Synchronous Machine Dynamic Simulation

Soham Karanjikar

Performed: April 7th 2020
Written: April 11th 2020

1 INTRODUCTION

The objective of this lab was to complete a simulation in a software of our choice to see the behaviour of a synchronous motor. It gave us exposure to using forward Euler method at a large scale and at small increments and see the behavior of our machine. We simulated suddenly adding loads onto our motor and seeing if it can maintain stability, then we are asked to see if equal area criteria gives us the same maximum sudden load.

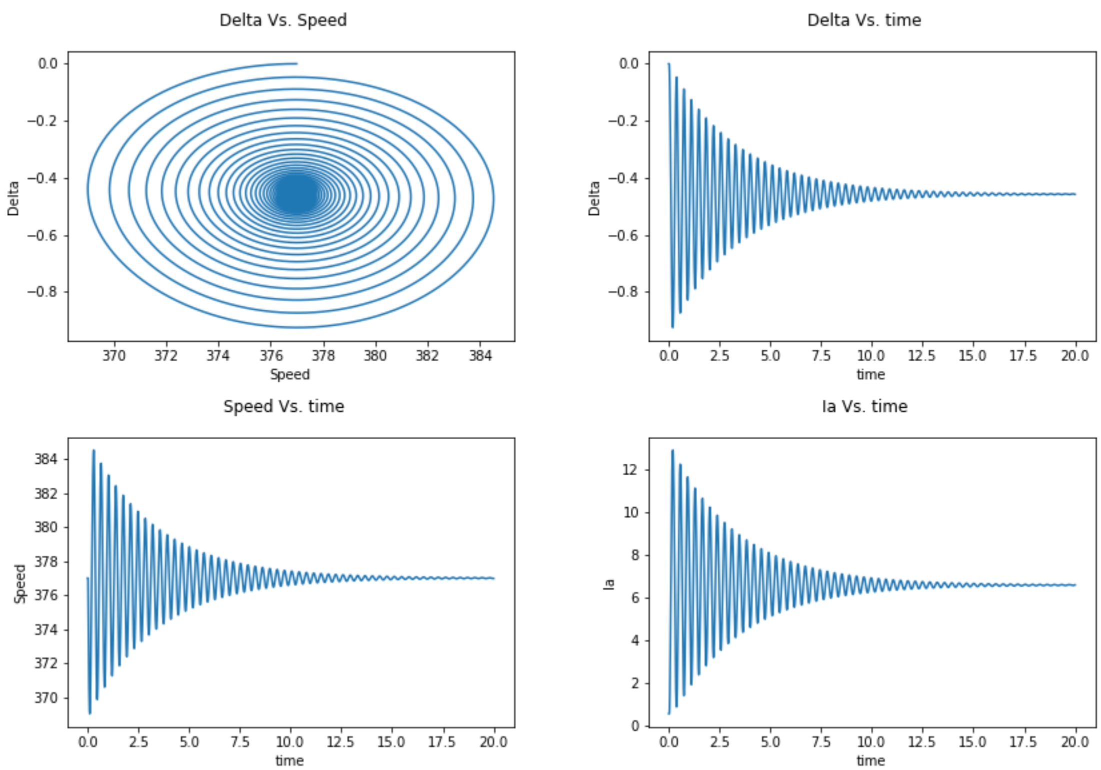
All data/measurements taken are provided in the appendix at the bottom of the document.

2 RESULTS

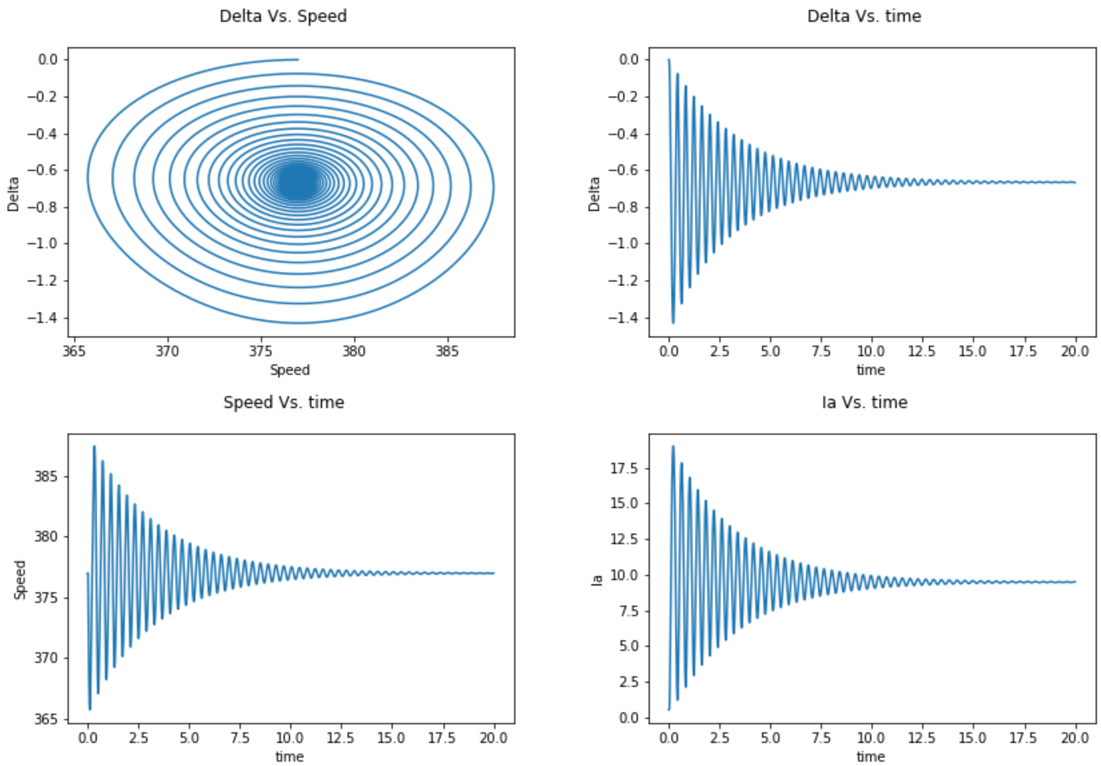
2.1 *Data of different Sudden Loads*

The data collected from the first part of the lab is below, with the data we plot to form the graphs.

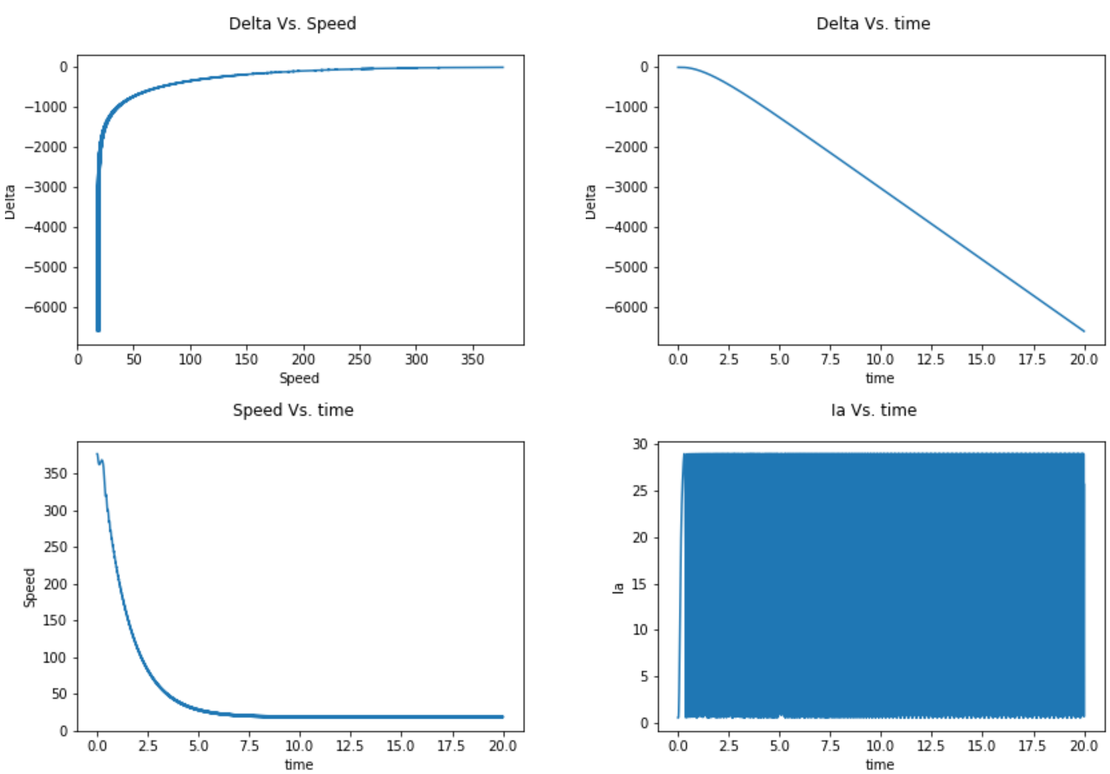
Adding a sudden load of 2500W gives us the following results:



Adding a sudden load of 3500W gives us the following results:

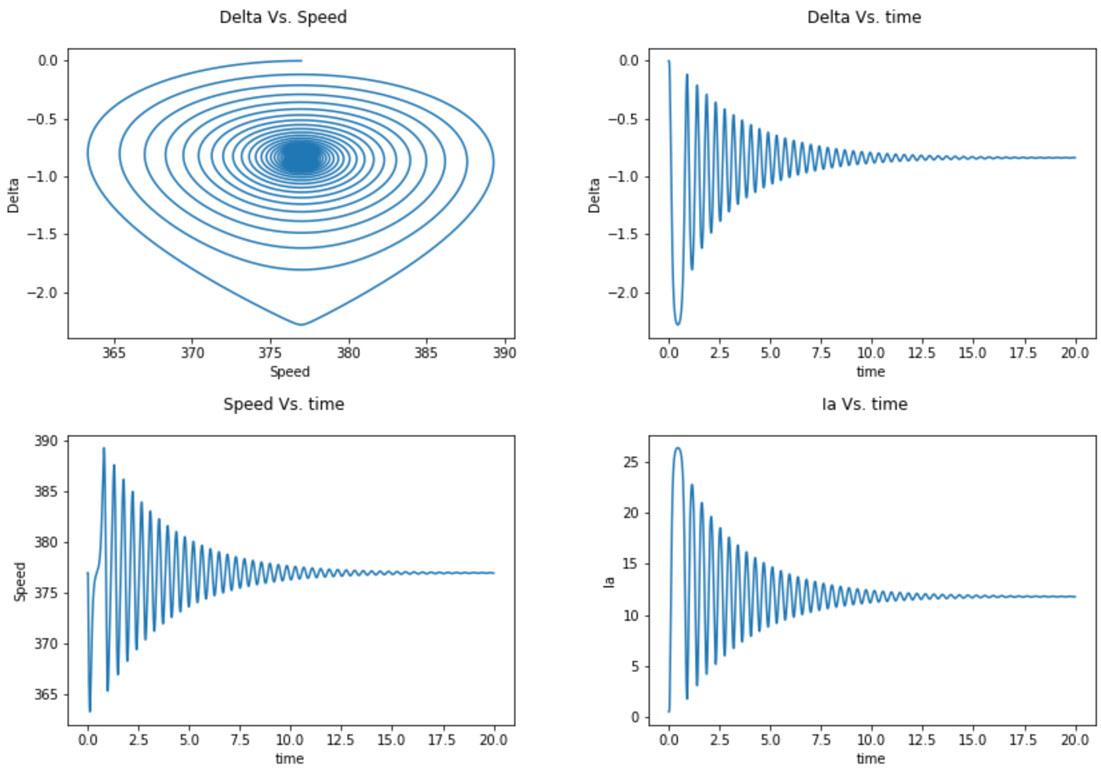


Adding a sudden load of 4500W gives us the following results:

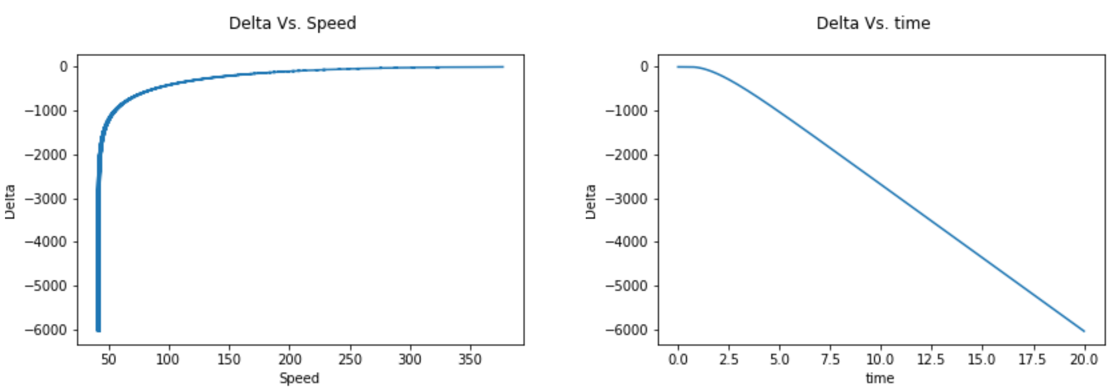


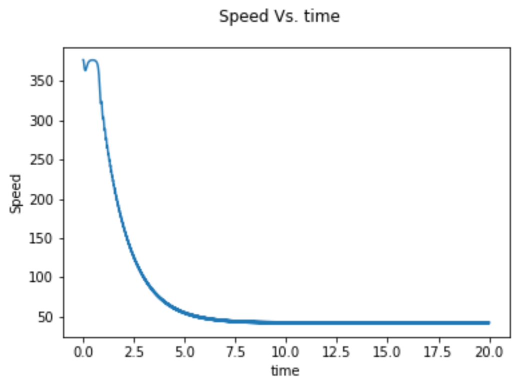
Using the same simulation, the maximum sudden load I found to be 4209W. We can also use the equal area criteria to see what the max loading will be and compare the two readings.

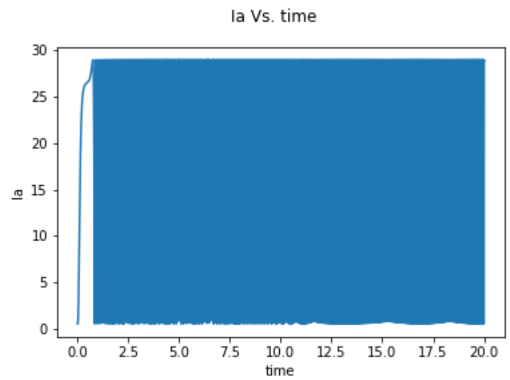
At 4209W there are the curves:



At 4210W the system becomes unstable and these are the curves:







To use the Equal Area criteria we solve the following integral:

$$\int_0^{\sin^{-1}(-0.00017656\,x)} (x + \sin(-t) \times (-5663.74))\,dt = \int_{\sin^{-1}(-0.00017656\,x)}^{\pi - \sin^{-1}(-0.00017656\,x)} (5663.74\sin(-t) - x)\,dt$$

This equation checks for what sudden load will the two areas will be exactly equal in the equal area criteria, at their max value.

Solving the equation, the max load comes to 4104W which is relatively close to what we predicted from our simulation.

3 CONCLUSION

This lab gave us a good look into synchronous machines simulations and how to use the Euler method to calculate sudden loading effects. This lab was different to the others as it was all code, but very helpful as it helped connect our learning in a different way. Overall, I enjoyed the lab as I had never done something like this before.

4 REFERENCES

[1] P.W. Sauer, P.T. Krein, P.L. Chapman, *ECE 431 Electric Machinery Course Guide and Laboratory Information*, University of Illinois at Urbana-Champaign, 2005.

5 APPENDIX

Code:

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

def xfrange(start, stop, step):
    i = 0
    while start + i * step < stop:
        yield start + i * step
        i += 1

ws = 120*np.pi
wi = 120*np.pi
If = 1.2
Lasf = .4
P = 6
xs = 9
Vt = 230/np.sqrt(3)
D = .3
J = .4
Ef = ws*Lasf*If/np.sqrt(2)
deltaT = 0
Te = (3*(Vt)*Ef/((2/P)*ws*xs))*np.sin(-deltaT)
Td = -D*(2/P)*(wi-ws)
dDelta = wi - ws
Pm = 0
Tm = -Pm/((2/P)*(ws))
dwi = (1/J)*(6/2)*(Te+Td+Tm)
wiarr = np.array([])
deltaTarr = np.array([])
Iaarr = np.array([])
t = np.arange(0,20.0005,.0005)

for i in xfrange(0, 20.0005, .0005):
    if(i == .01):
        Pm = 4210
    wi = wi + dwi*.0005
    deltaT = deltaT + dDelta*.0005
    wiarr = np.append(wiarr,wi)
    deltaTarr = np.append(deltaTarr,deltaT)
    Ia = abs((Vt - (Ef*np.cos(-deltaT)+Ef*1j*np.sin(-deltaT)))/xs)
    Iaarr = np.append(Iaarr, Ia)
    Te = (3*(Vt)*Ef/((2/P)*ws*xs))*np.sin(-deltaT)
    Td = -D*(2/P)*(wi-ws)
    Tm = -Pm/((2/P)*(ws))
    dDelta = wi - ws
    dwi = (1/J)*(P/2)*(Te+Td+Tm)
```