Západočeská Univerzita v Plzni
Fakulta Aplikovaných Věd

MS1 2. semestrální práce

modelování v diagnostice - zpracování signálu

Filip Jašek

Předmět: KKY/MS1 (Modelování a simulace)
Vyučující: Ing. Hajšman Václav, Ph.D., Ing. Liška Jindřich, Ph.D., Ing. Janeček Petr, Ph.D.
Cvičící: Ing. Liška Jindřich, Ph.D. Datum: 23. května 2022

1 Zadání

Zadání 2. semestrální práce z předmětu MS1

modelování v diagnostice - zpracování signálu

1. Načtěte signál ze souboru signal.mat do Matlabu

2. Zobrazte časový vývoj signálu (vzorkovací frekvence je 80kHz)

3. Určete časové parametry signálu - střední hodnotu signálu, energii signálu a efektivní hodnotu

3. Určete frekvenční parametry - zobrazte spektrum signálu. Které frekvence jsou v signálu dominantní?

4. Implementujte metodu krátkodobé Fourierovy transformace v Matlabu.

5. Ověřte princip neurčitosti - zvolte krátkou (např. 256 vzorků) a dlouhou (např. 4096 a více vzorků) okénkovou funkci a výsledky zobrazte formou spektrogramu. Jaký je rozdíl mezi oběma spektrogramy? V čem spočívá princip neurčitosti při časo-frekvenčním zpracování signálů?

6. Nalezněte časo-frekvenční událost v datech, kolik událostí se v signálu nachází a v jakém čase nastaly?

7. Vytvořte zprávu shrnující získané výsledky formou zobrazení a vysvětlujícího textu. V závěru zprávy uveďte kód z Matlabu, který jste použili pro získání výsledků.

Základní funkce v Matlabu doporučené pro zpracování semestrální práce (podrobnější informace viz dokumentace/help Matlabu):

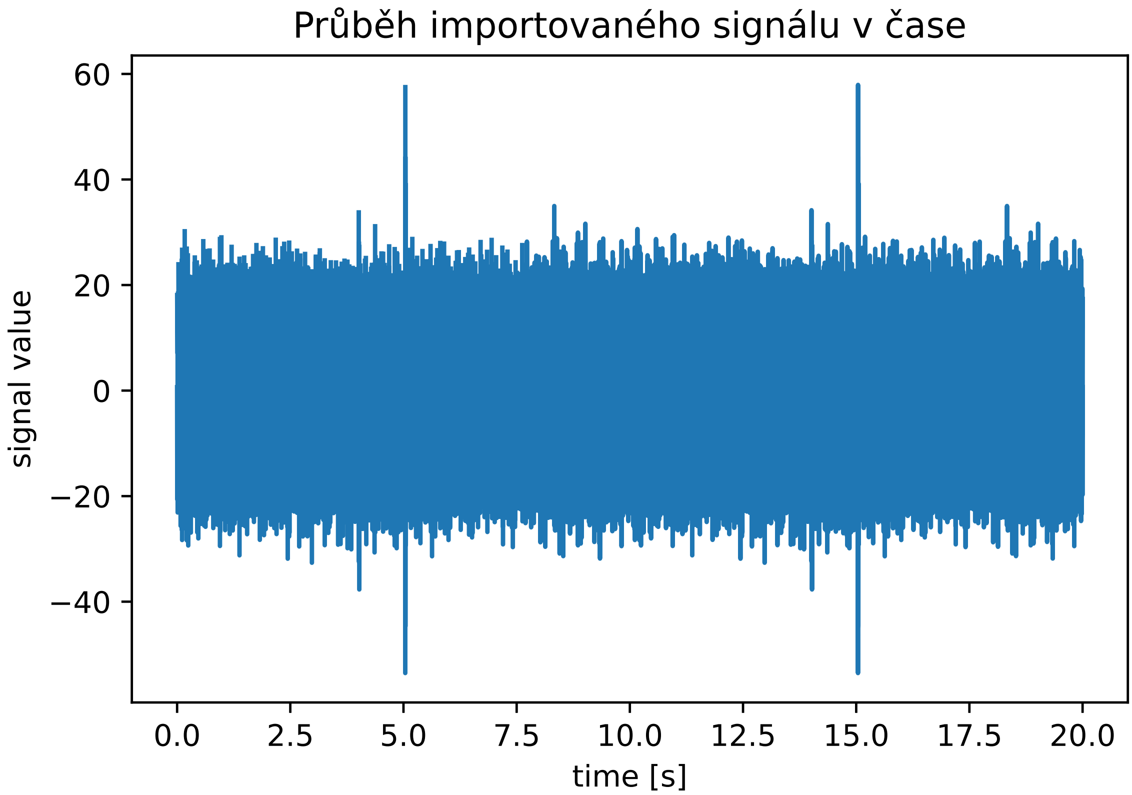
- load, size, length, for cyklus, ...
- plot, xlabel, ylabel, grid, title,...
- fft, abs, mean, sqrt, hanning, ...
- imagesc, waterfall, caxis, colorbar, ...
- ...

2 Vypracování

Samotné vypracování bylo provedeno v Pythonu za pomocí knihoven pandas, numpy a matplotlib. Tento přístup byl zvolen na základě faktu, že provedení v Pythonu umožňuje replikovat postup kýmkoliv bez ohledu na vlastnictví licence programu Matlab.

2.1 Načtení a zobrazení signálu

Z poskytnutého signálu ve formátu .mat byl extrahován signál do standardního csv souboru, který byl pak následně zpracován. Z poskytnuté informace o frekvenci $f = 80kHz$ stanovíme periodu $T = 1/f = 0.0000125s$ a vytvoříme časovou osu s časovými přírůstky rovné periodě T. Zjistíme tak, že vzorek signálu byl odebrán na časovém intervalu $20s$. Průběh zadaného signálu v čase je zobrazen na následujícím obrázku 1.



Obrázek 1: Průběh zadaného signálu v čase.

2.2 Určení časových parametrů signálu

Střední hodnotu signálu určíme jako aritmetický průměr na konečném intervalu pomocí vzorce

$$\bar{x} = \frac{1}{N} \cdot \sum_{t=0}^{N} x(t),$$

kde $\bar{x}$ je požadovaná střední hodnota, $N = 1600000$ je počet vzorků a $x(t)$ hodnota signálu v časovém okamžiku $t \in (0, N)$

Energii vypočteme za pomoci vektorového součinu samotného signálu se sebou a vydělením frekvencí nebo použitím následujícího vzorce.

$$E = T \cdot \sum_{t=0}^{N} x(t)^2$$

Násobení periodou je nutné jelikož výpočet vychází ze spojitého vzorce

$$E = \int_{-\infty}^{+\infty} |x(t)|^2 dt,$$

kde se počítá integrál z nekonečně malého přírůstku, zatímco pomocí sumy jen umocníme diskrétní časové okamžiky. Vynásobením periody tak stanovíme šířku jednotlivých obdélníků dané vzorkovací frekvencí f.

Výkon se pak dopočte jako

$$P = \frac{1}{T \cdot N} \cdot E,$$

jelikož je E již díky předchozímu výpočtu popsáno v závislosti na čase, je nutné podobně jako ve spojitém případě

$$P = \lim_{X \to \infty} \frac{1}{2X} \int_{-X}^{+X} |x(t)| dt$$

vydělit celkovým časem ve kterém byla energie spočtena.

Efektivní hodnota je pak jednoduchá odmocnina vypočteného výkonu

$$x_{ef} = \sqrt{P}.$$

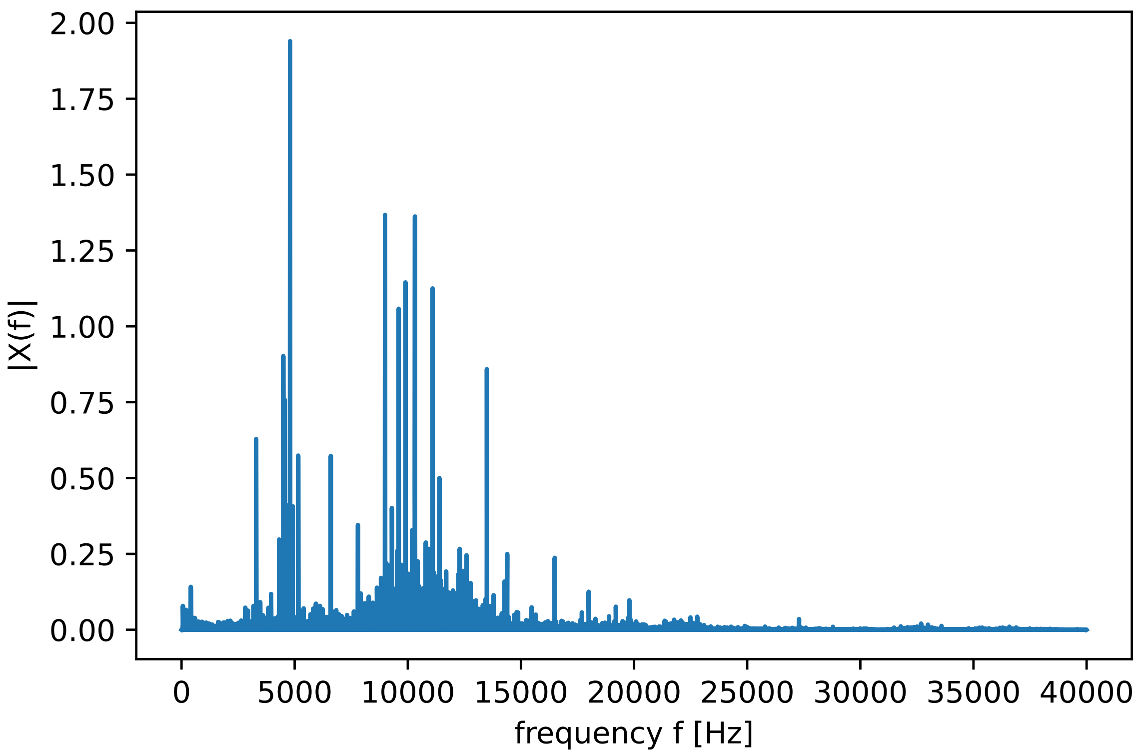
Pro zadaný signál vyšly následující výsledky:

```
stredni hodnota signalu: -0.0003
energie signalu: 1025.6469
vykon signalu: 51.2823
efektivni hodnota signalu: 7.1612
```

2.3 Určení frekvenčních parametrů

Pomocí absolutní hodnoty Fourierovy transformace byl získán symetrický graf spektra signálu od frekvence $-40kHz$ do $+40kHz$, který byl sečten do podoby v obrázku 2 obsahující pouze kladné frekvence.

V získaném grafu lze pozorovat, že dominantní frekvence jsou $4798.2Hz$ a v okolí $10kHz$.



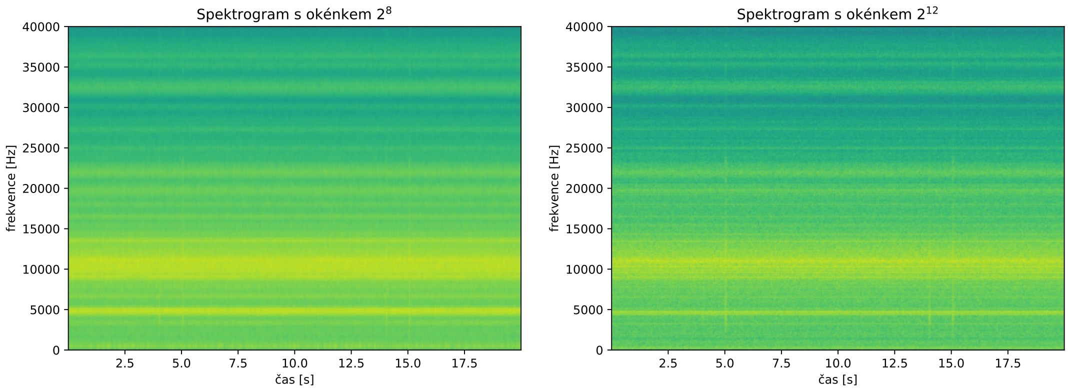
Obrázek 2: Frekvenční spektrum zadaného signálu pro kladné frekvence.

2.4 Ověření principu neurčitosti

Princip neurčitosti spočívá v tom, že čím přesněji budeme chtít znát jednu vlastnost (zde signálu) tím nepřesněji budeme znát jinou. V našem případě zvětšováním okénkové funkce získáme přesnější informace o událostech v čase, ale na úkor přesnosti intenzit frekvencí. Při tvorbě spektrogramu počítáme s tím, že signál je na krátkém časovém okamžiku stacionární. Tento časový úsek definujeme pomocí velikosti okénka, na kterém se spočítá jedna průměrná hodnota podle typu okénka.

Jak je ale z obrázku 3 patrné, zvětšováním okénka jsme sice zvýraznili některé události v čase, ale zároveň ztratili jemnost měřítka ve frekvencích.

Zde se dokonce podařilo větším okénkem 2^{12} utlumit i jednu časovou událost v čase $4s$. Pro potvrzení časových událostí jsem použil spektrogram vytvořený menším okénkem 2^8. Sečetl jsem hodnoty jednotlivých okénkách v čase a získal informaci, že signál obsahuje 4 časo-frekvenční události v časech 4, 5, 14 a 15s.



Obrázek 3: Ověření principu neurčitosti za pomoci okénkové funkce.

3 Zdrojový kód

```python
# -*- coding: utf-8 -*-

#importy
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

"""## 1) Načtení dat signálu"""

#stazeni signalu z githubu
#! git clone https://github.com/Fiiila/MS1_2022.git

#nacteni signalu pomoci pandas
signal_pd = pd.read_csv("MS1_2022/signal.csv",header=None)
#prevod na numpy
signal = signal_pd.to_numpy()
signal = signal[:,0]

"""## 2) Zobrazení časového vývoje signálu"""

#vytvoreni casove osy na zaklade frekvence signalu
freq = 80000 #Hz
print(f"frequency of signal: {freq} Hz")
T = 1/freq
print(f"signal T: {T} s")
time = np.array(range(0,len(signal)))*T
plt.figure()
plt.plot(time, signal)
plt.xlabel("time [s]")
plt.ylabel("signal value")
plt.title("Průběh importovaného signálu v čase")

"""## 3) Určení časových parametrů signálu"""

#zobrazeni parametru signalu
print(f'stredni hodnota signalu: {signal.mean():.4f}')
energy = signal.dot(signal)*T
print(f'energie signalu: {energy:.4f}')
power = (1/(T*len(signal)))*energy
print(f'vykon signalu: {power:.4f}')
print(f'efektivni hodnota signalu: {np.sqrt(power):.4f}')

"""## 4) Určení frekvenčních parametrů signálu"""

# spektrum signálu
spectrum = np.fft.fft(signal)
power_spectrum = abs(spectrum)/len(signal)
frequencies = np.fft.fftfreq(len(spectrum),1/freq)
i = frequencies>0
plt.figure()
plt.plot(frequencies[i],2*power_spectrum[i])
plt.xlabel("frequency f [Hz]")
plt.ylabel("|X(f)|")
plt.savefig("power_spectrum.eps", format="eps")
# dominantní frekvence
print(f'dominantni frkvence: {frequencies[np.argmax(abs(spectrum))]}Hz')

"""## 5) Ověření principu neurčitosti"""

#okenkova funkce
plt.figure(figsize=(15,5))
# mensi okenko
plt.subplot(1,2,1)
plt.title("Spektrogram s okénkem $2^8$")
plt.xlabel("čas [s]")
plt.ylabel("frekvence [Hz]")
spectrogram1 = plt.specgram(signal,Fs=freq,NFFT=2**8,mode="default", window=plt.mlab.window_hanning)
# vetsi okenko
plt.subplot(1,2,2)
plt.title("Spektrogram s okénkem $2^{12}$")
plt.xlabel("čas [s]")
```

```
plt.ylabel("frekvence [Hz]")
spectrogram2 = plt.specgram(signal,Fs=freq,NFFT=2**12,mode="default", window=plt.mlab.window_hanning)

"""## 6) Nalezení časo-frekvenční události

Na vykreslených grafech lze pozorovat 4 časo-frekvenční události v časech 4s, 5s, 14s a 15s.
"""

#nalezeni caso-frekvencnich udalosti
spectrogram = spectrogram1
#suma jednotlivych sloupcu
column_sum = np.sum(spectrogram[0],axis=0)
#vykresleni sum v jednotlivych casovych okamzicich
plt.figure()
plt.scatter(np.arange(0,(len(time)/2**12),(len(time)/2**12)/len(column_sum))*(T*2**12),column_sum)
#vypsani maxim
print(f"casy ve kterych byla nalezena maxima: {np.argpartition(column_sum,-4)[-20:]*((len(time)/2**12)/len(column_sum))*(T*2**12)}")
```