

Matematično-fizikalni praktikum

4. Fourierova Analiza

Tilen Šket, 28221057

Navodila:

Izračunaj Fourierov obrat Gaussove porazdelitve in nekaj enostavnih vzorcev, npr. mešanic izbranih frekvenc. Za slednje primerjaj rezultate, ko je vzorec v intervalu periodičen (izbrane frekvence so mnogokratniki osnovne frekvence), z rezultati, ko vzorec ni periodičen (kako naredimo Gaussovo porazdelitev 'periodično' za FT?). Opazuj pojav potujitve na vzorcu, ki vsebuje frekvence nad Nyquistovo frekvenco. Napravi še obratno transformacijo (2) in preveri natančnost metode. Poglej, kaj se dogaja z časom računanja, kako je odvisen od števila vzorčenj? Po Fourieru analiziraj 2,3 s dolge zapise začetka Bachove partite za violino solo, ki jih najdeš na spletni strani Matematičnofizikalnega praktikuma. Signal iz začetnih taktov partite je bil vzorčen pri 44 100 Hz, 11 025 Hz, 5512 Hz, 2756 Hz, 1378 Hz in 882 Hz. S poslušanjem zapisov v formatu `mp3` ugotovi, kaj se dogaja, ko se znižuje frekvenca vzorčenja, nato pa s Fourierovo analizo zapisov v formatu `txt` to tudi prikaži.

14. november 2024

Kazalo

1	Uvod	3
2	Algoritem za DFT	3
3	Uporaba DFT	4
3.1	Potujitev (<i>Aliasing</i>)	4
3.2	Gaussova krivulja	4
3.3	Analiza Bachove partite	6
4	Zaključek	8

1 Uvod

Fourierjeva transformacija je eden izmed najpomembnejših, če ne najpomembnejše, odkritje. Vendar v svoji navadni, zvezni, obliki izven listov polnih teorije nima tolikšnega pomena, saj signalov v praksi ne moremo meriti zvezno, ampak le diskretno. Obstaja pa tudi njegova diskretna enačica, ki se presnetljivo imenuje diskretna Fourierova transformacija (DFT).

Pri tej nalogi smo obravnavali uniformno vzorčenje na interval Δ . Diskretni verziji transformacije in njenega inverza, smo zapisali na naslednji način:

$$h_k = h(t_k), \quad t_k = k\Delta, \quad k = 0, 1, 2, \dots, N-1,$$

$$H_n = \sum_{k=0}^{N-1} h_k \exp(2\pi i k n / N), \quad n = -\frac{N}{2}, \dots, \frac{N}{2}, \quad (1)$$

$$h_k = \frac{1}{N} \sum_{n=0}^{N-1} H_n \exp(-2\pi i k n / N). \quad (2)$$

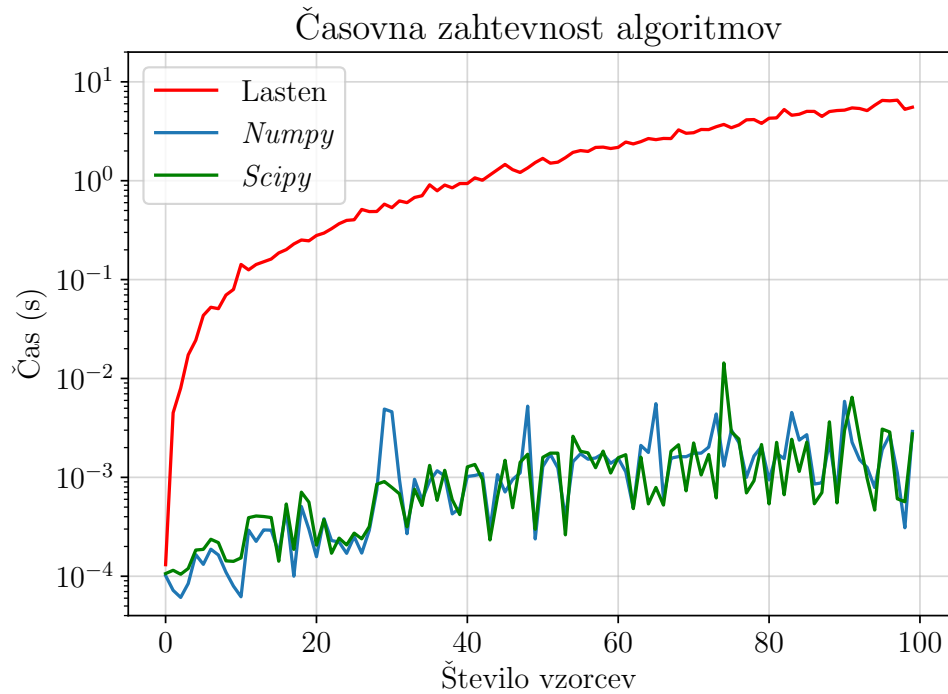
Ta algoritem množico točk iz časovne domene prenese v frekvenčno in če naš signal ne vsebuje frekvenc višjih od Nyquistove, to je $f_{\text{Nyquist}} = \frac{1}{2\Delta}$, potem s tako transformacijo nismo izgubili nič informacije in lahko po inverzni transformaciji rekonstruiramo originalni signal. Če pa naš signal vsebuje tudi višje frekvence, potem dobimo pojav potujitve (*Aliasing*).

Torej imamo diskretno verzijo transformacije, ki nam iz časovne domene slika v frekvenčno na območje $[-f_{\text{Nyquist}}, f_{\text{Nyquist}}]$ in inverzno transformacijo, ki nam omogoča rekonstrukcijo originalnih signalov.

2 Algoritem za DFT

Za boljše razumevanje samega delovanja DFT, sem se sam lotil kodiranja algoritma in njegovega inverza po enačbah 1 in 2. Časovna zahtevnost mojega algoritma je $O(N^2)$, torej ne spada v rang FFT algoritmov, kateri imajo časovno zahtevnost $O(N \ln N)$. Čeprav teoretično poznam obe časovni zahtevnosti, sem se lotil le te tudi preveriti. In sicer sem primerjal svoj DFT algoritem s FFT algoritmoma iz Python knjižnic *Numpy* in *Scipy*. Primerjal sem jih na vsoti naključnih sinusnih signalov. Rezultate sem izrisal na graf na sliki 1.

Od tukaj dalje sem uporabljal algoritem iz knjižnjice *Numpy*.



Slika 1: Graf časovne zahtevnosti različnih algoritmov DFT. Primerjan je moj lasten, naiven, algoritem z algoritmoma iz knjižnic *Scipy* in *Numpy*.

3 Uporaba DFT

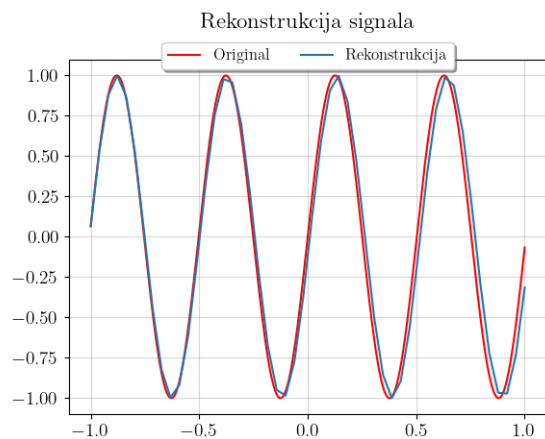
3.1 Potujitev (*Aliasing*)

Najprej sem vzel sinusni signal z eno frekvenco in se zanimal, kaj se dogaja pri različnih velikostih te frekvence glede na Nyquistovo frekvenco. Vzel sem dva primera in sicer enega s polovično frekvenco Nyquistove in drugega s trikratno. Odločitve sem sprejel na podlagi tega, da se potujitev dobro vidi grafično. Rekonstrukciji in originalna signala so prikazani na slikah 2a in 2b.

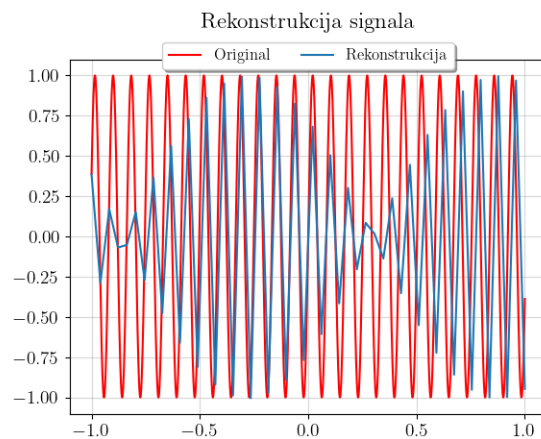
Poglejmo si še primer potujitve za signal, ki je vsota večih sinusoid. Primer je prikazan na slikah 3a in 3b. V obeh primerih sem uporabil isti signal in le spremenil frekvenco vzorčenja in opazimo, pojav potujenja, ko je frekvenca premajhna.

3.2 Gaussova krivulja

Nato sem algoritem uporabil na Gaussovi krivulji, kakor je bilo predvideno v navodilih. Po teoriji vemo, da se Gaussova krivulja po Fourierovi transformaciji transformira v Gaussovo

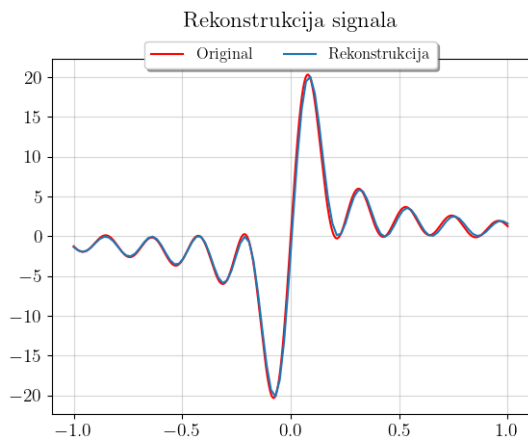


(a) Frekvenca nižja od Nyquistove.

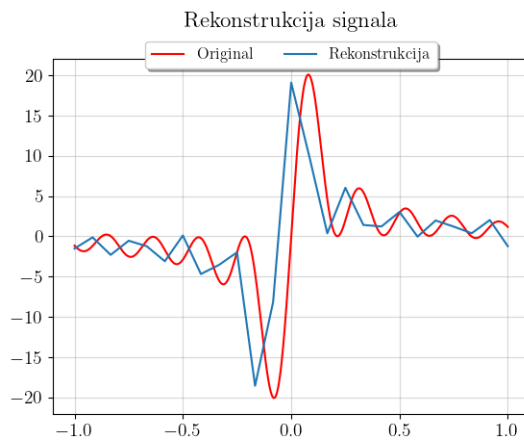


(b) Frekvenca višja od Nyquistove.

Slika 2: Slika rekonstrukcije dveh sinusoid, kjer ima ena frekvenco manjšo in druga večjo od Nyquistove. Opazimo pojav potujitve, kjer vsebuje rekonstrukcija nižje frekvence, ki jih originalni signal ne.



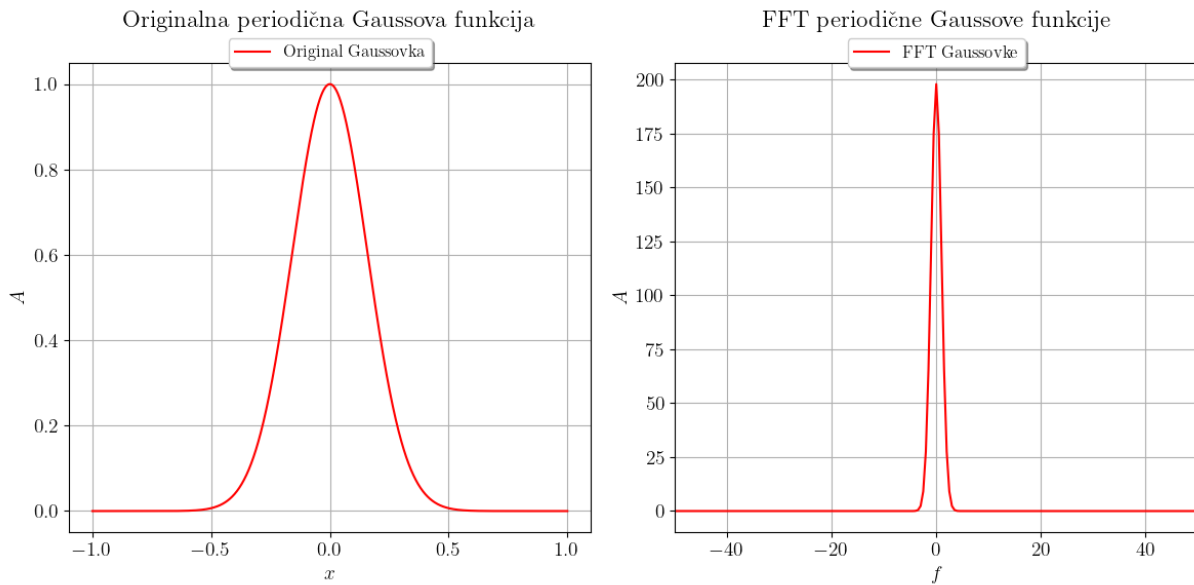
(a) Frekvence nižja od Nyquistove.



(b) Frekvence višja od Nyquistove.

Slika 3: Sliki rekonstrukcije signala, ki je vsota sinusoid. Razlika v obeh primerih je le v Nyquistovi frekvenci, kjer je enkrat nižja in enkrat višja od vseh frekvenc v vsoti. Ponovno opazimo pojav potujenja.

krivuljo v drugi domeni. To vidimo tudi na sliki 4, kjer sta izrisani Gaussova krivulja in njena transformiranka. Gaussovo krivuljo lahko naredimo tudi periodično, kakor vidimo na sliki 5.

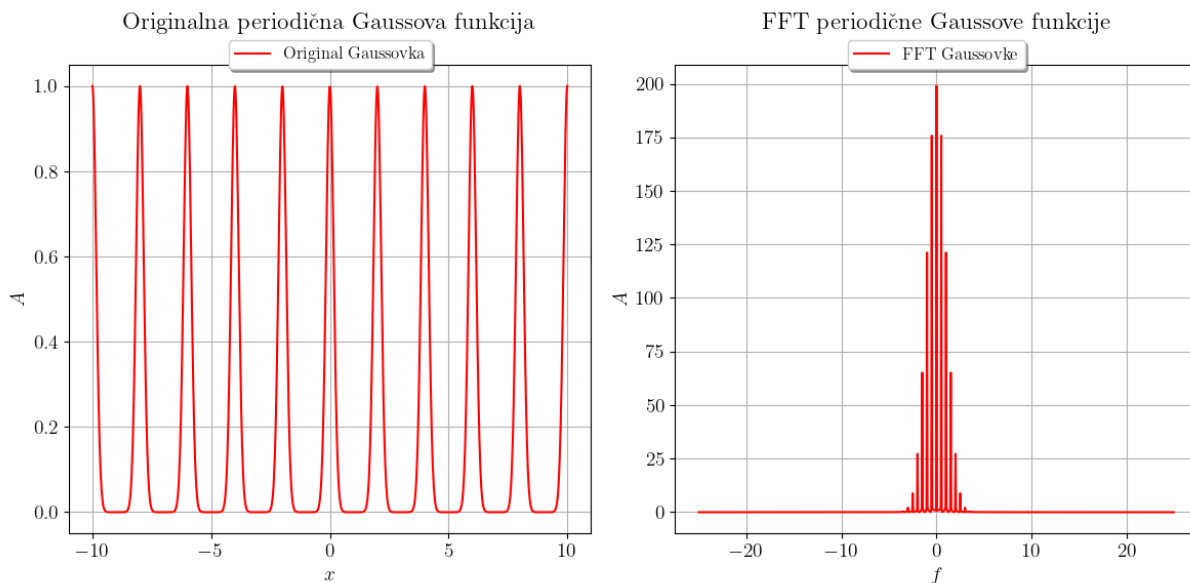


Slika 4: Grafa Gaussove krivulje in njene Fourierove transformiranke.

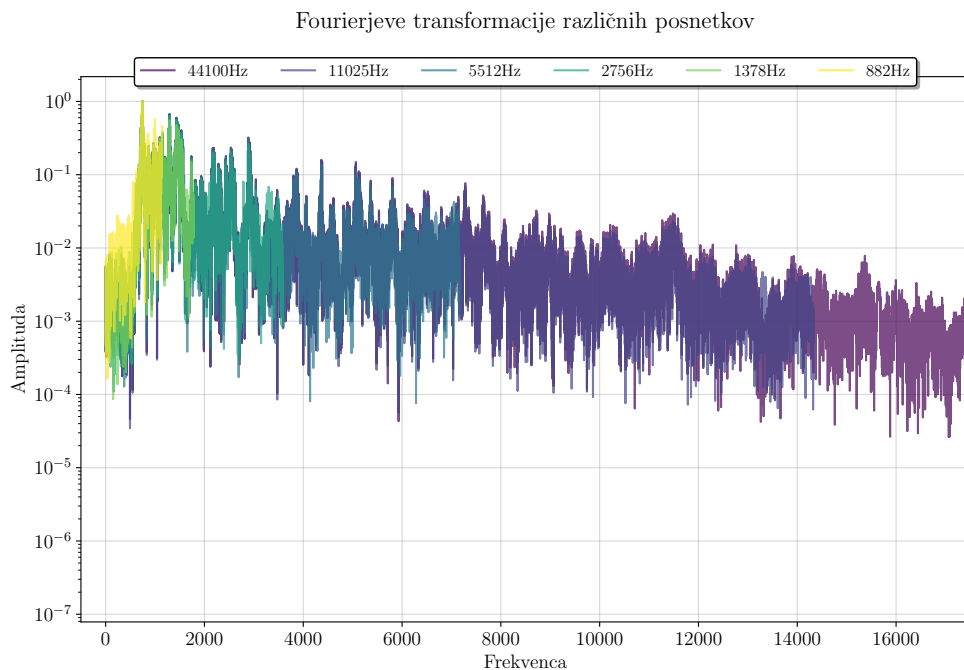
3.3 Analiza Bachove partite

Na spletni učilnici predmeta, smo dobili 6 posnetkov začetka Bachove partite za violino solo. Signal je v teh posnetkih bil vzorčen pri različnih frekvencah, in sicer pri 44 100 Hz, 11 025 Hz, 5512 Hz, 2756 Hz, 1378 Hz in 882 Hz. Pri poslušanju zveni vsaka višja stopnja vzorčenja bolj ostra oz. definirana.

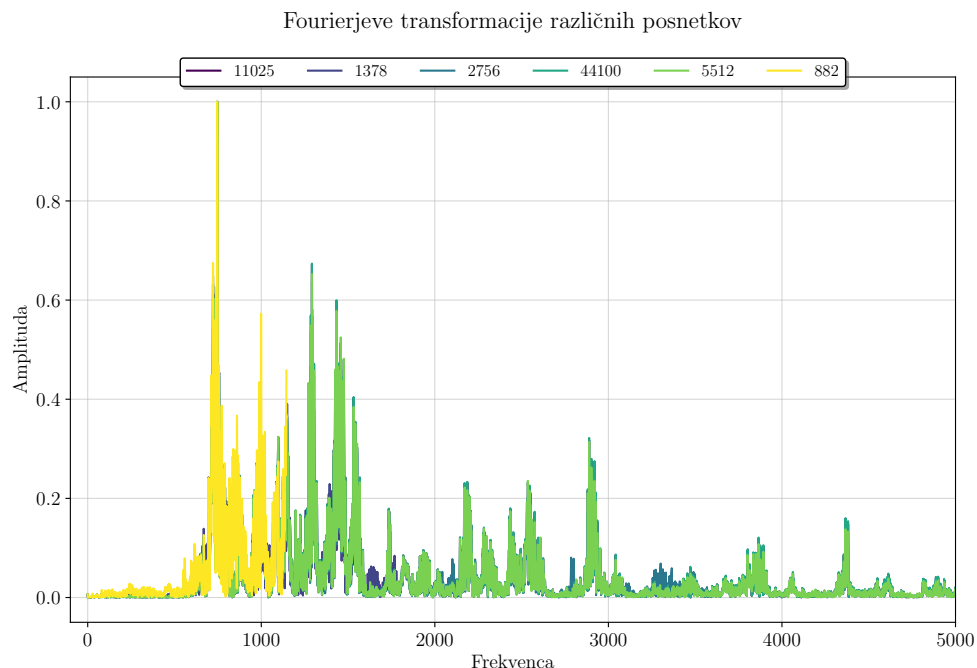
Podatke, ki sem jih dobil, sem izrisal na logaritemski skali, na sliki 6 in na linearno skali na sliki 7. Opazimo lahko, kako vsaka višja frekvenca vzorčenja doda večje frekvenčno območje. Vendar pa se vse meje med frekvencami znotraj slušnega območja, kar lahko potrdim s tem, da sem med vsakim parom posnetkov slišal občutno razliko.



Slika 5: Grafa periodične Gaussove krivulje in njene Fourierove transformiranke.



Slika 6: Grafi aplitud v odvisnosti od frekvenc izrisani eden na drugem. Hitro vidimo, da z rastjo frekvence vzorčenja, dobimo večje frekvenčno območje.



Slika 7: Grafi aplitud v odvisnosti od frekvenc izrisani eden na drugem. Hitro vidimo, da z rastjo frekvence vzorčenja, dobimo večje frekvenčno območje.

4 Zaključek

Ob tej nalogi sem zaradi želje po globljem razumevanju Fourierove transformacije in še posebej FFT algoritmov, ki so po besedah Derek-a Muller-ja, lastnika Youtube kanala Veritasium, najpomembnejši algoritmi, kar nekaj časa na začetku sedel ob navodilih s svojo tablico in odprto Wikipedijo in ostalimi viri in si poskusil obrazložiti delovanje algoritma na grafične in drugačne načine. Menim, da sedaj, ko pišem zaključek, te algoritme razumem veliko bolje kot prej, vendar mi manjka še kar nekaj intuicije in predvsem izkušenj.

Rad bi še parafraziral izjavo, ki sem jo prebral v lanskem komentarju k tej nalogi in mi je skozi celotno reševanje ždela v mislih. In sicer je eden izmed študentov zapisal, da smo dobili orodje s katerim lahko začnemo 'tolč' po vsaki obliki podatkov, kar mislim, da je za FFT odlična izjava.

Še nekaj časovnih statistik, kot vedno. Za celotno nalogo sem porabil nekje 10h, kar je nekoliko več kot za zadnji dve. Se je pa od zadnjih dveh obdržala stalnica, da pišem zaključek že kar krepko v četrtek, že nekaj ciklov zaspanosti kasneje, kot bi si to želel priznati.