ECE 351-51

MARCH 22, 2022

# Lab 8

*Submitted By :*
Kennedy Beach

# Contents


1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2
2 Equations . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2
3 Methodology . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2
4 Results . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3
5 Error Analysis . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5
6 Questions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5
7 Conclusion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

# 1 Introduction

The purpose of this lab was to hand-calculate the coefficient expressions of a Fourier series function, then use Python to display the series at different intervals.

# 2 Equations

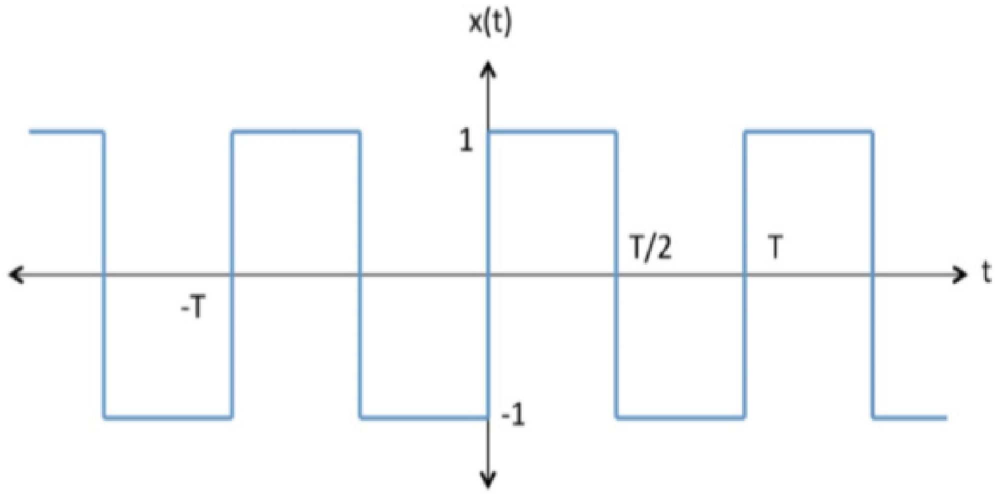


Figure 1: Given Square Wave

The coefficients and general expression of the Fourier series found are given below:

$$a_k = 0$$

$$b_k = \frac{2}{k\pi}(1 - cos(k\pi))$$

$$x(t) = \sum_{k=1}^{N} \frac{2}{k\pi}(1 - cos(k\pi)) * sin(k\frac{2\pi}{T}t)$$

# 3 Methodology

The first part involved defining the functions for the $a_k$ and $b_k$ coefficients. The equations are seen above.The values for $a_0, a_1, b_1, b_2,$ and $b_3$ are seen in the output in the Results section.

```python
import numpy as np
import matplotlib.pyplot as plt

def a_k(k):
    return 0;

def b_k(k):
    b_k = 2/(k*np.pi)*(1 - np.cos(k*np.pi))
    return b_k
```

Listing 1: Coefficient functions

The Fourier series expression, x(t), was then defined as a function.

```python
def fourier_series(N,T,t):
    omega = 2*np.pi/T
    y = np.zeros(t.shape)
    for i in range(1,N+1):
        y = y + (b_k(i)*np.sin(i*omega*t))
    return y
```

Listing 2: Fourier series function definition

Lastly, the Fourier series was plotted for N = 1,3,15,50,150,1500 from 0 to 20 seconds, where T was 8 seconds.

```python
steps = 1e-5
t = np.arange(0, 20 + steps, steps)
x1 = fourier_series(1,8,t)
x3 = fourier_series(3,8,t)
x15 = fourier_series(15,8,t)
x50 = fourier_series(50,8,t)
x150 = fourier_series(150,8,t)
x1500 = fourier_series(1500,8,t)
```

Listing 3: Steps and Fourier series variables that were plotted

# 4 Results

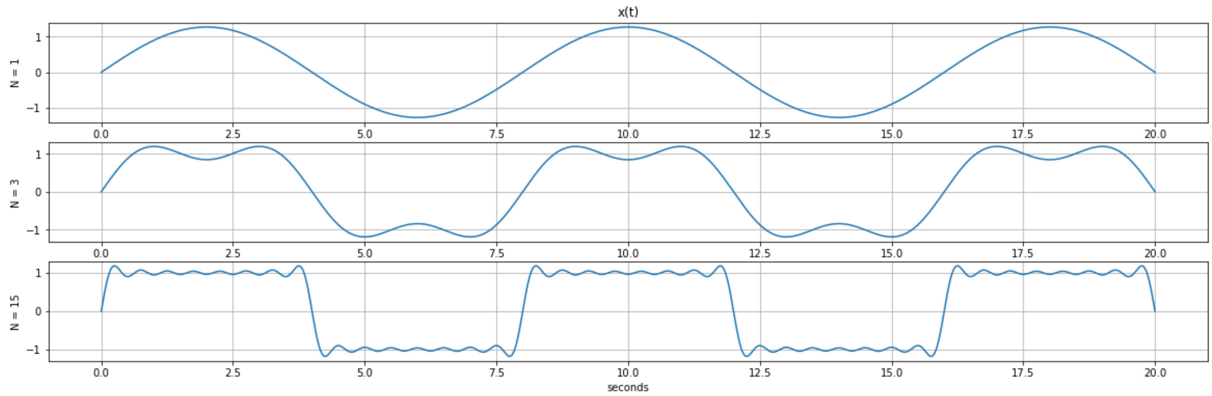


Figure 2: Plots of Fourier series where N = 1,3,15

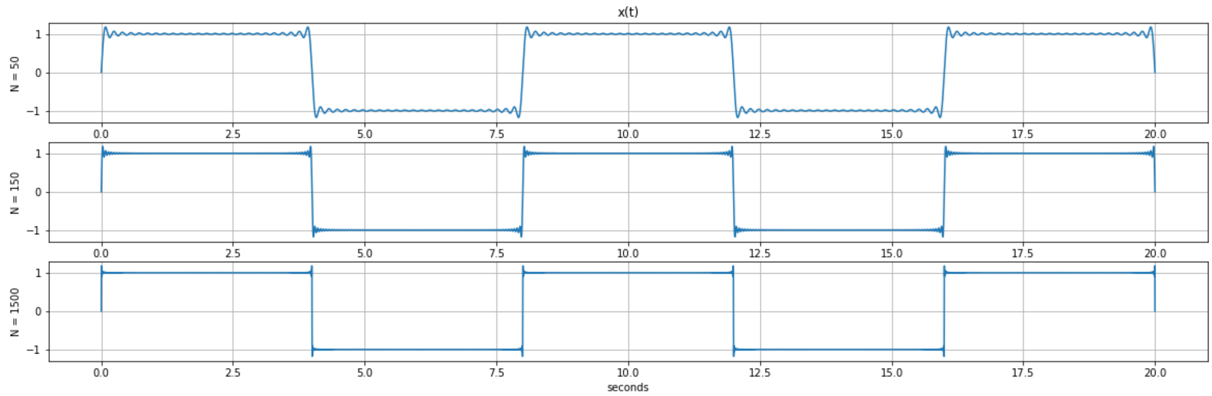


Figure 3: Plot of Fourier series where N = 50,150,1500

```
In [10]: runfile('D:/College/Spring 2022/ECE_351/Lab8/Beach_Kennedy_ECE351_Lab8.py', wdir='D:/College/Spring 2022/ECE_351/Lab8')
a0:  0
a1:  0

b1:  1.2732395447351628
b2:  0.0
b3:  0.4244131815783876
```

Figure 4: Printed output of $a_0, a_1, b_1, b_2$, and $b_3$

## 5 Error Analysis

There was an initial issue where I had y in Fourier series function as an array and was adding the $i^{th}$ indexes. I had to add all of y together with the next $i^{th}$ $b_k * sin(i * omega * t)$ term or else I would get "ValueError: setting an array element with a sequence".

## 6 Questions

**1. Is x(t) an even or an odd function? Explain why.**

x(t) is an odd function since it is not symmetrical about the y-axis.

**2. Based on your results from Task 1, what do you expect the values of a2, a3, . . . , an to be? Why?**

The value of $a_k$ is always 0 since that is what the equation simplified to. Also, since x(t) is an odd function, $a_k$ will be zero since the integral of a cosine (sine) will be multiplied by a cosine which equates to 0.

**3. How does the approximation of the square wave change as the value of N increases? In what way does the Fourier series struggle to approximate the square wave?**

As N increases, the approximation gets closer to resembling a square wave. The Fourier series struggles to eliminate the vertical edges that go above the horizontal plateau of the square wave.

**4. What is occurring mathematically in the Fourier series summation as the value of N increases?**

More summations of the terms will occur, which leads to a plot that looks more like a square wave. The individual summations get smaller and smaller as N, and k, increase.

**5. Leave any feedback on the clarity of the expectations, instructions, and deliverables.**

Everything was clear on what needed to be done and turned in.

# 7 Conclusion

This lab was very simple since it simply showcased how increasing the number of summations in a Fourier series will approximate to a square wave. The Python and LaTeX code are seen in `https://github.com/Eniac618/ECE351_Code` and `https://github.com/Eniac618/ECE351_Reports` respectively.