

ECE 351 - 52

SIGNALS AND SYSTEMS 1

LAB 7

Submitted By :
Owen Blair

Contents

1	Important Notes	2
2	Part 1 Deliverables	2
3	Part 2 Deliverables	3
4	Results	4
5	Equations	5
6	Listings	6
7	Questions	8
	7.1 Question 1	8
	7.2 Question 2	9
	7.3 Question 3	9
	7.4 Question 4	9

1 Important Notes

It is important to note that `plt.show()` is needed at the end of the python code to properly show the plots of each function. It was not included in every section of code that plots a function(s) because it is assumed that its included at the end the code.

The web address to the GitHub where \LaTeX code is stored is here:
https://github.com/Blairis123/ECE351_Reports

The web address to the GitHub where the Python code is here:
https://github.com/Blairis123/ECE351_Code

2 Part 1 Deliverables

The purpose of part 1 is to use python to perform an analysis on a block diagram as you would by hand. This starts by factoring the functions $G(s)$, $A(s)$, and $B(s)$ along with identifying the poles of each function. The hand found functions can be found in the equations section. The zeros and poles of each function are as follows:

```
1 Zeros for A = -4
2 Poles for A = -3, -1
3 Zeros for B = -14, -12
4 Poles for B = NA (No denominator to have poles)
5 Zeros for G = -4
6 Poles for G = 8, -4, -2
```

Then the factored function's poles and zeros are checked against an output from the `scipy.signal.tf2zpk()` function which are as follows:

```
1 Zeros for A
2 [-4.]
3
4 Poles for A
5 [-3. -1.]
6
7 Zeros for B
8 [-14. -12.]
9
10 Poles for B
11 []
12
13 Zeros for G
```

```
14 [-4.]  
15  
16 Poles for G  
17 [ 8. -4. -2.]
```

Next the open loop transfer function is found and can be seen in the equation section. Considering the found open loop transfer function it can be found it is unstable. This is evidenced by the fact that the open loop transfer function has a positive pole, putting a pole of the function in the right side of the plane. having a pole on the right side of the plane means that the function in question will be unstable. The instability of the open loop transfer function can be seen in Figure 1. Figure 1 is the step response of the open loop transfer function and it can be seen going to infinity and not approaching a real number as time goes to infinity. The code needed to make the calculation and plot the step response is available in Listing 1.

3 Part 2 Deliverables

Part 2 continues the exploration of using Python to do an analysis of block diagrams but this time the closed loop transfer function is considered. The closed loop transfer function solved symbolically can be found in the equations section. Using the convolve and tf2zpk functions from `scipy.signal` the closed loop transfer was found and can be seen in the equation section. The closed loop transfer function will be stable because it has no poles on the right hand side of the plane. The plot of the step response can be found in Figure 2. The plot in Figure 2 supports the closed loop transfer function being stable because the function can be seen approaching a real number. This is a good indication that the system is stable. The code used for plotting the step response and finding the transfer function can be found in listing 2.

4 Results

Figure 1: Open Loop Transfer Function Step Response

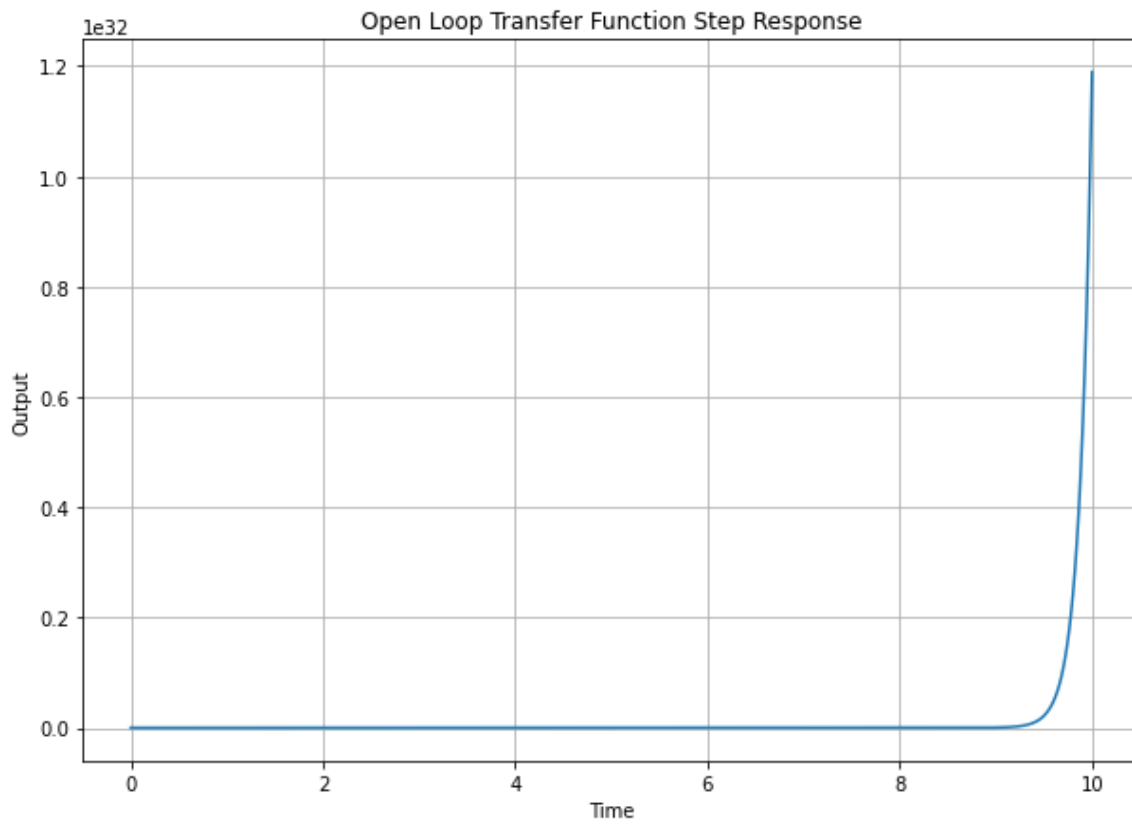
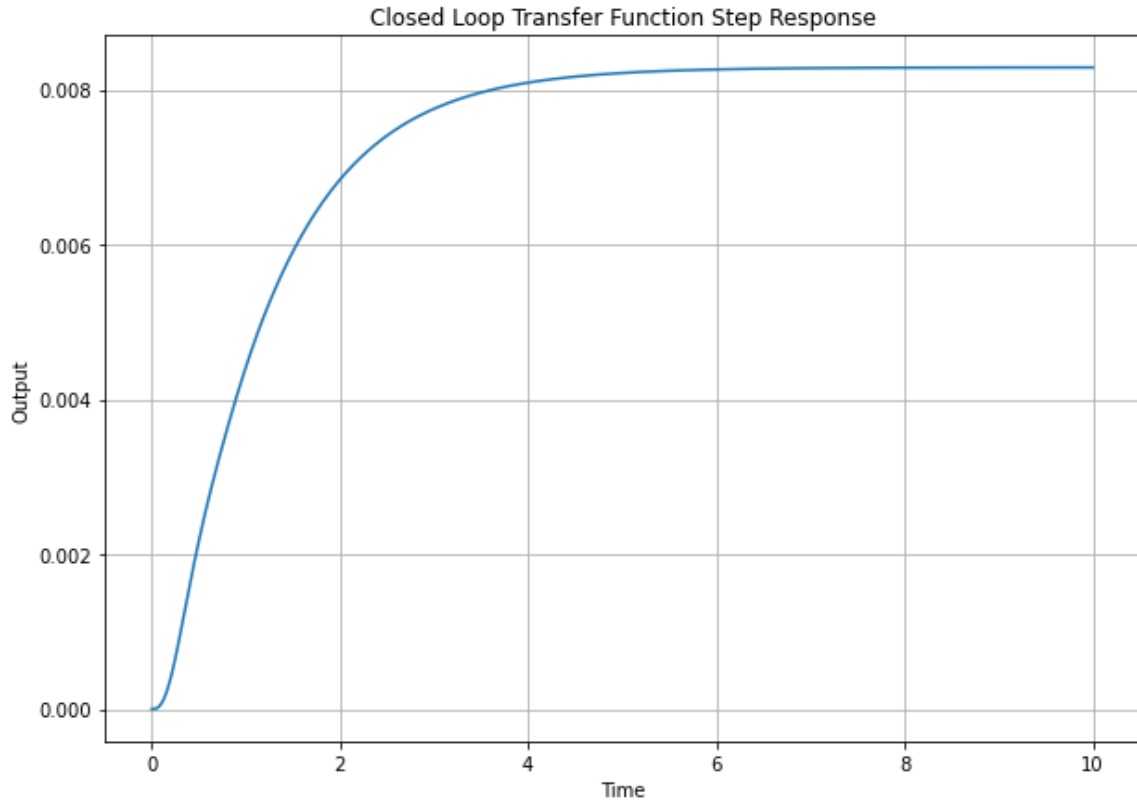


Figure : Closed Loop Transfer Function Step Response



5 Equations

Equation 1: Factored $A(s)$

$$A(s) = \frac{s + 4}{(s + 1)(s + 3)}$$

Equation 2: Factored $B(s)$

$$B(s) = (s + 14)(s + 12)$$

Equation 3: Factored $G(s)$

$$G(s) = \frac{s + 9}{(s + 4)(s + 2)(s - 8)}$$

Equation 4: Factored Open Loop Transfer Function

$$H_{T_{open}} = \frac{(s+4)(s+9)}{1+GB}$$

Equation 5: Symbolic Closed Loop Transfer Function

$$H_{T_{closed}} = \frac{AG}{1+GB}$$

Equation 6: Scipy.signal Found Transfer Function

$$H_{T_{closed}} = \frac{s^2 + 13s + 36}{2s^5 + 41s^4 + 500s^3 + 2995s^2 + 6878s + 4344}$$

6 Listings

Listing 1: Part 1

```

1  #-----END FUNCTIONS-----#
2
3      #THIS WILL BE THE SAME FOR THE ENTIRE LAB!
4
5      #Define step size
6  steps = 1e-2
7
8      #t for part 1
9  start = 0
10 stop = 10
11      #Define a range of t_pt1. Start @ 0 go to 20 (+a step) w/
12      #a stepsize of step
13  t = np.arange(start, stop + steps, steps)
14
15  #-----PART 1, OPEN-LOOP-----#
16      #Find the roots and the poles
17  A_num = [1, 4]
18  A_den = [1, 4, 3]
19
20  B_num = [1, 26, 168]
21  B_den = [1]
22
23  G_num = [1, 4]
24  G_den = [1, -2, -40, -64]
25
26  zeros_A, pole_A, gain_A = sig.tf2zpk(A_num, A_den)
27  zeros_B, pole_B, gain_B = sig.tf2zpk(B_num, B_den)
28  zeros_G, pole_G, gain_G = sig.tf2zpk(G_num, G_den)

```

```

29
30 print("Zeros for A")
31 print(zeros_A)
32 print("")
33 print("Poles for A")
34 print(pole_A)
35 print("")
36 print("Zeros for B")
37 print(zeros_B)
38 print("")
39 print("Poles for B")
40 print(pole_B)
41 print("")
42 print("Zeros for G")
43 print(zeros_G)
44 print("")
45 print("Poles for G")
46 print(pole_G)
47 print("")
48
49 H_open_num = [1, 9]
50 H_open_den = [1, -2, -37, -82, -48]
51
52 zero_H_open, pole_H_open, gain_H_open = sig.tf2zpk(H_open_num,
53     H_open_den)
54
55 print("Poles of the open loop transfer function")
56 print(pole_H_open)
57 print("")
58
59 stepTOpen, stepHOpen = sig.step((H_open_num, H_open_den), T = t)
60
61 plt.figure(figsize=(10,7))
62 plt.plot(stepTOpen, stepHOpen)
63 plt.grid()
64 plt.xlabel("Time")
65 plt.ylabel("Output")
66 plt.title("Open Loop Transfer Function Step Response")

```

Listing 2: Part 2

```

1 #-----PART 2, CLOSED-LOOP
2     -----#
3     #Finding the Numerator and Denomenator of transfer function
4 H_clo_num = sig.convolve([1,4],[1,9])
5
6 part_den = sig.convolve([1,1],[1,3])

```



```

7 H_clo_den = sig.convolve(part_den,[2,33,362,1448])
8
9 print("Closed loop numerator = ", H_clo_num)
10 print("Closed loop denemonator = ", H_clo_den)
11 print()
12
13     #This is what the num and den should look like
14 #H_clo_num = [1, 13, 36]
15 #H_clo_den = [2, 41, 500, 2995, 6878, 4344]
16
17 zeros_H_clo, pole_H_clo, gain_H_clo = sig.tf2zpk(H_clo_num,
18     H_clo_den)
19
20 print("Poles of the closed loop transfer function:")
21 print(pole_H_clo)
22 print("")
23
24     #Define transfer function!
25 stepTClosed, stepHClosed = sig.step((H_clo_num, H_clo_den), T = t)
26
27     #Make plots for pt1
28 plt.figure(figsize=(10,7))
29 plt.plot(stepTClosed, stepHClosed)
30 plt.grid()
31 plt.xlabel("Time")
32 plt.ylabel("Output")
33 plt.title("Closed Loop Transfer Function Step Response")

```

7 Questions

7.1 Question 1

Convolvoing the factored terms of the denominator and numerator results in the expanded form of the denominator and denominator. This is because the convolution of two functions is the same as the multiplication of the two functions Laplace transform. The lab 3 user defined convolution would not work. This can be shown through testing. The following is an output using the code from lab 3:

```

1 runfile('D:/U_of_I/Fall_2021/ECE-351/Lab3/Lab3_OwenBlair.py', wdir
2     ='D:/U_of_I/Fall_2021/ECE-351/Lab3')
3
4 Test1 input:  [1, 4]
5 Test2 input:  [1, 9, 4, 2]
6
7 User convolve:  [ 9. 40. 18.  8.  0.]
8
9 sig.convolve output:  [ 1 13 40 18  8]

```

The correct answer is $(s + 4)(s_9^3 s^2 + 4s + 2) = s^4 + 13s^3 + 40s^2 + 18s + 8$ which is given by the `sig.convolve` function and not the user defined function from lab 3.

7.2 Question 2

The difference between the open loop system and the close loop system is that closed loop includes the feedback loop through $B(s)$ and the open loop does not account for the feedback through $B(s)$. Stability differs in each case because the denominator of the open loop transfer function includes a pole that resides within the right hand side of plane, where the closed loop transfer function has poles that all exist within the left side of the plane. If a function does not have all of its poles on the left side of the plane then it will be unstable.

7.3 Question 3

The difference between `scipy.signal.residue()` and `scipy.signal.tf2zpk()` is that the residue command computes the partial fraction expansion where the `tf2zpk` command returns the zeros, poles and the gain from the two input arrays. This information can be found in the documentation for both functions.

7.4 Question 4

It is possible for an open loop system to be stable. Using the $A(s)$ from the equations section, and a modified $G(s)$ where $G(s) = \frac{s+9}{(s+4)(s+2)(s+8)}$ would be a stable system because all of its poles are on the left side of the plane. Using a similar method it can be shown that a closed loop system can be unstable. Using the system described in the lab and the functions $G(s)$, $B(s)$, and a modified $A(s)$, where the denominator of $A(s)$ is $(s + 1)(s - 3)$, you can find that the transfer function is unstable. This can be seen by the transfer function having a pole on the right side of the plane, caused by the $(s - 3)$ term in $A(s)$.