

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

ISS Projekt 2021/22

7. januára 2022

Jana Kováčiková (xkovac59)

# Štandardné zadanie

## 1 Základy

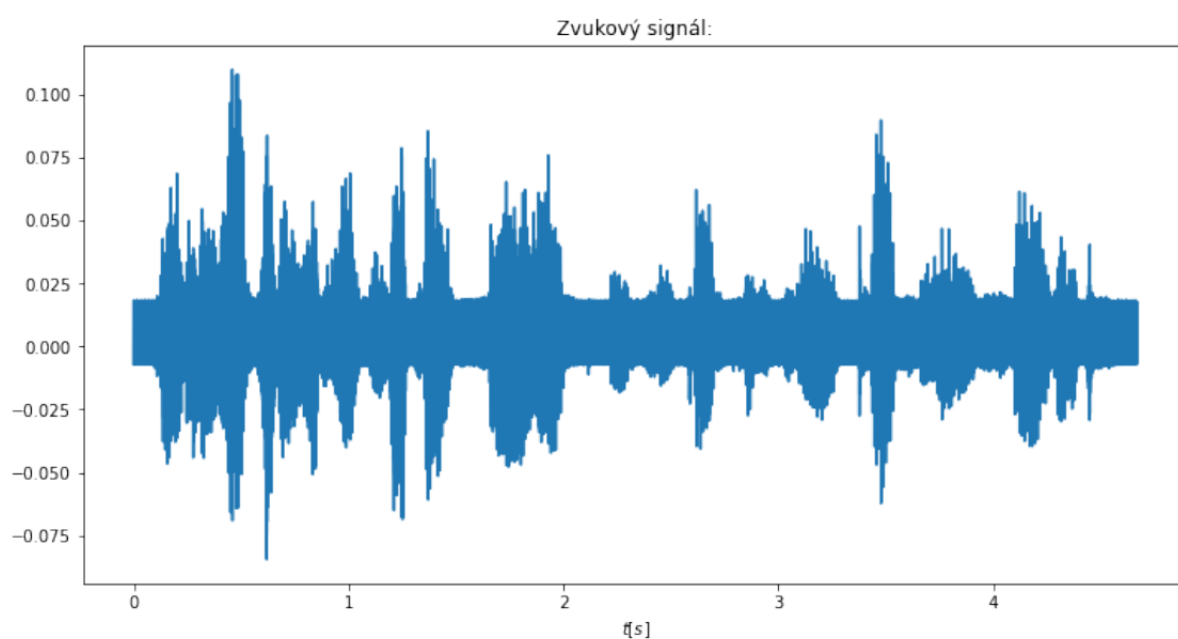
Dĺžka vo vzorkoch = 74752

Dĺžka v sekundách = 4.672

**Minimálna a maximálna hodnota:**

(-0.084442138671875, 0.109954833984375)

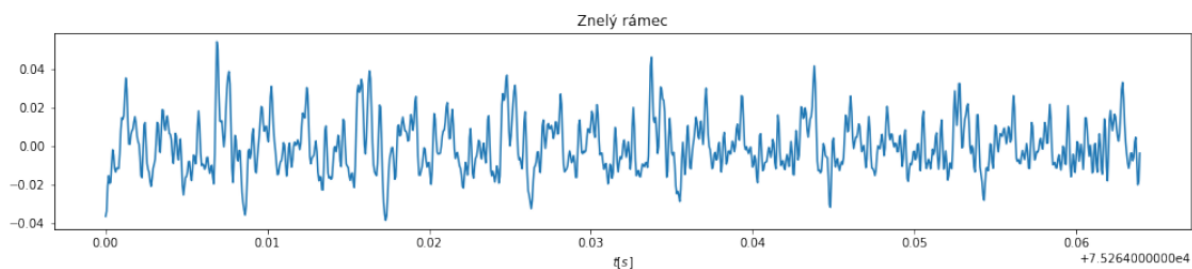
**Graf vstupného signálu xkovac59.wav:**



## 2 Predzpracovanie a rámce

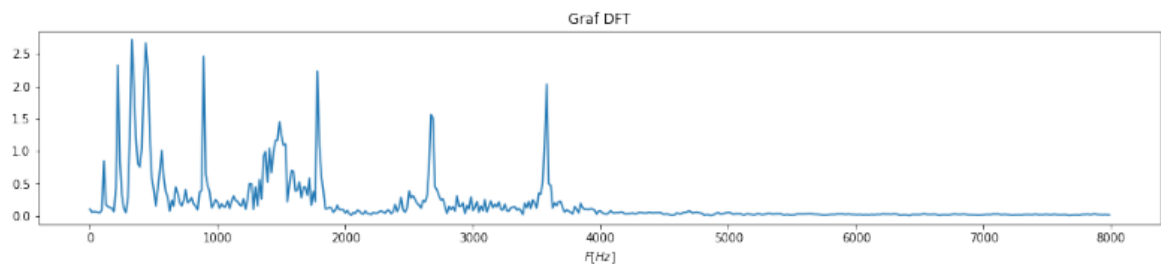
Signál som rozdelila na úseky o dĺžke 1024 vzorkov s prekrytím 512 vzorkov podľa zadania. Následne som zvolila znelý rámec číslo 22, ktorý má periodický charakter.

**Graf znelého rámca číslo 22:**



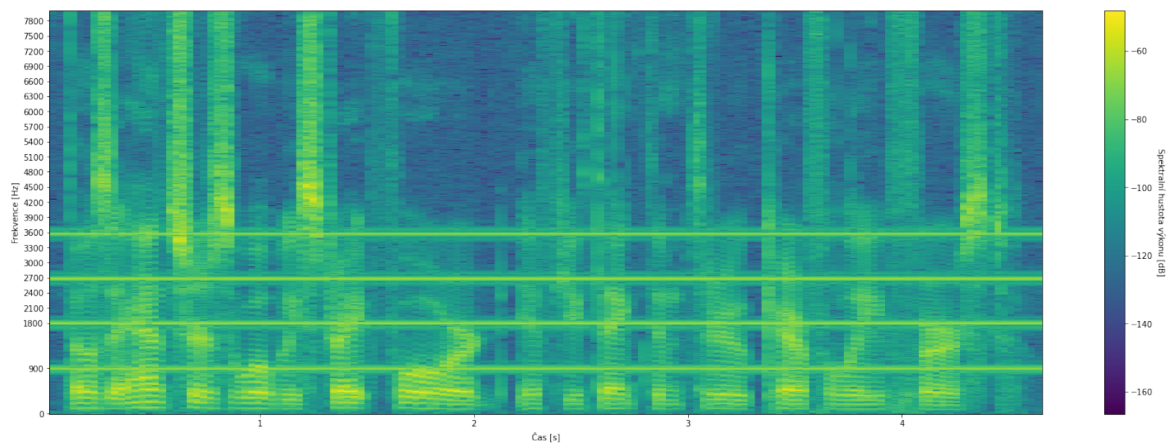
### 3 DFT

Graf diskkrétnej Fourierovej transformácie zobrazený pre frekvencie od 0 do  $F_s/2$ :



### 4 Spektrogram

Spektrogram vstupného signálu xkovac59.wav:



### 5 Určenie rušivých frekvencií

Rušivé frekvencie som určila ručne odčítaním zo spektrogramu. Pomocou

```
plt.yticks(np.append(np.array([0,900]), np.arange(1800, 8000, 300)))
```

som si zobrazila detailnejší popis osi y, aby sa dali rušivé frekvencie odčítať čo najpresnejšie.

$$f1 = 900 \text{ Hz}$$

$$f2 = 1800 \text{ Hz}$$

$$f3 = 2700 \text{ Hz}$$

$$f4 = 3600 \text{ Hz}$$

Ako môžeme vidieť, cosinusovky sú harmonicky vzťahované -  $f2$  je dvojnásobkom  $f1$ ,  $f3$  je trojnásobkom  $f1$  a  $f4$  je štvornásobkom  $f1$ .

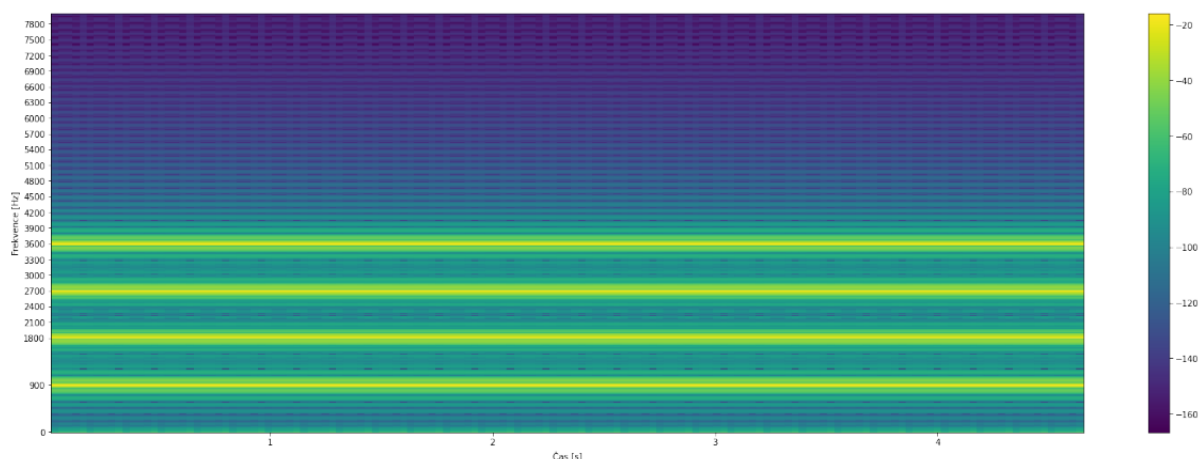
## 6 Generovanie signálu

Z rušivých frekvencií z úlohy 5 som vytvorila štyri cosinusovky pomocou funkcie `np.cos`, ktoré som následne sčítala do premennej `wav` a vygenerovala z nej signál.

```
1 cos_f1 = np.cos(2*np.pi*f1*np.array(fSamples))
2 cos_f2 = np.cos(2*np.pi*f2*np.array(fSamples))
3 cos_f3 = np.cos(2*np.pi*f3*np.array(fSamples))
4 cos_f4 = np.cos(2*np.pi*f4*np.array(fSamples))
5
6 wav = cos_f1+cos_f2+cos_f3+cos_f4
7 wavfile.write("4cos.wav", fs, wav.astype(np.float32))
```

Zo signálu `4cos.wav` bolo potrebné vygenerovať spektrogram a zistiť, či som frekvencie určila a signál vygenerovala správne. Spektrogram naznačuje, že generovanie bolo úspešné.

**Spektrogram signálu `4cos.wav`:**



## 7 Čistiaci filter

V tejto úlohe som navrhla sadu filtrov na potlačovanie rušivých frekvencií. Pre implementáciu som využila 3. alternatívu - **návrh 4 pásmových zádrží**.

```
1 #filter f1
2 N1, Wn1 = buttord([(f1-90)/(fs/2), (f1+90)/(fs/2)], [(f1-30)/(fs/2), (f1+30)/(fs/2)], 30, 50)
3 b1, a1 = butter(N1, Wn1, 'bandstop')
4
5 #filter f2
6 N2, Wn2 = buttord([(f2-90)/(fs/2), (f2+90)/(fs/2)], [(f2-30)/(fs/2), (f2+30)/(fs/2)], 30, 50)
7 b2, a2 = butter(N2, Wn2, 'bandstop')
8
9 #filter f3
10 N3, Wn3 = buttord([(f3-90)/(fs/2), (f3+90)/(fs/2)], [(f3-30)/(fs/2), (f3+30)/(fs/2)], 30, 50)
11 b3, a3 = butter(N3, Wn3, 'bandstop')
12
13 #filter f4
```

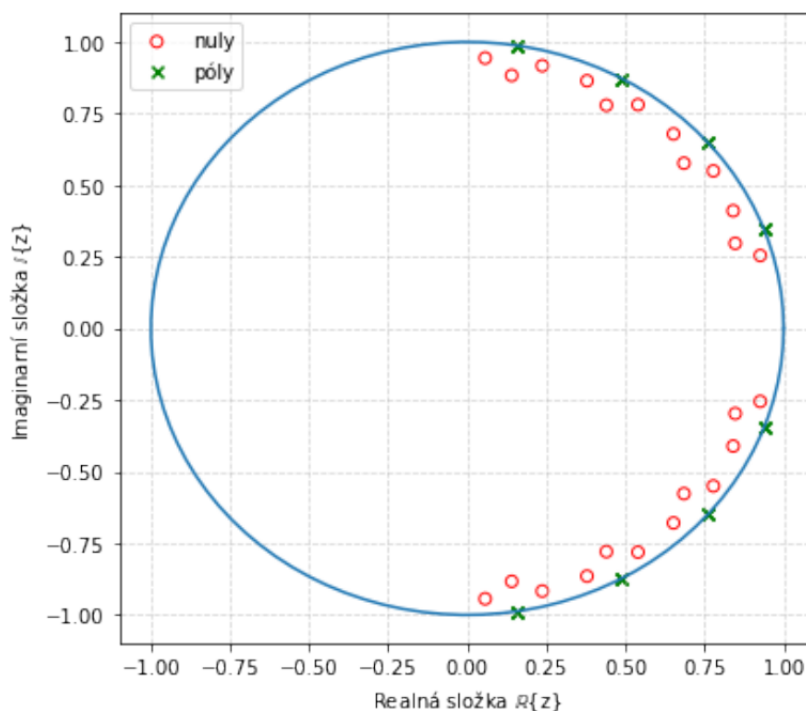
```

14 N4, Wn4 = buttord([(f4-90)/(fs/2), (f4+90)/(fs/2)], [(f4-30)/(fs/2), (f4+30)/(fs
/2)], 30, 50)
15 b4, a4 = butter(N4, Wn4, 'bandstop')

```

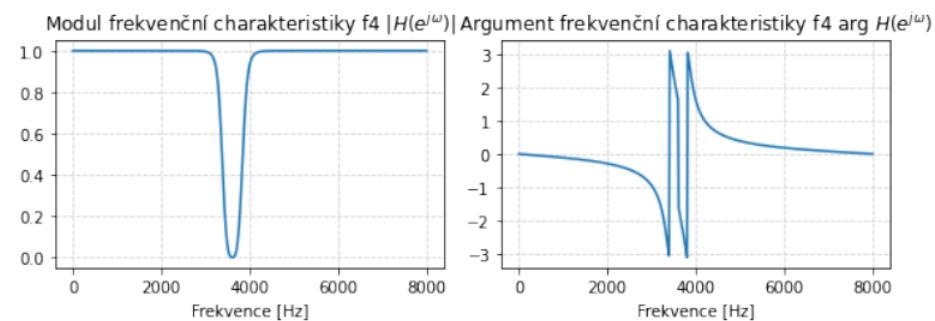
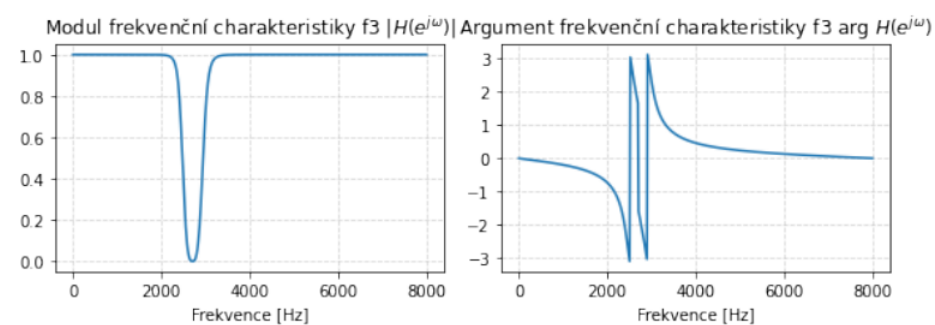
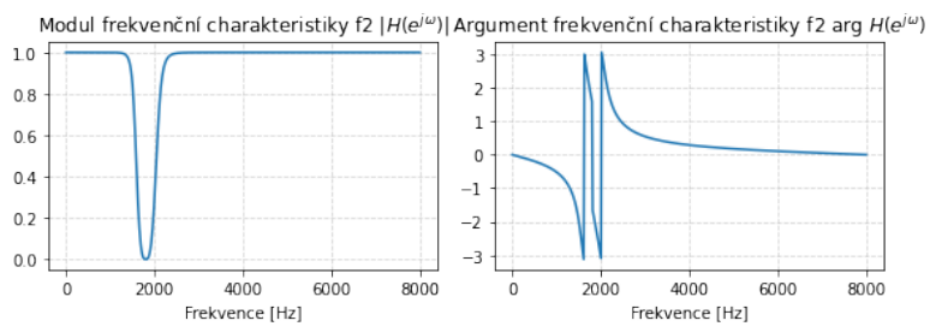
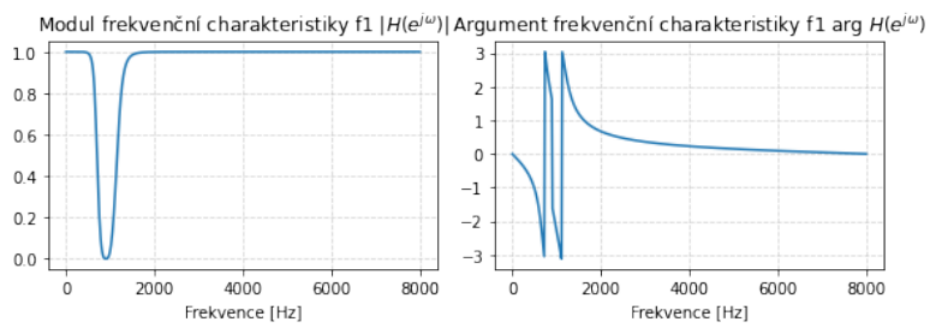
## 8 Nulové body a póly

Graf núl a pólov sady filtrov v komplexnej rovine:



## 9 Frekvenčná charakteristika

Vypočítala som frekvenčné charakteristiky sady filtrov a zobrazila som ich modulovú aj argumentovú časť. Detailnejšie zobrazenie osi x sa mi nepodarilo zobrazíť, no podľa priložených grafov môžeme približne vidieť, že filtre potláčajú rušivý signál na správnych frekvenciách.



## 10 Filtrácia

Vyfiltrovaný signál je vo slušnom dynamickom rozsahu:  $(-0.07672428673281753, 0.07756215666089547)$ . Filtrovanie signálu v tento moment považujem za dokončené, nakoľko vo vypočutej nahrávke nie je počuť rušivé signály, ale čistú a zrozumiteľnú reč.

## Záver

Implementáciu projektu považujem za úspešnú nakoľko sa mi podarilo splniť takmer všetky úlohy zo štandardného zadania: načítanie vstupného signálu a jeho následné zobrazenie, rozdelenie signálu na rámce a vybranie znelého rámca s ktorým som ďalej pracovala, zobrazenie logaritmického výkonového spektrogramu, určenie rušivých frekvencií, generovanie signálu zo 4 cosinusoviek rušivého signálu, navrhnutie sady čistiacich filtrov cez návrh štyroch pásmových zádrží, výpočet a následné zobrazenie nulových bodov a pólov, výpočet a zobrazenie frekvenčných charakteristík filtrov a nakoniec samotná filtrácia signálu.

Pri práci na projekte som sa naučila veľa nových vecí - prvýkrát som pracovala v programovacom jazyku Python, konkrétne v Jupyter Notebooku. Ďalej som moje poznatky z prednášok a cvičení mala možnosť uplatniť v reálnom živote pri spracovaní zadaného signálu, čo hodnotím veľmi pozitívne.