Predictive Maintainence Assignment(B19ME039)

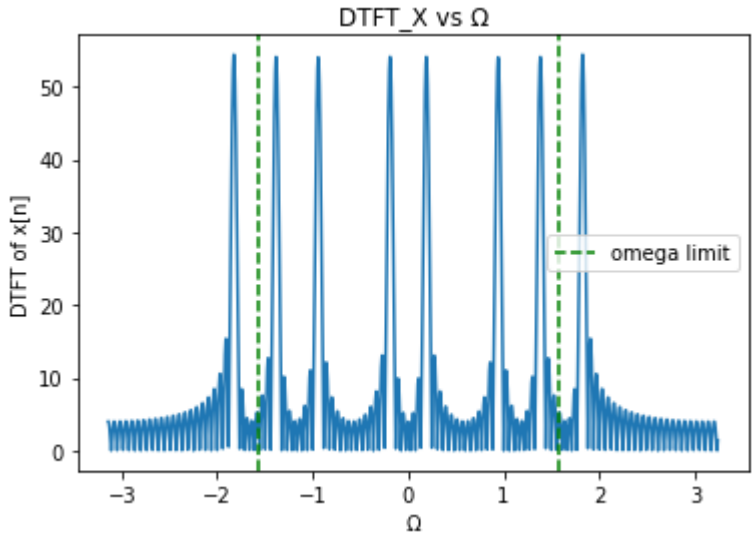
We are sampling x(t) at a frequency of 100Hz, i.e., taking one sample every 0.01 second or 100 samples per second. We know that all frequency components in the captured signal from a properly functioning device should be <=25Hz. We know that **Ω** = **ωT**. Here T = 1/f = 1/100 = 0.01. **ω** = 2πf, now since we need f<=25Hz for a properly functioning device, **ω** <= 2π x 25 = 50π. So ω <= 50π. Now, since Ω = ωT, so if we put ω <= 50π condition in this equation, Ω <= 50π x 0.01 = π/2 = 1.57. So for a healthy device, the magnitude of DTFT of x[n] should be zero everywhere except in the interval -π/2 <= Ω <= π/2 for the frequency components to be <=25Hz.

We will first calculate DTFT of x[n] using X(e^jΩ) = Σx[n]e^(-jΩn) and then plot X(e^jΩ) vs Ω and check if magnitude of X(e^jΩ) is non-zero in Ω intervals other than [-π/2, π/2] too. If it is non-zero other than [-π/2, π/2] interval, then device is not functioning properly.

Numpy code for the assignment

```
import pandas as pd
import numpy as np

# importing x[n] values and storing them in variable x
x = pd.read_csv("/content/signal_assignment1 - Sheet1.csv")

# creating a list of n (the values of n in x[n]) from 0 to 100
n = np.arange(0, 101).reshape(101,1)


# creating Ω values from [-π, π]
omega = np.arange((-1)*np.pi, np.pi+0.1, 0.001)

# assigning a variable t the values of -jΩn in the RHS of the
# expression X(e^jΩ) = Σx[n]e^(-jΩn).
t = (1j)*(-1)*n*omega

# finding the e^(-jΩn) values in the RHS of X(e^jΩ) = Σx[n]e^(-jΩn).
e = np.exp(t)

# taking transpose of the matrix 'e' formed above
e = e.T

# finally calculating the DTFT of x[n] with all the omega values.
DTFT_X = np.dot(e,x)
```

```
# plotting the graph
import matplotlib.pyplot as plt

# imposing the limit of Ω for a properly functioning device as π/2 as calculalted
omega_limit = 1.57
plt.plot(omega, abs(DTFT_X))
plt.axvline(omega_limit, color='green', linestyle='dashed', label='omega limit')
plt.axvline(-omega_limit, color='green', linestyle='dashed')
plt.xlabel('Ω')
plt.ylabel('DTFT of x[n]')
plt.title('DTFT_X vs Ω')
plt.legend()
plt.show()
```



Since X(e^jΩ) also has non-zero magnitudes in intervals of Ω other than [-π/2, π/2] too, therefore we are recieving signals having some frequency components > 25Hz, so the device is not functioning properly and requires maintainence.