

ISS - Projekt

2022/23

Lukáš Večerka, xvecer30

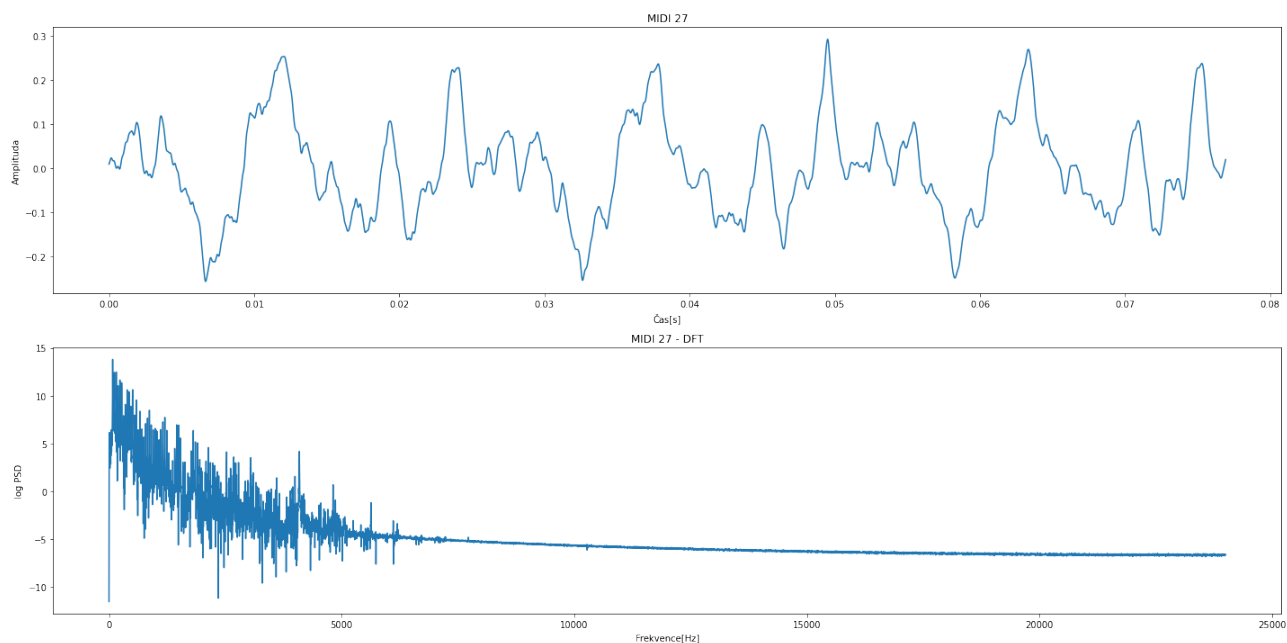
16. prosince 2022

Obsah

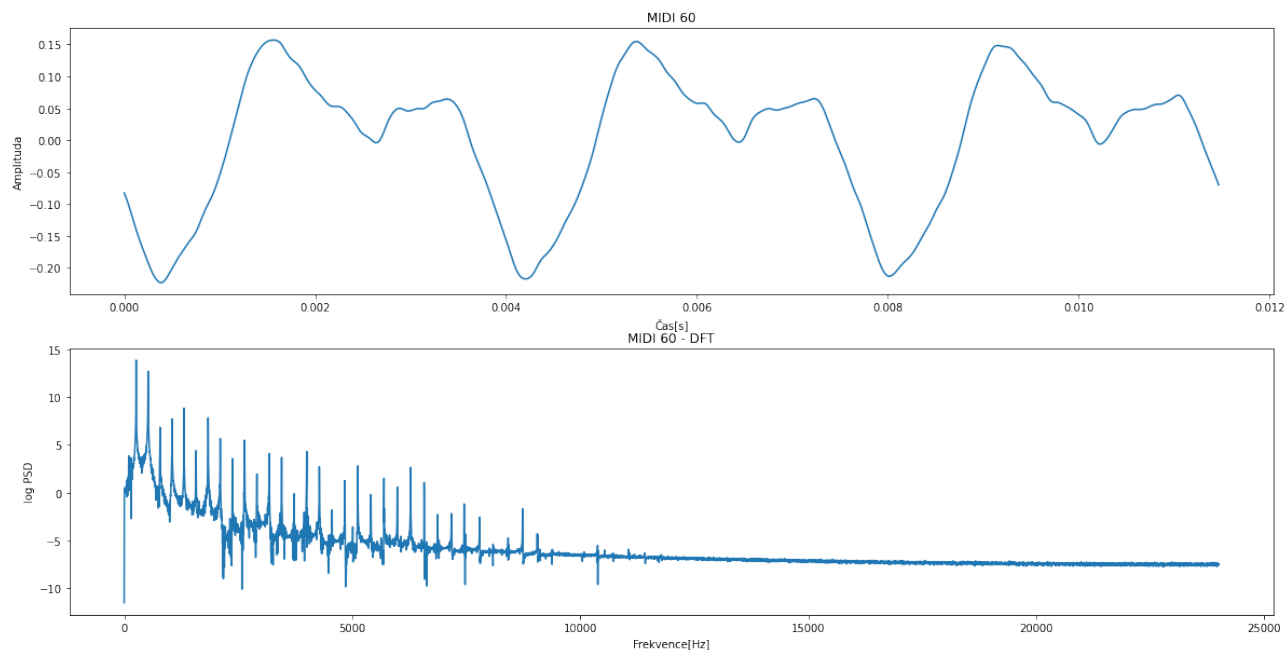
1	Úkol 4.1 - Základy	2
2	Úkol 4.2 - Určení základní frekvence	3
3	Úkol 4.3 - Zpřesnění odhadu základní frekvence	5
4	Úkol 4.4 - Reprezentace klavíru	6
5	Úkol 4.5 - Syntéza tónu	8
6	Úkol 4.6 - Generace hudby	9
7	Úkol 4.7 - Spektrogram	9

1 Úkol 4.1 - Základy

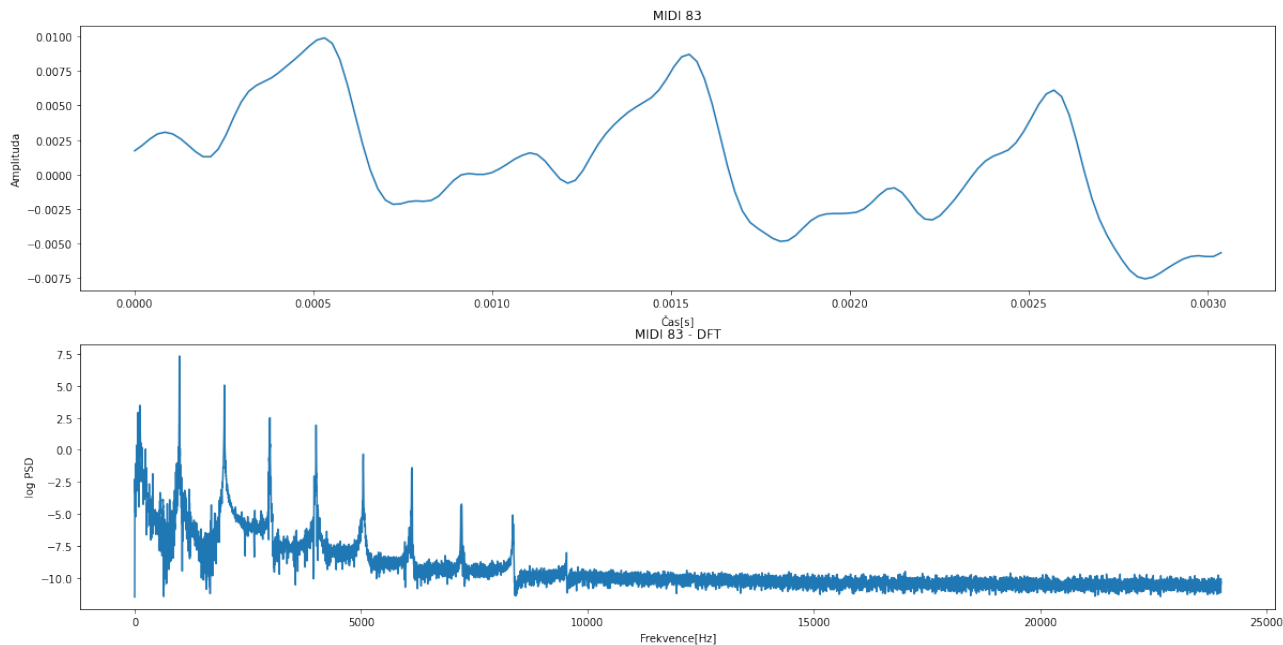
Pomocí algoritmu uvedeného ze zadání jsem si nahrál vzorky dlouhé 0.5 s každého tónu. Přeskočil jsem 0.25 s od začátku, kde zní úder kladívka. Následně jsem pro mé tři dané tóny provedl DFT a vykreslil jsem 3 periody daného tónu a jeho spektrum.



Obrázek 1: MIDI 27 - signál a spektrum (3 periody)



Obrázek 2: MIDI 60 - signál a spektrum (3 periody)

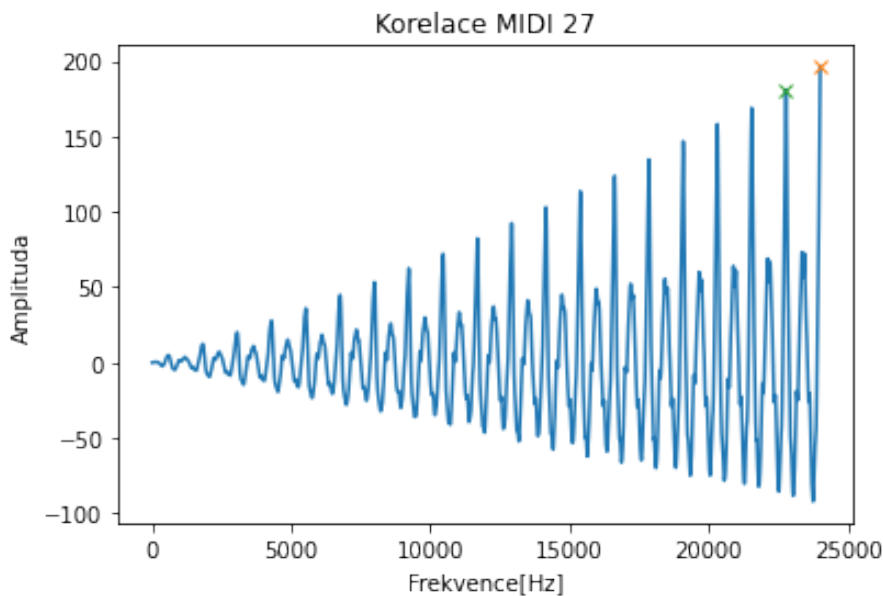


Obrázek 3: MIDI 83 - signál a spektrum (3 periody)

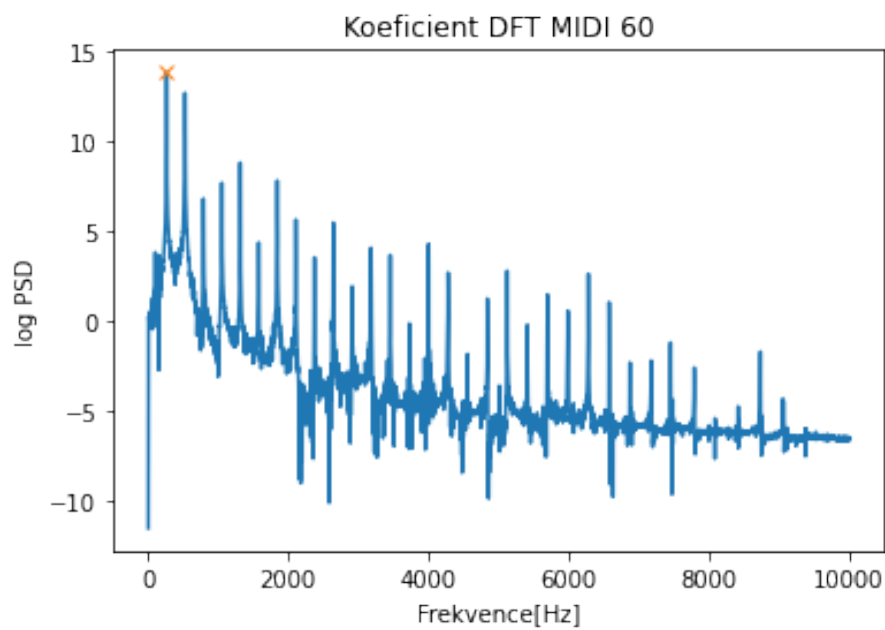
2 Úkol 4.2 - Určení základní frekvence

Nejdříve jsem vyzkoušel určení základní frekvence pro všechny tóny provést pomocí DFT, ale zjistil jsem, že tato metoda nefunguje pro nižší tóny od MIDI 24-41. Pro tyto tóny se vyplatí odhadnout základní frekvence pomocí autokorelace, která zase naopak nefunguje na tóny o vyšších frekvencích. Proto tedy provádím odhad tónů 24-41 pomocí autokorelace a pro tóny 41-108 používám DFT.

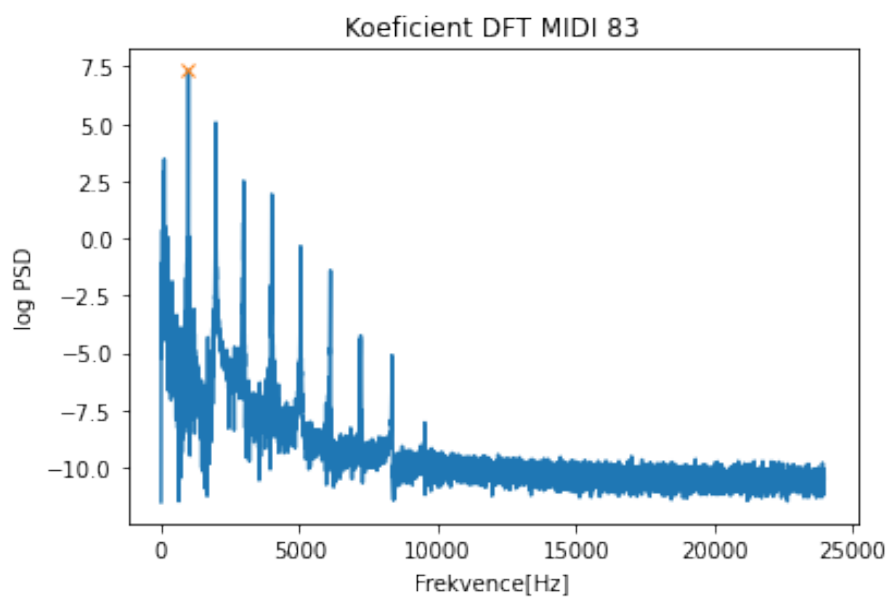
U metody DFT zjišťujeme odhad tak, že zjistíme na které frekvenci leží největší koeficient DFT. U autokorelace se zjišťuje odhad pomocí rozdílu frekvencí mezi největším a druhým největším koeficientem autokorelace.



Obrázek 4: Zjištění základní frekvence tónu 27 pomocí autokorelace



Obrázek 5: Největší koeficient MIDI 60



Obrázek 6: Největší koeficient MIDI 83

Porovnání odhadu frekvence se skutečnou frekvencí:

Tón	Vypočtená f_0	Skutečná f_0
27	38.961	38.89
60	262.0	261.63
83	988.0	987.77

Můžeme vidět, že díky tomu, že u odhadu základní frekvence pomocí metody DFT dostáváme zaokrouhlená čísla, odhady nejsou tak přesné jako například při autokorelaci. Proto je lepší odhady u DFT zpřesnit pomocí DTFT.

3 Úkol 4.3 - Zpřesnění odhadu základní frekvence

Pro zpřesnění odhadu jsem využil metodu DTFT. Vypočítám koeficienty pro frekvence rozložené na 100 centech okolo odhadnuté původní frekvence. Interval zjišťuji tak, že si najdu levou a pravou hranici za pomoci poměru 50 centů. Implementaci DTFT jsem přebíral z přednášky a trochu ji doupřávil. DTFT počítám pouze pro tóny vyšší jak 41 protože pro nižší tóny má metoda horší výsledky jak autokorelace.

```
for tone in range(42, MIDIT0+1):
    max_freq = basicFreq1[tone]
    if tone in range (53,56):
        max_freq = max_freq//2

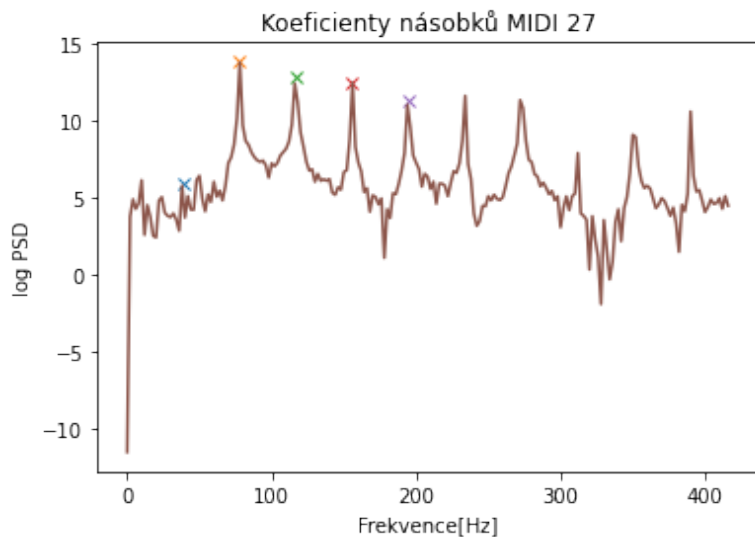
    ffrom = max_freq/1.029302 # 50 cents = f2/f1 = 1.029302
    fto = max_freq*1.029302
    n = np.arange(0,N)
    fsweep = np.linspace(ffrom, fto, 1000)
    A = np.zeros([1000, N],dtype=complex)
    for k in np.arange(0,1000):
        A[k,:] = np.exp(-1j * 2 * np.pi * fsweep[k] / Fs * n)
    Xdtft = np.matmul(A,xall[tone].T)
    precisefmax = fsweep[np.argmax(np.abs(Xdtft))]
    basicFreq1[tone]=precisefmax
```

Porovnání odhadnutých frekvencí se skutečnými:

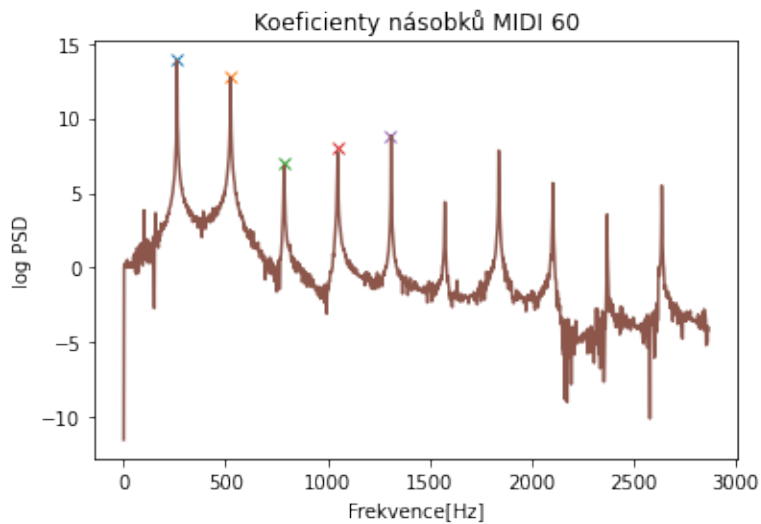
Tón	Původní f_0	Zpřesněná f_0	Skutečná f_0
27	38.961	-	38.89
60	262.0	261.768	261.63
83	988.0	987.527	987.77

4 Úkol 4.4 - Reprezentace klavíru

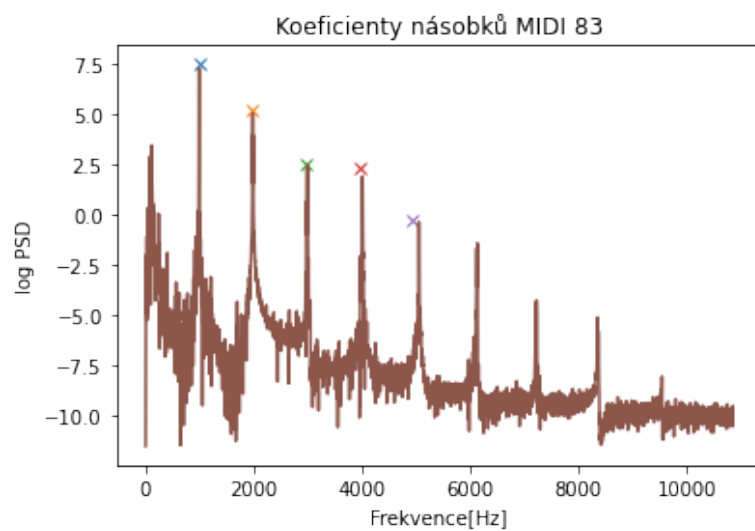
Postupoval jsem tak, že jsem si zjistil největší koeficienty na násobcích odhadnuté základní frekvence a uložil jsem si jejich modul a fázi do pole deseti prvků.



Obrázek 7: Vyznačené koeficienty pro MIDI 27



Obrázek 8: Vyznačené koeficienty pro MIDI 60



Obrázek 9: Vyznačené koeficienty pro MIDI 83

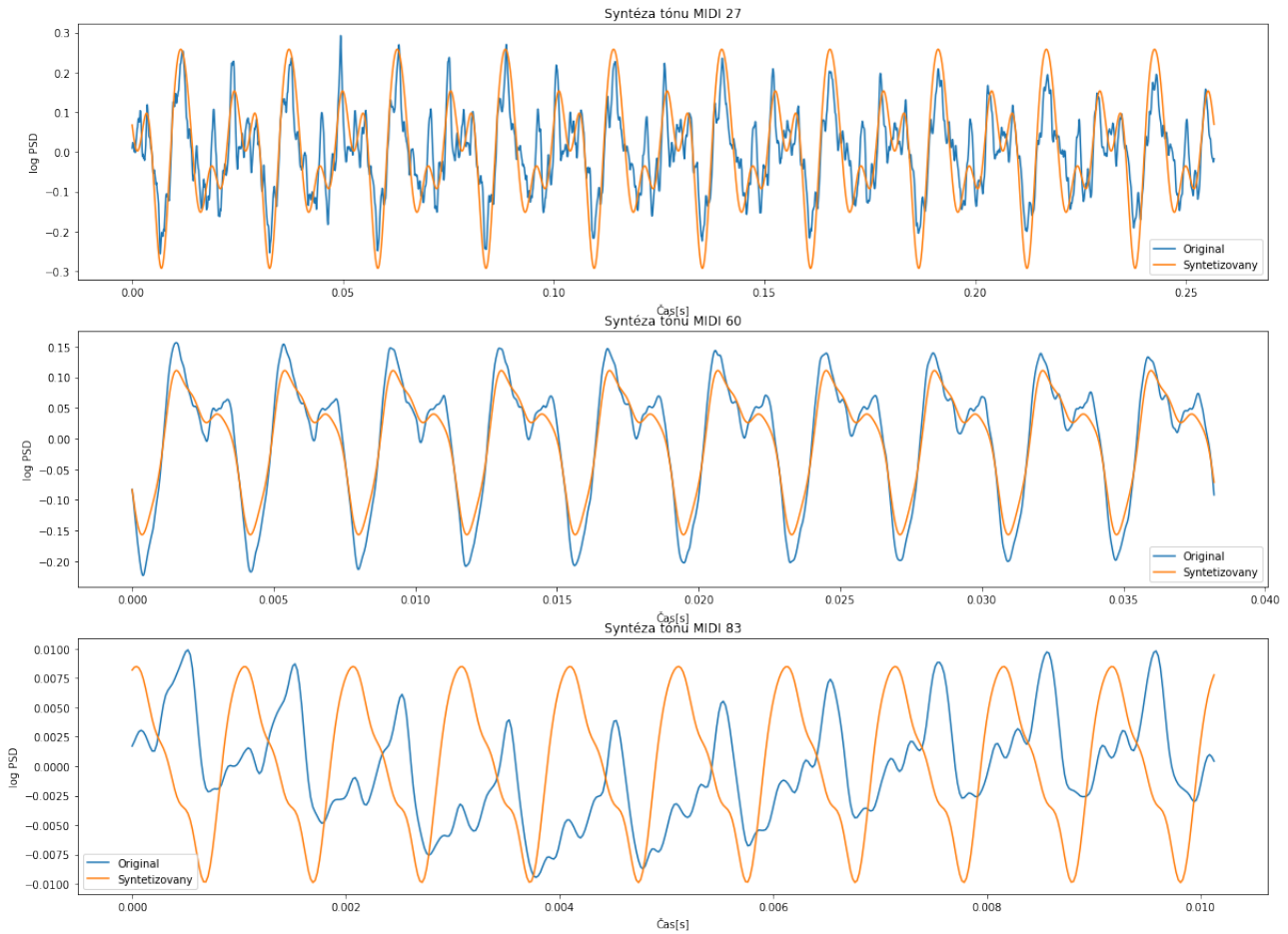
5 Úkol 4.5 - Syntéza tónu

Rekonstrukci tónu jsem provedl pomocí prostého sčítání cosinusovek. K výpočtu jsem použil tento vzorec:

$$y(t) = \sum_{k=1}^K r_k \cos(2\pi k f_0 t + \phi_k)$$

Kde r_k je modul koeficientu k , $k f_0$ je frekvence násobků odhadnuté základní frekvence a ϕ_k je fáze koeficientu.

Pokud nám signály nezačínají stejnou fází, je možné je synchronizovat pomocí korelace. Koreluje se originální signál s námi rekonstruovaným signálem, a následně zjistíme o kolik je tón posunutý od středu korelace. V mém případě jsem signály synchronizovat nemusel, protože mi vycházely zarovnané, snížila se mi zároveň i rychlost výpočtu syntézy signálu, protože korelace je výpočetně náročná. Syntetizovaný signál je třeba normalizovat a vynásobit amplitudou originálního signálu.



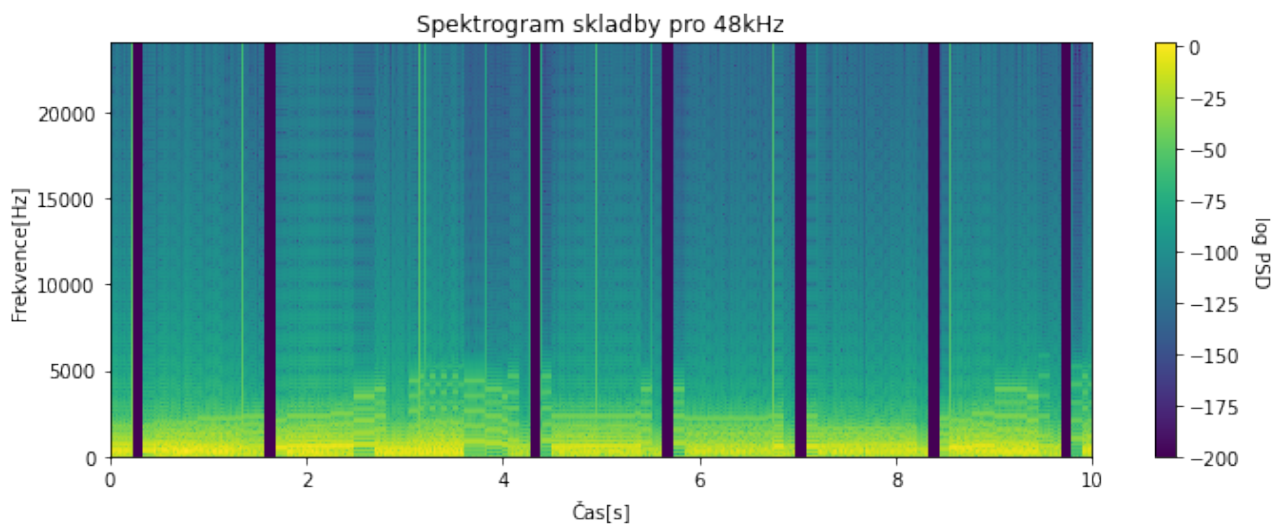
Obrázek 10: Srovnání syntézy tónů s původními signály

6 Úkol 4.6 - Generace hudby

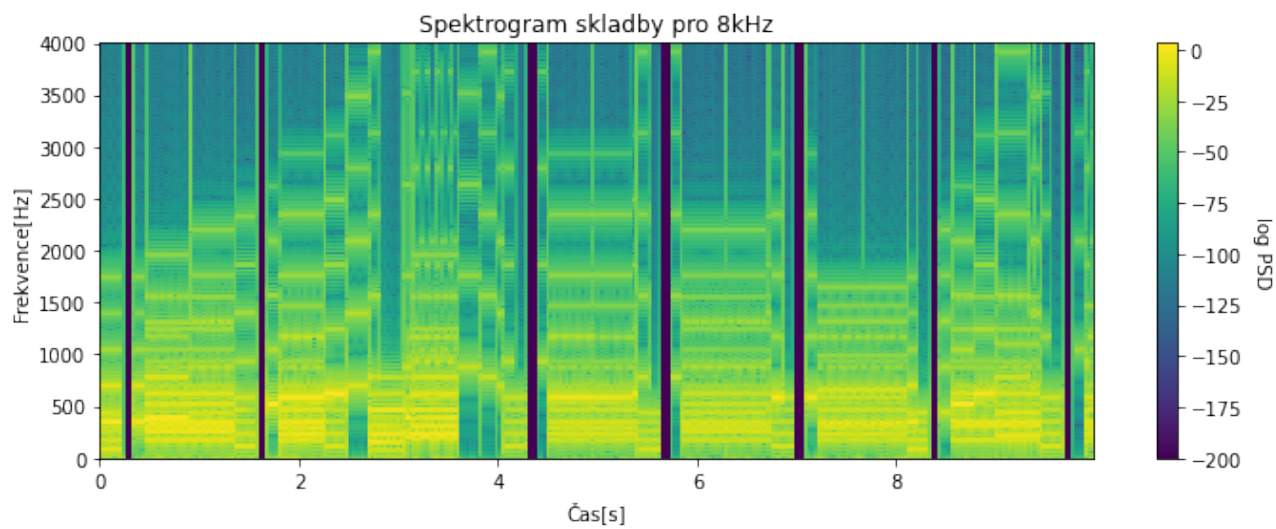
Nejdříve bylo potřeba zjistit délku nahrávky. To jsem provedl tak, že jsem si ze souboru s danou skladbou zjistil největší hodnotu ze sloupce `do [ms]`. Dále postupně parsujeme soubor se skladbou a syntetizujeme dané tóny ze sloupce `MIDI`. Syntetizovaný tón následně přičteme do výsledného zvuku na příslušné místo, které zjistíme hranicí od `[ms]` a do `[ms]`. Hlasitost řeším tak, že daný signál vynásobím hodnotou která je ve sloupci `hlasitost`. Potom signál uložím jako `.wav` soubor, s danou vzorkovací frekvencí. Před tím než signál uložím ho normalizuji. Při generování s vzorkovací frekvencí 8kHz musíme dát pozor, aby jsme negenerovali harmonické signály nad $\frac{F_s}{2}$. Ve výsledné skladbě můžeme slyšet lupání, které je způsobeno tzv. attackem. Ten se mi už nepodařilo zlepšit.

7 Úkol 4.7 - Spektrogram

Implementaci pro vykreslení spektrogramu jsem převzal z přednášky. U spektrogramů můžeme vidět že tóny se generují maximálně do frekvence $\frac{F_s}{2}$. Tmavě modré čáry vyjadřují čas, ve kterém nehraje žádný tón.



Obrázek 11: Spektrogram pro skladbu s vzorkovací frekvencí 48kHz



Obrázek 12: Spektrogram pro skladbu s vzorkovací frekvencí 8kHz