LMS filtriranje signalov

Predloge za predavanja, 2019

Dušan Ponikvar

Fakulteta za matematiko in fiziko

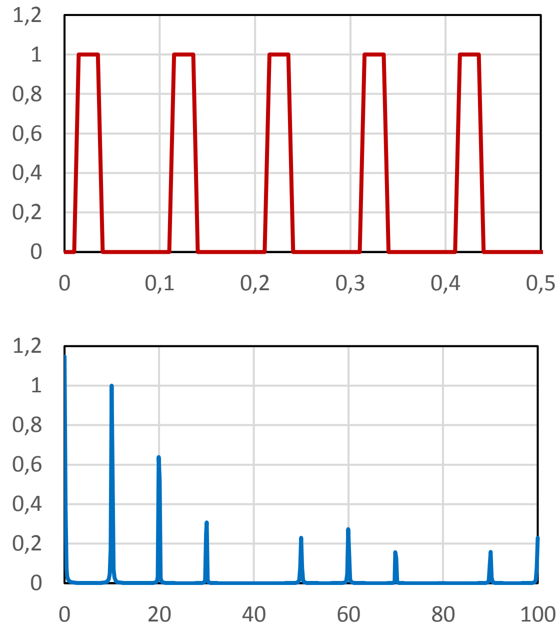
Jadranska 19, Ljubljana

LMS filtriranje signalov

Signali prenašajo informacije in tem signalom so vedno dodane tudi moteče komponente, ki otežujejo dekodiranje informacij. Moteče komponente odstranimo s filtriranjem. Do sedaj smo filtrirali signale tako, da smo iz celotnega frekvenčnega spektra zajetega signala odstranili tiste dele spektra, kjer ni koristnih informacij. Tako smo ohranili vse lastnosti koristnega dela signala in hkrati odstranili ali vsaj zmanjšali delež motečih komponent. Ta metoda je dobra, kadar je koristni signal omejen na (majhen) del frekvenčnega spektra.

Signali niso nujno harmonske oblike. Kadar je informacija kodirana v obliki signala, ta oblika pa po Fourieru zaseda več ločenih delov frekvenčnega spektra, je naloga težja. Informacija je na primer lahko kodirana v širino periodičnega pravokotnega sunka (PWM, pulzno širinska modulacija); pravokotni signal po Fourieru zaseda vse cele mnogokratnike osnovne frekvence periodičnega signala. Ali lahko filtriramo tako, da ohranimo v frekvenčnem spektru le vse cele mnogokratnike frekvence osnovnega signala, vse ostalo pa izločimo?

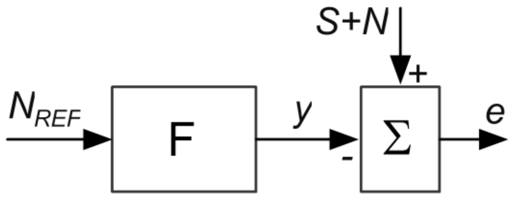
Podobno je težko odstraniti moteči signal, ki zaseda več delov spektra zajetega signala. Za zgled naj spet služi moteči pravokotni signal, ki ga po Fourieru lahko razstavimo v množico komponent, te so med sabo razmaknjene za dvakratnik osnovne frekvence pravokotnega signala. Kako izločiti tak moteči signal tako, da hkrati čim manj spremenimo koristni del spektra in s tem koristni signal?



Slika 1: Signal pulzno-širinsko moduliranega signala in njegov amplitudni spekter. Zgornji diagram: abscisa je čas v sekundah, ordinata je normirana velikost signala. Spodnji diagram: abscisa je frekvenca v Hertzih, ordinata je normirana amplituda

Z običajnim filtriranjem pač ne moremo rešiti zgornjih dveh problemov, zato poskusimo drugače. Če bi poznali obliko motečega signala v drugem navedenem primeru zgoraj, njegovo amplitudo, frekvenco in fazo, bi ga iz lahko iz zajetega signala enostavno odšteli in uporabili preostali del zajetega signala. Problem je le ta, da vseh lastnosti motečega signala ne poznamo, zato ne znamo odštevati. Poskusili bomo najti metodo, po kateri bo mogoče avtomatsko določiti vse lastnosti motečega signala in ga izločiti po najboljših močeh.

Osnovna bločna shema potrebnega sistema je na sliki 2. Zajeti signal $S+N$ vsebuje koristni del S in moteči del N . Za filtriranje moramo poznati vsaj približne lastnosti motnje N, na primer njeno obliko in frekvenco, kar navadno ni posebej težko. Za primer: če motnjo povzroča električno omrežje, je frekvenca moteče (ali motečih, lahko jih je več) komponente zagotovo celoštevilčni mnogokratnik frekvence 50 Hz. Vzorec motnje označimo z N_{REF} in na splošno ni enak moteči komponenti N zajetega signala, ima pa enako obliko in frekvenco. Ko vzorec motnje vodimo skozi filter F (morda FIR filter), ga priredimo; spremenimo mu velikost, fazo in obliko, njegove frekvence pa filter ne more spremeniti. Modificirano verzijo vzorca motečega signala $y=F(N_{REF})$ nato odštejemo od zajetega signala $S+N$. Če smo imeli srečo s koeficienti filtra F oziroma lastnostmi tega filtra, je modificirana verzija vzorca motečega signala y ravno enaka moteči komponenti N v zajetem signalu $S+N$ in po odštevanju dobimo le koristni signal, označimo ga z e. Če nismo imeli sreče in motečega signala nismo dovolj dobro ponaredili, je treba popraviti filtrske koeficiente tako, da bo v signalu e moteče komponente čim manj. Lastnosti filtra (vrednosti koeficientov filtrskega jedra) bo treba iterativno popravljati proti optimalni vrednosti, kar bo za nas počel računalnik. Potrebujemo pa dober kriterij za določanje velikosti preostanka moteče komponente N v izhodnem signalu e in metodo, ki bo na podlagi rezultatov tega kriterija znala izbrati boljše vrednosti koeficientov v filtru.

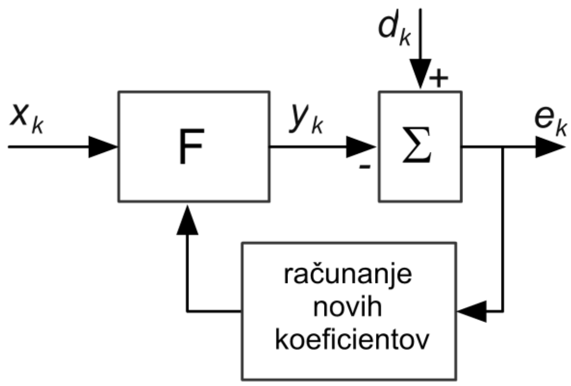


Slika 2: Osnovna bločna shema za idejo filtra z odštevanjem motečega signala

Na podlagi zgoraj povedanega sestavimo pravo bločno shemo filtra z odštevanjem na sliki 3, ki jo bo treba implementirati v računalniku. Tak sistem lahko uporabimo na različne načine, kar bo prikazano v nadaljevanju tega poglavja, zato bomo signale poimenovali na bolj univerzalen način. Signali so:

- d_k: željeni signal (»desired signal«),
- x_k: referenčni signal,
- y_k: modificirani referenčni signal in
- e_k: napaka (»error signal«).

Do optimalnih lastnosti filtra se dokopljemo iterativno, zato vsakemu od zgoraj navedenih signalov pridamo indeks k za oznako koraka iteracije. Pri iskanju optimalnih lastnosti filtra se zanašamo na to, da se lastnosti signalov le malo spreminjajo v času, kar označujemo s pojmoma ergodičnost (osnovne lastnosti signala je mogoče razbrati iz kateregakoli poljubnega, a dovolj dolgega niza vzorcev tega signala) in stacionarnost (osnovne lastnosti signalov se s časom ne spreminjajo).



Slika 3: Bločna shema filtra za odštevanje motečega signala

Potrebujemo še kriterij, po katerem lahko izračunamo trenutno kvaliteto odštevanja signalov. Signal d_k je željeni signal v k-tem koraku, x_k pa referenčni signal v istem koraku. Referenčni signal x_k moramo spremeniti s filtriranjem F tako ($y = F(x)$, da bo čim bolj podoben željenemu. Če to dosežemo, je napaka e_k v tem koraku enaka nič ($e_k = d_k - y_k = 0$). Seveda nas ne zanima samo napaka v posameznem koraku, saj se motilni in vhodni signal s časom spreminjata in sistem se mora spremembam prilagajati, zato je pomembna tudi povprečna vrednost napake. Če je nič tudi ta lahko trdimo, da je filter dobro spremenil referenčni signal in je torej modificirani signal enak željenemu.

Iščemo torej tisto vrednost koeficientov F, za katero je napaka e minimalna. Če izberemo za kriterij o velikosti napake kvadrat napake v posameznem koraku, je iskanje matematično lažje, zato:

$$e_k{}^2 = \;(d_k - y_k)^2$$

Če se odločimo in za filtriranje uporabimo od prej znano FIR strukturo, je izhodni signal y_k filtra podan s konvolucijo med zaporednimi vzorci vhodnega signala x v filter in koeficienti filtrskega jedra f_m:

$$y_k = \sum_{m=0}^{M-1} x_{k-m} \cdot f_m$$

Pri tem velja, da je izdelani FIR filter kavzalnega značaja, saj njegov izhodni signal y_k temelji le na trenutnem in preteklih vzorcih vhodnega signala x. Izpeljava sistema za fitriranje bo enostavnejša, če konvolucijsko enačbo zapišemo v matrični obliki:

$$\boldsymbol{X_k} = \begin{bmatrix} x_k \\ x_{k-1} \\ \vdots \\ x_{k-M+1} \end{bmatrix} \qquad \boldsymbol{F} = \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_{M-1} \end{bmatrix} \qquad \Rightarrow \qquad y_k = \boldsymbol{X_k^T} \cdot \boldsymbol{F} = \boldsymbol{F}^T \cdot \boldsymbol{X_k}$$

V takem zapisu napako e_k izračunamo:

$$e_k = d_k - y_k = d_k - \boldsymbol{X_k^T F} = d_k - \boldsymbol{F^T X_k}$$

Za kriterij o velikosti napake v posameznem koraku pa smo raje zapisali kvadrat zgornjega izraza. Kadar je napaka velika, želimo bolj posegati v sistem in s tem hitreje izboljševati lastnosti sistema. Zato računamo:

$${e_k}^2 = (d_k - y_k)^2 = (d_k - \boldsymbol{F^T X_k})^2 = {d_k}^2 - 2 d_k \boldsymbol{F^T X_k} + (\boldsymbol{F^T X_k})^2$$

Prilagajanje je dobro, če je povprečna vrednost napake čim manjša, zato za kriterij kakovosti prilagajanja definiramo povprečno vrednost kvadrata napake (kriterijsko finkcijo J, ki je pravzaprav kvadrat efektivne vrednosti napake RMS):

$$J = \langle {e_k}^2 \rangle = \langle {d_k}^2 \rangle - \langle 2 d_k \boldsymbol{F^T X_k} \rangle + \langle \boldsymbol{F^T X_k X_k^T F} \rangle$$

$$J = \langle {d_k}^2 \rangle - \langle 2 d_k \boldsymbol{X_k^T} \rangle \boldsymbol{F} + \boldsymbol{F}^T \langle \boldsymbol{X_k X_k^T} \rangle \boldsymbol{F}$$

Kriterijska funkcija je torej kvadratna funkcija koeficientov filtrskega jedra, taki pa lahko analitično poiščemo minimum in na ta način ob poznavanju vseh lastnosti vhodnih signalov izberemo optimalne vrednosti filtrskih koeficientov $\boldsymbol{F}$. Gradient kriterijske funkcije je:

$$\nabla J_k = \frac{\partial J_k}{\partial \boldsymbol{F}} = \langle -2 d_k \boldsymbol{X}_k \rangle + \boldsymbol{F}^T \langle \boldsymbol{X_k X_k^T} \rangle$$

Od tod izračunamo optimalne vrednosti filtrskih koeficientov $\boldsymbol{F_{OPT}}$:

$$\langle -2 d_k \boldsymbol{X}_k \rangle + \boldsymbol{F}^T \langle \boldsymbol{X_k X_k^T} \rangle = 0 \qquad \Rightarrow \qquad \boldsymbol{F_{OPT}^T} = \langle 2 d_k \boldsymbol{X}_k \rangle \ \langle \boldsymbol{X_k X_k^T} \rangle^{-1}$$

Če bi torej natančno poznali vse signale, bi optimalne vrednosti filtrskih koeficientov lahko izračunali z množenjem, transponiranjem, povprečenjem in inverzijo matrik. Take operacije bi morali izvajati dokaj pogosto, kar je računsko zelo intenzivno in zato praktično neizvedljivo. Treba bo najti drugo, enostavnejšo pot za izbiro optimalnih vrednosti koeficientov filtra.

Minimum poljubne kvadratne funkcije $f(x)$ lahko iščemo tudi iterativno (po korakih). Pri tem v vsakem koraku velja:

$$x_{MIN\,k} = x_{MIN\,k-1} - \mu \cdot \frac{d\,f(x)}{dx}$$

Hitrost približevanja minimumu določa velikost konstante μ (in strmina funkcije v točki preverjanja). Ta mora biti dovolj majhna zato, da minimuma funkcije ne preskočimo s predolgim korakom in hkrati dovolj velika, da je približevanje minimumu zmerno hitro. V našem primeru, ko korakoma iščemo minimum kvadrata napake ${e_k}^2$ in s tem optimalne vrednosti filtrskih koeficientov $\boldsymbol{F}$, pišemo (polovička olajšuje nadaljnji račun):

$$\boldsymbol{F}_k = \boldsymbol{F}_{k-1} - \frac{\mu}{2} \cdot \frac{\partial\, {e_k}^2}{\partial \boldsymbol{F}}$$

Za računanje torej potrebujemo parcialne odvode funkcije kvadrata napake, ki predstavlja kriterij kakovosti prilagajanja referenčnega signala željenemu, po posameznih koeficientih filtrskega jedra $\boldsymbol{F}$. Potrebujemo torej parcialne odvode kvadrata napake:

$$\frac{\partial {e_k}^2}{\partial \boldsymbol{F}^T} = \begin{bmatrix} \frac{\partial {e_k}^2}{\partial f_0} \\ \frac{\partial\, {e_k}^2}{\partial f_1} \\ \vdots \\ \frac{\partial\, {e_k}^2}{\partial f_{M-1}} \end{bmatrix}$$

Posamezen parcialni odvod določimo:

$$\frac{\partial {e_k}^2}{\partial f_0} = 2e_k \frac{\partial e_k}{\partial f_0} = 2e_k \frac{\partial}{\partial f_0}\left(d_k - \boldsymbol{X}_{\boldsymbol{k}}^T \boldsymbol{F}\right) = -2e_k x_k$$

$$\frac{\partial {e_k}^2}{\partial f_1} = 2e_k \frac{\partial e_k}{\partial f_1} = 2e_k \frac{\partial}{\partial f_1}\left(d_k - \boldsymbol{X}_{\boldsymbol{k}}^T \boldsymbol{F}\right) = -2e_k x_{k-1}$$

$$\vdots$$

Vektor parcialnih odvodov je zato:

$$\frac{\partial {e_k}^2}{\partial \boldsymbol{F}^T} = \begin{bmatrix} -2e_k x_k \\ -2e_k x_k - 1 \\ \vdots \\ -2e_k x_{k-M+1} \end{bmatrix} = -2e_k \boldsymbol{X}_k$$

Po teh poenostavitvah in izračunih napišemo končni izraz za iterativno izračunavanje optimalnih filtrskih koeficientov $\boldsymbol{F}_k$, v vsakem koraku k bo treba izračunati izraz:

$$\boldsymbol{F}_k = \boldsymbol{F}_{k-1} + \mu \cdot e_k \boldsymbol{X}_k$$

Novo vrednost filtrskih koeficientov $\boldsymbol{F}_k$ torej dobimo tako, da prejšnjo vrednost teh koeficientov $\boldsymbol{F}_{k-1}$ popravimo za uteženo vrednost zaporednih vrednosti signala x, pri tem je utež enaka produktu napake v tem koraku e_k in velikosti koraka μ. Spet velja, da korak μ ne sme biti prevelik, ker bi sicer lahko optimalne vrednosti koeficientov preskočili. Hkrati korak μ ne sme biti premajhen, ker je sicer približevanje optimalni vrednosti prepočasno. Vrednost koraka μ navadno izberemo z nekaj poskusi in mora biti prilagojen lastnostim opazovanih signalov. Vrednost koraka se da določiti tudi analitično, vendar moramo pri tem poznati lastnosti (strmino naraščanja) funkcije napake, česar tu ne bomo izpeljevali.

Filtriranje po opisanem algoritmu zato opravimo po spodnjem <u>receptu</u>:

a) - Inicializirajmo vektor koeficientov filtrskega jedra $\boldsymbol{F}_k$; vektor naj ima zadostno število elementov *M*, primerna začetna vrednost za koeficiente je tudi 0, saj bo algoritem iterativno poiskal optimalne vrednosti.
- Inicializirajmo vektor prejšnjih vrednosti referenčnega signala $\boldsymbol{X}_k$; vektor naj ima enako število elementov *M*, primerna začetna vrednost je tudi 0; v ta vektor bomo shranjevali trenutno in prejšnje vrednosti referenčnega signala x.
- Izberimo vrednost koraka μ.

b) Pomaknimo v vektorju $\boldsymbol{X}_k$ shranjene vrednosti prejšnjih izmerkov referenčnega signala x za eno mesto naprej, v sproščeno prvo mesto pa shranimo pravkar pomerjeno novo vrednost x_k. Sprejmimo tudi novo vrednost željenega signala d_k.

c) Izračunajmo novo izhodno vrednost filtra y_k, uporabimo formulo za konvolucijo:

$$y_k = \sum_{m=0}^{M-1} x_{k-m} \cdot f_m$$

d) Izračunajmo napako e_k:

$$e_k = d_k - y_k$$

e) Popravimo vrednosti filtrskih koeficientov:

$$\boldsymbol{F}_k = \boldsymbol{F}_{k-1} + \mu \cdot e_k \boldsymbol{X}_k$$

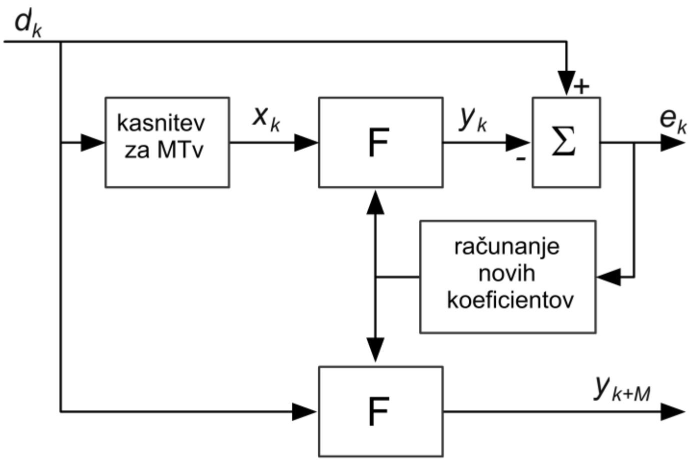
f) Ponovimo od koraka b) za naslednji vzorec vhodnih signalov

Takšen način filtriranja je v literaturi znan pod imenom LMS (»Least Mean Square«) filtriranje, saj optimalne vrednosti koeficientov filterskega jedra iščemo po metodi najmanjših kvadratov napake.

Opisano način filtriranja uporabljamo v različne namene, nekateri so podani v spodnji listi.

a) <u>Predvidevanje prihodnosti, slika 4</u>

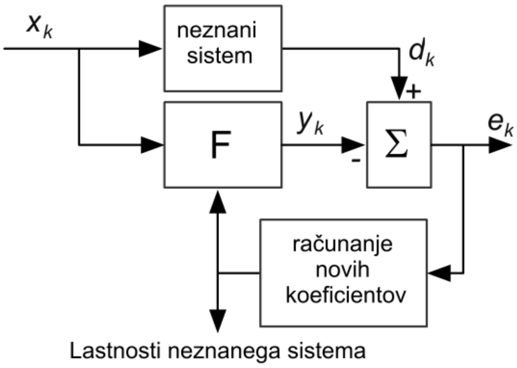
Zgornja polovica filtra spremeni zakasnjeno verzijo vhodnega signala d_k tako, da je y_k enak d_k; filter torej iz zakasnjene verzije vzorcev vhodnega signala napove trenutno vrednost vhodnega signala, to stori s pomočjo filtriranja s primernimi koeficienti. Če te iste koeficiente uporabimo za filtriranje ne-zakasnjene verzije vhodnega signala, je izhodni signal spodnje polovice filtra y_{k+M} napoved prihodnosti. Seveda je potrebno biti pri vsakem napovedovanju prihodnosti previden; tule se zanašamo na ergodičnost in časovno invariantnost lastnosti vhodnega signala.



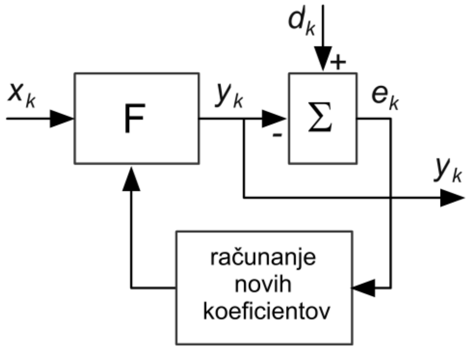
Slika 4: Shema filtra za napovedovanje prihodnosti

b) <u>Ovrednotenje lastnosti neznanega sistema, slika 5</u>

Vzemimo, da imamo neznani sistem, katerega lastnosti nas zanimajo. Tak neznani sistem vzbujamo s signalom x_k in hkrati s pomočjo opisanega filtriranja priredimo vzbujalnemu signalu signal y_k tako, da sta odziva merjenega sistema d_k in filtra F enaka ($y_k = d_k \rightarrow e_k = 0$). Takrat trdimo, da so lastnosti filtra in fizikalnega sistema enake, torej lahko tudi fizikalni sistem popišemo kot FIR filter s pravkar izračunanimi koeficienti.



Slika 5: Shema filtra za ovrednotenje lastnosti sistema



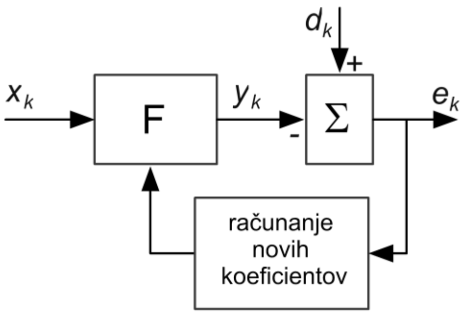
Slika 6: Shema filtra za iskanje signala z delno znanimi lastnostmi

c) <u>Iskanje najboljšega približka signala z delno znanimi lastnostmi, slika 6</u>

Na vhod $d_k$priključimo vhodni signal, ki vsebuje motilne komponente, na vhod x_k pa vzorec signala, ki ga iščemo. Sistem se bo prilagodil tako, da bo minimiziral napako e_k, torej bo iz vhodnega signala $d_k$odstranil vse, kar se da odstraniti na podlagi vzorčnega signala x_k. Takrat trdimo, da je izračunani signal y_k najboljši približek v signalu d_k vsebovanega vzorca signala x_k.

d) <u>Izločanje motenj z deloma poznanimi lastnostmi, slika 7</u>

Tokrat na vhod d_k priključimo signal z motnjami, na vhod x_k pa vzorec motnje. Ta vzorec mora biti po svojem frekvenčnem spektru čim bolj podoben motnjam, ki jih želimo izločiti, vsekakor pa mora vsebovati harmonske komponente pri vseh frekvencah, kjer pričakujemo motnje in nič harmonskih komponent pri frekvencah, ki jih želimo v izhodnem signalu obdržati nespremenjene. Sistem bo s filtriranjem prilagodil signal y_k tako, da bo vrednost napake e_k minimalna, torej ne bo vsebovala komponent, ki so v vzorčnem signalu.



Slika 7: Shema filtra za izločanje moten j z delno znanimi lastnostmi

Vir: Adaptive signal processing, B.Widrow, S.Stearns, 1985, Prentice-Hall