

Block Diagrams and System Stability

Cameron Williams

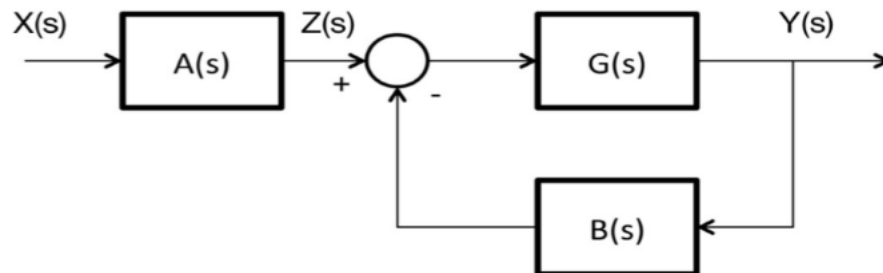
ECE 351-51

Lab Report 7

10 March 2020

1 Introduction

The objective of this lab was to practice analyzing Laplace-domain block diagrams using Python to determine system stability. For the purposes of this lab, the following block diagram and equations will be used.



$$G(s) = \frac{s + 9}{(s^2 - 6s - 16)(s + 4)}$$

$$A(s) = \frac{s + 4}{s^2 + 4s + 3}$$

$$B(s) = s^2 + 26s + 168$$

2 Methodology

I began by factoring the provided equations $G(s)$, $A(s)$, and $B(s)$. The factored form of each of these equations may be seen in equations (1), (2), and (3) of the results section respectively. From the factored equations, it is apparent that: $G(s)$ has a root at -9 and poles at 8, -2, and -4; $A(s)$ has a root at -4 and poles at -1 and -3; and $B(s)$ has roots at -14 and -12 (no poles). I used the `scipy.signal.tfzpk()` function on $G(s)$ and $A(s)$ to confirm the roots and poles I found for them. I also used `numpy.roots()` to confirm the roots that I found for $B(s)$. I printed the results of these functions in the Python console, which resulted in the following:

```
G(s) has  
zeroes: [-9.]  
poles: [ 8. -4. -2.]
```

```
A(s) has  
zeroes: [-4.]  
poles: [-3. -1.]
```

```
B(s) has  
zeroes: [-14. -12.]
```

Using the factored form of $A(s)$ and $G(s)$, I determined the open-loop response of the system described in the block diagram. The equation describing the open-loop response may be seen in equation (4) of the results section. The open-loop response is not stable because it has a positive pole. A plot of the open-loop step response may be seen in Figure 1 of the results section. I plotted this by first using the `scipy.signal.convolve()` to find the numerator and denominator of the open-loop step response then using the `scipy.signal.step()` function to find the step response. The plot agrees with my previous observation of instability since the plot diverges to infinity.

Next, I determined the closed-loop response of the system. The symbolic equation for the closed-loop response may be seen in equation (5) of the Results section. Using the symbolic equation and the `scipy.signal.convolve()` function, I determined the numerator

and denominator of the closed-loop response. I then used the closed-loop numerator and denominator in the `scipy.signal.tf2zpk()` function to determine the roots and poles of the closed-loop response. I once again printed the results in the Python console and found the following:

```
Closed Loop has  
zeroes: [-9. -4.]  
poles: [-5.16237064+9.51798197j -5.16237064-9.51798197j -6.17525872+0.j  
-3.          +0.j          -1.          +0.j          ]
```

Unlike the open-loop response, the closed-loop response has no positive poles and is thus stable. A plot of the closed-loop step response may be seen in Figure 2 of the Results section. The plot agrees with my analysis of stability because it converges.

3 Results

$$G(s) = \frac{(s + 9)}{(s - 8)(s + 2)(s + 4)} \quad (1)$$

Factored form of provided equation G(s). It is apparent that it has root -9 and poles 8, -2, and -4.

$$A(s) = \frac{(s + 4)}{(s + 1)(s + 3)} \quad (2)$$

Factored form of provided equation A(s). It is apparent that it has root -4 and poles -3 and -1.

$$B(s) = (s + 14)(s + 12) \quad (3)$$

Factored form of provided equation B(s). It is apparent that it has roots -14 and -12.

$$Y(s) = \frac{X(s)(s + 4)(s + 9)}{(s - 8)(s + 2)(s + 4)(s + 1)(s + 3)} \quad (4)$$

Open-loop transfer function of the system described by the block diagram provided.

$$H(s) = \frac{(A_{num})(G_{num})}{((G_{den}) + (B_{num})(G_{num}))A_{den}} \quad (5)$$

Symbolic description of the closed-loop transfer function of the system described by the block diagram provided.

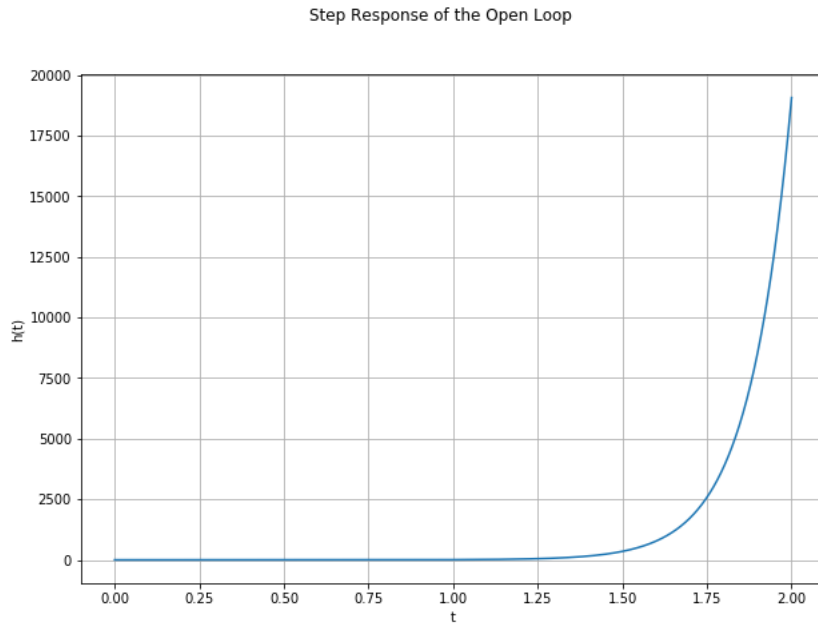


Figure 1

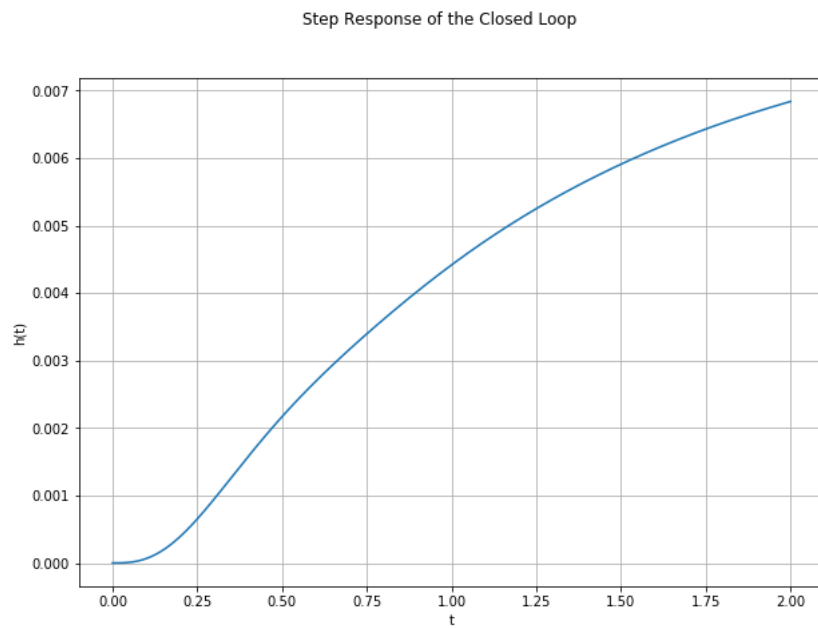


Figure 2

Conclusion

The `scipy.signal.convolve()` and `scipy.signal.tf2zpik()` functions are incredibly useful for processing and analyzing block diagrams and their component equations. With proper use, they can be used to analyze systems that would be infeasible or incredibly cumbersome to analyze by hand.

Questions

1. In Part 1 Task 5, why does convolving the factored terms using `scipy.signal.convolve()` result in the expanded form of the numerator and denominator? Would this work with your user-defined convolution function from Lab 3? Why or why not?

Using `scipy.signal.convolve()` in this context results in the expanded form of the numerator and denominator because this is the default behavior of the function in this context. In contrast to this, the user-defined convolution created in Lab 3 is programmed to only output an array of sums of the convolution at each inputted x-value of the time array provided.

2. Discuss the difference between the open- and closed-loop systems from Part 1 and Part 2. How does stability differ for each case, and why?

As noted before, the open-loop system has a positive pole and is thus unstable whereas the closed-loop system does not have a positive pole and is thus stable. The instability caused by the positive pole in the open-loop system is eliminated by the negative feedback loop present in the closed-loop system.

3. What is the difference between `scipy.signal.residue()` used in Lab 6 and `scipy.signal.tf2zpk()` used in this lab?

While the `residue` and `tf2zpk` produce similar results, `tf2zpk` is designed to work for linear problems (such as the one provided) whereas `residue` can be used to compute the partial fraction expansion of non-linear systems. `tf2zpk` stands for transfer function to Z, P, and K and is named as such because it accepts a transfer function as input and outputs the transfer function's zeroes, poles, and gain. I surmise that using `tf2zpk` is probably more efficient computationally since the problem domain it addresses is smaller (linear vs. non-linear calculations).

4. Is it possible for an open-loop system to be stable? What about for a closed-loop system to be unstable? Explain how or how not for each.

It is possible for an open-loop system to be stable. In this case, the open-loop system would have been stable if the $G(s)$ equation simply did not have the $(s-8)$ term in the

denominator. Similarly, not all closed-loop systems are stable. The instability causing (s-8) term just as easily could have been present in the $A(s)$ part of the system in which case the closed-loop system would have been unstable just as the open-loop system was.

5. Leave any feedback on the clarity/usefulness of the purpose, deliverables, and expectations for this lab.

The purpose, deliverables, and expectations for this lab were communicated clearly.