

# PRIMJENA KALMANOVOG FILTRA ZA FUZIJU SENZORA

---

**Mostarac, Marija**

**Undergraduate thesis / Završni rad**

**2021**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:200:454747>

*Rights / Prava:* [In copyright](#) / [Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-11-14**

*Repository / Repozitorij:*

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU  
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I  
INFORMACIJSKIH TEHNOLOGIJA**

**Sveučilišni studij**

**PRIMJENA KALMANOVOG FILTRA ZA FUZIJU  
SENZORA**

**Završni rad**

**Marija Mostarac**

**Osijek, 2021.**

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK**Obrazac Z1P - Obrazac za ocjenu završnog rada na preddiplomskom sveučilišnom studiju**

Osijek, 16.09.2021.

Odboru za završne i diplomske ispite

**Prijedlog ocjene završnog rada na  
preddiplomskom sveučilišnom studiju**

<b>Ime i prezime studenta:</b>	Marija Mostarac
<b>Studij, smjer:</b>	Preddiplomski sveučilišni studij Elektrotehnika i informacijska tehnologija
<b>Mat. br. studenta, godina upisa:</b>	4560, 24.07.2018.
<b>OIB studenta:</b>	38981157991
<b>Mentor:</b>	Prof. dr. sc. Kruno Miličević
<b>Sumentor:</b>	
<b>Sumentor iz tvrtke:</b>	
<b>Naslov završnog rada:</b>	Primjena Kalmanovog filtra za fuziju senzora
<b>Znanstvena grana rada:</b>	<b>Elektrostrojarstvo (zn. polje elektrotehnika)</b>
<b>Predložena ocjena završnog rada:</b>	Izvrstan (5)
<b>Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:</b>	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 2 bod/boda Jasnoća pismenog izražavanja: 1 bod/boda Razina samostalnosti: 3 razina
<b>Datum prijedloga ocjene mentora:</b>	16.09.2021.
<b>Datum potvrde ocjene Odbora:</b>	22.09.2021.
Potpis mentora za predaju konačne verzije rada u Studentsku službu pri završetku studija:	Potpis:
	Datum:

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK**IZJAVA O ORIGINALNOSTI RADA**

Osijek, 27.09.2021.

**Ime i prezime studenta:**

Marija Mostarac

**Studij:**

Preddiplomski sveučilišni studij Elektrotehnika i informacijska tehnologija

**Mat. br. studenta, godina upisa:**

4560, 24.07.2018.

**Turnitin podudaranje [%]:**

5

Ovom izjavom izjavljujem da je rad pod nazivom: **Primjena Kalmanovog filtra za fuziju senzora**

izrađen pod vodstvom mentora Prof. dr. sc. Kruno Miličević

i sumentora

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija. Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

# SADRŽAJ

<b>1. UVOD .....</b>	<b>1</b>
<b>1.1. Zadatak završnog rada .....</b>	<b>2</b>
<b>2. PREGLED PODRUČJA TEME .....</b>	<b>3</b>
<b>2.1. Općenita primjena filtra u estimaciji stanja .....</b>	<b>4</b>
<b>2.2. Izvod jednadžbi Kalmanovog filtra .....</b>	<b>6</b>
<b>2.3. Algoritam Kalmanovog filtra .....</b>	<b>9</b>
2.3.1. Faze izvođenja algoritma .....	9
2.3.3. Vizualizacija algoritma .....	10
<b>2.4. Primjer MATLAB simulacije .....</b>	<b>13</b>
<b>3. FUZIJA SENZORA.....</b>	<b>21</b>
<b>3.1. Odlike fuzije senzora.....</b>	<b>22</b>
<b>3.2. Algoritam fuzije senzora .....</b>	<b>24</b>
3.2.1. Prilagodna jednadžbi Kalmanovog filtra .....	26
<b>3.3. MATLAB primjer primjene Kalmanovog filtra za fuziju senzora .....</b>	<b>26</b>
<b>4. ZAKLJUČAK.....</b>	<b>29</b>
<b>LITERATURA .....</b>	<b>30</b>
<b>SAŽETAK .....</b>	<b>32</b>
<b>ABSTRACT.....</b>	<b>32</b>
<b>ŽIVOTOPIS .....</b>	<b>33</b>
<b>PRILOG – Kôd MATLAB simulacije .....</b>	<b>34</b>

## 1. UVOD

Kalmanov filtar je *optimalni algoritam procjene*. Često se koristi za estimaciju to jest optimalnu procjenu unutarnjih stanja sustava koja se ne mogu izravno mjeriti pri čemu se u obzir uzima prisustvo šumova i eventualnih poremećaja u poticaju. Za postojanje navedenog rekurzivnog algoritma, koji od godine objavljivanja 1960. nije izgubio na značaju, zaslužan je Rudolph Emil Kalman. Njegova prva implementacija bila je procjena trajektorija do Mjeseca i natrag u sklopu NASA-inog projekta Apollo gdje je razlučivao valjane podatke od nevaljanih.

Estimacija podrazumijeva procjenu trenutnog stanja na temelju dostupnih mjerenja. Razlikuje se prediktivna i neprediktivna procjena. Procjena koja se temelji samo na podacima od prethodnog koraka naziva se prediktivna dok neprediktivna uz navedene podatke prethodnog koraka uključuje i izmjerene vrijednosti aktualnog koraka. Problemi koji se javljaju prilikom estimacije su često netočni i neprecizni rezultati mjerenja uzrokovani prisustvom šumova. Važnost Kalmanovog filtra proistječe upravo iz toga. Naime, termin filtar upotrebljava se zbog uklanjanja šuma iz podataka.

U situaciji kada željeno stanje sustava nije moguće izmjeriti izravno, koristi se Kalmanov filtar za pronalazak najbolje procjene unutarnjeg stanja na temelju indirektnog mjerenja. U tom slučaju se do relevantne veličine dolazi posredno, izračunavanjem na osnovu mjerljivih veličina. Kada su dostupna mjerenja različitih senzora koja podliježu šumovima, Kalmanov filtar se koristi za kombiniranje senzornih podataka iz pojedinih izvora – fuzija senzora, kako bi pronašao najbolju procjenu parametara od interesa.

Za shvaćanje što je ustvari Kalmanov filtar te kako se koristi, u radu će se promatrati sustavi opisani odgovarajućim matematičkim modelom. Koristit će se pojmovi usvojeni tijekom dosadašnjeg sveučilišnog obrazovanja naročito iz područja vjerojatnosti, automatskog upravljanja te analiza mreža i vladanja sustava. Proći će se kroz jednadžbe i algoritam te primjer MATLAB simulacije za primjenu filtra.

## 1.1. Zadatak završnog rada

Osnovni zadatak jest pokazati kako se stanja linearnog sustava mogu procijeniti upotrebom Kalmanovog filtra te kako se isti koristi prilikom kombiniranja mjerenja različitih senzora. Problem se svodi na to da se na temelju poznatih vrijednosti izlaznih veličina sustava  $y$  i upravljačkih veličina  $u$  odredi vrijednosti varijabli stanja  $x$ , tj. na potragu optimalnog linearnog procjenitelja slučajne varijable izlaza u skupu uzastopnih vremenskih trenutaka. U tu svrhu je potrebno objasniti vremenski diskretne iteracije sadržane algoritmom od kojih svaka podrazumijeva dva koraka, predikciju te ažuriranje to jest korekciju, čiji je cilj odrediti matricu Kalmanovog pojačanja tako da se dobije najmanja moguća greška procjene. Konačno se slika upotpunjuje izvedenom simulacijom u programskom jeziku visoke razine – MATLAB-u gdje se promatra hoće li upotrebom filtra procjena stanja biti točnija te hoće li primjena većeg broja senzora u vidu kombinacije mjerenja poboljšati procjenu smanjenjem utjecaja šuma.

## 2. PREGLED PODRUČJA TEME

Algoritmi za procjenu stanja sustava u trenutnom koraku prije Kalmanovog koristili su podatke od svih ranijih koraka. Prednost Kalmanovog filtra leži u rekurzivnosti, tj. nije potrebno pamtiti podatke iz svih prošlih koraka nego samo najbolju estimaciju prethodnog stanja rekurzivno određenu na osnovu dotadašnjih mjerenja. Osnovna verzija Kalmanovog filtra koja je prva objavljena 1960. godine proizašla je iz činjenice da se za prikaz i razmatranje vladanja bilo kojeg sustava koristi linearni matematički model. U korak s time R. E. Kalman u prvobitnom algoritmu, za aproksimaciju vremenski diskretnih veličina koristio je linearnu funkciju [1]. Budući da je on omogućio procjenu kako prošlih, tako i sadašnjih ali i budućih stanja sustava čak i na osnovi nepreciznog matematičkog modela, nikada nije izgubio na značaju.

Već iduće godine, filter je doživio svoju veoma bitnu modifikaciju i primjenu na inicijativu NASA-e sudjelovanjem u projektu Apollo. Dakle, kombinacija teorije linearnog filtra dr. Kalmana s linearnim metodama perturbacije već prethodno korištenih, dala je znanstvenicima NASA-e potencijalno rješenje problema nelinearne navigacije, kao i nekih problema koje predstavljaju ograničenja brzine i pohrane korištenog računala IBM 704 [2].

Danas postoje brojne modifikacije originalnog linearnog diskretnog filtra pa se tako za rješavanje estimacije kod nelinearnih sustava koristi prošireni Kalmanov filter (engl. *Extended Kalman filter* – *EKF*) gdje se putem procjene stanja i kovarijance koraka  $k-1$  linearizira procjena stanja u narednom,  $k$ -tom koraku primjenom Taylorovog reda točnije, parcijalnim derivacijama nelinearnih funkcija procesa i mjerenja [3]. Uz EKF posebno se ističe još i nederivirajući Kalmanov filter (engl. *Unscented Kalman filter* – *UKF*) koji se također koristi za nelinearne sustave primjenjujući nelinearnu transformaciju i preslikavanje linearizacijom čime se u usporedbi s EKF-om dobivaju točniji rezultati. Algoritam UKF-a je složeniji jer uz inicijalizaciju te diskretizaciju koja podrazumijeva predikciju i korekciju, uključuje i izračun sigma točaka. [4,5]

Kao optimalni estimator, Kalmanov filter pronašao je brojne uporabe u sveobuhvatnoj okolini, od obrade i procesiranja raznovrsnih signala, digitalnih izračunavanja u sustavima upravljanja, preko praćenja putanje i položaja objekata, navigacije, široko primijenjene robotike, do različitih vrsta zaštite primjerice od poplava.[6, 7, 8, 9]



## 2.1. Općenita primjena filtra u estimaciji stanja

Estimator stanja ili procjenitelj stanja je sustav koji pruža procjenu unutarnjeg stanja određenog stvarnog sustava, na temelju mjerenja ulaza i izlaza navedenog fizičkog sustava. Uobičajeno je računalno implementiran i pruža osnovu mnogim praktičnim primjenama [10]. Poznavanje stanja sustava neophodno je za rješavanje brojnih problema teorije upravljanja, npr. stabiliziranje sustava pomoću povratnih informacija. U većini praktičnih slučajeva stanje sustava ne može se utvrditi izravnim promatranjem nego ga treba procijeniti posrednim tehnikama. Npr. kod svemirskih letjelica poznavanje iznosa unutarnje temperature komore za sagorijevanje pogonskog goriva je od krucijalne važnosti ali do te informacije nije moguće doći izravnim mjerenjem. Ako bi senzor za mjerenje navedene vrijednosti bio postavljen unutar komore, visoka temperatura bi uzrokovala njegovo topljenje, zbog čega se on postavlja na hladniju površinu izvan komore ali u njezinoj neposrednoj blizini. Sada taj senzor može mjeriti samo vanjsku temperaturu koja predstavlja izlaz. Dakle, potrebno je odrediti vrijednost varijabli stanja  $x$  na temelju poznatih izlaznih  $y$  i upravljačkih  $u$  veličina [11]. Procijenjene varijable označavaju se s  $\hat{x}$ . Ukoliko je poznat matematički model promatranog sustava kao i ulaz u sustav koji ujedno predstavlja ulaz modela, dobiva se procijenjena vrijednost izlaza te je uz postojanje svih potrebnih jednadžbi moguć izračun i procijenjene vrijednosti traženog stanja sustava. Ako bi matematički model bio savršeno precizan i točan te ako bi početni uvjeti stvarnog sustava i modela bili identični, tada bi izmjerene i pretpostavljene vrijednosti izlaza u potpunosti odgovarale jedne drugima, analogno tome i pretpostavljeno stanje sustava bi se moglo uzeti u obzir kao realno, tj. pravo stanje sustava kojem se teži budući da ga nije moguće izravno mjeriti. No, u realnosti matematički model predstavlja samo aproksimaciju stvarnog sustava i nije apsolutno precizan što ovisi o tome koliko dobro se napravi točna identifikacija parametara modela [11]. Osim toga, obično se sustav pretpostavlja linearnim, a uvijek postoje veće ili manje nelinearnosti te su točni početni uvjeti najčešće nepoznati. Također, sustav može biti pod utjecajem poremećaja ili šumova koje se ne mogu modelirati i uzeti u obzir kod formiranja modela. Kao rezultat toga procijenjena vrijednost izlaza ne odgovara stvarnoj izmjerenoj vrijednosti izlaza gdje onda estimator stanja dobiva svoje mjesto upotrebe. Cilj koji stoji iza principa rada procjenitelja stanja jest postići podudaranje izlaza (pretpostavljene i stvarne izmjerene vrijednosti) kako bi se prema tome postigla i jednakost unutarnjih veličina što se ostvaruje **minimiziranjem razlike između procijenjene i izmjerene vrijednosti** izlaza tako da procijenjena vrijednost željenog unutarnjeg stanja sustava konvergira stvarnoj.

Zbog toga se uvodi sustav povratne veze gdje se greška odnosno odstupanje vrijednosti procjene i mjerenja pokušava eliminirati upotrebom matrice/pojačanja  $K$  estimatora. Pitanje koje se prirodno nameće jeste kako izabrati pojačanje da bi se optimalno minimizirala razlika procijenjenih i stvarnih veličina izlaza. Matematička podloga prethodno teorijski objašnjenog principa rada estimatora stanja izgleda ovako, gdje  $u$  označava ulaz,  $y$  predstavlja izlaz, a unutarnje stanje koje se želi estimirati dano je s  $x$  [11]:

Jednadžbe koje opisuju proces:

$$\dot{x} = Ax + Bu \quad (2-1)$$

$$y = Cx + Du \quad (2-2)$$

Jednadžbe estimatora:

$$\dot{\hat{x}} = A\hat{x} + Bu + K(y - \hat{y}) \quad (2-3)$$

$$\hat{y} = C\hat{x} + Du \quad (2-4)$$

Razlika vrijednosti  $x - \hat{x}$  označava se s  $e$ . Oduzimanjem jednadžbi (2-1) i (2-3) kao i (2-2) i (2-4) dobiva se izraz za dinamiku pogreške.

$$\dot{x} - \dot{\hat{x}} = Ax - A\hat{x} + Bu - Bu + K(y - \hat{y}) \quad (2-5)$$

$$y - \hat{y} = C(x - \hat{x}) + Du - Du \quad (2-6)$$

Sređivanjem izraza, izlučivanjem i uvođenjem  $e$  umjesto  $x - \hat{x}$  dinamika pogreške zadovoljava jednadžbu:

$$\dot{e} = (A - KC)e \quad (2-7)$$

$$y - \hat{y} = Ce \quad (2-8)$$

Rješenje jednadžbe (2-7) jest eksponencijalna funkcija  $e(t) = e^{(A-KC)t}e(0)$  što znači da ako je uvjet  $(A-KC) < 0$  tada će  $e$  težiti nuli tijekom vremena, točnije kada  $t$  teži u beskonačno, prema tome će i  $\hat{x}$  konvergirati prema  $x$ . Valja napomenuta da će i bez prisustva člana  $KC$  navedena eksponencijalna funkcija biti padajuća no, važnost postojanja pojačanja  $K$  unutar povratne veze jest ta da se može kontrolirati brzina pada eksponencijalne funkcije i osigurati brža eliminacija odstupanje/pogreške. Što prije pogreška opada ili se korigira, to brže procijenjena vrijednost konvergira stvarnoj vrijednosti [11]. Optimalni način izbora pojačanja  $K$  izvodi se upotrebom

Kalmanovog filtra kao jednog od najznačajnijih estimatora stanja dizajniranih za stohastičke sustave.

## 2.2. Izvod jednadžbi Kalmanovog filtra

Brojni procesi predstavljaju se determinističkim modelima. Međutim, ponašanje realnog sustava nikada nije potpuno determinističko jer na njega djeluju različiti poremećaji - šumovi. Proces koji u sebi sadržava elemente slučajnosti naziva se stohastički. Takav proces koji je vremenski diskretan i nije u potpunosti deterministički može se prikazati sljedećom linearnom stohastičkom jednadžbom stanja procesa [3, 12]:

$$x_k = Ax_{k-1} + Bu + w_k \quad (2-9)$$

I jednadžbom mjernog sustava:

$$y_k = Cx_k + v_k \quad (2-10)$$

Gdje je  $x_k$  vektor stanja u trenutku  $k$ ,  $A$  je tranzicijska matrica do stanja u trenutku  $k$  iz stanja u trenutku  $k-1$ , a zajedno sa  $x_k$  čini deterministički dio jednadžbe. Parametar  $u$  je vektor ulaza u koraku  $k$  dok  $B$  simbolizira matricu ulaza. Parametar  $w_k$  predstavlja šum procesa koji je slučajan i može biti prisutan zbog, npr. utjecaja vjetera ili ako se radi o procjeni lokacije automobila može označavati promjenu brzine vozila.

$y_k$  karakterizira vektor izlaznih veličina tj. mjerenja, matrica  $C$  opisuje povezanost između izlaza i stanja, a slučajni vektor  $v_k$  jest mjerni šum [13].

U cilju jednostavnije izvedbe pretpostavlja se da su matrice  $A$ ,  $B$  i  $C$  nepromjenjive u vremenu. Vektori  $w_k$  i  $v_k$  su međusobno neovisni bijeli šumovi, a pomoću teorije vjerojatnosti pretpostavlja se da se ponašaju prema određenoj funkciji gustoće vjerojatnosti koja je u slučaju Kalmanovog filtra Gaussova ili normalna funkcija razdiobe s nultim matematičkim očekivanjem i vrijedi izraz [3, 14] :

$$v \sim N(0, R) \quad (2-11)$$

što znači da ako se, npr. mjeri položaj automobila 100 puta na istoj lokaciji, a zatim se rezultati mjerenja prikazuju grafički, šum tih očitavanja će poprimiti vrijednosti koje su na pravcu uglavnom pozicionirane oko sredine (oko nule), dok se nekoliko očitavanja nalaze raspršena to jest udaljenija od nule što rezultira Gaussovom raspodjelom koja je opisana kovariancom. Budući da je riječ o

sustavu s jednim izlazom, kovarijanca je skalarna i jednaka je varijanci mjernog šuma. Slično vrijedi i za šum procesa. Pretpostavlja se da su kovarijance konstante[11].

$$w \sim N(0, Q) \quad (2-12)$$

Sada se zna da je sustav podložen šumu, te zato mjerenje ne odražava sasvim stvarno stanje sustava, npr. položaj automobila. Stoga ako je poznat matematički model sustava (automobila) ulaz u sustav se može provesti kroz njega kako bi se estimirala željena veličina u ovom slučaju položaj. Budući da se procjenjuje varijabla  $x_k$  koja je nepouzdana zbog postojanja procesnog šuma, sama procjena također neće biti savršena. Zbog toga u primjenu stupa Kalmanov filter koji **kombinira mjerenje i predikciju/procjenju** kako bi se došlo do adekvatne estimacije stanja sustava, tj. položaja automobila u prisustvu procesnog i mjernog šuma. Ova optimalna procjena pronalazi se uzimajući u obzir funkcije vjerojatnosti predviđanja i mjerenja, skaliranjem rezultata i izračunavanjem srednje vrijednosti rezultirajuće funkcije gustoće vjerojatnosti. U tu svrhu treba uvesti pojam Bayesovog pravila. Postupak se svodi na algoritam od dva koraka, a to su predikcija i korekcija. Korak kojim se dolazi do *a priori* procjene, a podrazumijeva predviđanje ponašanja sustava u narednom koraku naziva se predikcija nakon čega slijedi podešavanje predikcije na temelju podataka dobivenih mjerenjem čime se dolazi do *a posteriori* procjene. Za izvođenje Kalmanovog filtra, a prema Bayesu potrebno je uvesti neke nove varijable [3, 13, 14]

Na temelju stanja procesa u koraku k-1 izračunava se *a priori* estimacija koju se označava sa  $\hat{x}_k^- \in R^n$ , dok se na temelju poznatih rezultata mjerenja  $y_k$  izračunava *a posteriori* estimacija sa  $\hat{x}_k \in R^n$ . Greške *a priori* i *a posteriori* estimacije iskazuju se kao [13]:

$$\hat{e}_k^- = x_k - \hat{x}_k^- \quad (2-13)$$

$$\hat{e}_k = x_k - \hat{x}_k \quad (2-14)$$

Kovarijanca greške *a priori* estimacije dana je sa:

$$P_k^- = E[e_k^- e_k^{-T}] \quad (2-15)$$

Kovarijanca greške *a posteriori* estimacije:

$$P_k = E[e_k e_k^T] \quad (2-16)$$

Konačno na temelju funkcija gustoće vjerojatnosti predviđanja i mjerenja dolazi se do jednadžbe Kalmanovog filtra koja ima oblik:

$$\hat{x}_k = A\hat{x}_{k-1} + Bu_k + K_k(y_k - C(A\hat{x}_{k-1} + Bu_k)) \quad (2-17)$$

gdje prvi dio  $(A\hat{x}_{k-1} + Bu_k)$  predviđa trenutno stanje pomoću procjene stanja iz prethodnog vremenskog koraka i trenutnog ulaza – *a priori* procjena  $\hat{x}_k^-$ . Sada jednadžba prima ovaj oblik:

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - C\hat{x}_k^-) \quad (2-18)$$

gdje drugi dio jednadžbe koristi mjerenje  $y_k$  i pripaja ga predikciji  $\hat{x}_k^-$  kako bi se ažurirala ili korigirala *a priori* procjena nakon čega se dobiva rezultat tj. *a posteriori* procjena. Dakle, *a posteriori procjena* je linearna kombinacija *a priori* procjene i umnoška odgovarajućeg koeficijenta  $K$  to jest *matrice Kalmanovog pojačanja* i *razlike* između dobivenog mjerenja i predikcije koja se naziva *inovacija* [11, 13].

Optimalna vrijednost matrice Kalmanovog pojačanja koja daje najbolju estimaciju odnosno minimizira kovarijancu greške *a posteriori* estimacije određuje se tako da se najprije izraz (2-10) supstituira u (2-17). Dobiveni izraz uvrsti se u (2-16). Cilj jest minimizirati kovarijancu greške *a posteriori* estimacije što je jednako minimiziranju traga matrice navedene kovarijance. S obzirom da je trag matrice jednak tragu transponirane matrice, da bi se postiglo minimiziranje vrši se deriviranje s  $K$  te izjednačavanje s nulom a preko diferencijala konačno se dobiva izraz [3]:

$$K_k = \frac{P_k^- C^T}{C P_k^- C^T + R} \quad (2-19)$$

Ako se sada promotre dva krajnja slučaja, kada kovarijanca mjernog šuma  $R$  teži nuli i kada kovarijanca greške *a priori* procjene  $P_k^-$  teži nuli, utvrđuje se da će u prvom slučaju nakon uvrštavanja 0 na mjesto  $R$  izrazi u brojniku i nazivniku kratiti te će Kalmanovo pojačanje biti jednako inverznoj vrijednosti  $C$  što iznosi 1. Uvrštavanjem vrijednosti  $C^{-1}$  u izraz (2-18) dolazi se do zaključka da je  $\hat{x}_k = y_k$ , znači izračunavanje trenutnog stanja sustava potječe samo od mjerenja [3, 13]:

$$\lim_{R_k \rightarrow 0} K_k = C^{-1} \quad (2-20)$$

U drugom slučaju matrica  $K$  će biti nula, ponovnim uvrštavanjem u (2-18) dobiva se  $\hat{x}_k = \hat{x}_k^-$ . *A priori* estimacija stanja je velike točnosti te mjerenje nije potrebno uzimati u obzir jer ako je predikcija točna tada *a priori* procjenu nije potrebno korigirati [3, 11]:

$$\lim_{P_k^- \rightarrow 0} K_k = 0 \quad (2-21)$$

### 2.3. Algoritam Kalmanovog filtera

Postupak izvođenja primjene Kalmanovog filtera odvija se kako je prethodno navedeno, kroz dvije faze. Prvi korak ili faza je predikcija u kojoj se odvija preslikavanje stanja sustava iz prethodnog vremenskog koraka  $k-1$  u naredni korak  $k$  za što je zadužena grupa jednadžbi vremenskog osvježavanja. Time se dobiva *a priori* procjena stanja. Zatim se na temelju izvedenog mjerenja vrši korekcija *a priori* estimacije te se dobiva *a posteriori* procjena stanja pomoći jednadžbi osvježavanja mjerenjem. Nakon vremenske projekcije *a posteriori* procjena postaje *a priori* procjena narednog koraka. Zapravo se algoritam Kalmanovog filtera može objasniti beskonačnom petljom u kojoj se konstantno izmjenjuju dvije grupe jednadžbi s ciljem procjene trenutnog stanja sustava [13].

Jednadžbe vremenskog osvježavanja ili predikcije glase ovako:

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k \quad (2-22)$$

$$P_k^- = AP_{k-1}A^T + Q \quad (2-23)$$

Jednadžbe korekcije ili osvježavanja mjerenjem:

$$K_k = \frac{P_k^- C^T}{CP_k^- C^T + R} \quad (2-24)$$

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - C\hat{x}_k^-) \quad (2-25)$$

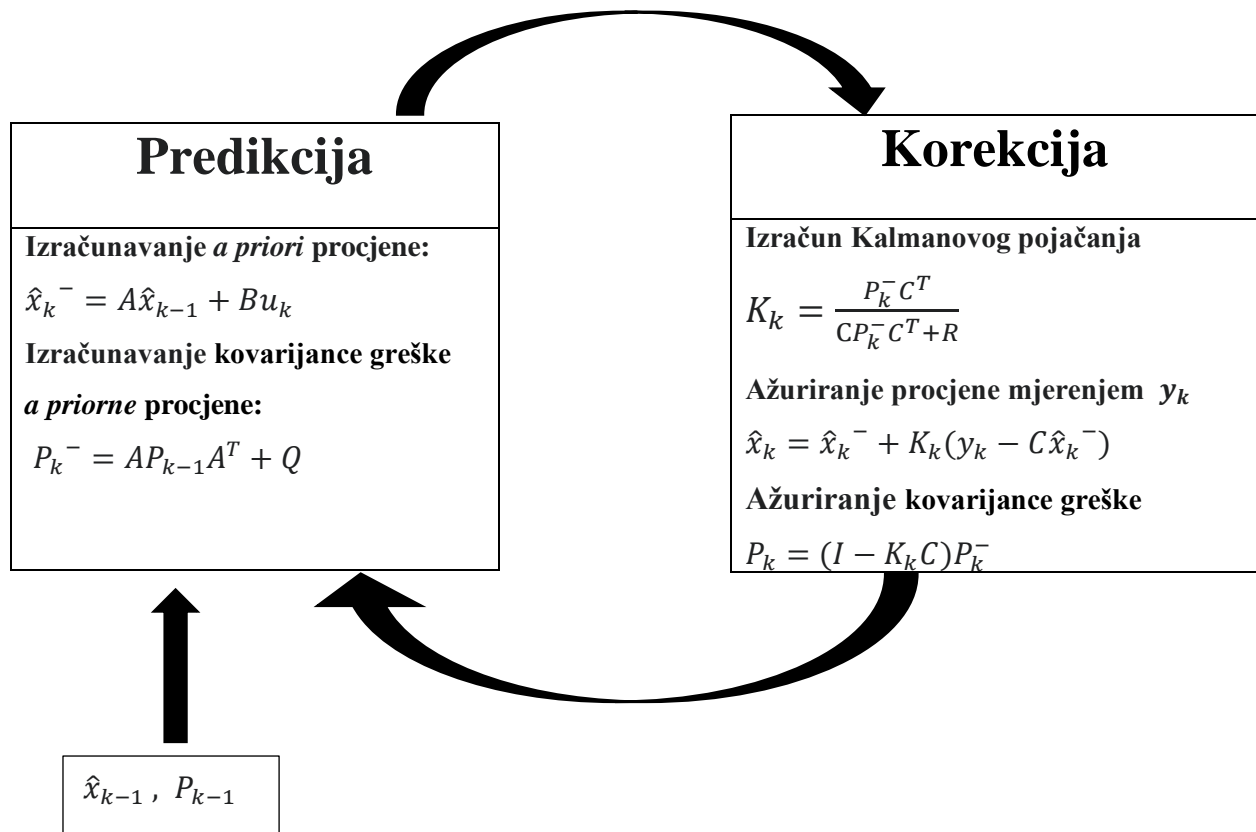
$$P_k = (I - K_k C)P_k^- \quad (2-26)$$

Iz ovih jednadžbi je uočljivo da za estimaciju trenutnog stanja sustava algoritam ne treba sve prošle informacije, jedino treba stanje sustava i kovarijancu greške *a posteriori* procjene prethodnog vremenskog koraka ili iteracije te trenutno mjerenje. To je ono što ustvari čini Kalmanov filter rekurzivnim.

#### 2.3.1. Faze izvođenja algoritma

Algoritam prolazi kroz nekoliko etapa od kojih prva podrazumijeva određivanje *a posteriori* procjene  $\hat{x}_{k-1}$  u koraku  $k-1$  te kovarijancu greške *a posteriori* procjene u istom vremenskom koraku  $P_{k-1}$ . Nakon toga se na temelju tih podataka vrši predviđanje stanja sustava u trenutku  $k$  čime se dobiva *a priori* procjena stanja  $\hat{x}_k^-$  (2-22) i kovarijanca pogreške *a priori* estimacije  $P_k^-$  (2-23). Slijedi izračun Kalmanovog pojačanja  $K$  i izvođenje mjerenja koje daje vrijednost  $y_k$ .

Konačno se postupak dobivanja željenog stanja završava korigiranjem *a priori* procjena pri čemu se vrijednosti prethodno spomenutih veličina uvrštavaju u izraze (2-25) i (2-26) te se dobivaju *a posteriori* vrijednosti koje su spremne za ulazak u novu iteraciju [3].



Sl. 2.1. Shema postupka estimacije stanja sustava

### 2.3.2. Vizualizacija algoritma

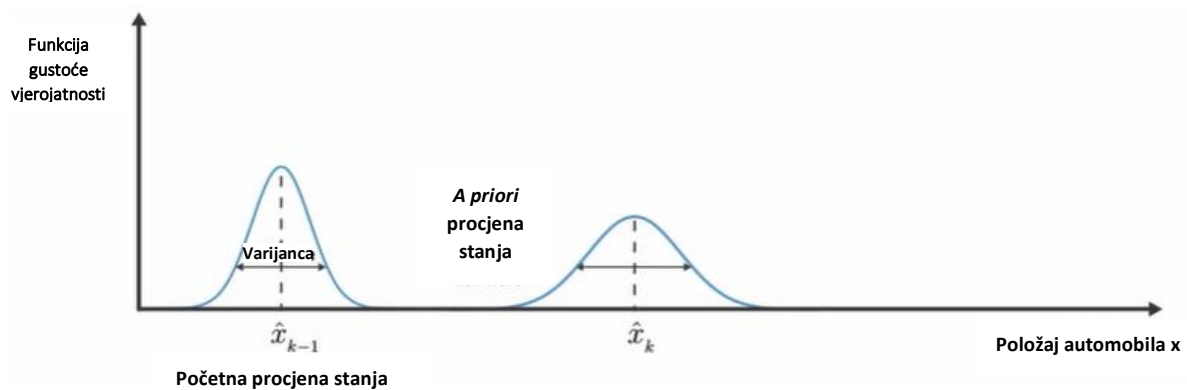
Do sada je ustanovljeno da se putem posrednih tehnika dolazi do vrijednosti veličina koje nisu izravno mjerljive. Posredne tehnike često opisuju preciznost određene informacije funkcijama vjerojatnosti. Algoritmi vjerojatnosne procjene stanja sustava računaju gustoću vjerojatnosti nad svim mogućim stanjima u kojima se objekt može naći u trenutnom koraku. Princip rada Kalmanovog filtra može se objasniti pomoću funkcije gustoće vjerojatnosti. U svrhu lakšeg razumijevanja razmotrit će se primjer određivanja položaja automobila predložen grafičkim prikazom ovisnosti funkcije gustoće vjerojatnosti o položaju u određenom trenutku  $k - x_k$  [11, 14].

Prikazana zvonolika funkcija gustoće vjerojatnosti (f.g.v.) daje procjenu položaja-stanja u trenutku  $k-1$ . Najveća vrijednosti f.g.v. na grafu prikazana okomitom crtkanom linijom, odgovara najboljoj estimaciji položaja tj. automobil se vjerojatno nalazi u sredini prikazane razdiobe.



Sl. 2.2. Grafički prikaz f.g.v. položaja automobila u koraku k-1 [11]

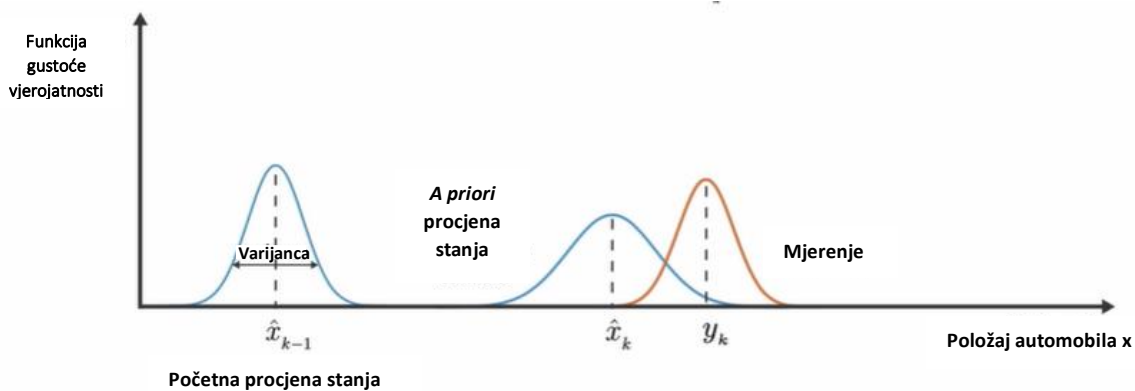
U sljedećem vremenskom koraku  $k$  tj. koraku predikcije, na osnovu prethodnog stanja u koraku  $k-1$  i njegove kovarijanse greške *a posteriori* procjene dobiva se f.g.v. pomaknuta prema modelu gibanja no nesigurnost u procjeni raste što se prikazuje većom varijancom (desni graf na slici 2.3.) – *a priori* procjena. Razlog porasta varijance (grafički šira zvonolika funkcija) leži u nepreciznosti modela koje mogu biti i korisne. Npr. između koraka  $k-1$  i  $k$  automobil može preći drugačiju udaljenost nego što je predviđeno modelom zbog primjerice ubrzavanja ili usporavanja uzrokovanih reljefom ili mogućim preprekama na putu [11, 14].



Sl. 2.3. Grafički prikaz f.g.v. *a priori* procjene položaja automobila u koraku  $k$  [11]

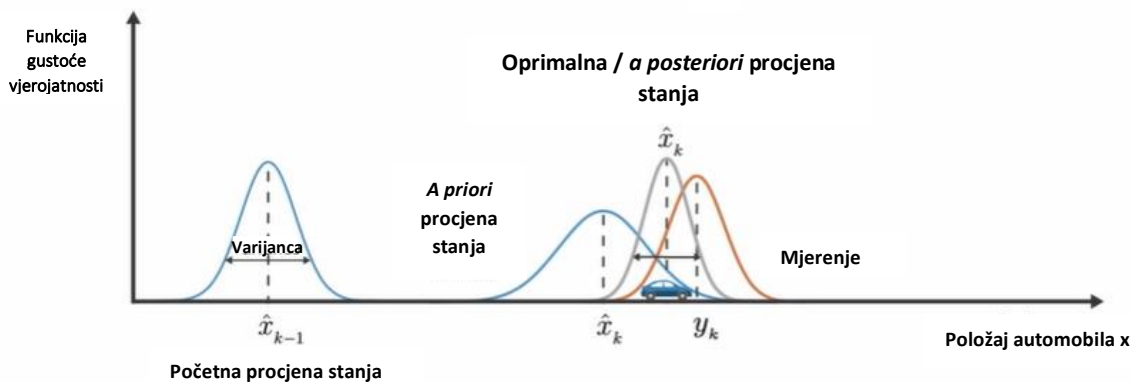
Nakon drugog koraka novi izvor informacija o položaju automobila dolazi od mjerenja  $y_k$ . Varijancu funkcije gustoće vjerojatnosti mjerenja predstavlja prisustvo šumova.





Sl. 2.4. Grafički prikaz f.g.v. mjerenja u koraku  $k$  [11]

Uzimanjem u obzir funkciju gustoće vjerojatnosti predikcije (na grafu *a priori* procjena stanja) i mjerenja- korekcija, dobiva se nova f.g.v. koja ima manju varijancu od bilo koje ranije estimacije i sredina nove funkcije daje optimalnu procjenu pozicije vozila – *a posteriori* procjena [11, 14].



Sl. 2.5. Grafički prikaz f.g.v. *a posteriori* procjene položaja automobila u koraku  $k$  [11]

## 2.4. Primjer MATLAB simulacije

Kako je već navedeno kroz rad, postoje brojne vrste i implementacije Kalmanovog filtra a primjer u programskom jeziku visoke razine odnosi se na primjenu stacionarnog Kalmanovog filtra na vremenski diskretan sustav. Cilj primjera jest usporediti filtrirani odziv odnosno izlaz sa stvarnim izlazom procesa. Model u prostoru stanja je matematički prikaz fizičkog sustava kao skupa ulaznih, izlaznih i varijabli stanja povezanih diferencijalnim jednadžbama prvog reda. Varijable stanja definiraju vrijednosti izlaznih varijabli. Kontinuirani linearni vremenski nepromjenjivi sustavi u prostoru stanja opisuju se sada već poznatim jednadžbama (2-1) i (2-2). Vremenski diskretni sustavi u prostoru stanja mogu se prikazati sljedećim jednadžbama gdje je  $n$  korak uzorkovanja:

$$x[n + 1] = Ax[n] + Bu[n] + Gw[n] \quad (2-27)$$

$$y[n] = Cx[n] + Du[n] + Hw[n] + v[n] \quad (2-28)$$

Dakle, cilj je dizajnirati Kalmanov filter kako bi se procijenila stvarna vrijednost izlaza procesa  $y_t[n] = y[n] - v[n]$  na temelju mjerenja u prisustvu šuma  $y[n]$  gdje  $v[n]$  predstavlja mjerni šum točnije, kako bi se pokazao utjecaj filtra na **pogrešku** uzrokovanu mjernim šumom. Za tu procjenu, stacionarni Kalmanov filter koristi iduće izraze:

$$\hat{x}[n + 1|n] = A\hat{x}[n|n - 1] + Bu[n] + Bw[n] \quad (2-29)$$

$$\hat{x}[n|n] = \hat{x}[n|n - 1] + M_x(y[n] - C\hat{x}[n|n - 1]) \quad (2-30)$$

$$\hat{y}[n|n] = C\hat{x}[n|n] \quad (2-31)$$

gdje je  $\hat{x}[n + 1|n]$  procijenjena vrijednost  $x[n]$  s obzirom na prošla mjerenja do  $y[n]$ , analogno tome i  $\hat{x}[n|n - 1]$  je procijenjena vrijednost  $x[n]$  s obzirom na prošla mjerenja do  $y[n-1]$ .

$\hat{x}[n|n]$  i  $\hat{y}[n|n]$  su procijenjene vrijednosti stanja i mjerenja ažurirane na temelju posljednjeg mjerenja a  $M_x$  je optimalno inovacijsko pojačanje odabrano za minimiziranje stacionarne kovarijance pogreške procjene [15, 16].

Na samom početku izvedbe simulacije najprije je potrebno unutar *Command Window*-a MATLAB-a (vidjeti kôd) definirati vrijednosti matrica A, B, C i D koje su konstantne i vremenski neovisne. Matrica stanja A je dimenzija  $N_x \times N_x$  odnosno 3x3. Matrica B koja povezuje stanja i ulaze je dimenzija  $N_x \times N_u$  tj. 3x1. C povezuje stanja i izlaze dimenzija  $N_y \times N_x = 1 \times 3$  a matrica D koja povezuje izlaze i ulaze dimenzija  $N_y \times N_u$  jednaka je 0 budući da ulazni šum, šum procesa, nema direktnog utjecaja na izlaz.

<b>Linija</b>	<b>Kod</b>
1:	>> A = [1.1269    -0.4940    0.1129 1.0000            0            0 0    1.0000            0];
2:	>> B = [-0.3832 0.5919 0.5191];
3:	>> C = [1 0 0];
4:	>> D = 0;

Također je postavljena varijabla za vrijeme uzorkovanja (u kodu Ts) koja označava da je proces diskretan, a vrijednost -1 naznačava da vrijeme uzorkovanja nije točno određeno.

<b>Linija</b>	<b>Kod</b>
5:	>> Ts = 1;

Naredni korak podrazumijeva specificiranje modela procesa s aditivnim šumom w. U MATLAB-u se ovakvi procesi mogu generirati pomoću naredbe *sys* čija definicija za model u prostoru stanja izgleda ovako: *sys = ss(a,b,c,d)*, (prostor stanja eng. *state-space* - SS). Unutar naredbe definirani su ulazi procesa preko '*InputName*', {'u' 'w'}, te izlazi procesa '*OutputName*', 'y' [16].

<b>Linija</b>	<b>Kod</b>
6:	>> sys = ss(A, [B B], C, D, Ts, 'InputName', {'u' 'w'}, 'OutputName', 'y');

Kovarijance procesnog šuma  $Q$  i mjernog šuma  $R$  su skalarne vrijednosti veće od nula kako je i navedeno u kôdu. Kalmanov filter u stacionarnom stanju koji sadrži implementirane jednadžbe predikcije i korekcije nazvan sa `kalmf`, projektira se pomoću naredbe `kalman` kojoj se kao argumenti predaju prethodno definirani model procesa te kovarijance.

<b>Linija</b>	<b>Kod</b>
7:	<code>&gt;&gt; Q = 2.3;</code>
8:	<code>&gt;&gt; R = 1;</code>
9:	<code>&gt;&gt; [kalmf, L, ~, M, Z] = kalman(sys, Q, R);</code>

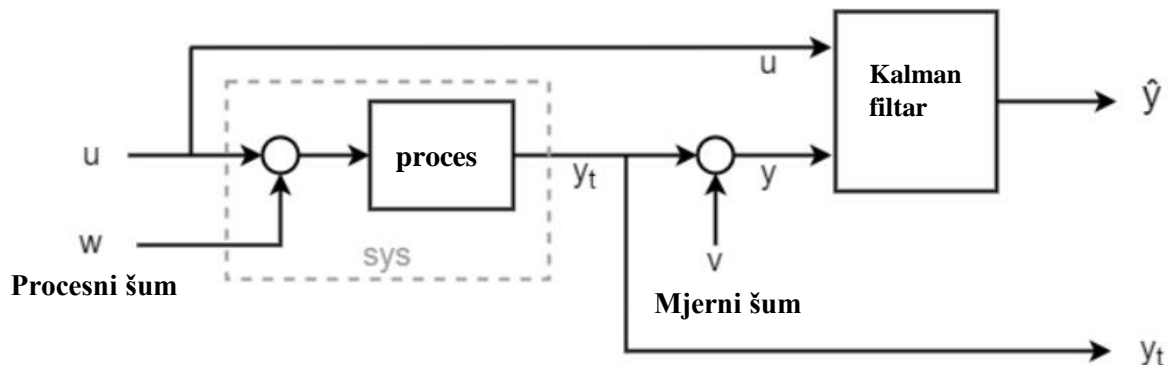
Kako bi se izlazu procesa dodao i utjecaj mjernog šuma v koristi se naredba `sumblk`. Ulazi u filter su ulaz procesa i izlaz procesa sa prisustvom procesnog i mjernog šuma a da bi se to ostvarilo potrebno je povezati `sys` i filter za što se u MATLAB-u koristi naredba `connect` pri čemu se dobiva model za simulaciju [15].

<b>Linija</b>	<b>Kod</b>
11:	<code>&gt;&gt; sys.InputName = {'u', 'w'};</code>
12:	<code>&gt;&gt; sys.OutputName = {'yt'};</code>
13:	<code>&gt;&gt; vIn = sumblk('y=yt+v');</code>
14:	<code>&gt;&gt; kalmf.InputName = {'u', 'y'};</code>
15:	<code>&gt;&gt; kalmf.OutputName = 'ye';</code>
16:	<code>&gt;&gt; SimModel = connect(sys, vIn, kalmf, {'u', 'w', 'v'}, {'yt', 'ye'});</code>

Prvi odziv ovako definiranog filtra jest procijenjena/filtrirana vrijednost izlaza  $\hat{y}$  tj. u kodu  $y_e$  a ostali su procijenjena stanja  $\hat{x}$ . Budući da je u primjeru od interesa vrijednost izlaza, samo je ona zadržana.

<b>Linija</b>	<b>Kod</b>
10:	<code>&gt;&gt; kalmf = kalmf(1, :);</code>

Kompletan prikaz sustava koji se simulira dan je na slici 2.6.



Sl. 2.6. Blokovski dijagram sustava za simulaciju [15]

Da bi se simuliralo filtriranje generirani su signal ulaza u obliku poznatog sinusnog vektora kao i vektori šumova procesa i mjerenja koristeći iste vrijednosti kovarijanci  $Q$  i  $R$  te naredbu *randn* koja generira slučajne skalare iz standardne normalne razdiobe.

<b>Linija</b>	<b>Kod</b>
17:	>> t = (0:100)';
18:	>> u = sin(t/5);
19:	>> rng(10, 'twister');
20:	>> w = sqrt(Q)*randn(length(t),1);
21:	>> v = sqrt(R)*randn(length(t),1);

Pomoću naredbe *lsim* simulira se odziv odnosno izlaz za korištene ulaze, nakon čega se izluči stvarni izlaz, filtrirani (procijenjeni) izlaz te se računa izmjereni izlaz (pod utjecajem mjernog šuma).

<b>Linija</b>	<b>Kod</b>
22:	>> out = lsim(SimModel, [u,w,v]);
23:	>> yt = out(:,1); % stvarni izlaz
24:	>> ye = out(:,2); % filtrirani izlaz
25:	>> y = yt + v; % izmjereni izlaz

Konačno pomoću krajnjih nekoliko linija koda (vidjeti linije 26-33) dolazi se do slikovite usporedbe stvarnog izlaza procesa i procijenjenog izlaza dobivenog pomoću filtra gornji grafički prikaz na slici 2.7. Pored toga dobiva se prikaz usporedbe pogrešaka. Pogreške koja se dobiva kao razlika stvarnog i mjerelog izlaza, te pogreške koja je jednaka razlici stvarnog i filtriranog izlaza.

**Linija Kod**

```

26:  >> clf
27:  >> subplot(211), plot(t,yt,'g',t,ye,'b--'),
28:  >> xlabel('Broj uzoraka'), ylabel('Izlaz')
29:  >> title('Odziv Kalmanovog filtra - Marija Mostarac')
30:  >> legend('Stvarni izlaz','Filtrirani izlaz')
31:  >> subplot(212), plot(t,yt-y,'y',t,yt-ye,'r--'),
32:  >> xlabel('Broj uzoraka'), ylabel('Pogreška')
33:  >> legend('Stvarni - mjereni izlaz','Stvarni - filtrirani izlaz')

```

Može se vidjeti da je primjerom dokazana zapravo definicija samog filtra, kako drugi graf na slici 2.7. pokazuje, Kalmanov filter evidentno smanjuje pogrešku točnije, minimizira razliku između izlaza  $y_t - y$  koja nastaje zbog prisustva mjernog šuma.



Sl. 2.7. Simulacijski odziv Kalmanovog filtra

Da bi se potkrijepila te tvrdnja odnosno za potvrdu smanjenja pogreške primijenjene su naredne linije kôda upravo za izračun pogreške kovarijance prije filtriranja i pogreške kovarijance nakon filtriranja [15]:

**Linija      Kod**

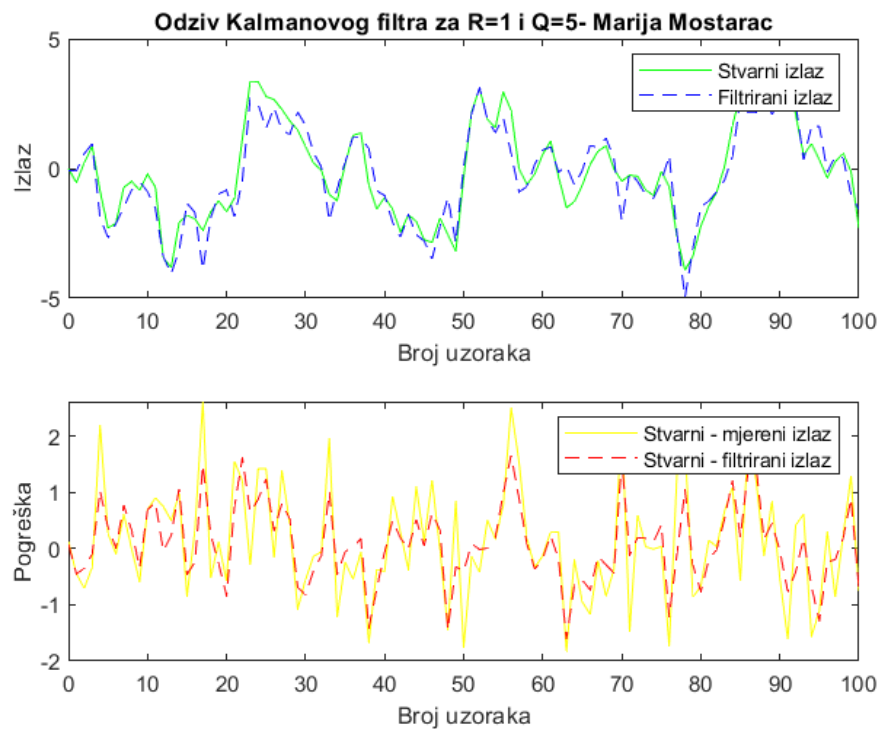
```
34:  >> PogrMjer = yt-y;
35:  >> PogrKovMjer = sum(PogrMjer.* PogrMjer)/length(PogrMjer)
36:  >> PogrFilt = yt-ye;
37:  >> PogrKovFilt = sum(PogrFilt.*PogrFilt)/length(PogrFilt)
38:  >> PogrKovMjer =
```

0.9871

```
39:  >> PogrKovFilt =
```

0.3479

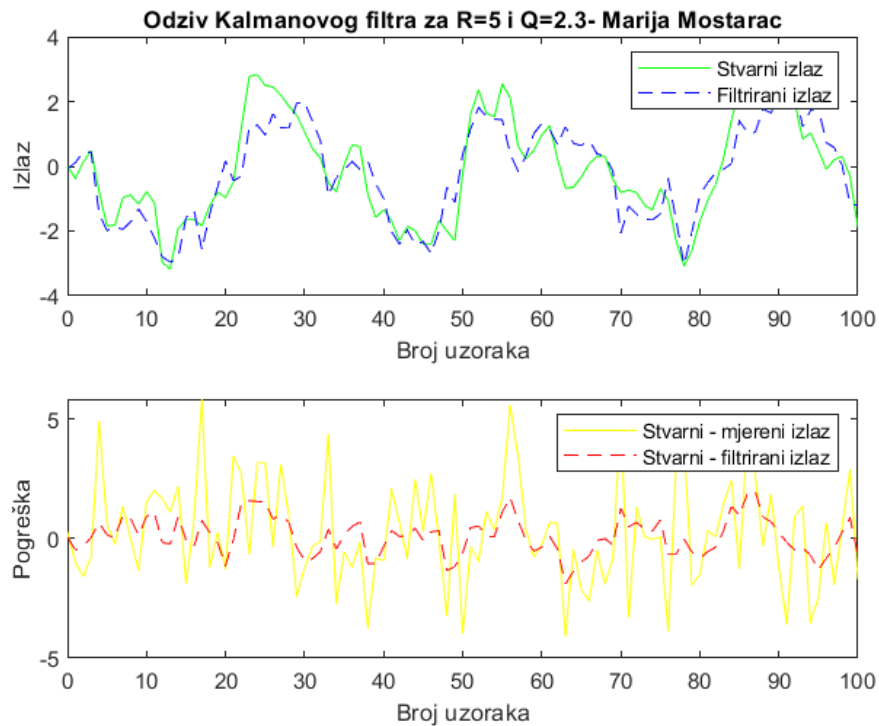
Pokazano je da je kovarijanca greške mjerenja veća nego kovarijanca greške filtriranja za odabrane vrijednosti parametara. Sada će se promotriti što se postiže, odnosno mijenja promjenom vrijednosti varijabli  $R$  i  $Q$ . Neka je  $R$  zadržao prvobitno postavljenu vrijednost u iznosu od 1 a vrijednost  $Q$  je povećana na 5, simuliranjem novonastalog modela dobiva se odziv prikazan na slici 2.8.



Sl. 2.8. Simulacijski odziv Kalmanovog filtra za isti iznos  $R$  i povećan  $Q$

Primjenom linija 34-38 iznos pogreške kovarijance mjerenja prima vrijednost 0.9871 i filtriranja 0.4816. Dakle, kovarijanca pogreške mjerenja zadržala je istu vrijednost prilikom upotrebe iste vrijednosti kovarijance mjernog šuma  $R = 1$ . Kovarijanca pogreške filtriranja povećanjem varijable  $Q$  također se povećala u odnosu na prethodni primjer što je vidljivo i na slici 2.8.

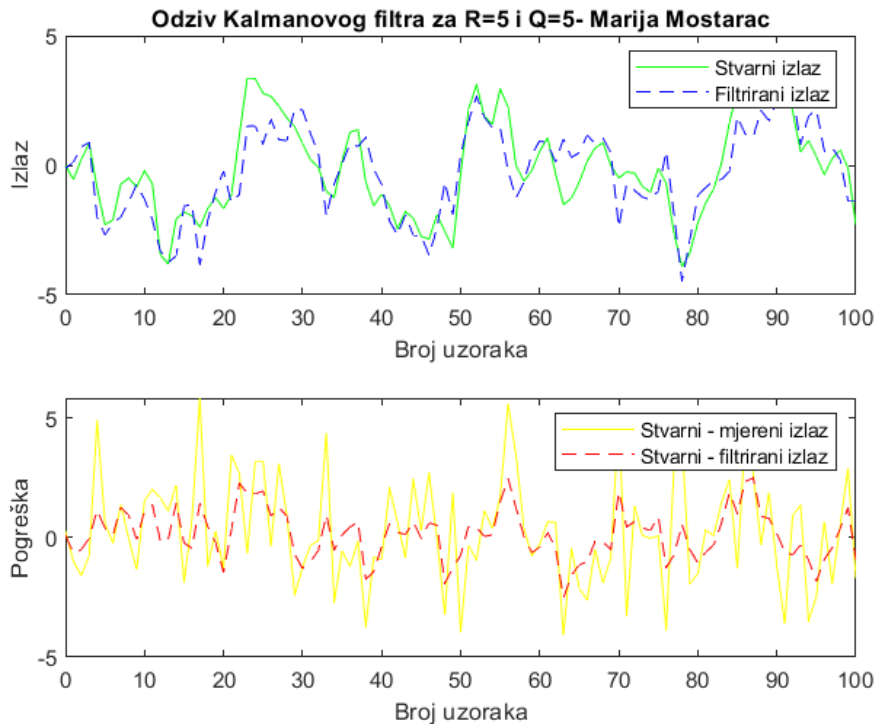
Ako se sada poveća  $R$  tako da mu je novi iznos 5 a  $Q$  zadrži istu vrijednost od 2.3, pogreška kovarijance mjerenja se poveća i sada je 4.9355, a filtriranja 0.6265.



Sl. 2.9. Simulacijski odziv Kalmanovog filtra za povećan iznos  $R$  i isti  $Q$

Povećanjem iznosa kovarijance i mjernog i procesnog šuma na vrijednost 5 greška kovarijance mjerenja iznosi 4.9355 a greška kovarijance filtriranja 1.0938.





Sl. 2.10. Simulacijski odziv Kalmanovog filtra za povećan iznos  $R$  i povećan  $Q$

U svim primjerima filter uspješno smanjuje pogrešku no, nakon proučavanja dobivenih rezultata simulacije s obzirom na promjene vrijednosti spomenutih varijabli pokazano je da povećanjem iznosa kovarijance mjernog šuma  $R$  i procesnog šuma  $Q$  dobivamo lošije rezultate. Preciznije rečeno, svrha filtra jeste da greška filtriranja bude što manja a prilikom povećanja iznosa kovarijance mjernog i procesnog šuma iznos kovarijance pogreške filtriranja raste u odnosu na prvi dati primjer. Također se opaža da primjenom većeg iznosa kovarijance procesnog šuma  $Q$  dobiveni stvarni signal uvelike odstupa od sinusnog valnog oblila. Povećanjem iznosa kovarijance mjernog šuma povećava se iznos pogreške mjerenja. Sve ovo navodi na razmišljanje da bi smanjenjem iznosa varijable  $Q$  stvarni signal bio približniji sinusnom valnom obliku kao što bi smanjenjem kovarijance mjernog šuma  $R$  filtrirani signal više konvergirao stvarnom.

### 3. FUZIJA SENZORA

Fuzija senzora je sastavni dio dizajna autonomnih sustava poput, npr. automobila koji se sami voze, radarskih stanica za praćenje objekata i brojnih drugih. Jedna od mogućih definicija fuzije senzora kaže da se navedeni termin odnosi na kombiniranje dvaju ili više izvora informacija/podataka tako da se postiže bolje razumijevanje sustava. Bolje razumijevanje sustava odnosi se može se reći, na rješenje ili izlaz koje je s vremenom dosljednije, točnije i pouzdanije nego što bi bilo da se dobiva putem mjerenja samo iz jednog izvora informacija. Izvor podataka su uglavnom senzori, a mjerenja dobivena iz istih pružaju upravo mogućnost razumijevanja sustava, kao npr. ubrzanje automobila ili udaljenost do nekoga objekta. Autonomni sustav treba interakciju sa svijetom oko njega i ukoliko želi biti uspješan postoje određene sposobnosti koje mora posjedovati a koje se mogu podijeliti u četiri glavna dijela. [17] Prvi dio podrazumijeva direktna mjerenja veličina iz okruženja pomoću senzora (engl. *Sense*). Ovaj dio **prikuplja informacije** iz sustava i vanjskog svijeta, za samovozeće automobile može uključivati radare, kamere i dr. Samo prikupljanje podataka iz senzora nije dovoljno dobro jer sustav mora biti sposoban interpretirati podatke i pretvoriti ih u oblik koji može biti razumljiv i koristan za djelovanje autonomnog sustava. Za to je zaslužan dio za zapažanje (engl. *Perceive*) koji **analizira/tumači podatke** primljene od senzora i daje im smisao. Idući dio je dio za planiranje (engl. *Plan*) koji shvaća što sustav želi učiniti ili dokle stići i **pronalazi put** za dostizanje cilja. Na kraju sustav izračunava najbolje radnje koje će ga na neki način natjerati da **slijedi taj put** što je zadaća kontrolera. Dio koji se bavi obradom podataka sadrži dvije važne odgovornosti. On je odgovoran za lokalizaciju i pozicioniranje sustava, daje odgovore na pitanja kao što su npr. gdje se nalazi sustav, što radi i slično. Također je odgovoran i za svijest o situaciji kao što je npr. otkrivanje drugih objekata u blizini i njihovo praćenje. Tu do izražaja dolazi upravo fuzija senzora jer ona ima doticaja sa obje navedene sposobnosti. To je proces uzimanja mjerenja iz više senzora, kombinirajući ih i pripajajući im informacije iz matematičkog modela s ciljem boljeg razumijevanja svijeta s kojim sustav može djelovati [17].

### 3.1. Odlike fuzije senzora

Postoje četiri različita načina na koje fuzija senzora može pomoći za obavljanje boljeg posla pri lokalizaciji i pozicioniranju sustava kao i otkrivanju i praćenju drugih objekata. Jedan od glavnih razloga primjene fuzije senzora jest **povećanje kvalitete podataka**. Uvijek je poželjno raditi s podacima koji imaju manje ili nikako zastupljeno prisustvo šuma, niži stupanj nesigurnosti i odstupanja od stvarnih vrijednosti. Primjer povećanja kvalitete podataka može se promotriti kroz upotrebu akceleratora. U pitanju je akcelerator položen na stol dakle, mjeri samo gravitacijsko ubrzanje. Da je ovaj senzor savršen izlaz bi imao konstantnu vrijednost u iznosu od  $9.81 \text{ m/s}^2$ , međutim stvarno mjerenje posjeduje šum. U kojoj količini će biti zastupljen šum ovisi o kvaliteti senzora i to prisustvo je nepredvidivo tj. ne može se ukloniti kalibracijom. Ako se doda još jedan akcelerator, može se smanjiti sveukupna količina šuma u signalu izračunavanjem prosječnog izlaza dvaju očitavanja. Sve dok šum nije u korelaciji preko senzora uzimajući izlaze obaju senzora smanjuje se kombinirani šum za faktor kvadratnog korijena broja senzora.[17] Dakle, u ovom slučaju sve što čini ovaj vrlo jednostavan algoritam fuzije je funkcija prosjeka.

Također se može smanjiti šum kombiniranjem mjerenja iz različitih vrsta senzora što može biti od koristi ako su korelirani izvori šuma. Na primjer, prilikom pokušaja mjerenja smjera mobilnog telefona prema sjeveru. Može se koristiti magnetometar pametnog telefona koji služi za određivanje geografskog smjera mjerenjem magnetskog polja. Uzimajući u obzir vrijednost jakosti magnetskog polja u ravninama X, Y i Z, aplikacije u pametnim telefonima mogu izračunati neto magnetsko polje oko telefona i korisniku dati okvirnu ideju o orijentaciji u 3D prostoru ali to će mjerenje također biti uz prisustvo šuma. Ako se namjerava smanjiti šum, može se posegnuti za dodavanjem drugog magnetometra. Međutim, doprinos šumova dolazi od pokretnih magnetskih polja koje stvara elektronika unutar samog telefona. To znači da će ovaj korelirani izvor šuma utjecati na svaki magnetometar, pa ako se izračuna prosjek mjerenja koja potječu od ovih dvaju senzora šum se neće smanjiti niti ukloniti. Jedno od rješenja problema jeste pomaknuti senzore iz dosega magnetnog polja što je teško učiniti s pametnim telefonom ili se može filtrirati mjerenje uporabom neke vrste filtra ali koji bi učinio mjerenje manje osjetljivim. Optimalna opcija u ovom primjeru jest napraviti fuziju magnetometra sa senzorom kutne brzine koji također sadrži šum ali upotrebom navedena dva različita tipa senzora smanjuje se vjerojatnost da su šumovi u korelaciji pa se mogu koristiti za međusobno kalibriranje. Postoji nekoliko različitih fuzijskih algoritama koji se mogu primijeniti za postizanje spomenutog kombiniranja ali Kalmanov je jedna od češćih i učinkovitijih metoda [17,18].

Iduća dobrobit fuzije senzora jest ta što **može povećati pouzdanost**. Ako se kao u primjeru akceleratora koriste dva jednaka senzora tada se u slučaju da jedan zakaže neće izgubiti mjerenje u potpunosti, naravno bit će manje kvalitete. Može se dodati i treći senzor iste vrste i algoritam fuzije će davati podatke bilo kojeg pojedinačnog senzora koji proizvodi mjerenje. U slučaju da jedan senzor iz bilo kojih razloga postane neupotrebljiv i dalje se posjeduju podatci iz preostala dva senzora. Dupliranje senzora učinkovit je način povećavanja pouzdanosti, no potreban je oprez jer kvar jednog senzora može utjecati na sve senzore u isto vrijeme. Tada fuzija senzora neće poboljšati mjerenje. Ovdje također može pomoći fuzija različitih senzora jer uzrok kvara jedne vrste senzora neće naškoditi drugoj vrsti senzora koja je otporna na takvu vrstu kvara te će informacija o mjerenju ostati očuvana [17].

Treća prednost fuzije senzora jeste **procjena veličina i stanja koja se ne mogu direktno mjeriti**. Nemjerena stanja ne znači da se ona ne mogu mjeriti općenito nego znači samo da sustav nema senzor za direktno mjerenje željene veličine. Npr. kamera ne može izmjeriti udaljenost do objekta u svom vidnom polju, veliki objekt koji je na većoj udaljenosti može imati jednak broj piksela kao i mali objekt na bližoj poziciji odnosno na manjoj udaljenosti. Dodavanjem druge kamere, algoritam fuzije sada dva senzora, će uspoređivati viđeno iz dva različita kuta i mjeriti relativne udaljenosti između objekata na dvije slike. Dakle, ovi senzori pojedinačno ne mogu mjeriti udaljenosti ali kombiniranjem mjerenja iz oba senzora to jest usporedbom slika od obje kamere dobiva se informacija o udaljenosti - dolazi se do željene veličine indirektnim mjerenjem [17].

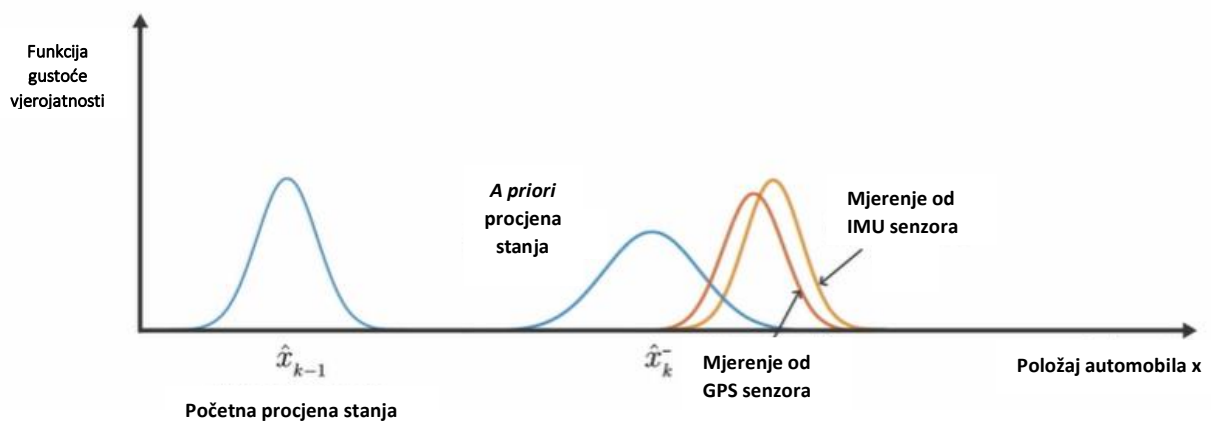
Fuzija senzora **povećava područje pokrivenosti**. Senzori kratkog dometa na automobilu koji se koriste za pomoć pri parkiranju su senzori koji mjere udaljenost do obližnjih objekata poput drugih parkiranih automobila kako bi vozača obavijestili kada se nalaze u neposrednoj blizini u svrhu zaštite od mogućeg oštećenja vozila. Svaki pojedinačni senzor može imati mali ograničen domet i usko vidno polje. Ako automobil mora imati potpunu pokrivenost sa sve četiri strane, potrebno je dodati dodatne senzore i kombinirati mjerenja kako bi se dobilo veće ukupno vidno polje. Izlaz se vjerojatno neće računati kao prosječno mjerenje jer je obično korisno znati koji senzor registrira objekt kako bi se dobila informacija gdje se taj objekt nalazi u odnosu na automobil. Algoritam koji spaja sve ove senzore zajedno u jedan koherentan sustav jest upravo fuzija senzora [17].

### 3.2. Algoritam fuzije senzora

Algoritam koji stoji iza ideje kombiniranja mjerenja iz više senzora za postizanje točnijeg krajnjeg rezultata ili procjene zapravo je jednak algoritmu koji se odnosi na Kalmanov filter, ranije opisan u sastavu poglavlja 2.3., 2.3.1, i 2.3.2. Cilj algoritma za fuziju senzora je proizvesti vjerodostojnu procjenu stanja objekta. Dakle, koriste se dva koraka s pripadnim jednačbama, a to su predikcija i korekcija. Predikcija uzima u obzir model kretanja i bavi se dinamikom objekta, npr. automobila - kroz korake u vremenu. Trenutno stanje automobila proizlazi iz informacija o njegovom stanju tijekom posljednjeg vremenskog koraka. Ovo predviđanje se zatim ažurira putem jednačbi ažuriranja/korekcije koje se bave dinamikom senzora, kombiniranjem pretpostavljenih vrijednosti iz koraka predikcije i izmjerenih vrijednosti dobivenih iz senzora. Na kraju se dobiva novi raspon mogućih vrijednosti stanja, koji se pretvara u ulaz za novu jednačbu predviđanja - i opet se kombinira sljedeće mjerenje senzora kako bi se ažuriralo predviđanje. To je upravo ono što je izneseno u poglavljima 2.3. i 2.3.2 u primjeru automobila. Budući da su algoritmi Kalmanovog filtra najraširenija primjena fuzije senzora i pružaju temelj za razumijevanje samog koncepta, fuzija senzora često je sinonim za Kalmanovo filtriranje [18].

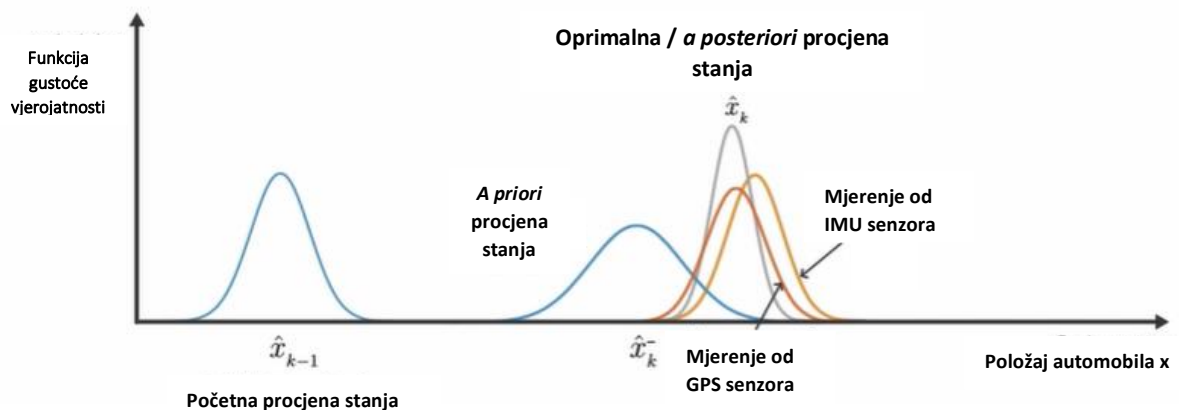
Za bolje razumijevanje algoritma fuzije te cjelokupne teorijske podloge koja se nalazi iza te ideje, promotrit će se ponovno primjer određivanja pozicije automobila ovoga puta korištenjem informacija iz više senzora. Jedan od senzora koji pomaže automobilu za određivanje lokacije jeste inercijska mjerna jedinica – IMU (engl. *Inertial measurement unit*) koja koristi brzinomjer i žiroskop za određivanje ubrzanja i kutne brzine automobila. Drugi senzor je odometar koji mjeri prijeđenu relativnu udaljenost, tj. broji okretaje kotača i prikazuje ih u obliku broja prijeđenih kilometara, on osigurava relativnu poziciju automobila. Treći senzor je globalni sustav za pozicioniranje (engl. *Global Positioning System, GPS*) - prijemnik koji prima signal od satelita za određivanje lokacije automobila na površini Zemlje i pruža apsolutnu poziciju automobila. Ukoliko automobil na putu do željenog cilja prolazi kroz tunel, postaje teško procijeniti njegovu poziciju preko GPS-a budući da je signal unutar tunela jako slab. U ovom slučaju informacija se uzima od IMU senzora koji daje ubrzanje pa da bi se došlo do pozicije potrebno je izvesti dvostruki integral ubrzanja. Jednostruki integral ubrzanja daje brzinu, potom integriranjem brzine dobiva se položaj. Ova informacija nije u potpunosti točna zbog pogrešaka koje se akumuliraju tijekom vremena. Za dobivanje bolje procjene stanja može se kombinirati mjerenje od IMU senzora zajedno sa mjerenjem odometra. Problem je što na mjerenje odometra može utjecati pritisak u gumama kotača automobila te stanje na cesti. Sveukupno, koristeći mjerenja iz IMU

senzora i odometra dobiva se relativna pozicija automobila. Ovi senzori daju češće ažuriranje podataka ali odstupaju od konstantnih vrijednosti dok GPS senzor daje rjeđe ažuriranja koja mogu biti u prisustvu šuma ali njime se dobiva potpuna lokacija. U ovom slučaju Kalmanov filter se koristi za optimalnu procjenu pozicije automobila fuzijom mjerenja svih senzora [11]. Za zaokruživanje kompletne priče, promotrit će se grafički prikaz iznesenih tvrdnji. Slika 3.1. prikazuje da se kako je navedeno u prvom koraku najprije koristi procjena stanja tijekom posljednjeg vremenskog koraka  $k-1$  da bi se predikcijom dobila procjena stanja u koraku  $k$ . Zatim se predikcija ažurira korekcijom, gdje se novi izvor informacija dobiva od mjerenja dvaju senzora GPS-a i IMU-a.



Sl. 3.1. Grafički prikaz f.g.v. mjerenja dvaju senzora u koraku  $k$  [11]

Ovoga puta s pomoću triju funkcija gustoće vjerojatnosti (*a priori* procjena stanja, mjerenje od GPS senzora i mjerenje od IMU senzora) dobiva se optimalna procjena stanja sustava – *a posteriori* procjena položaja automobila.



Sl. 3.2. Grafički prikaz f.g.v. *a posteriori* procjene u koraku  $k$

### 3.2.1. Prilagodba jednadžbi Kalmanovog filtra

Ukoliko postoje dva mjerenja koja potječu od dvaju senzora izraz (2-18) sada poprima novi oblik:

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - C\hat{x}_k^-) \quad \longrightarrow \quad \hat{x}_{k[1x1]} = \hat{x}_{k[1x1]}^- + K_{k[1x2]}(y_{k[2x1]} - C_{[2x1]}\hat{x}_{k[1x1]}^-)$$

Uočljivo je iz samog izraza:

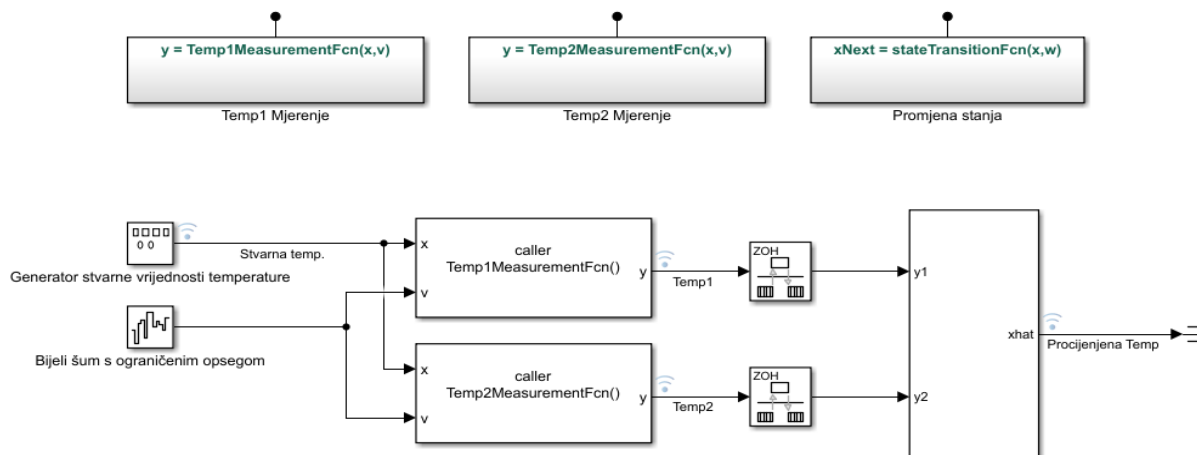
$$\hat{x}_{k[1x1]} = \hat{x}_{k[1x1]}^- + K_{k[1x2]}(y_{k[2x1]} - C_{[2x1]}\hat{x}_{k[1x1]}^-) \quad (3-1)$$

da su dimenzije parametara  $K_k$ ,  $y_k$ ,  $C$  promijenjene u odnosu na izraz kada mjerenje potječe samo iz jednog izvora [11]. Analogno tome, jednadžba Kalmanovog filtra za fuziju mjerenja iz  $n$  senzora izgleda ovako:

$$\hat{x}_{k[1x1]} = \hat{x}_{k[1x1]}^- + K_{k[1xn]}(y_{k[nx1]} - C_{[nx1]}\hat{x}_{k[1x1]}^-) \quad (3-2)$$

### 3.3. MATLAB primjer primjene Kalmanovog filtra za fuziju senzora

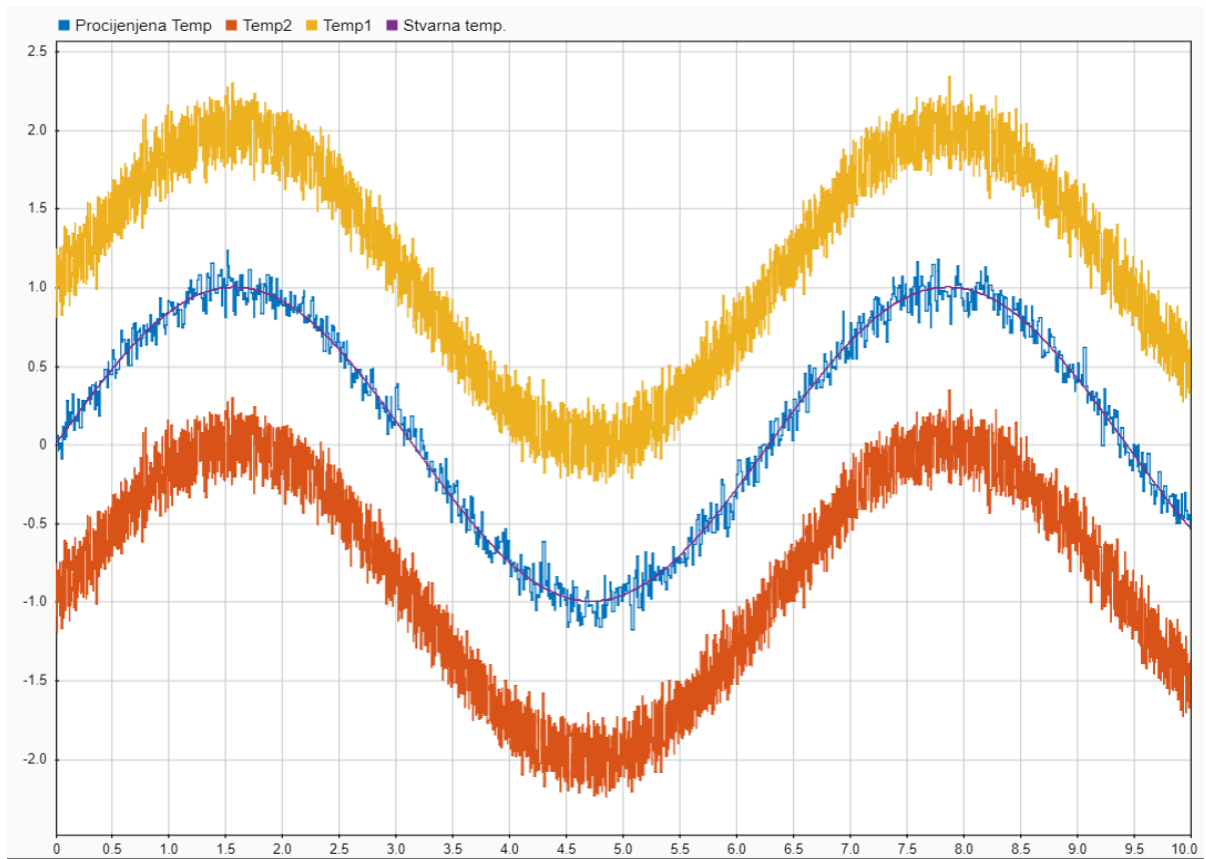
Primjer se odnosi na mjerenje temperature zraka prostorije pomoću dva senzora [20]. Oba mjerenja sadrže šum a njihovom kombinacijom nastoji se postići točniji rezultat. Simulacija se izvodi u *Simulinku* – programskom paketu razvijenom u MATLAB okružju koji se primjenjuje u svrhe modeliranja, simulacije i analize raznih dinamičkih sustava [19]. Predmet simuliranja u ovom paketu je blok dijagram kojim se modelira željeni sustav. Rezultati simulacije mogu se prikazati primjenom osciloskopa koji je implementiran unutar samog sučelja.



Sl. 3.3. Prikaz blokovskog dijagrama simuliranog unutar Simulinka [20]

Blok dijagram prikazuje generator stvarne vrijednosti temperature koji daje sinusni signal te će se promatrati odstupanje mjerenja senzora u odnosu na njega. Amplituda i frekvencija generatora signala su podešene na vrijednost 1. Ispod generatora nalazi se blok bijelog šuma s ograničenim opsegom (engl. *The band-limited white noise*) koji generira normalno distribuirane slučajne brojeve koji su prikladni za upotrebu u kontinuiranim ili hibridnim sustavima. Ovaj blok zajedno s prethodno spomenutim generatorom stvarne vrijednosti temperature predstavlja ulaze u oba bloka pozivatelja funkcija (engl. *Function caller block*). Blok pozivatelja funkcija poziva i izvršava funkciju definiranu unutar bloka, za izračunavanje izlaznih signala iz ulaznih. Svaki blok odgovara jednom od dvaju senzora i na osnovu šuma i stvarne vrijednosti temperature generiraju nove vrijednosti temperature u prisustvu šuma. Sada su dobivena mjerenja dvaju senzora koja je potrebno kombinirati te pomoću filtera ostvariti novo filtrirano mjerenje. Prije ulaza u filter nalaze se dva bloka tranzicije (engl. *Rate transition*) koji prenose podatke s izlaza bloka koji radi jednom brzinom na ulaz bloka koji radi drugom brzinom. Odnosno, pretvaraju signale iz vremenski diskretnih u kontinuirane. Koriste se radi bržeg odziva ili smanjenja memorijskih zahtjeva. Tip blokova je ZOH (engl. *Zero - order hold*) što podrazumijeva držanje svake vrijednosti uzorka u vremenskom trajanju od jednog intervala uzorka čime se postiže efekt pretvarača. Zatim je postavljen blok Kalmanovog filtra koji inicijalno nije imao mogućnost izvođenja fuzije senzora te se u blokovskom dijagramu u konačnici nalazi blok koji predstavlja prošireni Kalmanov filter. Unutar bloka definirana su oba mjerenja s pripadnim kovarijancama čije su vrijednosti postavljene na iznos 0.2 kao i utjecaj mjernog šuma. Pokretanjem simulacije generira se procijenjena/filtrirana vrijednost temperature kao rezultat kombinacije mjerenja dvaju senzora, sl. 3.4. Uočljivo je da se niti signal nazvan *Temp1* niti signal *Temp2*, koji prikazuju mjerenja senzora, ne podudaraju sa stvarnom vrijednošću temperature koja je predočena signalom nazvanim *Stvarna temp*. Dakle, mjerenja iz senzora zbog prisustva šuma odstupaju od stvarnog mjerenja. Također se vidi da filtrirana vrijednost temperature koja je nastala fuzijom senzora dakle, miješanjem signala *Temp1* i *2* odgovara stvarnoj vrijednosti signala zbog utjecaja Kalmanovog filtra na smanjenje zastupljenosti šuma. Dobivena izlazna vrijednost je veće pouzdanosti te povećane kvalitete i točnosti. (slika 3.4)





*Sl. 3.4. Prikaz signala nastalih simulacijom unutar Simulinka*

## ZAKLJUČAK

Za upravljanje dinamičkim sustavom, najprije je potrebno znati što i na koji način taj sustav radi. Za takve primjene nije uvijek moguće mjeriti svaku varijablu koja se želi kontrolirati, a Kalmanov filter pruža način za zaključivanje o informacijama koje nedostaju iz neizravnih mjerenja kao i mjerenja u prisustvu šumova. Svaki Kalmanov filter se primjenjuje u obliku procesa u dva koraka – predikcija i korekcija. U radu je utvrđeno na primjeru automobila da postoji više puteva kojima se može doći do informacije primjerice o lokaciji. To je moguće učiniti koristeći samo prethodna znanja kao što su: s koje je početne točke automobil krenuo, kolikom brzinom se vozio i sl., kako bi se predvidjelo gdje se nalazi. Pozicija se može izravno saznati, npr. putem mjerenja GPS-om ali pokazano je da se kombiniranjem navedenih načina, tj. smjenjivanjem predikcije i korekcije može dobiti točnija i preciznija procjena a to je zadaća primjene filtra. Kada dođe do novog mjerenja koje sadrži implementiran šum, filter uspoređuje predviđanje s mjerenjem i zatim ispravlja svoju procjenu na temelju relativnog povjerenja u obje informacije. Dakle, može se koristiti za procjenu stanja sustava kada ga nije moguće izravno mjeriti te za procjenu stanja kombiniranjem mjerenja iz različitih izvora – različitih vrsta senzora koja su podložna šumovima gdje u prvi plan dolazi primjena za fuziju senzora. Na primjeru MATLAB simulacije pokazano je da filter učinkovito smanjuje pogrešku minimiziranjem razlike između procijenjene i izmjerene vrijednosti čime je procjena stanja točnija. Također je pokazano da filter prilikom prikupljanja podataka od više senzora – fuzijom senzora, daje odziv koji ima osjetno nižu količinu zastupljenosti šuma i izraženije konvergira stvarnoj vrijednosti nego što bi se moglo postići upotrebom mjerenja od samo jednog senzora. Prilikom upotrebe Kalmanovog filtra za fuziju senzora jedinstvena funkcionalnost svakog od senzora ostaje očuvana a njihova se ograničenja i nedostaci poput osjetljivosti na vibracije reduciraju.

## LITERATURA

- [1] R.E. Kálmán, „A New Approach to Linear Filtering and Prediction Problems“, Research Institute for Advanced Study, Baltimore, Maryland, USA, 1960
- [2] <https://ntrs.nasa.gov/api/citations/19860003843/downloads/19860003843.pdf4> 12.7.2021
- [3] M. Alapić, „Kalmanovi filteri“, diplomski rad, Zagreb: Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet, 2019. <https://urn.nsk.hr/urn:nbn:hr:217:848768>
- [4] M. Matijašević, „Projektiranje estimatora stanja napunjenosti baterije temeljenog na Kalmanovom filtru“, diplomski rad, Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje, 2019. : <https://urn.nsk.hr/urn:nbn:hr:235:608969>
- [5] E. A. Wan, R. Van Der Merwe, „The unscented Kalman filter for nonlinear estimation“, in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium*, str. 153–158, 2000.
- [6] P. Pichlík, J. Zděnek, „Adhesion Force Detection Method Based on the Kalman Filter for Slip Control Purpose“, *Automatika*, vol. 57, br. 2, 2016: <https://doi.org/10.7305/automatika.2016.10.1152>
- [7] P. Pierro, C.A. Monje, N. Mansard, P. Soueres i C. Balaguer, „Open Solution for Humanoid Attitude Estimation through Sensory Integration and Extended Kalman Filtering“, *Automatika*, vol.56, br. 1, 2015 <https://doi.org/10.7305/automatika.2015.04.593>
- [8] Larsen, T. D., Hansen, K. L., Andersen, N. A., & Ravn, O. (1999). „Design of Kalman filters for mobile robots: Evaluation of the kinematic and odometric approach“, in *Proceedings of IEEE Conference on Control Applications* (Vol. 2). IEEE. <https://doi.org/10.1109/CCA.1999.801027>
- [9] T. Lončarek, „Estimacija brzine vrtnje sinkronog generatora vjetroagregata korištenjem Kalmanova filtra“, diplomski rad, Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva, 2014. <https://urn.nsk.hr/urn:nbn:hr:168:595540>
- [10] Westphal L.C. (1995) „State observers“. In: *Sourcebook of Control Systems Engineering*. Springer, Boston, MA. [https://doi.org/10.1007/978-1-4615-1805-1\\_25](https://doi.org/10.1007/978-1-4615-1805-1_25)
- [11] <https://www.mathworks.com/videos/series/understanding-kalman-filters.html> 20.8.2021

- [12] M. Kovandžić, „Kalmanov filter“, seminarski rad, Niš : Mašinski fakultet univerziteta u Nišu, 2018 [https://www.academia.edu/18134827/KALMANOV\\_FILTER](https://www.academia.edu/18134827/KALMANOV_FILTER)
- [13] Welch, G.; Bishop G.: An Introduction to the Kalman Filter, University of North Carolina, 2001. [http://www.cs.unc.edu/~welch/media/pdf/kalman\\_intro.pdf](http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf)
- [14] Š. Bašić, „Kalmanov filter“, seminarski rad, Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva, 2010.
- [15] <https://www.mathworks.com/help/control/ug/kalman-filtering.html> 31.8.2021
- [16] <https://www.mathworks.com/help/control/ref/ss.html> 31.8.2021
- [17] <https://www.mathworks.com/videos/series/understanding-sensor-fusion-and-tracking.html>
- [18] J.B. Gao, C.J. Harris „Some remarks on Kalman filters for the multisensor fusion“. In: *Information Fusion*, School of Mathematics, Statistics and Computer Sciences, University of New England, Armidale, NSW 2351, Australia, September 2002
- [19] I. Vujović „Modeliranje i simuliranje u elektrotehnici Matlab/Simulink“ Sveučilište u Splitu, Pomorski fakultet, Split 2014
- [20] <https://www.mathworks.com/matlabcentral/answers/417042-sensor-fusion-using-kalman-filter-simulink> 2.9.2021

## **SAŽETAK**

U radu je objašnjen pojam Kalmanovog filtra kao estimatora stanja sustava na temelju mjerenja uz prisustvo šumova. On omogućava mjerenje varijabli koje nisu izravno mjerljive kao i postizanje točnijih, preciznijih, kvalitetnijih i pouzdanijih podataka posredstvom fuzije senzora. U drugom poglavlju je iznesena teorijska podloga koja stoji iza definicije filtra. Objašnjen je algoritam filtra koji podrazumijeva izmjenu dva koraka a to su predikcija i korekcija s pripadnim jednadžbama, vizualizacijom svake faze algoritma i primjerom MATLAB simulacije. U trećem poglavlju definirane su karakteristike odnosno prednosti fuzije senzora, povučene su paralele između fuzije senzora i Kalmanovog filtra. Detaljno je objašnjena primjena filtra za fuziju senzora te su dobiveni zadovoljavajući rezultati simulacije kao rješenja postavljenog zadatka završnog rada.

Ključne riječi: estimator stanja, fuzija senzora, Kalmanov filter, korekcija, predikcija

## **ABSTRACT**

The paper explains the concept of the Kalman filter as an estimator of the system state based on measurements in the presence of noise. It enables the measurement of variables that are not directly measurable as well as the achievement of more accurate, precise, high-quality and more reliable data through sensor fusion. The second chapter presents the theoretical background behind the definition of a filter. The filter algorithm, which implies the change of two steps, namely prediction and correction is explained with the corresponding equations, visualization of each phase of the algorithm and an example of MATLAB simulation. In the third chapter, the characteristics and advantages of sensor fusion are defined, and parallels between sensor fusion and Kalman filter are drawn. The application of the filter for sensor fusion is explained in detail, and in the end satisfactory simulation results were obtained as a solution to the task of the final paper.

Keywords: state estimator, sensor fusion, Kalman filter, correction, prediction

## ŽIVOTOPIS

Rad je napisala Marija Mostarac. Rođena je 26. kolovoza 1999. godine u Žepču, BiH. Osnovnoškolsko obrazovanje stekla je u Osnovnoj školi „Žepče“ u Žepču. Bila je odlična učenica svih osam godina školovanja. Osim pohađanja nastave sudjelovala je na školskim i općinskim natjecanjima iz predmeta matematike i fizike, te postizala izvrsne rezultate. Godine 2014. upisala je srednju školu - Opću gimnaziju u Katoličkom Školskom centru „Don Bosco“ Žepče. Kroz čitavo školovanje postizala je odličan uspjeh i uzorno vladanje. Na školskim natjecanjima iz matematike sudjelovala je sve četiri godine a u drugom razredu je osigurala odlazak na federalno natjecanje u Stolcu, BiH. Stekla je priznanje učenika generacije za period 2014.-2018. godine. Nakon završene srednje škole 2018. godine upisala je preddiplomski sveučilišni studij Elektrotehnika i Informacijska tehnologija na Sveučilištu Josipa Jurja Strossmayera u Osijeku. Po završetku treće godine studija uručeno joj je priznanje za uspješnost u studiranju po Odluci Fakultetskog vijeća od 25. svibnja 2021. godine. Bila je licencirana mažoretkinja Bosne i Hercegovine te je uz sudjelovanje na brojnim smotrama nastupila na Europskom prvenstvu. Karateom se počela baviti 2007. godine. Osvajala je odličja na brojnim natjecanjima različitih razina, bila višestruka državna prvakinja čime je osiguravala svoje mjesto u karate reprezentaciji Bosne i Hercegovine. Školu karatea uspješno je završila 2015. godine postigavši zvanje majstor karatea odnosno nositelj crnog pojasa I-DAN, čime je stekla mogućnost treniranja nižih uzrasta. Služi se engleskim jezikom.

## PRILOG – Kôd MATLAB simulacije

```
Linija    Kod  
  
          >> A = [1.1269    -0.4940    0.1129  
1:         1.0000         0         0  
          0    1.0000         0];  
  
          >> B = [-0.3832  
2:         0.5919  
          0.5191];  
  
3:        >> C = [1 0 0];  
4:        >> D = 0;  
5:        >> Ts = 1;  
6:        >> sys = ss(A, [B B], C, D, Ts, 'InputName', {'u' 'w'}, 'OutputName', 'y');  
7:        >> Q = 2.3;  
8:        >> R = 1;  
9:        >> [kalmf, L, ~, M, Z] = kalman(sys, Q, R);  
10:       >> kalmf = kalmf(1, :);  
11:       >> sys.InputName = {'u', 'w'};  
12:       >> sys.OutputName = {'yt'};  
13:       >> vIn = sumblk('y=yt+v');  
14:       >> kalmf.InputName = {'u', 'y'};  
15:       >> kalmf.OutputName = 'ye';  
16:       >> SimModel = connect(sys, vIn, kalmf, {'u', 'w', 'v'}, {'yt', 'ye'});  
17:       >> t = (0:100)';  
18:       >> u = sin(t/5);  
19:       >> rng(10, 'twister');  
20:       >> w = sqrt(Q)*randn(length(t), 1);  
21:       >> v = sqrt(R)*randn(length(t), 1);  
22:       >> out = lsim(SimModel, [u, w, v]);  
23:       >> yt = out(:, 1);    % stvarni izlaz  
24:       >> ye = out(:, 2);   % filtrirani izlaz  
25:       >> y = yt + v;      % izmjereni izlaz  
26:       >> clf  
27:       >> subplot(211), plot(t, yt, 'g', t, ye, 'b--'),  
28:       >> xlabel('Broj uzoraka'), ylabel('Izlaz')  
29:       >> title('Odziv Kalmanovog filtra - Marija Mostarac')  
30:       >> legend('Stvarni izlaz', 'Filtrirani izlaz')  
31:       >> subplot(212), plot(t, yt-y, 'y', t, yt-ye, 'r--'),  
32:       >> xlabel('Broj uzoraka'), ylabel('Pogreška')  
33:       >> legend('Stvarni - mjereni izlaz', 'Stvarni - filtrirani izlaz')
```

**Linija**    **Kod**

```
34:    >> PogrMjer = yt-y;  
35:    >> PogrKovMjer = sum(PogrMjer.* PogrMjer)/length(PogrMjer)  
36:    >> PogrFilt = yt-ye;  
37:    >> PogrKovFilt = sum(PogrFilt.*PogrFilt)/length(PogrFilt)  
    >> PogrKovMjer =  
  
        0.9871  
38:    >> PogrKovFilt =  
  
39:          0.3479
```