

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

SEMINAR

Filteri s konačnim impulsnim odzivom

Fran Jelavić

Voditelj: *Ivan Đurek*

Zagreb, siječanj 2023.

SADRŽAJ

1. Uvod	1
2. FIR filter	2
2.1. Prednosti	2
2.2. Nedostatci	2
3. Pojasno-propusni filter	4
3.1. Vizualizacija	4
4. Implementacija	8
4.1. Zašto Python?	8
4.2. Zašto ne Python?	8
4.3. Kod	9
5. Zaključak	10
6. Literatura	11

1. Uvod

U obradi signala, filter s konačnim impulsnim odzivom (eng. *Finite Impulse Response* - FIR) vrsta je filtera korištena u širokom rasponu aplikacija, kao što su audio i video obrada, telekomunikacije i sustavi upravljanja.

Jedna specifična primjena FIR filtera je pojasno-propusno (eng. *band-pass*) filtriranje, koje se koristi za izolaciju određenog raspona frekvencija od signala.¹ U ovom seminaru istražiti ćemo upotrebu FIR filtera za pojasno-propusno filtriranje i pokazati kako se ovi filteri mogu dizajnirati i implementirati u Pythonu. Također ćemo raspravljati o karakteristikama pojasno-propusnih filtera i kako oni utječu na izlazni signal, te pokazati kako se izlazni signal može vizualizirati u vremenskoj i frekvencijskoj domeni.

Konačno, predstaviti ćemo primjer kako se pojasno-propusni filter može koristiti za izolaciju određenog raspona frekvencija od signala. Također ćemo raspravljati o nedostacima i ograničenjima korištenja pojasno-propusnog filtera.

Cilj ovog seminara je omogućiti razumijevanje principa i implementacije FIR pojasno-propusnih filtera te pokazati kako se oni mogu koristiti za analizu i obradu signala u različitim primjenama.

2. FIR filter

Filter s konačnim impulsnim odzivom (eng. *Finite Impulse Response* - FIR) vrsta je filtera čiji je impulsni odziv konačnog trajanja, jer se u konačnom vremenu postavlja na nulu.² To znači da će izlaz filtera biti nula nakon konačnog broja uzoraka. FIR filteri su nerekurzivni, što znači da je trenutna izlazna vrijednost određena isključivo trenutnim i prošlim ulaznim vrijednostima.³ Drugim riječima, izlaz FIR filtera u bilo kojem trenutku određen je ponderiranim zbrojem trenutnih i prošlih ulaza, gdje su težine koeficijenti filtera.

2.1. Prednosti

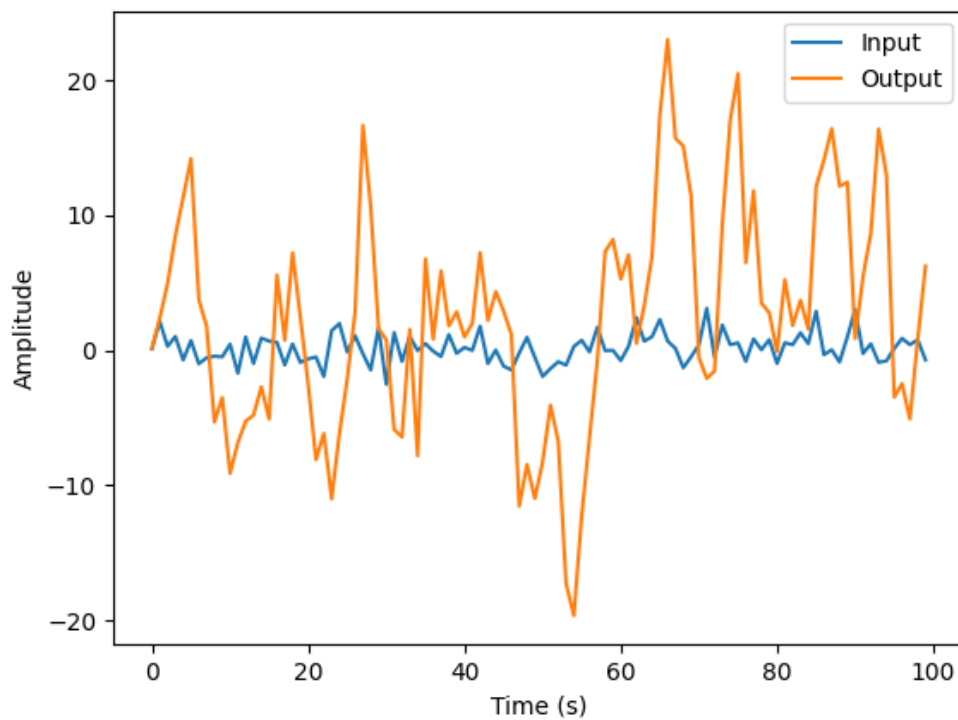
Nekoliko je prednosti korištenja FIR filtera. Jedna od glavnih prednosti je njihova linearna fazna karakteristika, što znači da je fazni odziv filtera linearan u odnosu na frekvenciju.⁴ To znači da je kašnjenje filtera konstantno na svim frekvencijama, što je važno u primjenama gdje kašnjenje signala treba minimizirati. Dodatno, FIR filteri su također stabilni i imaju predvidljiv odziv, što ih čini korisnim u sustavima upravljanja i drugim aplikacijama gdje je stabilnost važna.⁵

Još jedna prednost korištenja FIR filtera je njihova fleksibilnost. Mogu se odabirom odgovarajućih koeficijenata filtera dizajnirati tako da imaju širok raspon frekvencijskih odziva, kao što su niskopropusni (eng. *low-pass*), visokopropusni (eng. *high-pass*), pojasno-propusni (eng. *band-pass*) i pojasno-nepropusni (eng. *band-stop*) filteri. Osim toga, broj koeficijenata (tzv. eng. *taps*) u FIR filteru može se prilagoditi kako bi se postigla željena razina frekvencijske selektivnosti.⁶

2.2. Nedostatci

Međutim, postoje i neki nedostaci korištenja FIR filtera. Jedan od glavnih nedostataka je njihova računalna složenost jer zahtijevaju veliki broj množenja i zbrajanja. To ih može učiniti manje prikladnima za aplikacije u stvarnom vremenu gdje su računalni

resursi ograničeni.⁵ Dodatno, njihov impulsni odziv ograničenog je trajanja, što znači da im je odziv početnih uvjeta različit od nule. Drugim riječima, može doći do izlaza različitih od nule čak i kad je ulaz jednak nuli.³



Slika 2.1: Signal procesiran jednostavnim FIR filterom

3. Pojasno-propusni filter

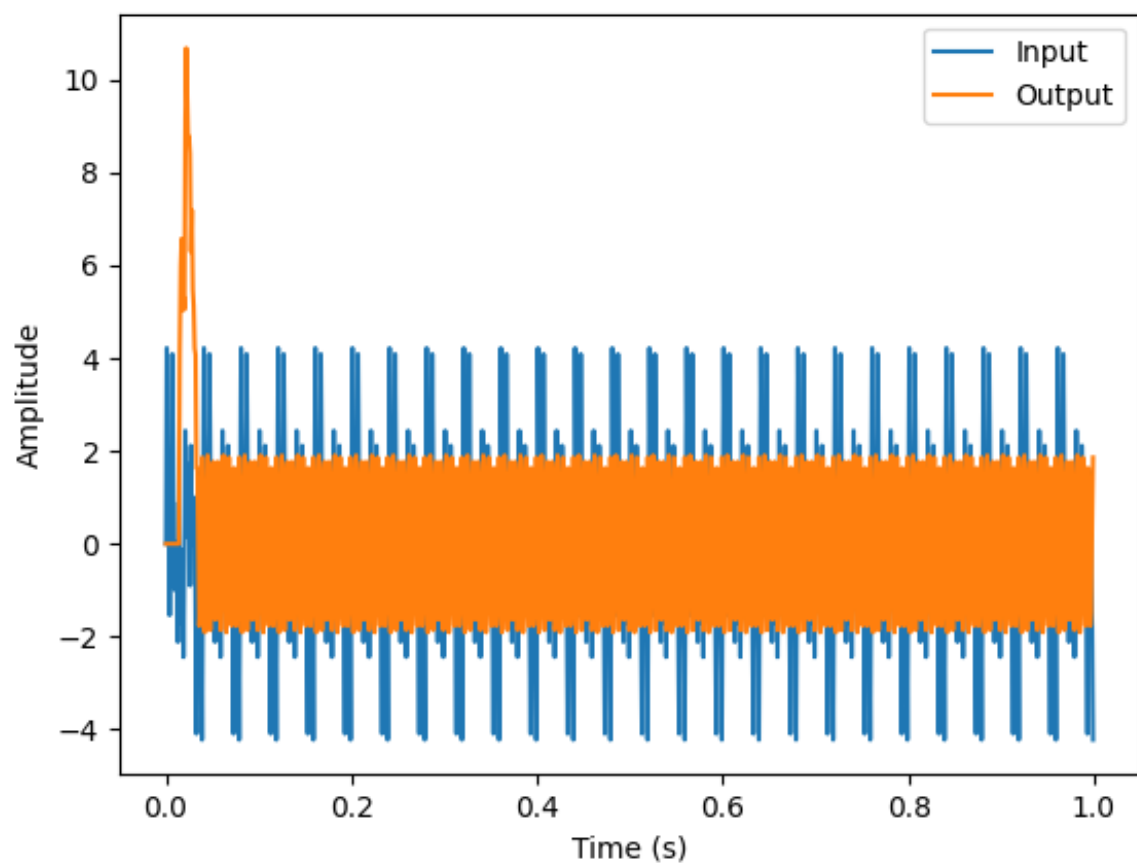
Pojasno-propusni (eng. *band-pass*) filteri koriste se za izolaciju određenog raspona frekvencija od signala, izostavljajući frekvencije izvan tog raspona.⁷

Koeficijenti filtera i broj odvoda mogu se odabrati kako bi se postigla željena razina frekvencijske selektivnosti. Također je potrebno pažljivo odabrati granične frekvencije, koje određuju raspon frekvencija koje će filter propustiti ili izostaviti. Dodatno, metoda dizajna filtera, kao što je metoda prozora (eng. *window method*), također može utjecati na performanse filtera.⁸

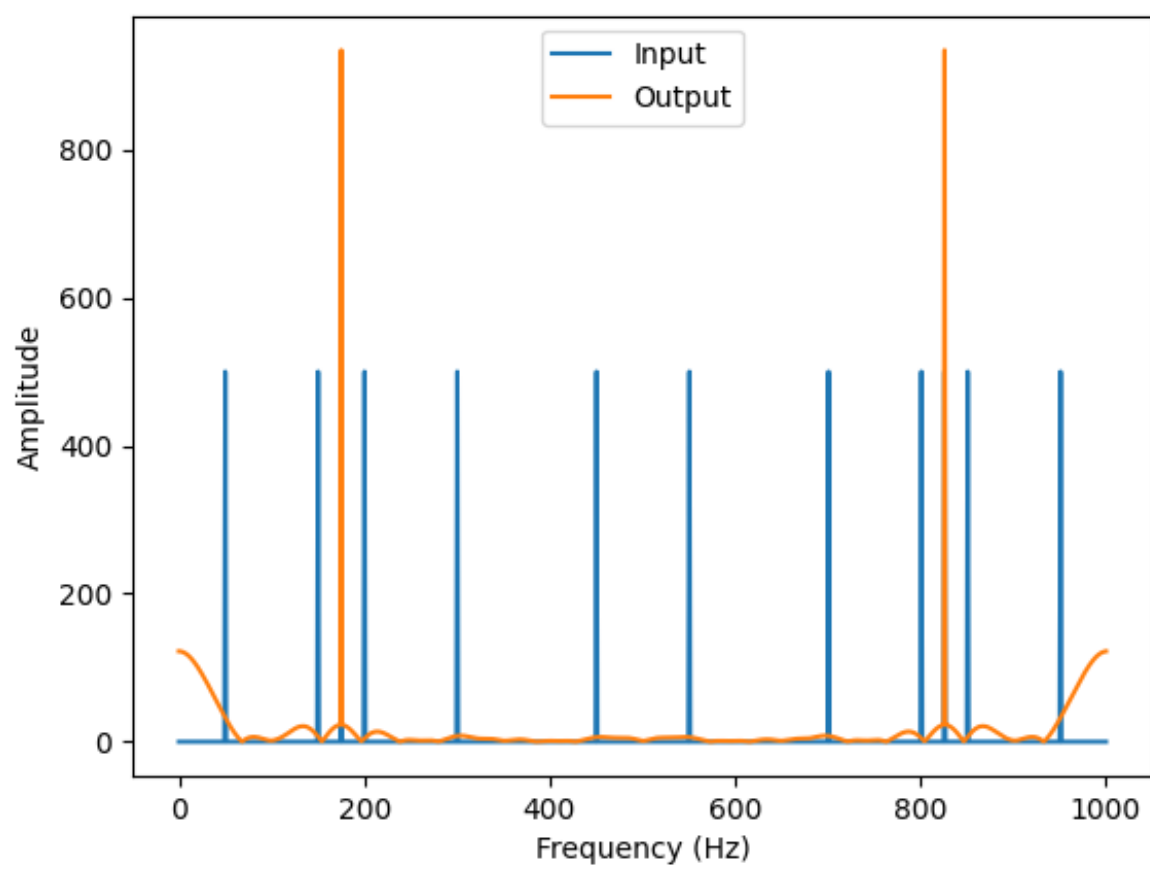
3.1. Vizualizacija

Kada vizualiziramo izlazni signal, važno je iscrtati vremensku i frekvencijsku domenu kako bismo vidjeli kako filter utječe na signal. U vremenskoj domeni možemo vidjeti amplitudu signala tijekom vremena, dok u frekvencijskoj domeni možemo vidjeti kako filter utječe na različite frekvencijske komponente signala.

U ovom seminaru predstaviti ćemo primjer kako se pojasno-propusni filter može koristiti za izolaciju određenog raspona frekvencija od signala. Ulazni signal sačinjen je od nekoliko sinusoida s različitim frekvencijama, a filter je dizajniran da propušta samo frekvencije između 100 i 300 Hz. Vidimo da izlazni signal sadrži pojačane frekvencije koje su unutar ovog raspona, dok su ostale frekvencije utišane.



Slika 3.1: Signal procesiran jednostavim pojasno-propusnim filterom u vremenskoj domeni



Slika 3.2: Signal procesiran jednostavim pojasno-propusnim filterom u frekvencijskoj domeni

U frekvencijskoj domeni možemo vidjeti kako filter utječe na različite frekvencijske komponente signala. U primjeru koji smo koristili, ulazni signal sadrži šest sinusoida s različitim frekvencijama, a to su 50 Hz, 150 Hz, 175 Hz, 200 Hz, 300 Hz i 450 Hz, s amplitudom 2. Pojasni filter je dizajniran da propušta samo frekvencije između 100 i 300 Hz.

Grafikon frekvencijske domene pokazuje da ulazni signal sadrži mnogo frekvencija, ali izlazni signal sadrži samo frekvencije koje su unutar raspona od 100-300Hz. Vidimo da izlazni signal ima veliki skok u amplitudi oko 175Hz, što je jedna od ulaznih frekvencija. To je zato što je frekvencija od 175 Hz u sredini frekvencija (150 Hz i 200 Hz) unutar propusnog pojasa i stoga se najviše pojačava.

Uzlazni nagibi vidljivi na stranama grafa frekvencijskog spektra poznati su kao "spektralno curenje" (eng. *spectral leakage*) te su rezultat efekta prozora koji se javlja prilikom izvođenja brze Fourierove transformacije (eng. *Fast Fourier Transform - FFT*) na signalu konačne duljine.³

Također je vrijedno napomenuti da se dijagram frekvencijskog spektra zrcali oko x-osi. To je zato što pravi signal ima Hermitovu simetriju te su negativne frekvencije samo odraz pozitivnih frekvencija.⁶ Ovaj graf prikazuje pomak signala udesno radi jasnijeg prikaza.

4. Implementacija

4.1. Zašto Python?

U ovom seminaru odlučili smo implementirati i predstaviti FIR pojasno-propusni filter koristeći programski jezik Python. Jedan od glavnih razloga za ovaj izbor je jednostavnost korištenja biblioteke Matplotlib, koja omogućuje stvaranje jasnih i informativnih vizualizacija ulaznih i izlaznih signala u vremenskoj i frekvencijskoj domeni. Ova biblioteka pruža jednostavno i intuitivno sučelje za iscrtavanje i oblikovanje grafikona, što je čini idealnim izborom za demonstracijske i prezentacijske svrhe.

4.2. Zašto ne Python?

Vrijedno je napomenuti da je za zadatke digitalne obrade signala (eng. *Digital Signal Processing* - DSP) C++ učinkovitija i brža opcija od Pythona. Međutim, u ovom slučaju, glavni cilj je brzo implementirati i predstaviti dokaz koncepta FIR pojasno-propusnog filtera, stoga je jednostavnost korištenja Matplotlib biblioteke i drugih korisnih biblioteka Python učinila boljim izborom. Sposobnost brzog iscrtavanja ulaznih i izlaznih signala na jasan i informativan način presudna je za razumijevanje ponašanja filtera i kako on utječe na signal.

4.3. Kod

```
def fir_filter(taps, input_signal):
    """
    Apply a FIR filter with coefficients b to input signal.
    """
    # Create an empty output signal array
    output_signal = np.zeros(len(input_signal))
    # Loop through each sample of the input signal
    for n in range(len(input_signal)):
        # Loop through each coefficient of the filter
        for k in range(len(taps)):
            if n-k < 0:
                continue
            # Perform the convolution sum
            output_signal[n] += taps[k] * input_signal[n-k]
    return output_signal

def band_pass_fir(lowcut, highcut, fs, numtaps, input_signal):
    """
    Design and apply a band-pass FIR filter to a signal.
    """
    # Design the filter
    nyquist = 0.5 * fs
    low = lowcut / nyquist
    high = highcut / nyquist
    # Create an array of taps filled with zeros
    taps = np.zeros(numtaps)
    # Calculate the center of the taps
    center = (numtaps - 1) / 2
    # Fill the center of taps with desired cutoff frequencies
    for i in range(int(center - (high - low) * center), \
        int(center + (high - low) * center)):
        taps[i] = 1
    # Apply the filter to the input signal
    filtered_signal = fir_filter(taps, input_signal)
    return filtered_signal
```

5. Zaključak

Filteri s konačnim impulsnim odzivom (FIR), posebno propusno-pojasni filteri, koristan su alat u digitalnoj obradi signala. Karakterizira ih njihov konačni impulsni odziv, linearni fazni odziv i sposobnost propuštanja te izostavljanja frekvencijskih pojasa. U dizajnu ovih filtera obično se koristi metoda prozora. Također je predstavljen izlaz *band-pass* filtera pomoću grafikona vremenske i frekvencijske domene. Napomenuto je razumijevanje karakteristika filtera za optimizaciju performansi. Dok je Python korišten u ovoj implementaciji zbog svoje jednostavnosti upotrebe crtanja grafikona, vrijedno je napomenuti da je C++ prikladniji za digitalnu obradu signala zbog svoje izvedbe. Općenito, FIR filteri imaju mnoge praktične primjene i njihovo dobro razumijevanje je važno u digitalnoj obradi signala.

6. Literatura

- ¹ Steven W. Smith. *Digital Signal Processing: A Practical Guide for Engineers and Scientists*. Pearson, 2010.
- ² Julius O. Smith. *Introduction to Digital Filters with Audio Applications*. W3K Publishing, 2003.
- ³ Li Tan and Jean Jiang. *Digital Signal Processing: Fundamentals and Applications*. CRC Press, 2010.
- ⁴ Emmanuel C Ifeachor and Barrie W Jervis. *Digital signal processing: a practical approach*. Pearson Education, 2002.
- ⁵ John R. Buck. *Digital Signal Processing: A Review Journal*. John R. Buck.
- ⁶ Jonathan Y Stein and Martin Vetterli. *Digital Signal Processing: A Computer Science Perspective*. Princeton University Press, 2011.
- ⁷ Sanjit K. Mitra. *Digital Signal Processing: A Computer-Based Approach*. McGraw-Hill Education, 2018.
- ⁸ John G Proakis and Dimitris K Manolakis. *Digital signal processing: principles, algorithms, and applications*. Pearson Education, 2014.

Filteri s konačnim impulsnim odzivom

Sažetak

Ovaj seminar raspravlja o korištenju filtera konačnog impulsnog odziva (FIR), posebno pojasno-propusnih filtera, u digitalnoj obradi signala. Ispituje kako dizajnirati te filtre i kako interpretirati izlaz u grafovima vremenske i frekvencijske domene. Također napominje važnost razumijevanja karakteristika filtera za optimizaciju performansi i korištenje pythona radi demonstracijskih svrha.

Ključne riječi: signali, FIR, pojasni propust, DSP, Python

Finite Impulse Response Filters

Abstract

This seminar discusses the use of Finite Impulse Response (FIR) filters, specifically band-pass filters, in digital signal processing. It examines how to design these filters and how to interpret the output in time and frequency domain graphs. It also notes the importance of understanding the filter's characteristics for performance optimization and the use of python for demonstration purposes.

Keywords: signals, FIR, band-pass, DSP, Python