

Analiza sygnału w dziedzinie czasu i częstotliwości

Ćwiczenie 5: Transformata falkowa

1. Wstęp

Celem ćwiczenia jest wprowadzenie w analizę czas-częstotliwość przy użyciu transformaty falkowej. Jako przykład sygnału niestacjonarnego wykorzystamy sygnał mowy oraz sygnał syntetyczny.

Użyte oprogramowanie: *Python ver. 3.9.7*

Użyte biblioteki: *numpy, scipy, matplotlib, ssqueezepy, math*

2. Kod źródłowy

```
import numpy as np
import matplotlib.pyplot as plt
import scipy.signal as sig
import scipy.io
import math
from ssqueezepy import cwt
from ssqueezepy.visuals import plot, imshow
from scipy.io import wavfile

plt.rcParams["figure.figsize"] = [15, 6]
plt.rcParams['font.size'] = '13'

f_d = [.038580777748, .126969125396, -.077161555496, -.607491641386,
       .745687558934, -.226584265197]
x = np.arange(1, len(f_d)+1)
f_g = -((-1)**x)*f_d

fig, (ax1, ax2) = plt.subplots(1, 2)
ax1.plot(x, f_d)
ax1.set(xlabel='czas [a.u.]', ylabel='amplituda [a.u.]', title='Filtr dolnoprzepustowy')
ax2.plot(x, f_g)
ax2.set(xlabel='czas [a.u.]', ylabel='amplituda [a.u.]', title='Filtr górnoprzepustowy')
```

```

plt.show()

N = 512
fs = 500
dt = 1/fs
t = np.arange(N)*dt
f1 = 20
f2 = 100

signal = sig.chirp(t, f1, t[-1], f2)

plt.plot(signal)
plt.xlabel('nr próbek')
plt.ylabel('amplituda [a.u.]')
plt.title('Sygnał świergotowy, liniowy 20-100 Hz')
plt.show()

Wx, scales = cwt(signal, 'morlet')
imshow(Wx, yticks=scales, abs=1,
        title="Skalogram sygnału świergotowego",
        ylabel="skala", xlabel="nr.próbki")

for i in range(int(math.log2(N))):

    splot_d = sig.convolve(signal, f_d, mode='same')
    splot_g = sig.convolve(signal, f_g, mode='same')

    x = np.arange(1, len(splot_d) + 1)

    fig, (ax1, ax2) = plt.subplots(1, 2)
    fig.suptitle(f'Wynik splotu filtrów z sygnałem świergotowym po {i+1}
iteracji')
    ax1.plot(x, splot_d)
    ax1.set(xlabel='nr próbek', ylabel='amplituda [a.u.]', title='Filtr
dolnoprzepustowy')
    ax2.plot(x, splot_g)
    ax2.set(xlabel='nr próbek', ylabel='amplituda [a.u.]', title='Filtr
górnoprzepustowy')
    plt.savefig(f'semestr2/ASwDCiC/lab5/shots/{i+1}.svg')
    plt.close()

    decym_splot_d = splot_d[:,2]
    decym_splot_g = splot_g[:,2]

    signal = decym_splot_d

wav_fname = 'semestr2/ASwDCiC/lab5/rabarbar8k.wav'
samplerate, signal = wavfile.read(wav_fname)

```

```

length = signal.shape[0] / samplerate
time = np.linspace(0., length, signal.shape[0])

plt.plot(time, signal)
plt.xlabel('czas [s]')
plt.ylabel('amplituda [a.u.]')
plt.title('Sygnał z pliku $\it{rabarbar8k.wav}$')
plt.show()

Wx, scales = cwt(signal, 'morlet')
imshow(Wx, yticks=scales, abs=1,
        title="Skalogram sygnału głosowego",
        ylabel="skala", xlabel="nr.próbki")

for i in range(int(math.log2(signal.shape[0]))):

    splot_d = sig.convolve(signal, f_d, mode='same')
    splot_g = sig.convolve(signal, f_g, mode='same')

    x = np.arange(1, len(splot_d) + 1)

    fig, (ax1, ax2) = plt.subplots(1, 2)
    fig.suptitle(f'Wynik splotu filtrów z sygnałem głosowym po {i+1} iteracji')
    ax1.plot(x, splot_d)
    ax1.set(xlabel='nr próbki', ylabel='amplituda [a.u.]', title='Filtr
dolnoprzepustowy')
    ax2.plot(x, splot_g)
    ax2.set(xlabel='nr próbki', ylabel='amplituda [a.u.]', title='Filtr
górnoprzepustowy')
    plt.savefig(f'semestr2/ASwDCiC/lab5/shots/{i+1}.svg')
    plt.close()

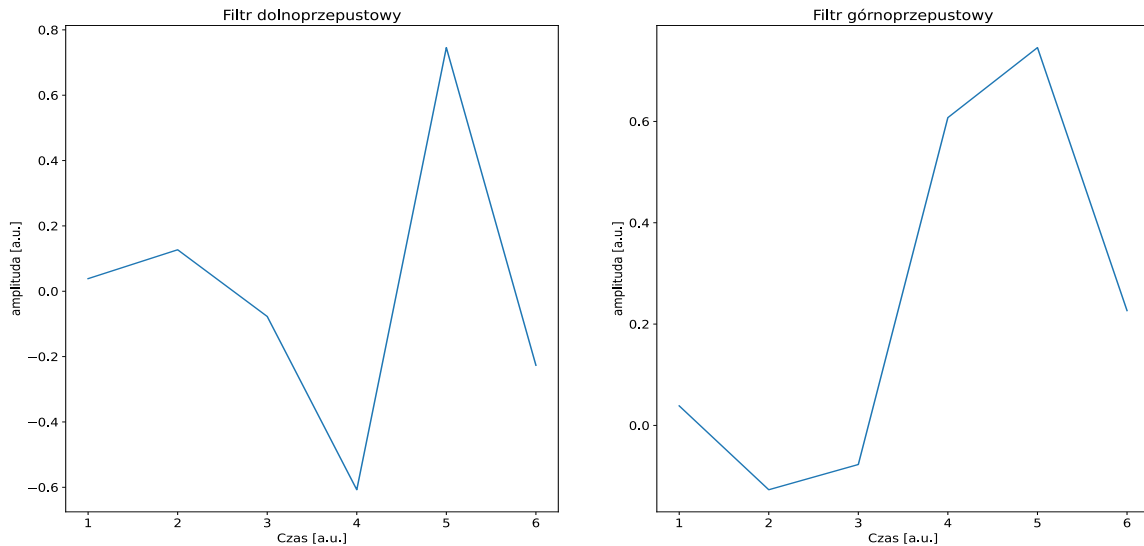
    decym_splot_d = splot_d[::2]
    decym_splot_g = splot_g[::2]

    signal = decym_splot_d

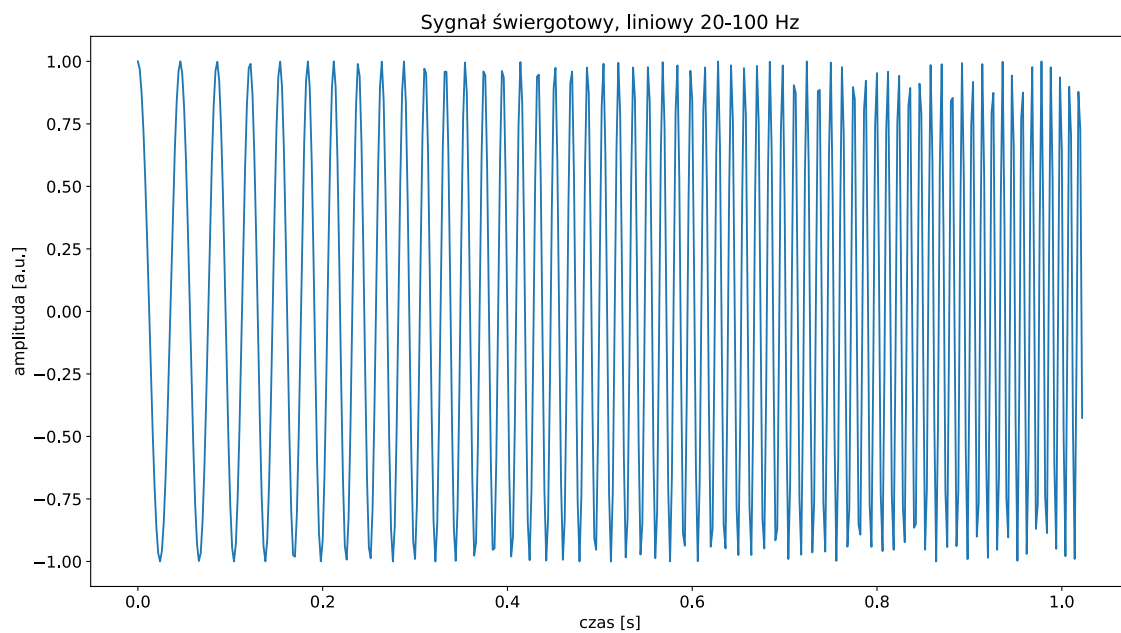
```

3. Wyniki

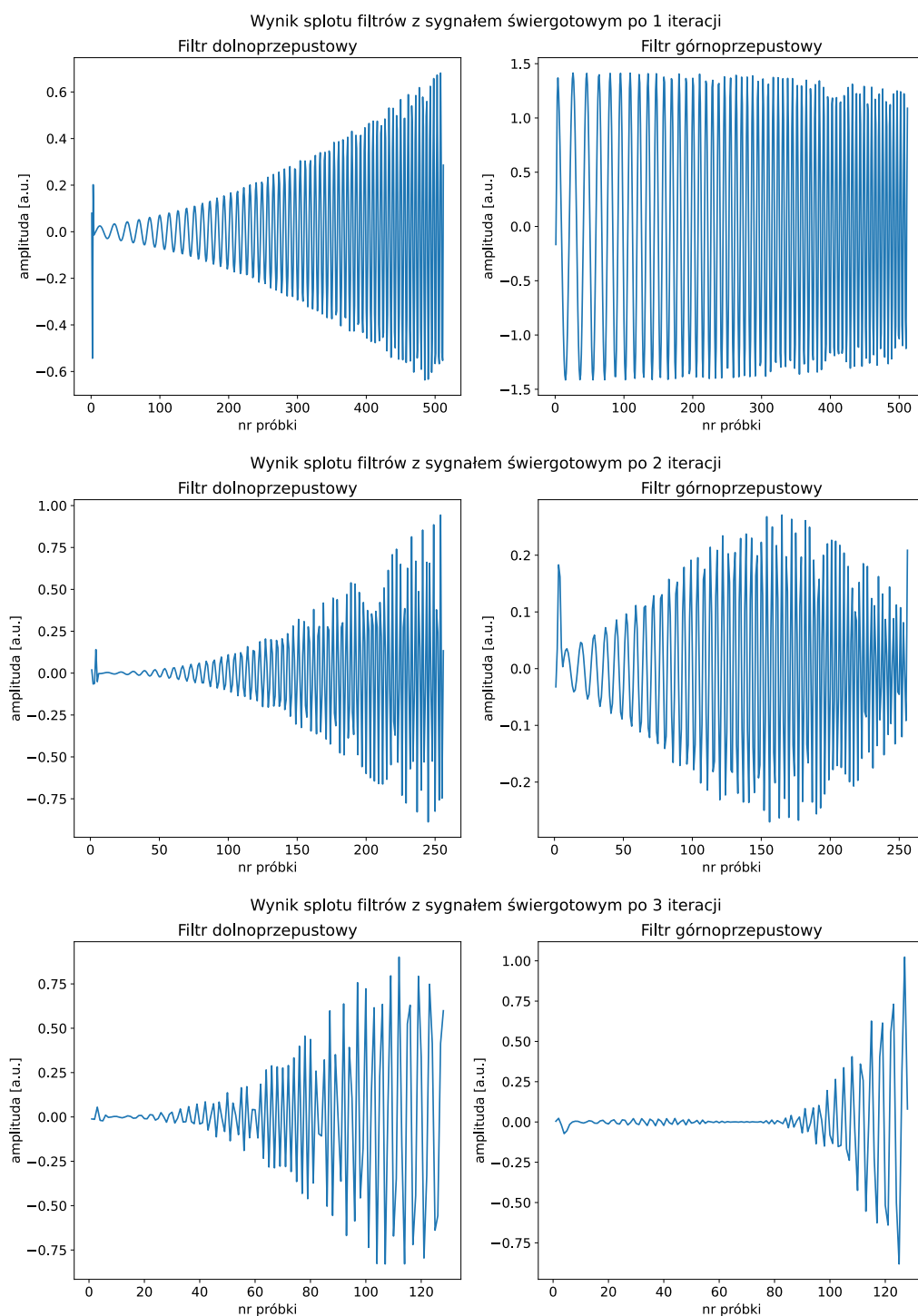
Wykres poniżej przedstawia falkę podstawową typu *Coiflet1* o współczynnikach podanych w instrukcji (lewa strona) oraz jej odbicie zwierciadlane (prawa strona).



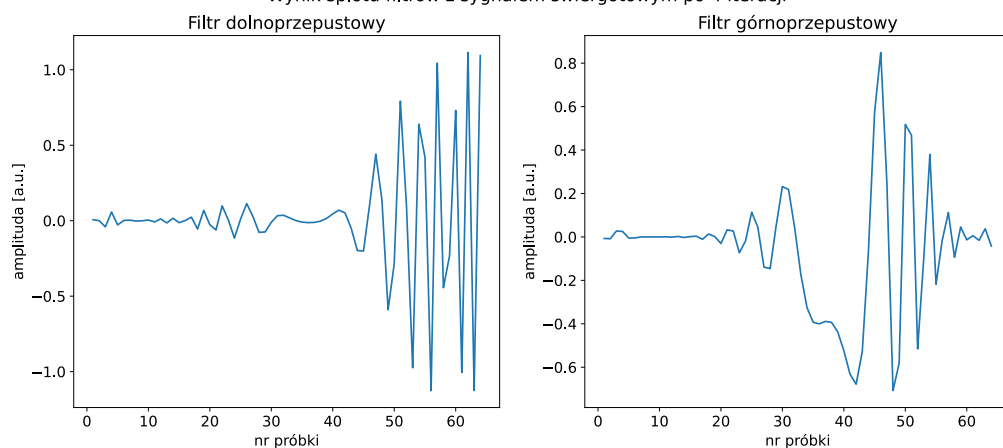
Pierwszą częścią ćwiczenia jest przeprowadzenie analizy falkowej na sygnale nietrywialnym, o długości 2^L . W tym celu wybrano sygnał świergotowy liniowy 20-100 Hz o długości 512, analizowany na laboratorium nr. 3. Sygnał ten jest przedstawiony na wykresie poniżej.



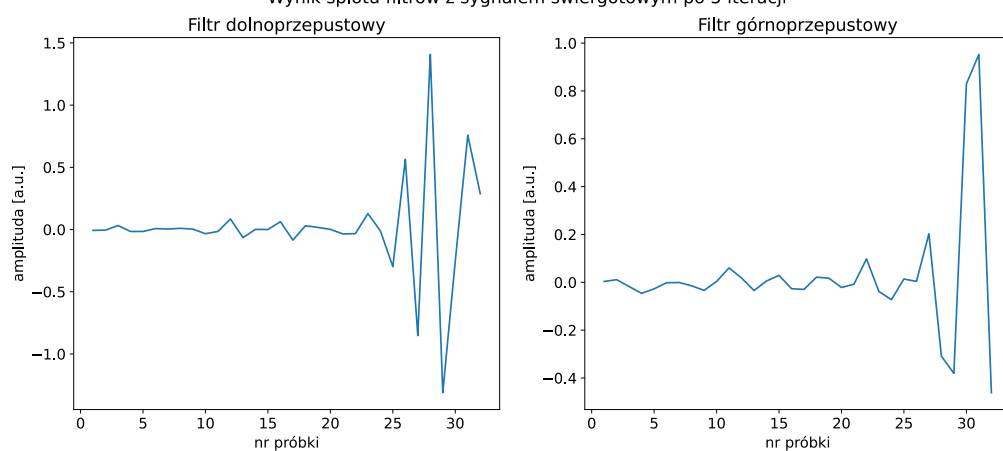
Wyniki splotu naszych filtrów z sygnałem świergotowym przedstawiają wykresy poniżej.



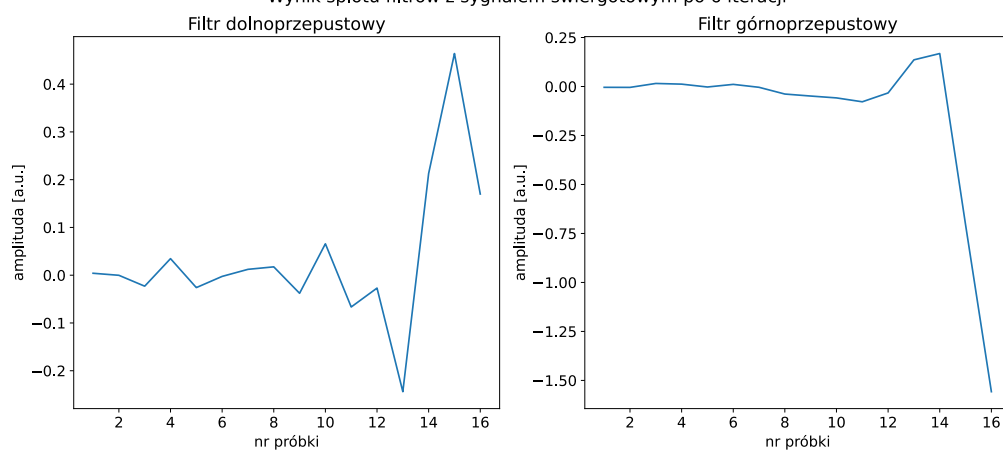
Wynik splotu filtrów z sygnałem świergotowym po 4 iteracji



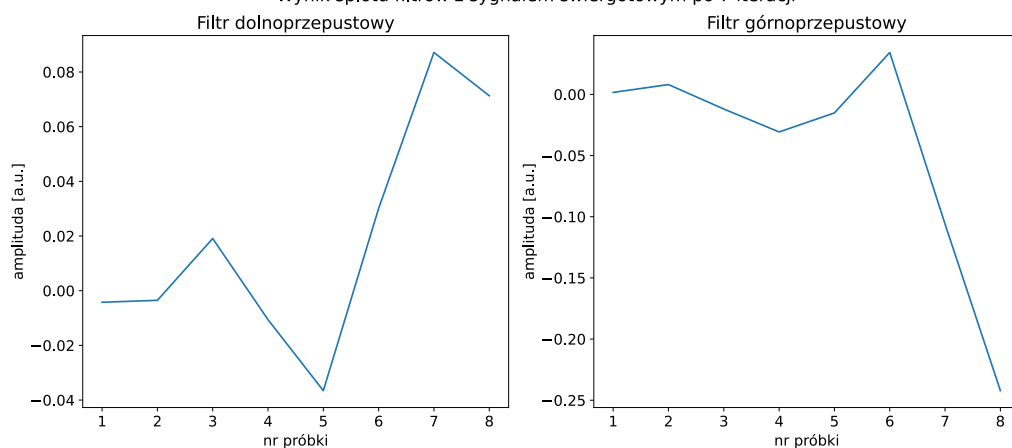
Wynik splotu filtrów z sygnałem świergotowym po 5 iteracji



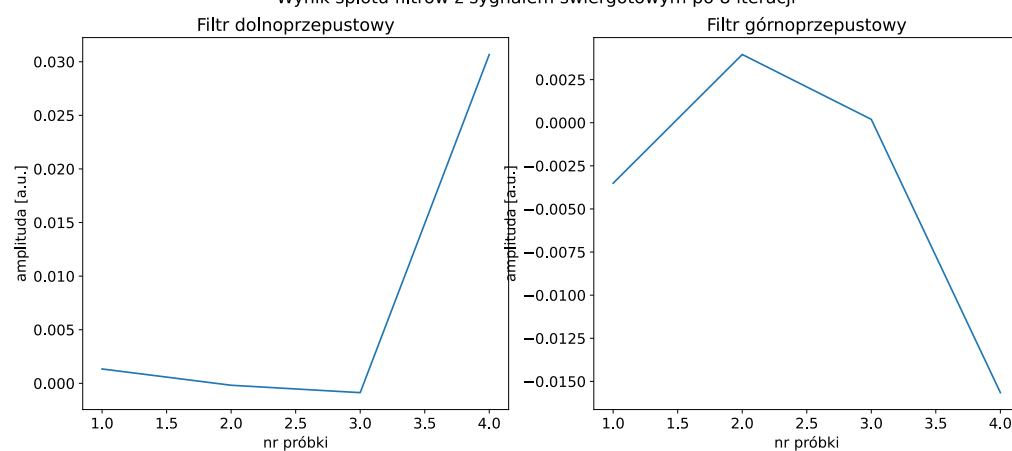
Wynik splotu filtrów z sygnałem świergotowym po 6 iteracji



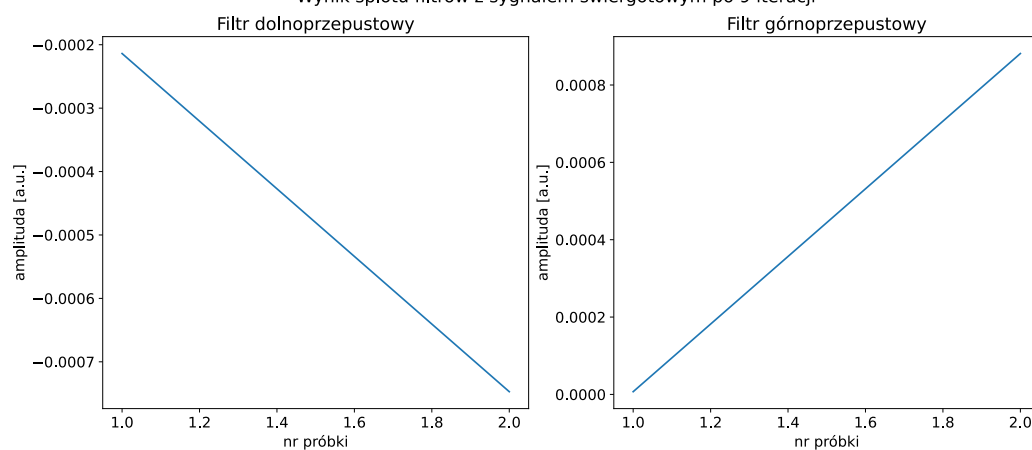
Wynik splotu filtrów z sygnałem świergotowym po 7 iteracji



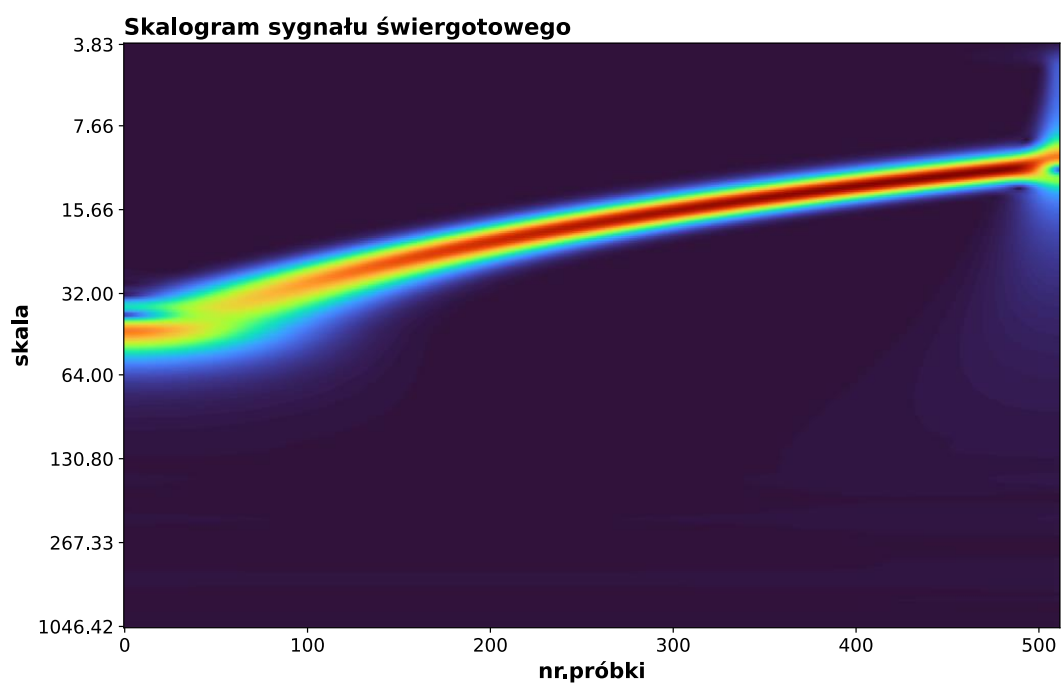
Wynik splotu filtrów z sygnałem świergotowym po 8 iteracji



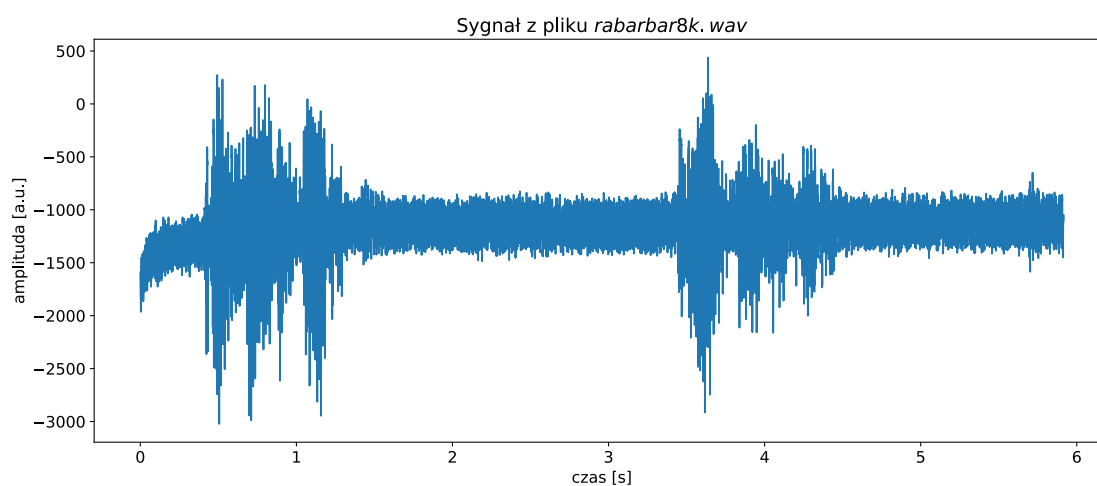
Wynik splotu filtrów z sygnałem świergotowym po 9 iteracji



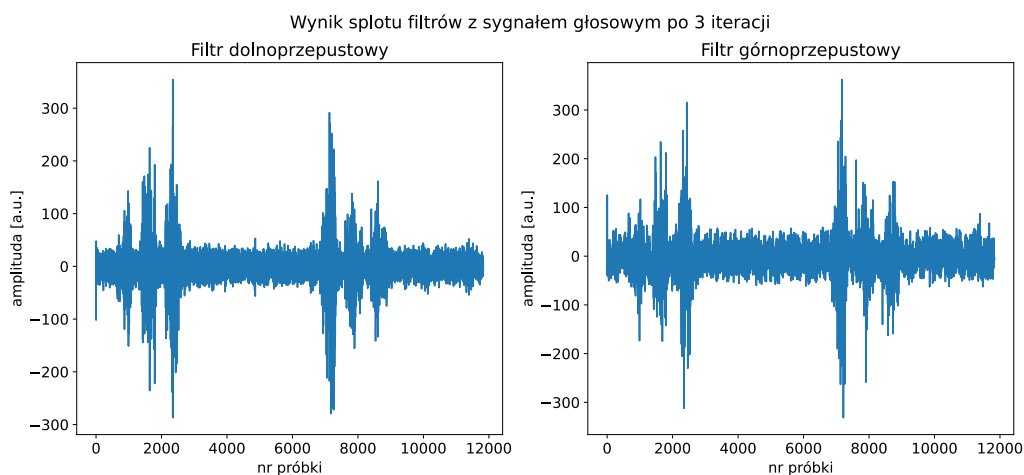
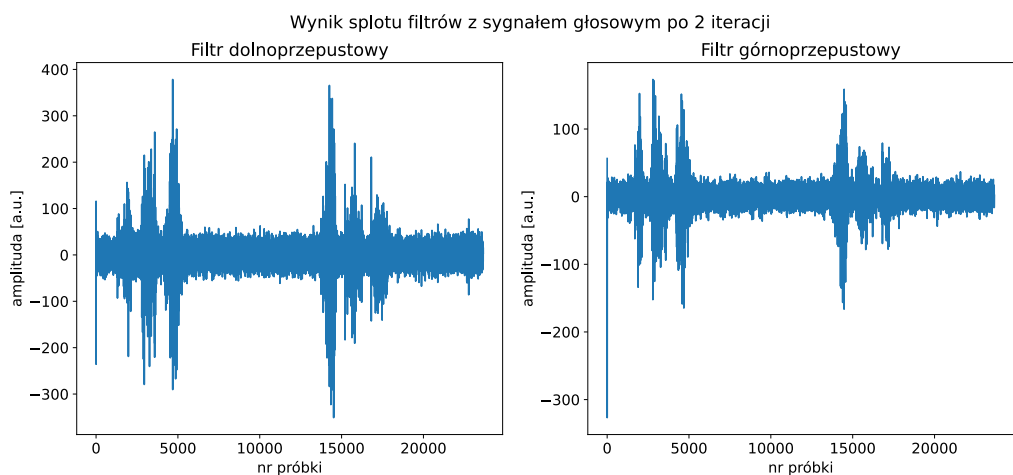
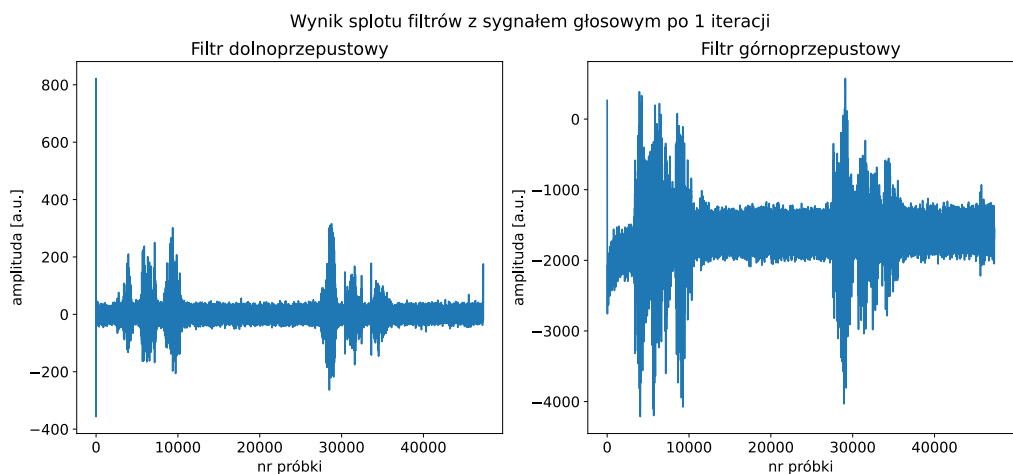
Skalogram sygnału świergotowego wykonany przy pomocy biblioteki *ssqueezepy* dla falki typu *morlet* przedstawia wykres poniżej (niestety przedstawiony w skali).



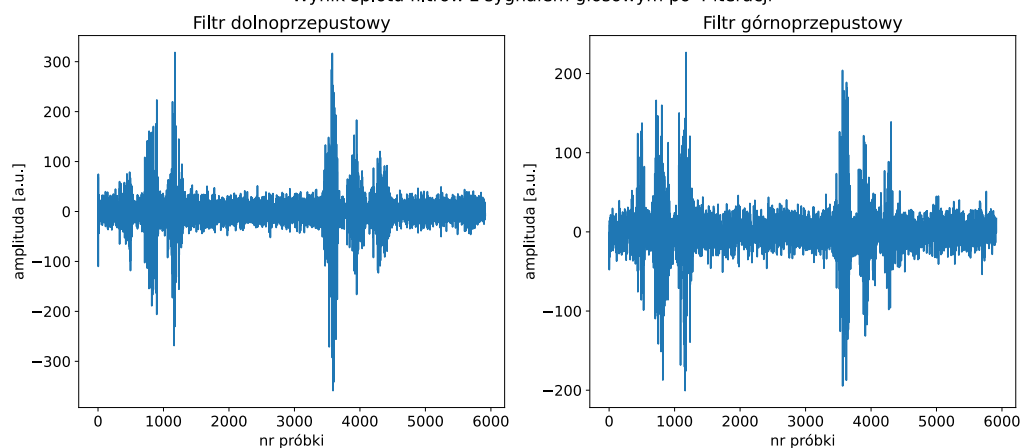
Kolejnym krokiem jest wczytanie sygnału głosowego z pliku *rabarbar8k.wav*. Sygnał ten jest przedstawiony na wykresie poniżej.



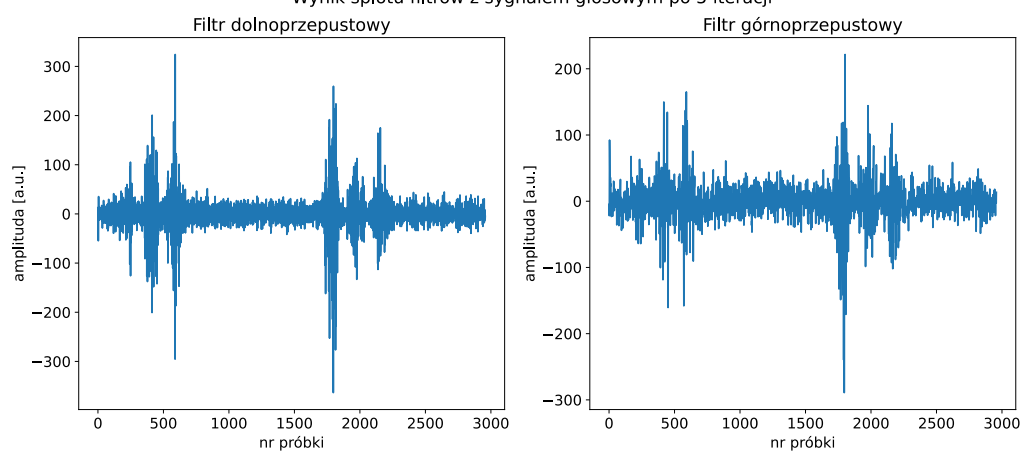
Powtarzając operacje wykonane na sygnale świergotowym, otrzymujemy następujące wyniki:



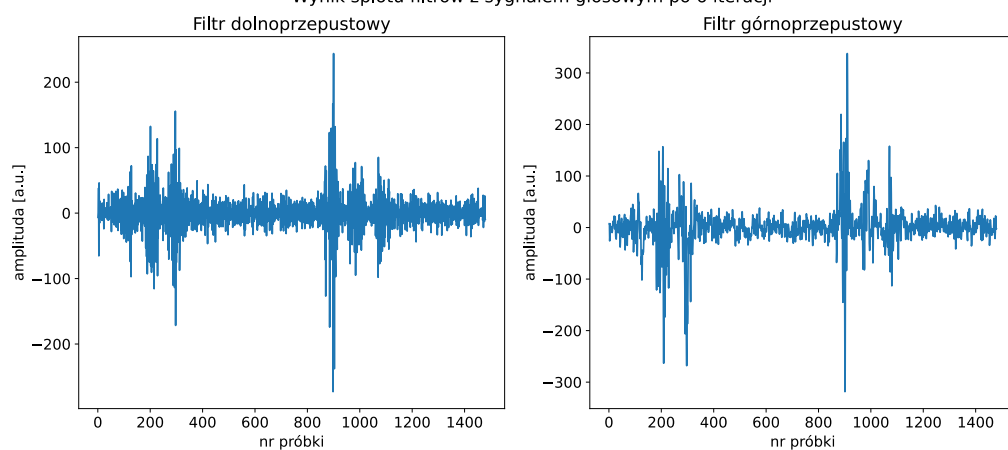
Wynik splotu filtrów z sygnałem głosowym po 4 iteracji



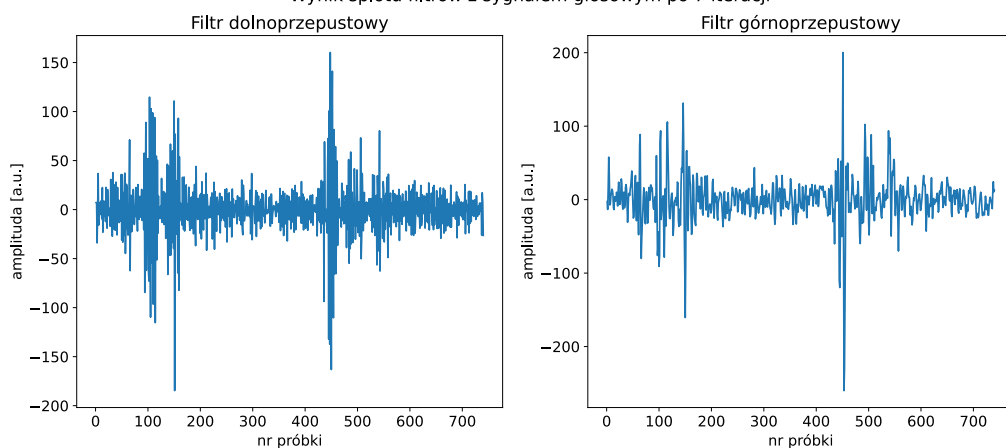
Wynik splotu filtrów z sygnałem głosowym po 5 iteracji



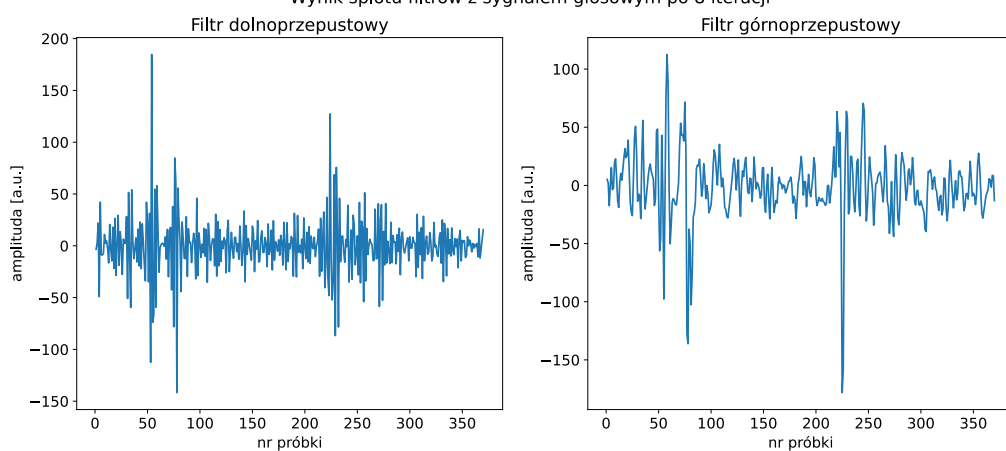
Wynik splotu filtrów z sygnałem głosowym po 6 iteracji



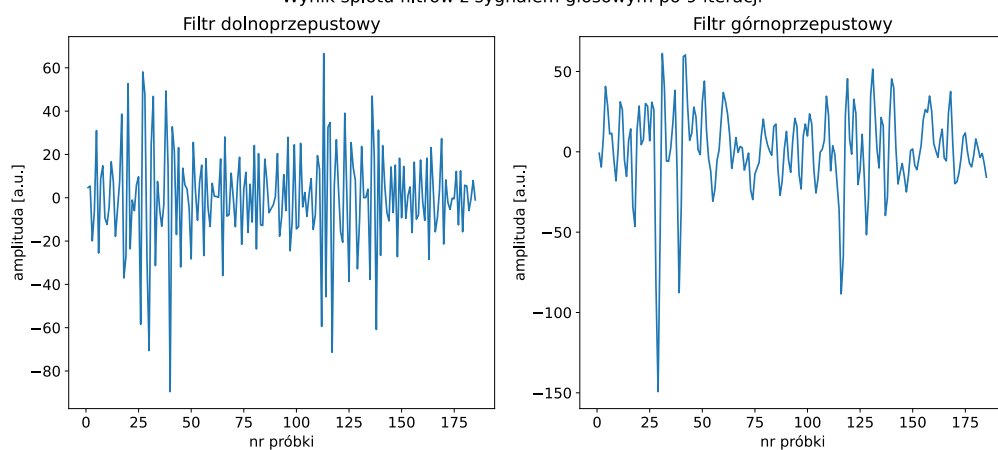
Wynik splotu filtrów z sygnałem głosowym po 7 iteracji



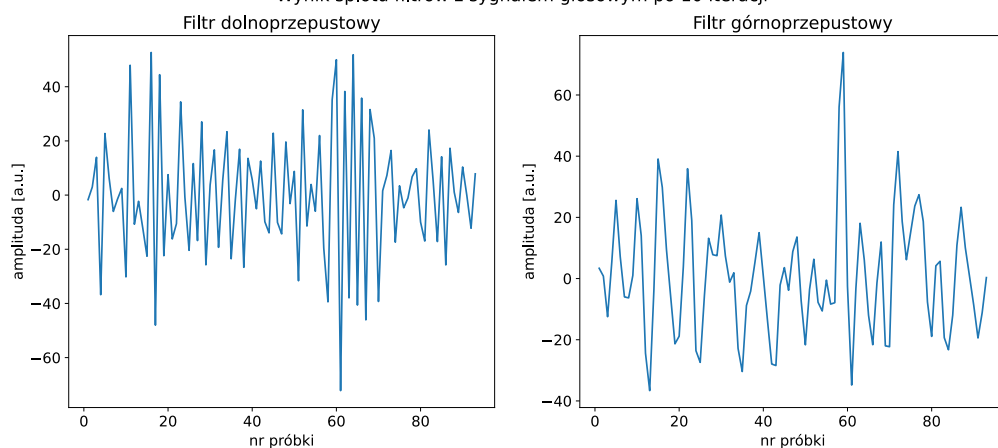
Wynik splotu filtrów z sygnałem głosowym po 8 iteracji



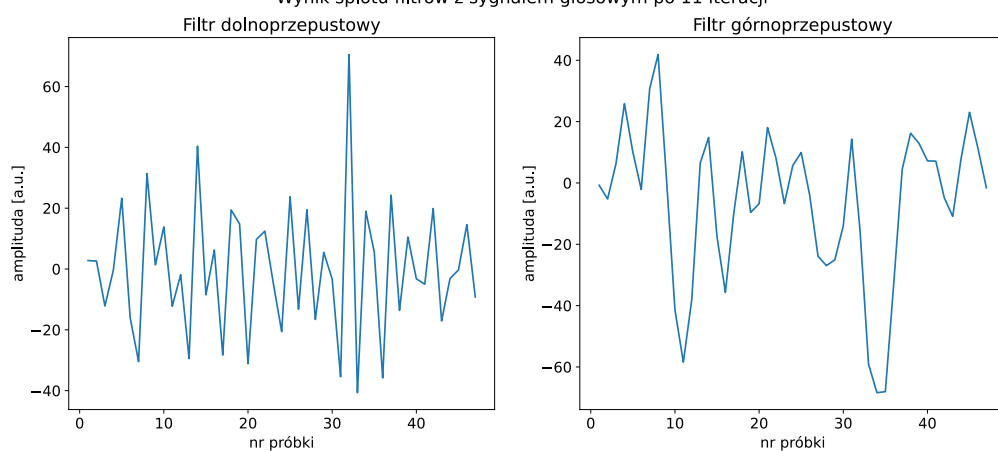
Wynik splotu filtrów z sygnałem głosowym po 9 iteracji



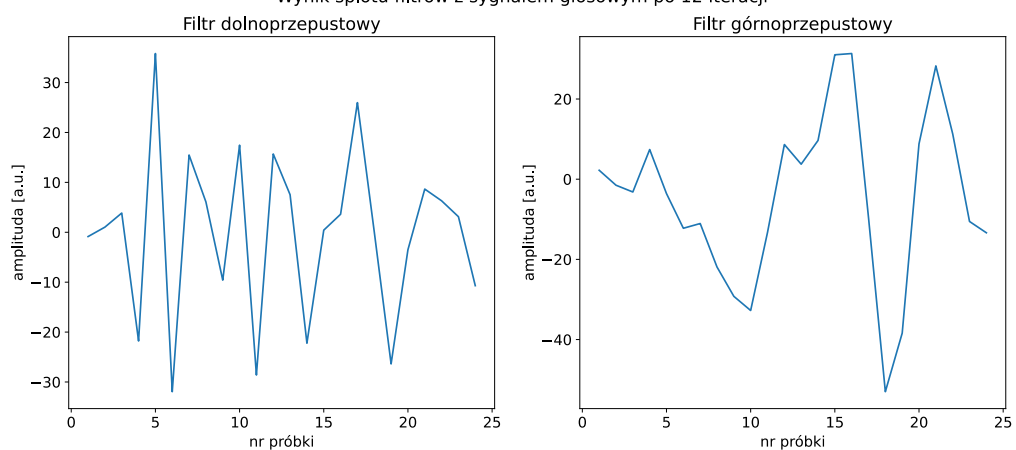
Wynik splotu filtrów z sygnałem głosowym po 10 iteracji



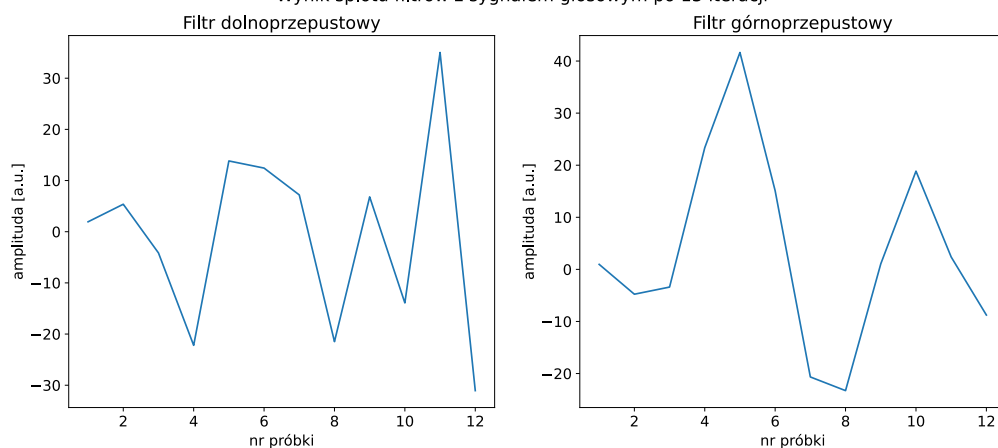
Wynik splotu filtrów z sygnałem głosowym po 11 iteracji



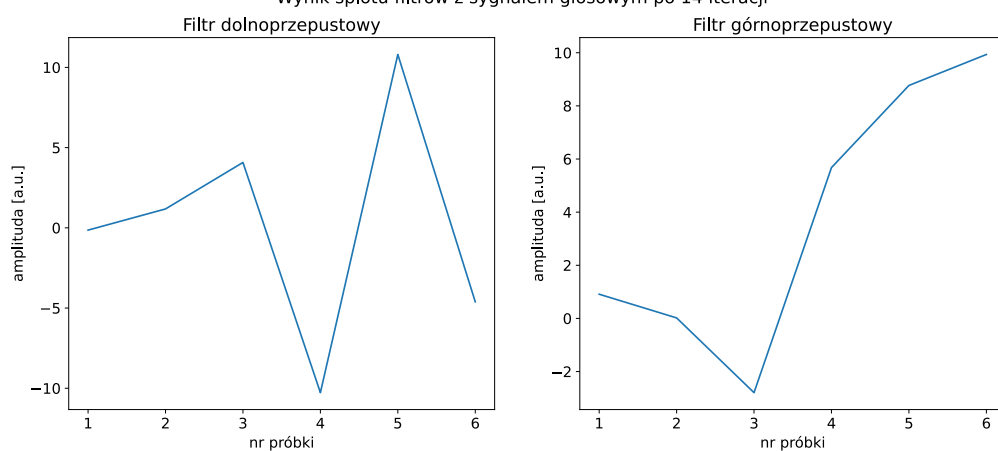
Wynik splotu filtrów z sygnałem głosowym po 12 iteracji



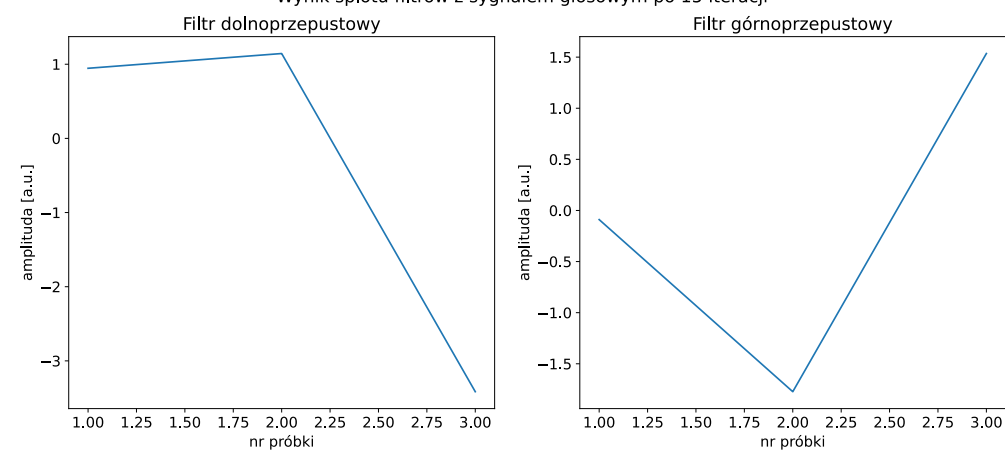
Wynik splotu filtrów z sygnałem głosowym po 13 iteracji



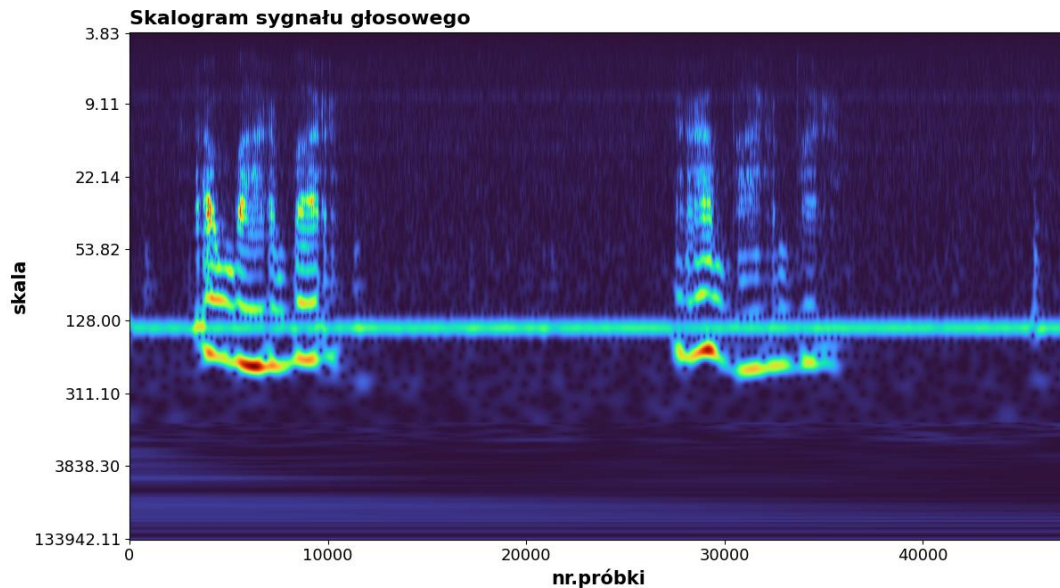
Wynik splotu filtrów z sygnałem głosowym po 14 iteracji



Wynik splotu filtrów z sygnałem głosowym po 15 iteracji



Skalogram dla sygnału głosowego (falka typu *morlet*) przedstawia wykres poniżej.



4. Wnioski

- Transformata falkowa umożliwia analizę sygnału w dziedzinie czas-skala.
- Transformata falkowa w odróżnieniu od transformaty Fouriera nie rozkłada sygnału w bazie fal sinusoidalnych, ale w bazie składowych będących odpowiednio przesuniętą i przeskalowaną falką podstawową.
- Falka jest falą o praktycznie ograniczonej długości i o zerowej wartości średniej.
- W transformacie falkowej sygnał przepuszczany jest na każdym poziomie przez parę filtrów: dolnoprzepustowy oraz górnoprzepustowy.
- Filtracja w dziedzinie częstotliwości jest równoważna operacji splotu w dziedzinie czasu.
- Odpowiedź układu w funkcji czasu można uzyskać przez operację splotu sygnału wejściowego z funkcją odpowiedzi na skok jednostkowy.
- Transformatę falkową można wykorzystać do kompresji plików (JPEG-2000).