

8. Naloga: Izdelava grafičnega uporabniškega vmesnika v okolju MATLAB za poljubno filtriranje EEG signalov s KEO spektralnimi filtri

Janez Božič

January 23, 2023

1 Uvod

Pri posnetkih merjenj aktivnosti možganov, lahko kmalu opazimo, da signali niso samo ravne črte, ki se ukrivijo samo pri aktivnosti točno določenega dela možganov. Kot pri večini ostalih merjenj, tudi EEG posnetki niso odporni na šum. Posledično, želimo pred nadaljno obdelavo in uporabo posnetkov, šum odstraniti. To storimo s filtri. Poznamo več vrst filtrov, vendar se bomo v tem delu osredotočili na spektralne filtre s končnim enotnim odzivom (KEO). V tem delu predlagamo grafično orodje, s katerim lahko filtriramo EEG signale z nizkoprepustnim, visokoprepustnim, pasovno prepustnim in pasovno zapornim filtrom. Za napredne uporabnike pa orodje omogoča tudi lastno definicijo koeficientov a in b , katera definirata filter. S tem je orodje prilagodljivo tudi za bolj napredne in zahtevne filtre od osnovnih KEO filtrov.

Kaj nam delo prinaša:

- Orodje za filtriranje EEG signalov s KEO filtri in osnovno podporo za filtriranje z NEO filtri
- Navodila za uporabo grafičnega vmesnika
- Primeri uporabe in rezultatov uporabe orodja

2 Metode

2.1 Filtri s Končnim Enotnim Odzivom (KEO)

2.1.1 Spektralni filtri

Naše orodje smo izdelali na osnovi KEO filtrov. KEO filtri (*ang. Finite impulse response (FIR) filters*) so del družine spektralnih filtrov [2]. Spektralni filtri transformirajo večkanalni signal $X(n)$ tako, da je vsak signal $y_i(n)$ v $Y(n)$ odvisen le od signala $x_i(n)$. V osnovi ločimo dva tipa spektralnih filtrov:

- Filtri z Neskončnim Enotnim Odzivom (NEO)
- Filtri s Filtri s Končnim Enotnim Odzivom (KEO)

V tem delu nas zanimajo digitalni filtri, ki so definirani z Linearno Diferenčno Enačbo s Konstantnimi Koeficienti (LDEKK). Sledijo enačbi:

$$y(n) = \sum_{k=1}^K a_k y(n-k) + \sum_{m=0}^M b_m x(n-m)$$

Kar loči NEO in KEO filtre je koeficient a , saj predstavlja rekurziven člen filtra. NEO filtri so rekurzivni filtri, impulzni odziv funkcije $h(n)$ na enotni vzorec $\delta(n)$ je neskončne dolžine, saj vsaj en koeficient $a_k \neq 0$. Za njih velja zgornja LDEKK enačba v celoti.

2.1.2 KEO filtri

KEO filtri so nerekurzivni filtri, ki imajo rekurziven člen (vsak koeficient a) enak 0 ($a_k = 0$). Pri KEO filtrih je impulzni odziv funkcije $h(n)$ na enotin vzorec $\delta(n)$ končne dolžine in za njih LDEKK enačba ne velja v celoti, saj lahko pokrajšamo celoten rekurziven del enačbe. Prilagojene enačba je tako:

$$y(n) = \sum_{m=0}^M b_m x(n-m)$$

Posledica okrajšave rekurzivnega dela formule je, da je odziv KEO filtra ($h(k)$) na enotin vzorec $\delta(n)$ določen s koeficientom b_m . Torej:

$$h(n) = b_n, \text{ če je } 0 \leq n \leq M \\ h(n) = 0, \text{ drugače}$$

Izvedemo lahko konvolucijo signala, kjer lahko enačbo za KEO filtre zapišemo:

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) = x(n) \times h(n)$$

Konvolucija signalov je pomembna za Fourierjevo transformacijo in tudi implementacijo filtra z uporabo konvolucijske vsote.

2.1.3 Lastnosti KEO filtrov

Iz izpeljave enačbe za KEO filtre, ki ne vsebuje rekurzivnega člena, lahko izrazimo njihove lastnosti:

- Z njimi lahko načrtamo vsak linearni časovno invariantni filter
- Impulzni odziv, $h(n)$, je končne dolžine in je definiran preprosto, kar s koeficienti b_m
- Filter načrtamo s primerno izbiro koeficientov b_m (visoko število potrebnih koeficientov)
- So vedno stabilni (nerekurzivni, ni povratne vezave)
- Faza (fazni odziv) je linearna (vse frekvence so na izhodu zakasnjene za isti čas), če je impulzni odziv simetričen ali antisimetričen
- Implementacija filtra z uporabo konvolucijske vsote je možna, $h(n) = b_n$, $n = 0, 1, 2, \dots, M$

2.1.4 Delovanje KEO filtrov

V delu smo implementirali:

- **Nizkoprepustni filter**
Filter prepušča samo nizke frekvence, nižje do frekvence, ki jo podamo.
- **Visokoprepustni filter**
Filter prepušča samo visoke frekvence, višje od frekvence, ki jo podamo.
- **Pasvno prepustni filter**
Filter prepušča samo frekvence med podanima mejnima frekvencama.
- **Pasovno zaporni filter**
Filter duši frekvence med podanima frekvencama (prepušča vse ostale, razen frekvenc na podanem intervalu).

Grafični prikaz delovanja filtrov si lahko ogledamo na Sliki 1.

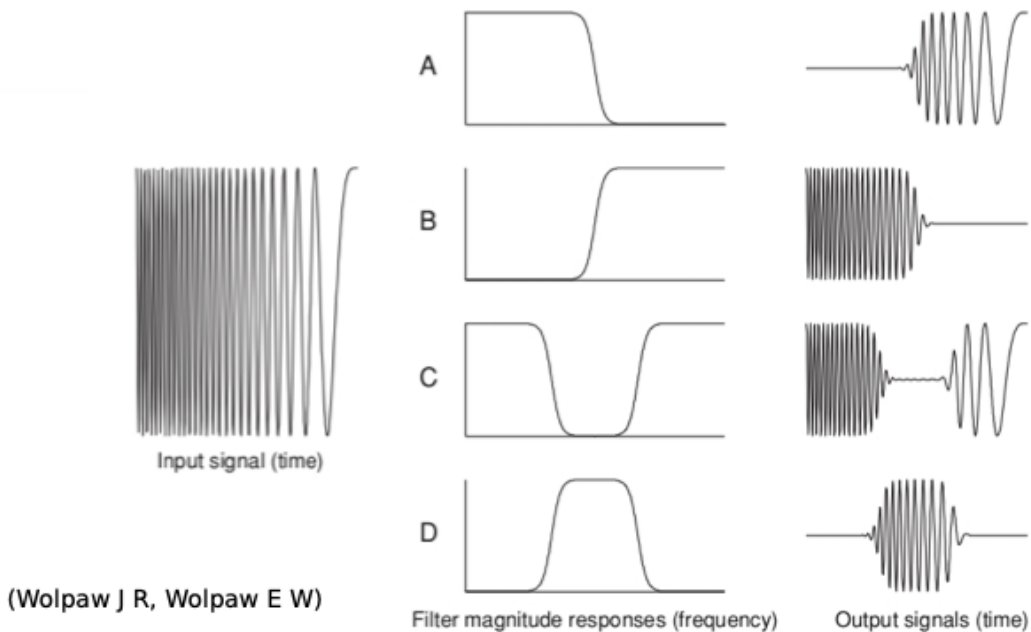


Figure 1: Primeri delovanja filtrov

2.2 Podatki

Podatke možganskih signalov smo pridobili iz podatkovne baze EEGMMI DS [1]. Podatkovna baza je sestavljena iz več kot 1500 EEG posnetkov dolžine 1-2 minuti, ki so bili posneti na 109 prostovoljnih. Vsak posnetek prikazuje delovanje možganov med izvajanjem ene izmed štirih nalog (stiskanje leve/desne pesti, zamišljanje stiskanja leve/desne pesti ali stopal, stiskanje obeh pesti in zamišljanje stiskanja obeh pesti ali stopal) ali mirovanja z bodisi odprtimi ali zaprtimi očmi.

Podatki so v EDF+ formatu, vsak posnetek vsebuje 64 signalov, vsak je posnet s 160 posnetki na sekundo, in 1 enotacijski kanal. Anotacije so v ločeni datoteki, katerim mi nismo posvečali večje pozornosti.

2.3 Implementacija

2.3.1 Uporaba orodja

Naša implementacija je narejena v programskem okolju MATLAB (v App Designer orodju). Aplikacija (na Sliki 2) je intuitivno zasnovana, saj nam omogoča večine funkcionalnosti preko bodisi miške bodisi tipkovnice (v orodni vrstici pod zavihkom "Help" in potem "Keyboard shortcuts" si lahko ogledamo vse bližnjice na tipkovnici).

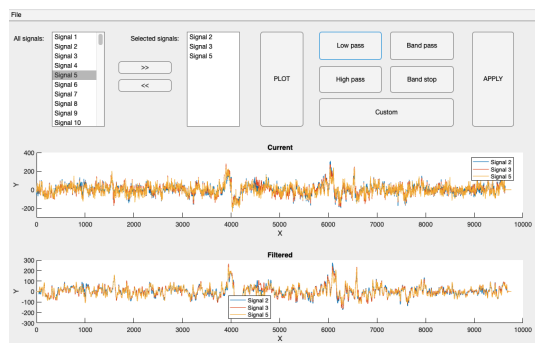


Figure 2: Okno aplikacije

Ko poženemo aplikacijo, se nam odpre glavo okno. Z izbiro zavihka "File" in potem "Open" (ali pritiskom tipke o) - Slika 3a, se nam odpre okno za izbiro .edf datoteke. Ko izberemo datoteko, lahko iz okna "All signals" izberemo signale, ki jih želimo prikazati in kasneje filtrirati. Signale izberemo z dvojnimi klikom na posamezno vrstico signala. Izbrani signali se prikažejo v oknu "Selected signals" (z dvojnimi klikom na signal v tem oknu, lahko odstranimo signal iz izbranih) - Slika 3b.

Ko imamo vse željene signale izbrane, kliknemo tipko "Plot" (ali kliknemo na del glavnega okna, da nismo v nobenem elementu in pritismo tipko p na tipkovnici) - Slika 3c. V zgornje okno (izmed dveh) se nam izrišejo signali, ki smo jih izbrali. Sedaj lahko izberemo filter s klikom na enega izmed gumbov za filtre (ali tipkami l,h,m,n,c). Ob izbiri filtra, se nam prikaže pojavno okno z nastavitvami parametrov filtra (Slika 3d). Za nizkoprepustni in visokoprepustni filter nastavimo stopnjo filtra in mejno frekvenco. Za pasovno prepustni in pasovno zaporni filter nastavimo stopnjo filtra in obe mejni frekvenci. Pri zadnjem, "Custom" filtru (filtru po meri) izberemo koeficienta a in b. Filter po meri nam omogoča uporabo NEO filtrov, saj nastavimo koeficiente a.

Ko filter izberemo in nastavimo parametre, se nam v spodnjem oknu izrišejo signali po filtriranju (Slika 3e). Če želimo filter potrditi (da ga lahko shranimo ali nad filtriranimi signali izvedemo nov filter), pritismo gumb "Apply" (ali tipko a na tipkovnici) - Slika 3f. Sedaj se filtrirani signali prestavijo iz spodnjega v zgornje okno, kar pomeni, da so le-ti trenutni v veljavi. Če želimo shraniti filtrirane signale v novo datoteko izberemo iz orodne vrstice zavihek "File" in "Save" (oz. tipko s na tipkovnici) - Slika 3g. Če smo delo že shranili, od odprtja aplikacije, potem že imamo nastavljeno datoteko za shranjevanje in jo lahko zamenjamo z izbiro "Save As", kjer se nam okno za shranjevanje ponovno odpre in nastavimo novo datoteko.

2.3.2 Kode implementacije

Za definicijo filtra uporabljamo funkcijo *fir1* (definiramo stopnjo filtra in frekvence, ki jih poda uporabnik, in vrsto filtra). Za aplikacijo filtra nad signali uporabljamo funkcijo *filter*, kjer podamo definiran filter in signale.

Za branje signalov uporabljamo funkcijo *rdsamp*, za branje podatkov o datoteki uporabljamo funkcijo *edfinfo*.

Za pisanje v .edf datoteko, uporabljamo funkcijo *edfwrite*, kateri moramo podati tudi nekatere podatke, ki jih prenesemo iz začetne datoteke (podatke o glavi).

Ostale funkcije in grafični vmesnik so standardne za okolje MATLAB in jih ni potrebno poudarjati.

3 Rezultati

Za testiranje orodja, smo naložili enega izmed posnetkov in poskusili filtriranje posameznih signalov. Rezultati so prikazani na Sliki 4. Lahko vidimo, da nizkoprepustni filter zgladi krivuljo, visokoprepustni pa poudari nihanje na visokih frekvencah. Da bi preverili delovanje filtrov še dodatno, smo naredili izris (v ločeno okno) močnostnih in amplitudnih spektrov, kjer smo preverili dušenje oz. prepustnost signala na specifičnih frekvencah. Uporabnik aplikacije filtrira signale in močnostnih spektrov ne vidi, ampak tako smo lahko preverili pravilnost filtrov.

Če neke signale filtriramo, shranimo in ponovno odpremo shranjeno datoteko, so rezultat že filtrirani signali. S tem smo preverili pravilnost funkcionalnosti shranjevanja.

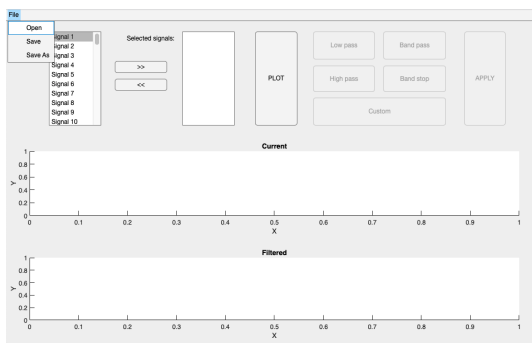
4 Diskusija

Izdelali smo grafični uporabniški vmesnik za poljubno filtriranje EEG signalov s KEO spektralnimi filtri. Filtriranje deluje pravilno in nam omogoča enostavno definicijo in aplikacijo filtrov nad EEG signali. Aplikacija omogoča vnos preko računalniške miške ali tipkovnice, samo izbor signalov za prikaz je mogoč samo z miško. Grafični vmesnik je zasnovan intuitivno, saj nima skritih podstrani in so vse možnosti uporabniku na voljo na prvi strani. Tudi omogočanje in onemogočanje gumbov z akcijami vodi uporabnika k pravilnemu zaporedju, saj v danem trenutku nima na voljo vseh gumbov (npr. preden apliciramo filter, moramo naložiti podatke - gumbi za filtre so onemogočeni pred izrisom osnovnih signalov). Tudi poskrbimo za neke predvidene napake, kjer bi recimo uporabnik želel izrisati signale, preden je sploh naložil datoteko, v tem primeru, se uporabniku pred izrisom odpre okno za izbor datoteke s signali (kot da bi najprej odprl datoteko in potem želel izrisati signale).

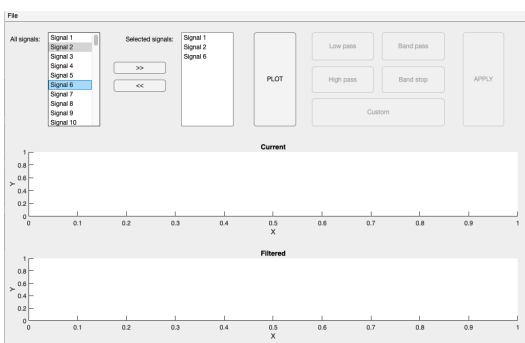
V prihodnje bi želeli še boljše podpreti grafični vmesnik z boljšo podporo za ustvarjanje poljubnih NEO filtrov, kjer sedaj samo podamo parametre, lahko pa bi še grafično spreminjali vnos. S tem bi dodali napredno podporo za filtre, čeprav to ni bil del naše osnovne naloge, ki smo si jo zadali v tem delu. Osnovna naloga je bila grafični vmesnik za KEO filtre, katerega smo izdelali, preverili in objavili za uporabo na javno dostopnem mestu GitHub [3], kjer bi si želeli, da bo (zaradi odprtokodne narave našega projekta) delo na projektu bilo še izboljšano s strani različnih članov odprtokodne skupnosti in seveda nas.

Literatura

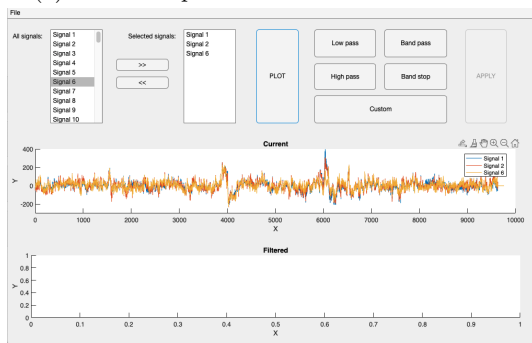
- [1] *EEG Motor Movement/Imagery Dataset*. URL: <https://www.physionet.org/content/eegmmidb/1.0.0> (visited on 01/23/2023).
- [2] *Gradivo predmeta KČR*. URL: https://ucilnica.fri.uni-lj.si/pluginfile.php/132806/mod_resource/content/46/8.Predobdelava_in_izlocanje_casovnih_ter_prostorskih_znacilk_II_new_xxx.pdf (visited on 01/23/2023).
- [3] *GUI for FIR filters in MATLAB*. URL: <https://github.com/janezbozic/GUI-for-FIR-filters-in-MATLAB> (visited on 01/23/2023).



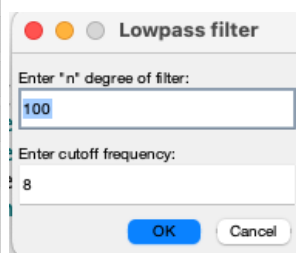
(a) Naložimo posnetek



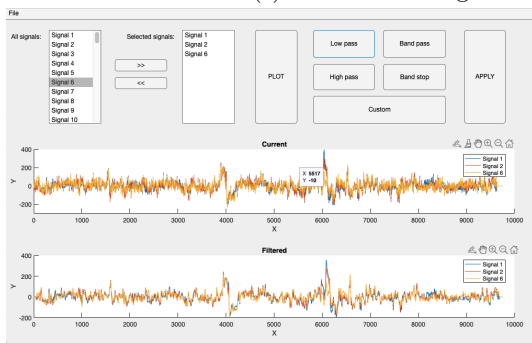
(b) Izberemo signale



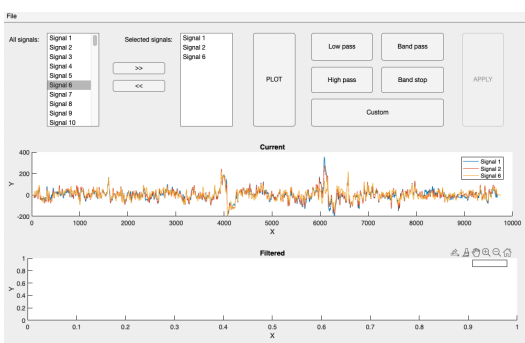
(c) Izris izbranih signalov



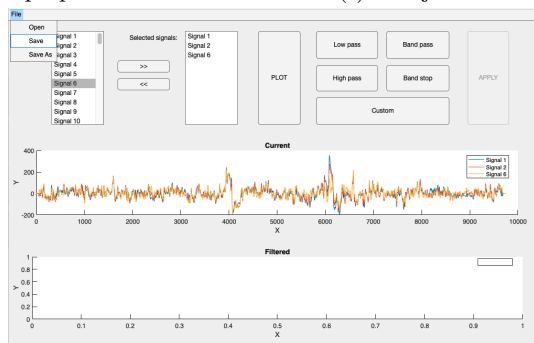
(d) Primer parametrov za filter



(e) Apliciran nizkoprepustni filter

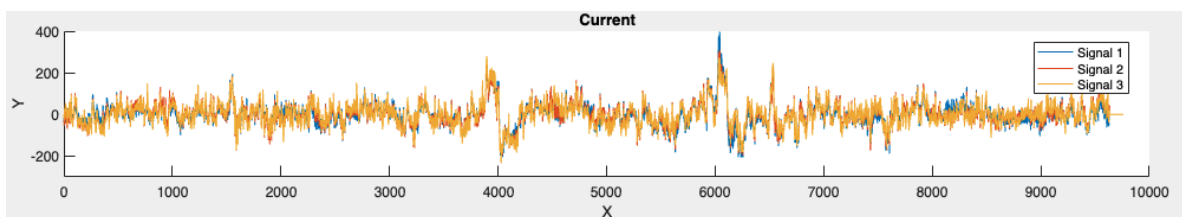


(f) Uveljavitev nizkoprepustnega filtra

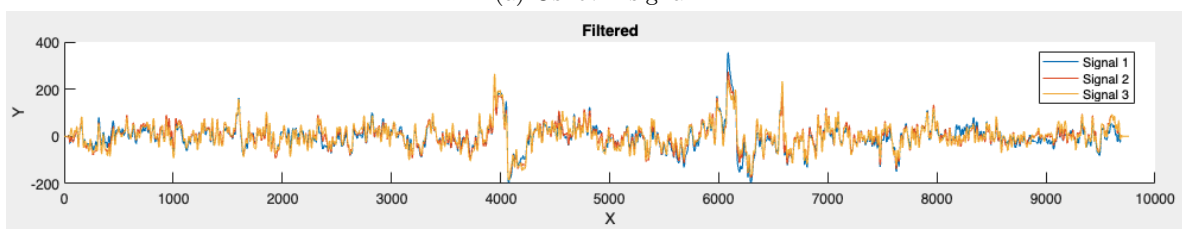


(g) Shrani signale

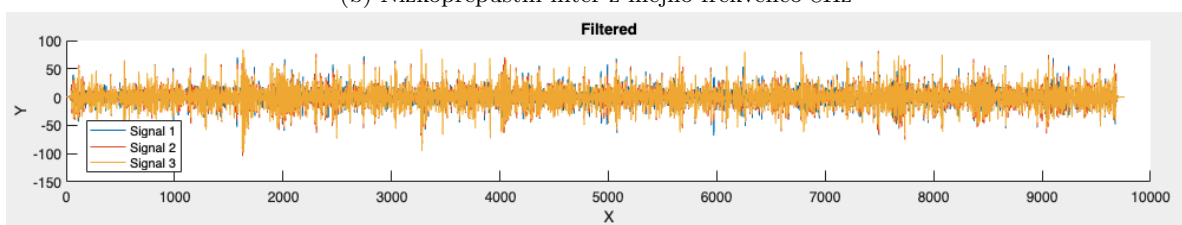
Figure 3: Akcije v aplikaciji



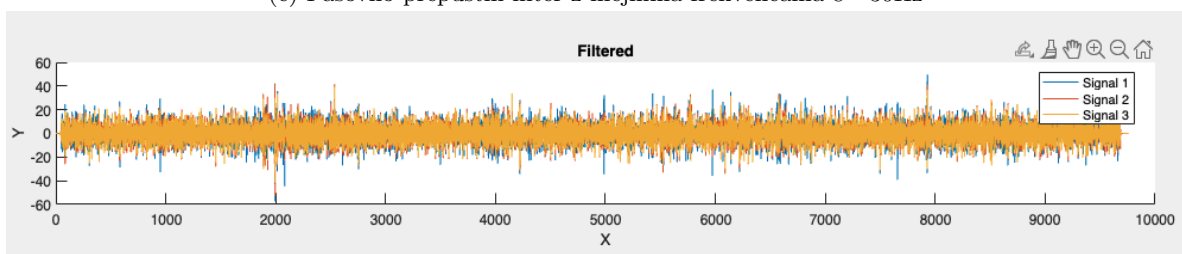
(a) Osnovni signali



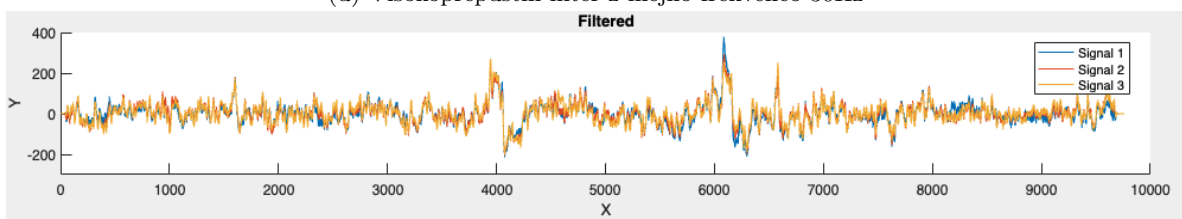
(b) Nizkoprepustni filter z mejno frekvenco 8Hz



(c) Pasovno prepustni filter z mejnima frekvencama 8 - 30Hz



(d) Visokoprepustni filter z mejno frekvenco 30Hz



(e) Pasovno zaporni filter z mejnima frekvencama 8 - 30Hz

Figure 4: Filtriranje signalov