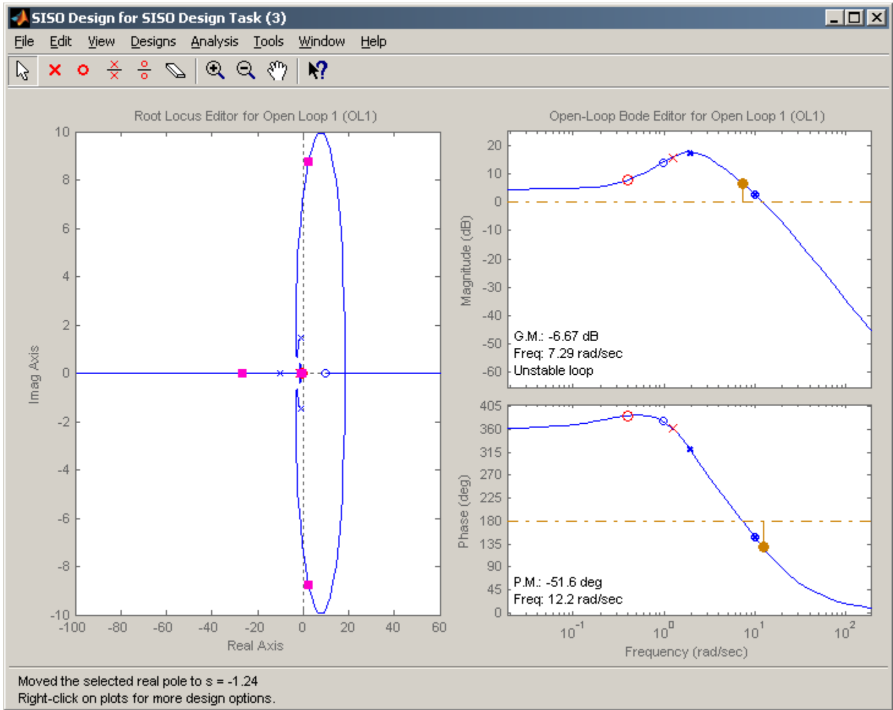


Figure 5: First Iteration with a Lead Compensator

Lead compensators can be distinguished by the placement of the poles (marked by x) in front of the zeroes (marked by o). From this it is observed that the addition of a lead compensator drastically reduces the phase angle. But since both the Gain and Phase crossover frequencies are so close together, it is hard to manipulate the Phase Margin without affecting the Gain Margin. In the next Iteration, the gain can be readjusted to manipulate the Phase and Gain margins into more acceptable values:

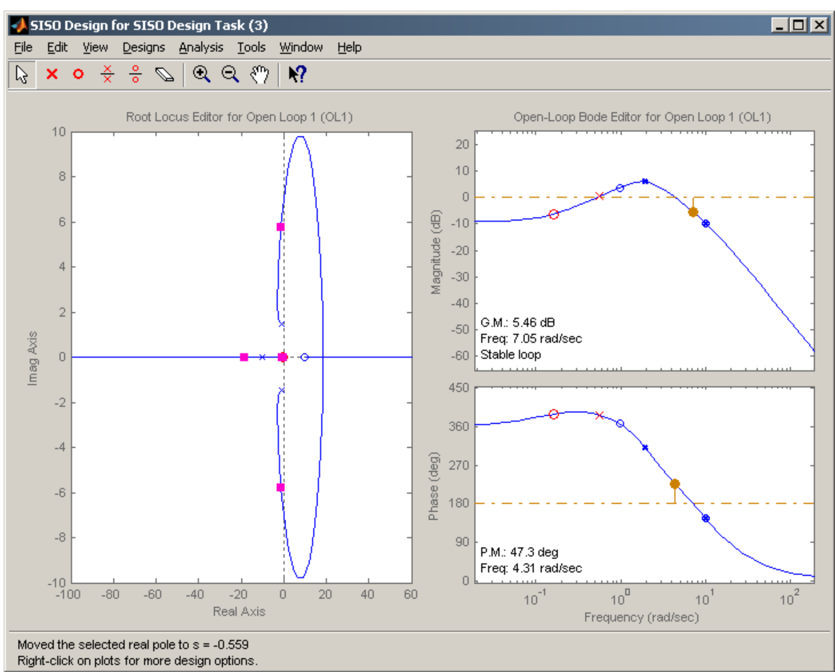


Figure 6: Second Iteration with Modified Gain After Addition of a Lead Compensator

Although the PM is satisfied, the GM is still below the minimum accepted value. Now is a good time to pause to consider what potential modifications are needed to arrive at better values. The addition of a lead compensator has not changed the slope of the “roll off” on the gain bode plot at all. A higher slope is desired in order to increase the GM without increasing the PM by too much. The picute below attempts to annotate this argument:

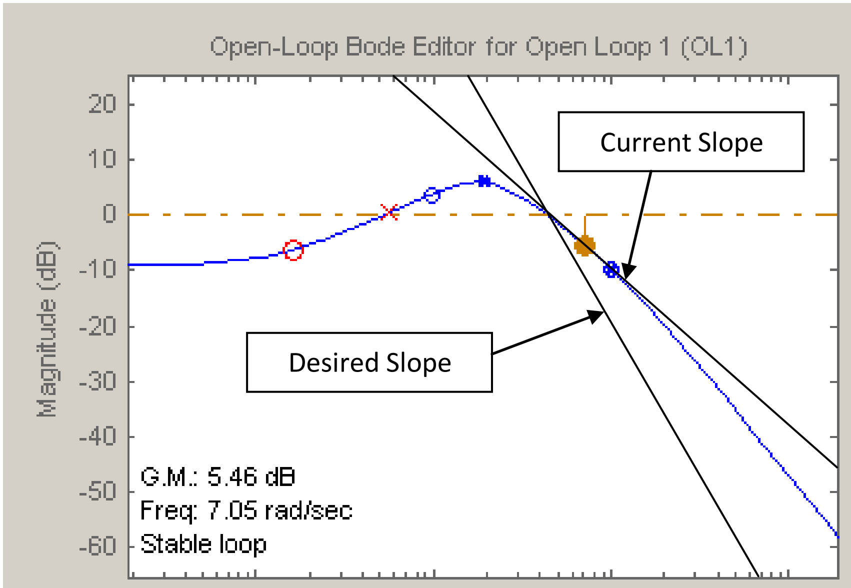


Figure 7: Annotated Explaination of “Roll Off” Slope

Unfortunately, a lead compensator does not increase the annotated slope after the crossover. Therefore, before continuing with the next design iteration, it is desired to take a step back to observe the effects of a lag compensator on the reference iteration (Figure 4):

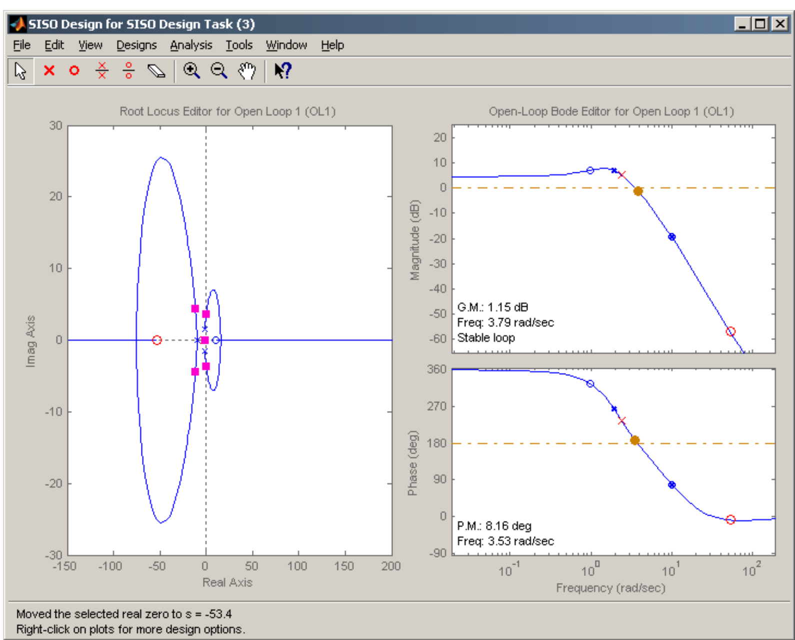


Figure 8: First Iteration with the Addition of a Lag Compensator to the Reference Iteration

By comparing Figure 8 and Figure 4 side-by-side, it becomes obvious that the lag compensator does indeed increase the slope of the “roll off”:

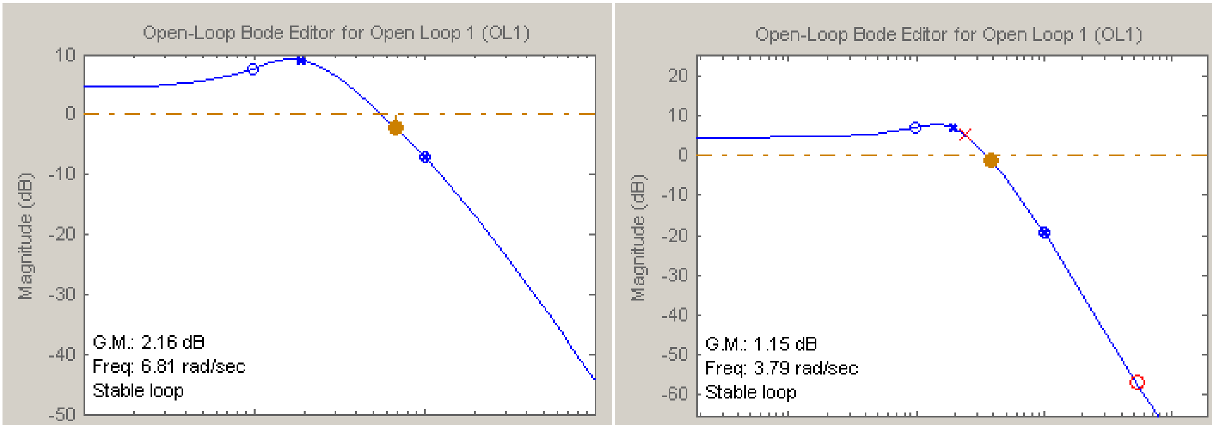


Figure 9: Side-by-Side Comparison of Figure 4 (Left) with Figure 8 (Right)

However, niether of the margins are within the desired bounds, therefore the gain must be iteratively adjusted.

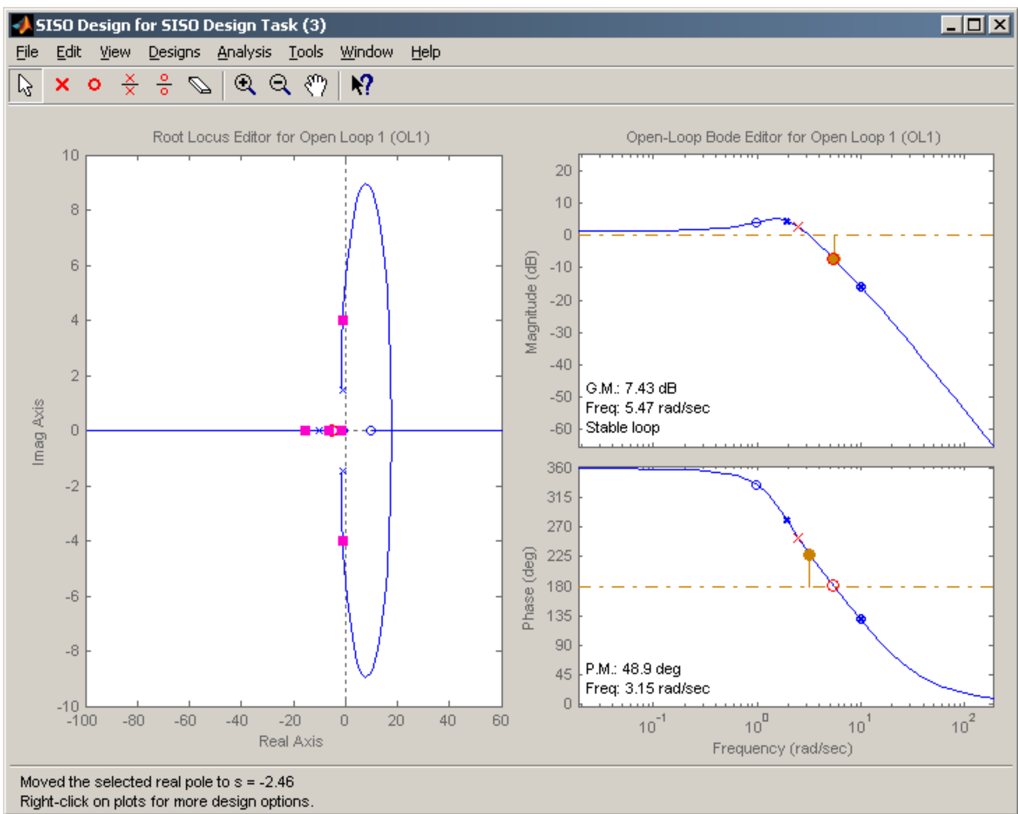


Figure 10: Second Iteration with Modified Gain after Addition of a Lag Compensator

The addition of a lag comensator has satisfied the requirements. However, the PM and GM are extremely close to their upper and lower bounds respectively. Perhaps the use of a lead compensator in tandem with a lag compensator will remedy this issue. The results of the next iteration are as follows:

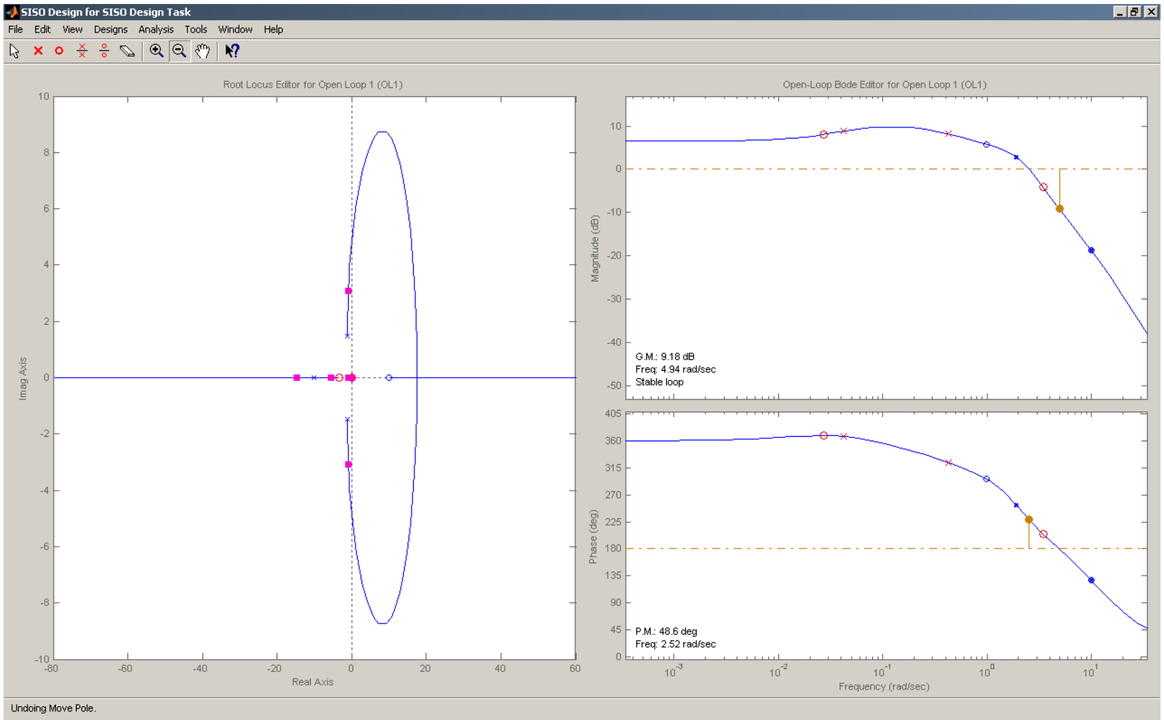


Figure 11: Third Iteration with Both Lead and Lag Compensators

The figure above shows that the requirements have been addressed in a better way without being too close to the margins. Before continuing with the next iteration, the frequency domain specification of the tracking error is left to be discussed.

Tracking error in the time domain is defined by the difference between the input and the output:

$$e(t) = r(t) - y(t)$$

The frequency domain characteristic of the tracking error remains the same once the equation above has been translated into the frequency domain. The requirement is to minimize tracking error for all frequencies below the gain crossover frequency. This is interpreted to mean that it is desired for the gain of the system to be minimized (reduced to 0dB) for all frequencies below the crossover point. This also means that the phase angle of the open loop system is to be kept as close to zero degrees as possible:

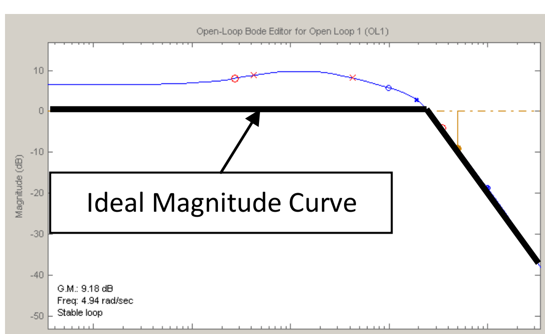


Figure 12: Annotated Ideal Magnitude Plot to Minimize Tracking Error

All of these requirements are addressed in the fourth and final iteration of the design. The final iteration was achieved by varying the poles and zeroes of both compensators along with the gain of the plant to achieve the best fitting result to that of Figure 12.

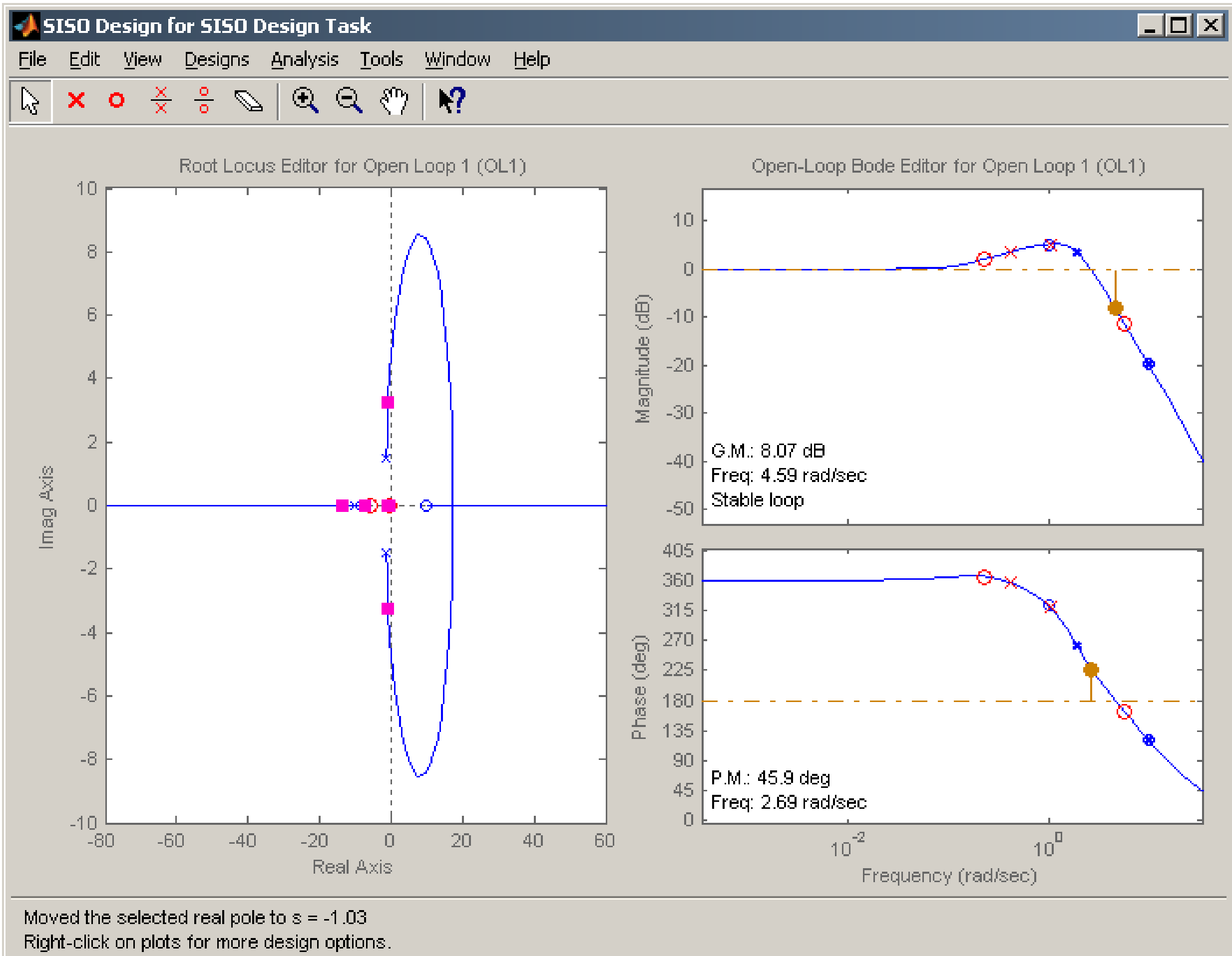
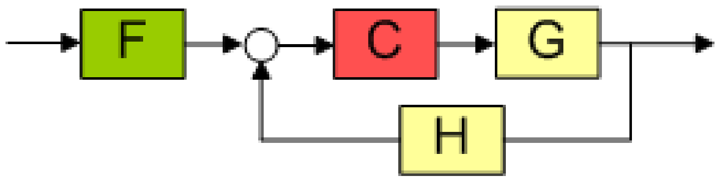


Figure 13: Final Iteration of the Control Element Design

This final iteration satisfies all of the requirements. The parameters of this iteration are listed in detail within the following Final Design section. See Appendix for the variability of these margins with respect to changing values of T_N.

Final Design

Design Configuration #1



Tunable Elements

C:

Parameter	Value
Gain	0.195868252682098
Zeros	[-0.225470988989424;-5.7135]
Poles	[-0.410257390605311;-1.0272]

F:

Parameter	Value
Gain	-0.03
Zeros	[]
Poles	[]

Fixed Elements

G:

Parameter	Value
Gain	2090
Zeros	[9.99999999651614;-0.987100000343892]
Poles	[-10+i*2.44965673222241e-007;-10-i*2.44965673222241e-007;-1.204+i*1.492;-1.204-i*1.492]

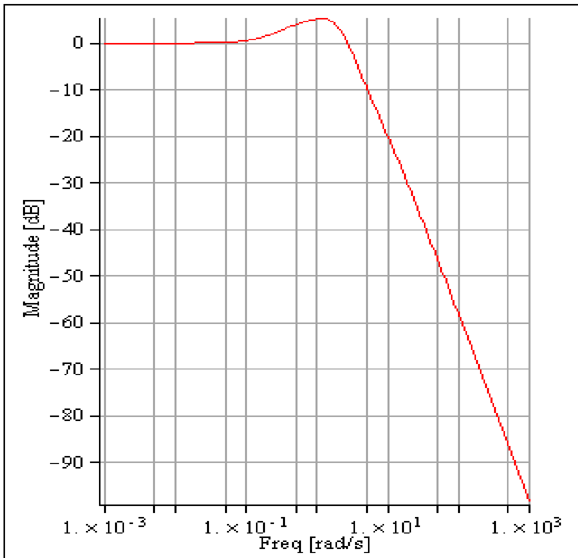
H:

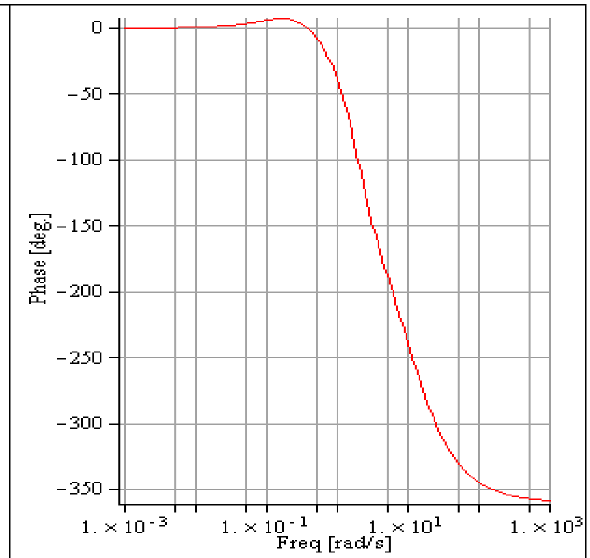
Parameter	Value
Gain	1
Zeros	[]
Poles	[]

Appendix A

Varying Values of T_N

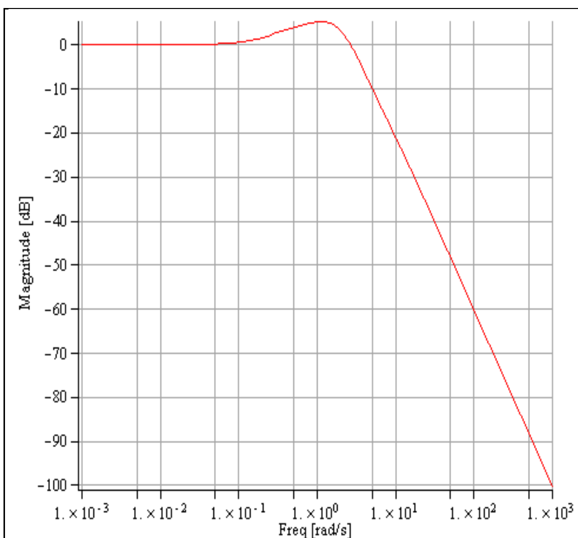
When $T_N = 0.1$:

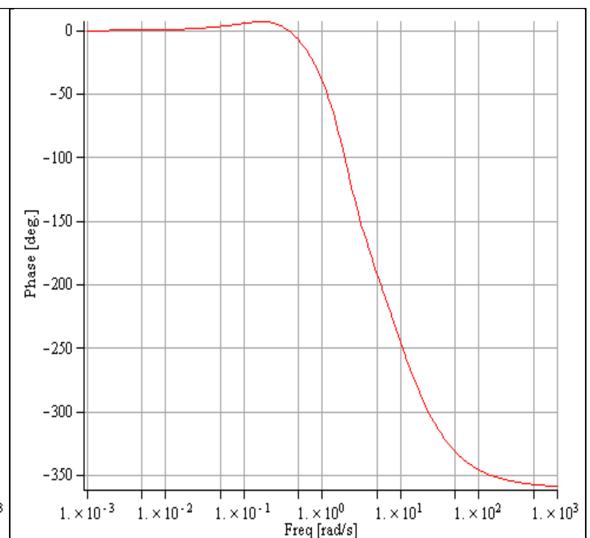




Gain Margin: 8.07 dB at: 4.59 rad/s	Phase Margin: 45.9 deg at: 2.69 rad/s

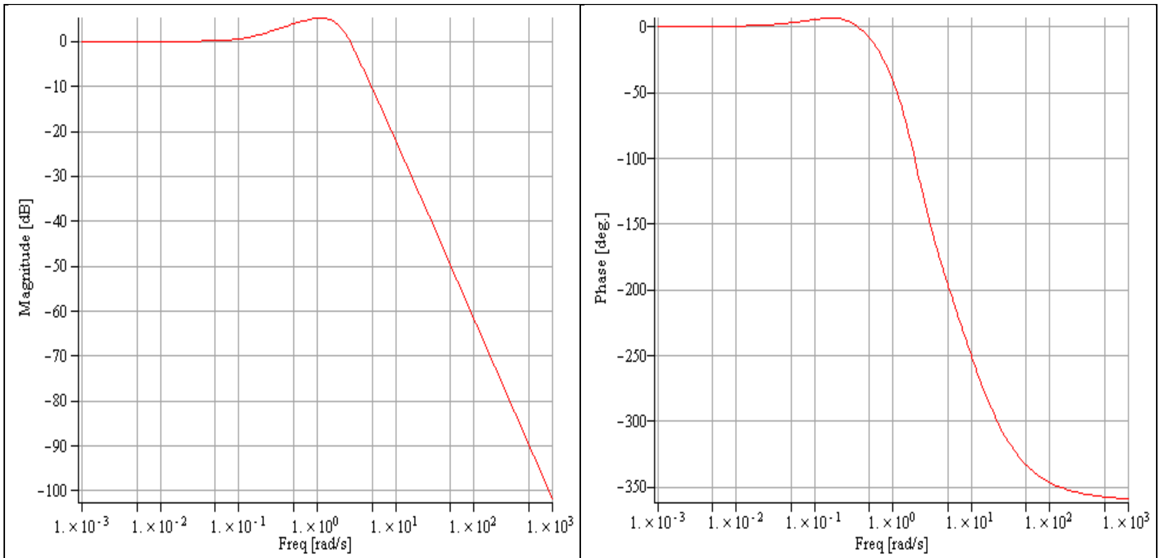
When $T_N = 0.125$:





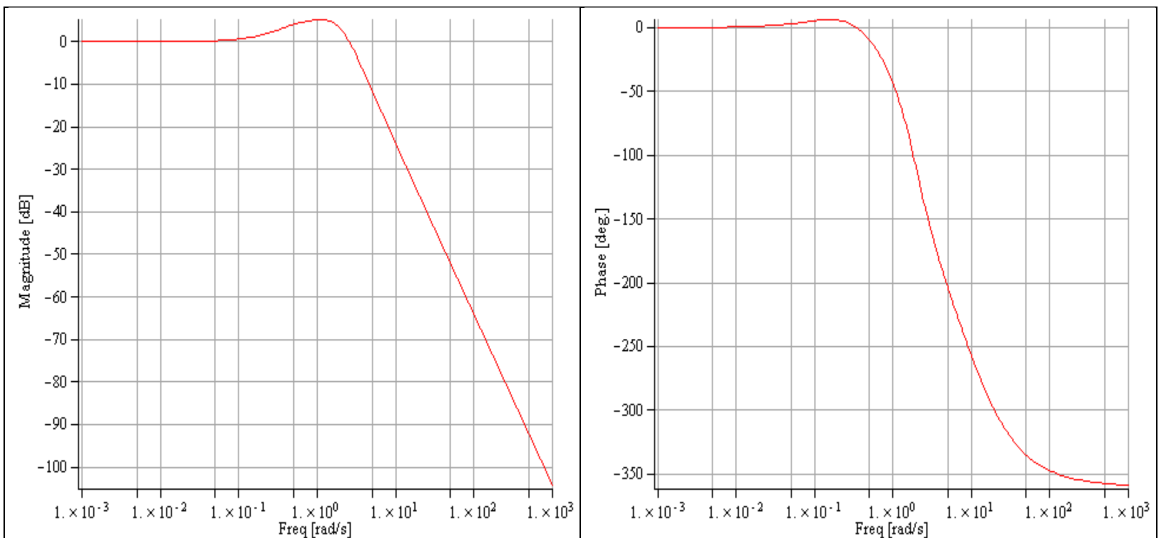
Gain Margin: 7.43 dB at: 4.30 rad/s	Phase Margin: 43.6 deg at: 2.66 rad/s

When $T_N = 0.15$:



Gain Margin: 6.93 dB	at: 4.08 rad/s	Phase Margin: 41.6 deg	at: 2.62 rad/s

When $T_N = 0.2$:



Gain Margin: 6.23 dB	at: 3.74 rad/s	Phase Margin: 38.7 deg	at: 2.55 rad/s