Relatório – ST7: T7: Projeto de Filtro FIR Passa-Baixas com Janela de Hamming

1. Enunciado

Today's Short Test (ST7): Design an FIR filter with the following specifications:

- $0.99 \leqslant |H(e^{j\omega})| \leqslant 1.01$, in the range $0 \leqslant \omega \leqslant 0.15\pi$
- $|H(e^{j\omega})| \leqslant 0.06$, in the range $0.45\pi \leqslant \omega \leqslant \pi$

Then, normalize the windowed filter and write down the difference equation to implement the filter you have just designed in a computer-based application.

53 / 109

2. Código Python

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

freq_pass_lim = 0.15 * np.pi
freq_stop_lim = 0.45 * np.pi
transition_width_rad = freq_stop_lim - freq_pass_lim
num_freq_points = 1024
angular_frequencies = np.linspace(0, np.pi, num_freq_points)


def criar_filtro_janelado_hanning(ordem_filtro):
    indices_tempo = np.arange(ordem_filtro + 1)
    ponto_central = ordem_filtro / 2
    freq_corte_ideal = (freq_pass_lim + freq_stop_lim) / 2
    sinc_ideal = np.sinc((freq_corte_ideal / np.pi) * (indices_tempo -
ponto_central))
    janela_hanning = 0.5 - 0.5 * np.cos(2 * np.pi * indices_tempo /
ordem_filtro)
    coeficientes_janelados = sinc_ideal * janela_hanning
    coeficientes_normalizados = coeficientes_janelados /
np.sum(coeficientes_janelados)
    return coeficientes_normalizados, indices_tempo


def verificar_especificacoes_filtro(coefs_filtro, indices_n_filtro):
    resposta_freq_H = np.array(
        [np.sum(coefs_filtro * np.exp(-1j * omega_i *
indices_n_filtro)) for omega_i in angular_frequencies])
    magnitude_H = np.abs(resposta_freq_H)
    ganho_min_passabanda = magnitude_H[angular_frequencies <=
freq_pass_lim].min()
    ganho_max_passabanda = magnitude_H[angular_frequencies <=
```

```
freq_pass_lim].max()
    atenuacao_max_rejeicao = magnitude_H[angular_frequencies >=
freq_stop_lim].max()
    return ganho_min_passabanda, atenuacao_max_rejeicao,
ganho_max_passabanda


ordem_atual_M = int(np.ceil(3.3 * np.pi / transition_width_rad))
if ordem_atual_M % 2:
    ordem_atual_M += 1

print("Iniciando busca pela ordem M do filtro (Janela de Hanning
aplicada):")
while True:
    coeficientes_finais, indices_n =
criar_filtro_janelado_hanning(ordem_atual_M)
    min_pb, max_sb, max_pb =
verificar_especificacoes_filtro(coeficientes_finais, indices_n)

    atende_min_pb = min_pb >= 0.99
    atende_max_pb = max_pb <= 1.01
    atende_max_sb = max_sb <= 0.06
    todas_atendidas = atende_min_pb and atende_max_pb and atende_max_sb

    print(f"Com M={ordem_atual_M}: GanhoMin PB={min_pb:.4f} ({'ok' if
atende_min_pb else 'X'}), "
          f"GanhoMax PB={max_pb:.4f} ({'ok' if atende_max_pb else
'X'}), "
          f"AtenMax SB={max_sb:.4f} ({'ok' if atende_max_sb else 'X'})
"
          f">> {'ATENDE TODAS' if todas_atendidas else 'NÃO ATENDE'}")

    if todas_atendidas:
        print(f"\nMenor Ordem M (par) que satisfaz as condições:
{ordem_atual_M}")
        print(f"  Ganho Mínimo na Banda de Passagem: {min_pb:.4f}")
        print(f"  Ganho Máximo na Banda de Passagem: {max_pb:.4f}")
        print(f"  Atenuação Máxima na Banda de Rejeição: {max_sb:.4f}")
        break
    ordem_atual_M += 2

print("\nCoeficientes finais do filtro h[n]:")
print(np.round(coeficientes_finais, 6))

resposta_H_final = np.array(
    [np.sum(coeficientes_finais * np.exp(-1j * omega_i * indices_n))
for omega_i in angular_frequencies])
frequencias_plot = angular_frequencies / np.pi

plt.figure(figsize=(10, 9))

plt.subplot(3, 1, 1)
plt.stem(indices_n, coeficientes_finais, basefmt=" ")
plt.title(f"Coeficientes do Filtro FIR (M={ordem_atual_M}, Janela de
Hanning)")
plt.xlabel("Índice n")
plt.ylabel("h[n]")
```

```
plt.grid(True)

plt.subplot(3, 1, 2)
plt.plot(frequencias_plot, np.abs(resposta_H_final),
color='dodgerblue')
plt.axvline(freq_pass_lim / np.pi, color='green', linestyle='--',
label='Borda Passa-Faixa')
plt.axvline(freq_stop_lim / np.pi, color='red', linestyle='--',
label='Borda Rejeita-Faixa')
plt.hlines([0.99, 1.01], 0, freq_pass_lim / np.pi, color='lightgreen',
linestyle=':')
plt.hlines([0.06], freq_stop_lim / np.pi, 1, color='lightcoral',
linestyle=':')
plt.title("Resposta em Magnitude |H(e^{jω})|")
plt.xlabel("Frequência Normalizada (×π rad/amostra)")
plt.ylabel("Magnitude |H|")
plt.ylim(-0.05, 1.15)
plt.legend()
plt.grid(True)

plt.subplot(3, 1, 3)
plt.plot(frequencias_plot, 20 * np.log10(np.abs(resposta_H_final) + 1e-
9), color='darkorange')
plt.axvline(freq_pass_lim / np.pi, color='green', linestyle='--',
label='Borda Passa-Faixa')
plt.axvline(freq_stop_lim / np.pi, color='red', linestyle='--',
label='Borda Rejeita-Faixa')
plt.hlines([20 * np.log10(0.99), 20 * np.log10(1.01)], 0, freq_pass_lim
/ np.pi, color='lightgreen', linestyle=':',
           label='Tolerância Passa-Faixa')
plt.hlines([20 * np.log10(0.06)], freq_stop_lim / np.pi, 1,
color='lightcoral', linestyle=':',
           label='Tolerância Rejeita-Faixa')
plt.title("Resposta em Magnitude em dB |H(e^{jω})|")
plt.xlabel("Frequência Normalizada (×π rad/amostra)")
plt.ylabel("Magnitude |H| (dB)")
plt.ylim(-80, 5)
plt.legend()
plt.grid(True)

plt.tight_layout()
plt.show()
```

3. Resultato

```
C:\Users\Andrei\PycharmProjects\pythonProject1\venv1\Scripts\python.exe C:\Users\Andrei\PycharmProjects\pythonProject1\ex7.py
Iniciando busca pela ordem M do filtro (Janela de Hanning aplicada):
Com M=12: GanhoMin PB=0.8734 (X), GanhoMax PB=1.0000 (ok), AtenMax SB=0.1173 (X)  >> NÃO ATENDE
Com M=14: GanhoMin PB=0.9079 (X), GanhoMax PB=1.0000 (ok), AtenMax SB=0.0782 (X)  >> NÃO ATENDE
Com M=16: GanhoMin PB=0.9461 (X), GanhoMax PB=1.0002 (ok), AtenMax SB=0.0488 (ok)  >> NÃO ATENDE
Com M=18: GanhoMin PB=0.9758 (X), GanhoMax PB=1.0072 (ok), AtenMax SB=0.0262 (ok)  >> NÃO ATENDE
Com M=20: GanhoMin PB=0.9930 (ok), GanhoMax PB=1.0109 (X), AtenMax SB=0.0106 (ok)  >> NÃO ATENDE
Com M=22: GanhoMin PB=1.0000 (ok), GanhoMax PB=1.0086 (ok), AtenMax SB=0.0064 (ok)  >> ATENDE TODAS

Menor Ordem M (par) que satisfaz as condições: 22
  Ganho Mínimo na Banda de Passagem: 1.0000
  Ganho Máximo na Banda de Passagem: 1.0086
  Atenuação Máxima na Banda de Rejeição: 0.0064

Coeficientes finais do filtro h[n]:
[-0.        0.        0.002276  0.006545  0.004117 -0.013403 -0.036444
 -0.033179  0.027192  0.139671  0.252881  0.300688  0.252881  0.139671
  0.027192 -0.033179 -0.036444 -0.013403  0.004117  0.006545  0.002276
  0.       -0.       ]
```

4. Gráfico da Resposta em Frequência