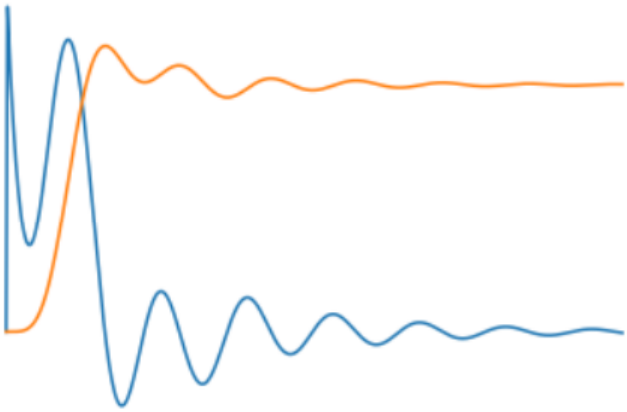


Project Report for ECE 351

Lab 6: Partial Fraction Expansion

Andrew Gibson

21 February 2023

https://github.com/gibs0630/ECE351_Code

https://github.com/gibs0630/ECE351_Reports

Contents

1 Introduction

With a linear differential equation that is complex or has many many terms, it is most likely easier to solve using Laplace Transforms. However, one of the key algebraic actions in the process is partial fraction expansion. once the terms are expanded, it is then trivial to inverse Laplace back, even if there are complex values. Using signal synthesizer software these can be done automatically to find a signal's output after going through a complex system.

2 Equations

Formula's used unit step function

$$u(x) = \begin{cases} 0 & t < 0 \\ 1 & t \geq 0 \end{cases}$$

complex inverse Laplace

$$\mathcal{L}^{-1}\left\{\frac{(|k|\angle k)}{s + (\alpha + j\omega)}\right\} = 2|k|e^{\alpha t}cos(\omega t + \angle k)u(t)$$

equations from the lab

$$y''(t) + 10y'(t) + 24y(t) = x''(t) + 6x'(t) + 12x(t)$$

$$y^{(5)}(t) + 18y^{(4)}(t) + 218y^{(3)}(t) + 2036y^{(2)}(t) + 9085y^{(1)}(t) + 25250y(t) = 25250x(t)$$

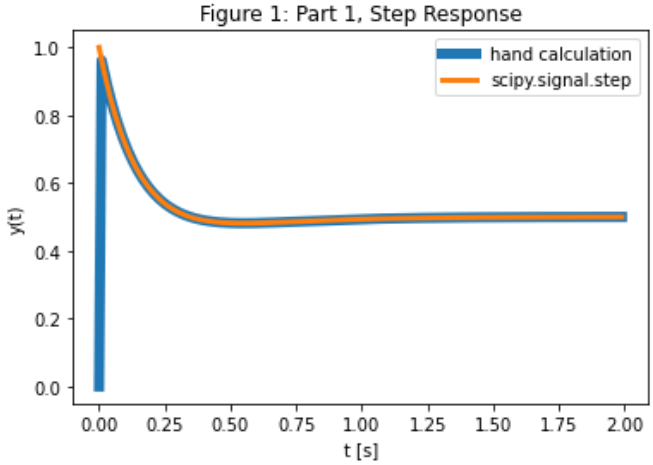
3 Methodology

This lab had us analyze differential equations via hand calculations and then create a model in python to compare them.

4 Results

Part 1

Figure 1: Part 1, Step Response



The purpose producing Figure 1 to show that the step response from the hand calculation was the same as the computed using the step function from the library skipy.

Figure 2: Partial Fraction Expantion of Simple Function

```
transfer function
R [ 2. -6.]
P [-4. -6.]
K [1.]

step response
R [ 0.5 -0.5  1. ]
P [ 0. -4. -6.]
K []
```

Figure 2 is the Partial Fraction Expansion output from the transfer function and step response using the residue function from the library skipy.

Part 2

Figure 3: Partial Fraction Expantion of Complex System

```
system
R [-1.45673077-4.12740385j -1.45673077+4.12740385j  2.14619635+0.j
  0.3836326 +0.97651934j  0.3836326 -0.97651934j]
P [ -3. +4.j  -3. -4.j -10. +0.j  -1.+10.j  -1.-10.j]
K []
```

Figure 3 is the Partial Fraction Expansion output complex system using the residue function from the library skipy. of note there are complex values for the poles (offset to the s in the denominator)and for the residue (numerator in s-space)

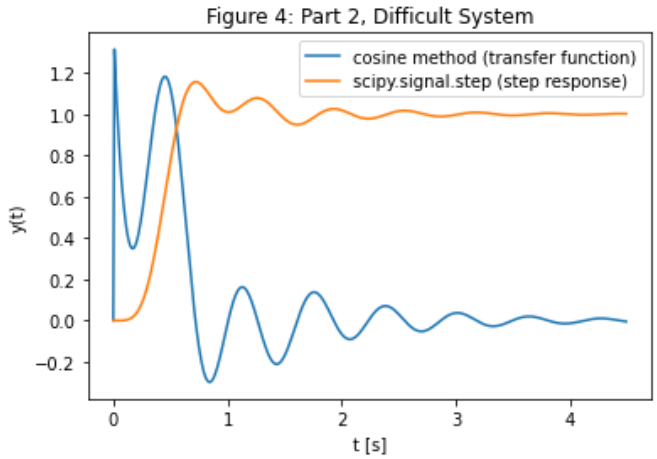


Figure 2 shows the cosine method computed from using complex terms from a partial fraction expansion. The step response would be the integral, of the transfer function, and that appears to be correct.

5 Questions

Q:1. For a non-complex pole-residue term, you can still use the cosine method, explain why this works.

A: 1. You can still use the cosine method because when both the numerator and denominator of the terms, "$a/(s+b)$", are in the real domain, then the angular frequency and angular displacement are 0. cosine of 0 is 1, which then harmlessly removes the term and we end up with out when the expected "$a * e^{-bt}$" equation.

Q:2. Leave any feedback on the clarity of the expectations, instructions, and deliverables.

A: 2. The provided book does not list a complete exhaustive list of identities that can be used to derive laplaces and their inverses, however there was a clear confusion in the idenitying what term is laplaced in the "The Cosine Method" section of the book as compared to the 4.3 task 3 of the lab. If there was a formula written out for accepting complex roots in isolation compared to when paired up with their complement, it woudl be easier to understand.