REPORT

Zajęcia: Analog and digital electronic circuits

Teacher: prof. dr hab. Vasyl Martsenyuk

Lab 3 and 4

17.02.2025

Topic: Windowing

Variant 15

Tobiasz Wojnar
Informatyka II stopień,
niestacjonarne,
1 semestr,
Gr. B

1. Problem statement:

Creation of tree sine signals with specified f_l, f_2 and f_3 frequencies. Maximum amplitude $|x[k]|_{max}$, using a sampling frequency f_s over the range

$0 <= k < N$

Two visualizations:

- DFT (Discrete Fourier Transform)
- DTFT (Discrete-Time Fourier fransform)

2. Input data:

- $f_l = 500$
- $f_2 = 500.25$
- $f_3 = 499.75$
- $|x[k]|_{max} = 4$
- $f_s = 800$
- $N = 2000$

3. Commands used (or GUI):

a) source code

```
# importing libraries
import numpy as np
import matplotlib.pyplot as plt
from numpy.fft import fft, ifft, fftshift
from scipy.signal.windows import hann, flattop
```

```
f1 = 500     # Hz
f2 = 500.25 # Hz
f3 = 499.75 # Hz
fs = 800 # Hz
N = 1600
```

```
k  = np.arange (N)
max_amplitude = 4

x1 = np.sin(2*np.pi * f1 / fs * k )*max_amplitude
x2 = np.sin(2*np.pi * f2 / fs * k )*max_amplitude
x3 = np.sin(2*np.pi * f3 / fs * k )*max_amplitude
```

```
# Generate window functions
w_rect = np.ones(N)
w_hann = hann(N, sym=False)
w_flattop = flattop(N, sym=False)

plt.plot(w_rect, 'C3o-', ms=3, label='Rect')
plt.plot(w_hann, 'C1o-', ms=3, label='Hann')
plt.plot(w_flattop, 'C7o-', ms=3, label='Flattop')
plt.xlabel('$k$')
plt.ylabel('window $w[k]$')
plt.xlim(0,N)
plt.legend()
plt.grid()
```

```
def fft2db(X):
    N = X.size
    Xtmp = 2/N *X
    Xtmp [ 0 ] *= 1/2
    if N % 2 == 0 :
        Xtmp [N//2] = Xtmp [N//2] /2
    return 20*np.log10(np.abs(Xtmp))
```

```
df = fs/N
f = np.arange(N)*df
```

```
def plotComparison(x_1, x_2, x_3, window_name,plotNr):
    plt.subplot( 3, 1, plotNr)

    plt.plot(f,  fft2db(x_1),  'C2o-',  ms=3,  label=f'f1  case
{window_name}')
    plt.plot(f,  fft2db(x_2),  'C3o-',  ms=3,  label=f'f2  case
{window_name}')
    plt.plot(f,  fft2db(x_3),  'C5o-',  ms=3,  label=f'f3  case
{window_name}')
```

```
    plt.xlim(275,275+50)
    plt.ylim(-50,20)

    plt.xticks(np.arange(275,275+50,5))
    plt.yticks(np.arange(-60,20,10))

    plt.xlabel('f/Hz')
    plt.ylabel('A / dB')

    plt.legend()
    plt.grid()
```

```
plt.figure(figsize=(16/1.5, 10/1.5))
plotComparison(x1_w_rect,x2_w_rect,x3_w_rect,'rect',1)
plotComparison(x1_w_hann,x2_w_hann,x3_w_hann,'hann',2)
plotComparison(x1_w_flattop,x2_w_flattop,x3_w_flattop,'flattop
',3)
```

```
def winDTFTdB(w):
    N  = w.size
    Nz = 100*N
    W  = np.zeros(Nz)

    W[0:N] = w
    W  = np.abs( fftshift (fft(W)) )
    W /= np.max(W) # normalize

    W = 20*np.log10(W)
    Omega = 2*np.pi/Nz*np. arange(Nz) - np.pi
    return Omega, W
```

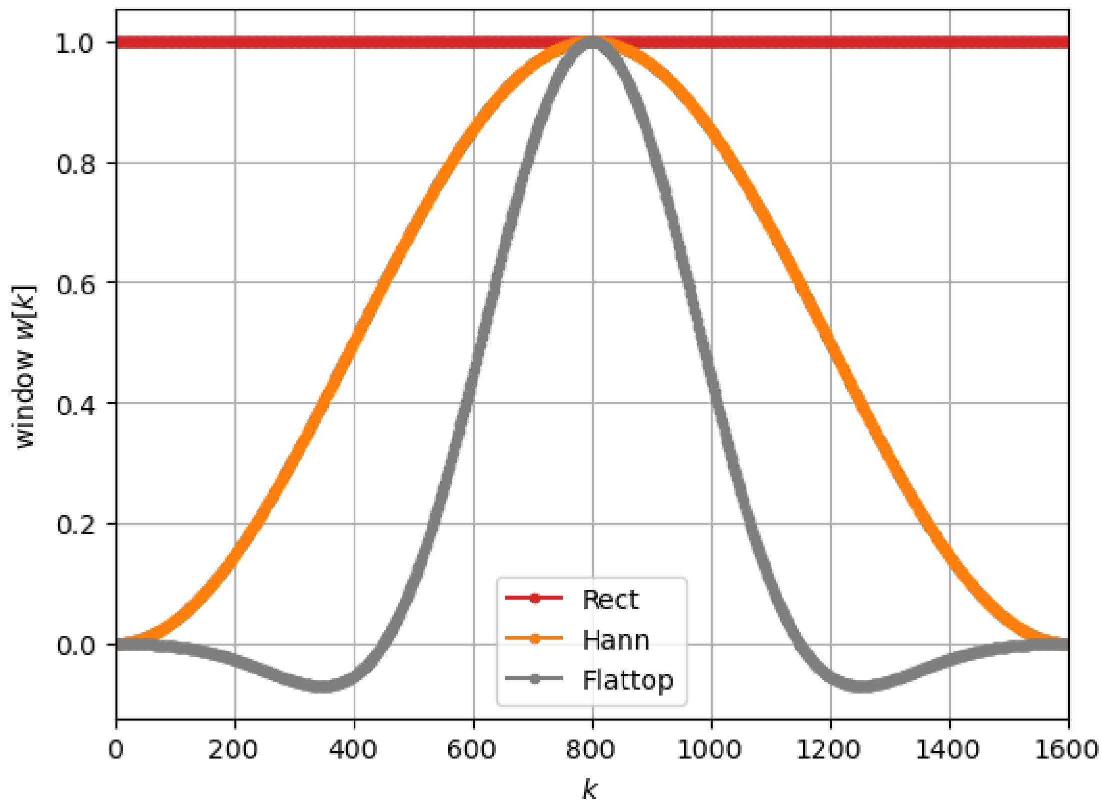
```
plt.plot([-np.pi,+np.pi], [-3.01,-3.01],'gray')
plt.plot([-np.pi,+np.pi], [-13.3,-13.3],'gray')
plt.plot([-np.pi,+np.pi], [-31.5,-31.5],'gray')
plt.plot([-np.pi,+np.pi], [-93.6,-93.6],'gray')

Omega, W = winDTFTdB(w_rect)
plt.plot(Omega, W, label='rect')
Omega, W = winDTFTdB(w_hann)
plt.plot(Omega, W, label='hann')
Omega, W = winDTFTdB(w_flattop)
```

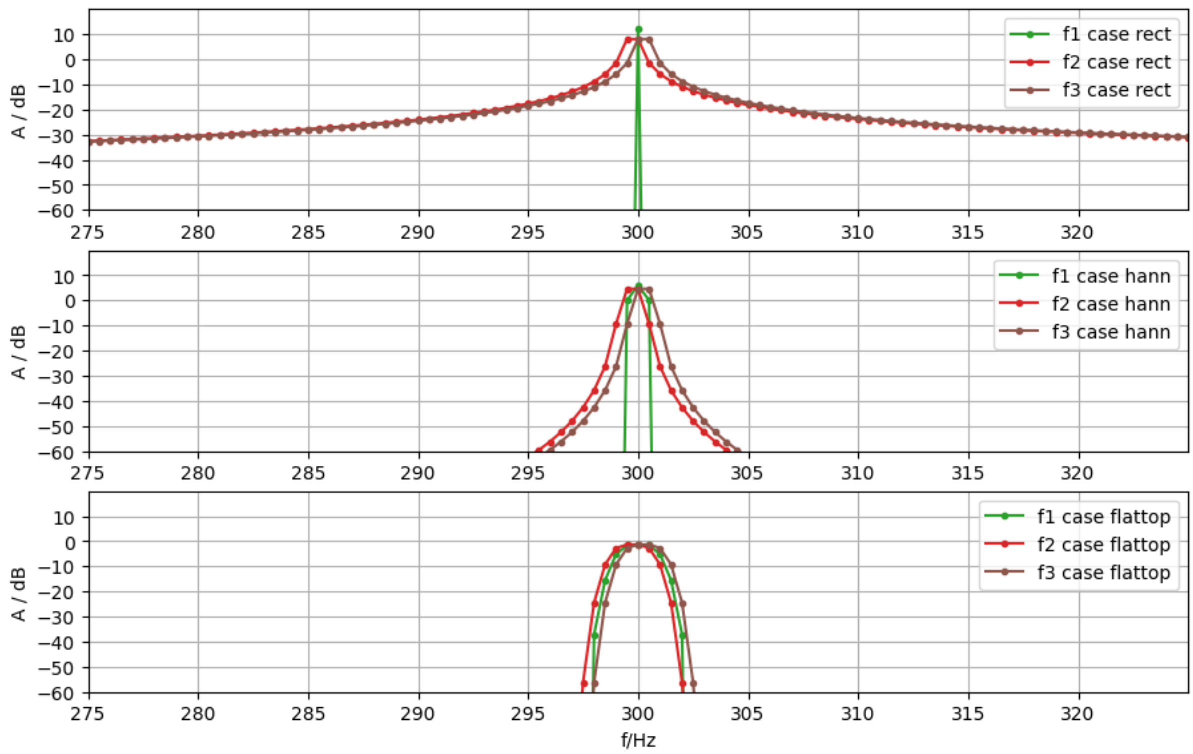
```
plt.plot(Omega, W, label='flattop')

plt.xlim(-np.pi/100,np.pi/100) # zoom into mainlobe
plt.ylim(-120,10)

plt.xlabel(r'$\Omega$')
plt.ylabel(r'|W($\Omega)| / dB$')

plt.legend()
plt.grid(True)
```

b) Link to remote repozytorium

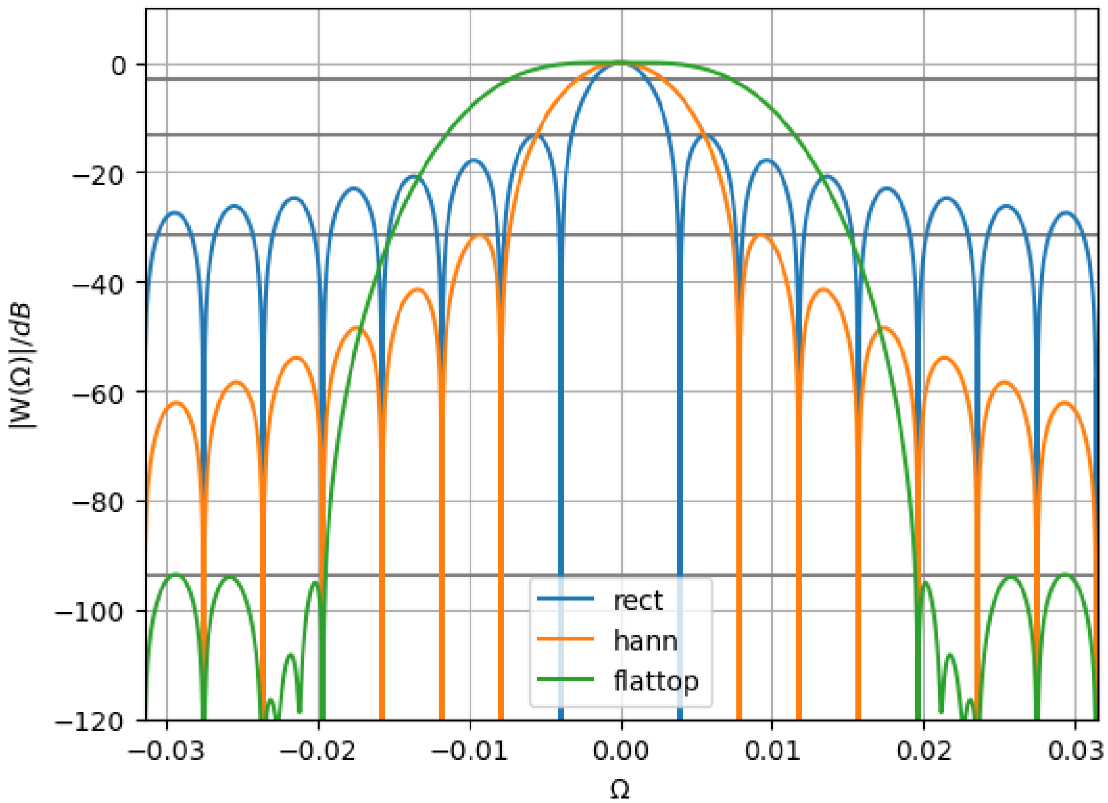
https://github.com/TobiaszWojnar/DSP

4. Outcomes:





f1 case rect
f2 case rect
f3 case rect
f1 case hann
f2 case hann
f3 case hann
f1 case flattop
f2 case flattop
f3 case flattop
A / dB
f/Hz



rect
hann
flattop
|W(Ω)|/dB
Ω

5. Conclusions

The selection of a window function for signal spectrum analysis is crucial and depends on the desired information. Each window has trade-offs affecting measurement quality. While the rectangular window provides the best frequency resolution (though suffers from sidelobes), the Hanning window offers a good balance between amplitude accuracy and spectral leakage reduction. Finally, the Flattop window prioritizes amplitude accuracy, even at the cost of reduced frequency resolution.

Rectangular Window

The rectangular window's advantage lies in its lack of signal modification, resulting in superior frequency resolution, essential for pinpointing specific frequencies. However, this benefit is offset by significant sidelobes in the resulting spectrum, potentially obscuring accurate signal analysis due to interference.

Hanning Window

The Hanning window offers a compromise between frequency resolution and sidelobe suppression. Its smoothing effect at signal boundaries reduces spectral leakage and improves amplitude accuracy. Although it sacrifices some frequency resolution compared to the rectangular window, the Hanning window's overall improvement in spectral quality makes it a popular choice in many applications.

Flattop Window

The Flattop window prioritizes amplitude accuracy, producing highly precise measurements due to its flat-topped spectral response. [1] This precision comes at the expense of a wider main lobe, reducing frequency resolution. However, for applications where amplitude accuracy is paramount, such as high-precision spectral measurements, the Flattop window is the optimal choice, provided the reduced frequency resolution is acceptable