

BLM6105-2 Uzaktan Algılama Dersi Ödev 2

24501100 – Aleyna Nil Uzunoğlu

Giriş

Bu çalışmanın amacı, belirli bir Ayırık Zamanlı Fourier Dönüşümü (DTFT) ile tanımlanan bir sistemin frekans yanıtını incelemektir.

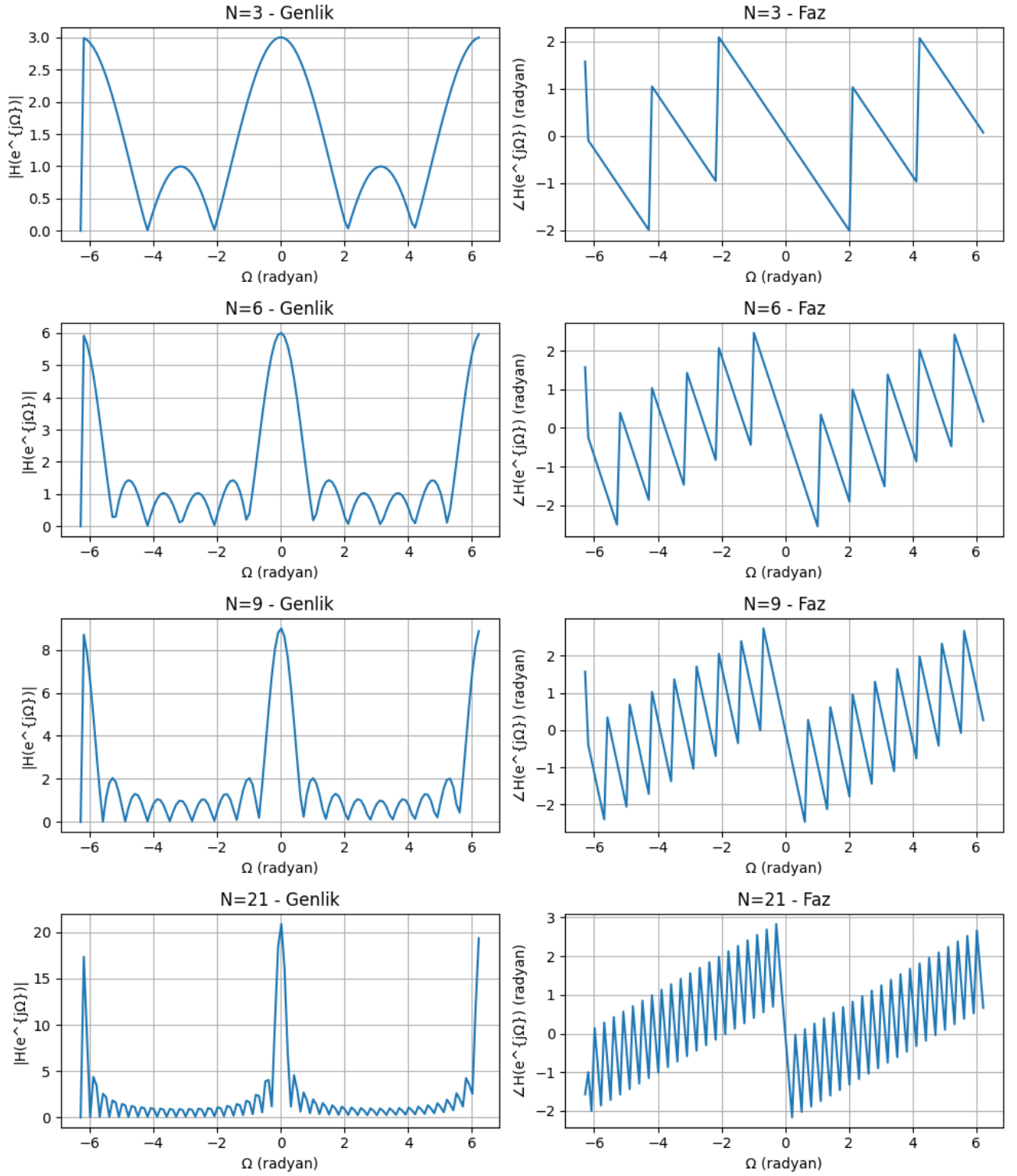
$$H(e^{j\Omega}) = \sum_{n=0}^{N-1} (e^{-j\Omega})^n = \frac{e^{-j\Omega N} - e^{-j\Omega 0}}{e^{-j\Omega} - 1} = \frac{e^{-j\Omega N} - 1}{e^{-j\Omega} - 1} = \frac{1 - e^{-j\Omega N}}{1 - e^{-j\Omega}}$$

Yöntem

Genlik ve faz spektrumlarının hesaplanması ve görselleştirilmesi için Python programlama dili ve NumPy ile Matplotlib kütüphaneleri kullanılmıştır.

- Frekans Vektörü:** Verilen aralıklarda (-2π ile 2π aralığında) 0.1 radyanlık adımlarla ayırık frekans değerleri (Ω) içeren bir NumPy dizisi oluşturulmuştur
 - $(\text{np.arange}(-2*\text{np.pi}, 2*\text{np.pi}, 0.1))$ satırında bu durum görülebilir.
- DTFT Hesaplanması:** Belirlenen her bir N değeri ($3, 6, 9, 21$) için, yukarıda verilen kapalı formül kullanılarak $H(e^{j\Omega})$ 'nin kompleks değerleri hesaplanmıştır.
- Kararlılık:** Verilen formülün paydasında bulunan terim, Ω 'nın 2π 'nin katları olduğu durumlarda sıfır olacağından dolayı, doğrudan hesaplamada sıfıra bölme hatasına yol açabilir. Bu sorunu aşmak için, paydanın mutlak değeri çok küçük (belirlenen bir epsilon $1e-12$ değerinden küçük) olduğunda, payda yerine bu küçük epsilon değeri kullanılarak kararlılık sağlanmıştır.
- Genlik ve Faz Hesabı:** Hesaplanan kompleks $H(e^{j\Omega})$ değerlerinin her biri için:
 - Genlik spektrumu $\text{np.abs}(H)$ fonksiyonu ile hesaplanmıştır.
 - Faz spektrumu $\text{np.angle}(H)$ fonksiyonu ile (radyan cinsinden) hesaplanmıştır.
- Görselleştirme:** Her bir N değeri için genlik ve faz spektrumları ayrı Matplotlib alt grafikleri (subplots) kullanılarak çizdirilmiştir. Grafikler, frekansa (Ω) karşı genlik ve fazı göstermektedir. Grafikler ilerleyen sayfalarda paylaşılmıştır.

DTFT $H(e^{j\Omega})$ için Genlik ve Faz Spektrumu



Grafik 1: $N = [3, 6, 9, 21]$ durumları için genlik ve faz spektrumları

Ek: main.py

```
import numpy as np

import matplotlib.pyplot as plt

Omega = np.arange(-2*np.pi, 2*np.pi, 0.1)

N_values = [3, 6, 9, 21]

fig, axs = plt.subplots(len(N_values), 2, figsize=(10, 12))

fig.suptitle('DTFT H(e^{j\Omega}) için Genlik ve Faz Spektrumu')

for idx, N in enumerate(N_values):

    #  $H(e^{j\Omega}) = (1 - e^{-j\Omega N}) / (1 - e^{-j\Omega})$ 

    numerator = 1 - np.exp(-1j * Omega * N)
    denominator = 1 - np.exp(-1j * Omega)

    epsilon = 1e-12
    denominator_safe = np.where(np.abs(denominator) < epsilon, epsilon, denominator)
    H = numerator / denominator_safe
    magnitude = np.abs(H)
    phase = np.angle(H)

    axs[idx, 0].plot(Omega, magnitude, label=f'N={N}')
    axs[idx, 0].set_title(f'N={N} - Genlik')
    axs[idx, 0].set_xlabel('\Omega (radyan)')
    axs[idx, 0].set_ylabel('|H(e^{j\Omega})|')
    axs[idx, 0].grid(True)
```

```
axs[idx, 1].plot(Omega, phase, label=f'N={N}')  
axs[idx, 1].set_title(f'N={N} - Faz')  
axs[idx, 1].set_xlabel('Ω (radyan)')  
axs[idx, 1].set_ylabel('∠H(e^{jΩ}) (radyan)')  
axs[idx, 1].grid(True)
```

```
plt.tight_layout()  
plt.savefig('DTFT_H_spectra.png')  
plt.show()
```