

Wolfman 0036XXXXXX 1.D_AUT	Fakultet elektrotehnike i računarstva Zagreb Zavod za automatiku i računalno inženjerstvo	23.5.2012
	Adaptivno i robusno upravljanje	
	Seminar 2: RLS metoda estimacije parametara procesa i adaptivni RST regulator	

2.1 Uvod

U sklopu ovog seminarskog rada biti će obrađena rekurzivna metoda estimacije parametara procesa, temeljena na metodi najmanjih kvadrata (u daljnjem tekstu RLS metoda, eng. *Recursive Least Square*). Ova metoda će biti iskorištena za online estimaciju parametara zadanog procesa. Estimirani parametri će se potom koristiti kao ulazni podaci za adaptivni RST regulator pomoću kojeg će se polovi zatvorenog kruga upravljanja postaviti tako da se postigne željeno vladanje.

2.2 RLS metoda

Zadan je diskretni proces prijenosnom funkcijom:

$$\frac{y(k)}{u(k)} = \frac{b_0 q^{-1} + b_1 q^{-2}}{1 + a_1 q^{-1} + a_2 q^{-2}}. \quad (2-1)$$

Izlaz ovog procesa može se opisati sljedećom rekurzivnom relacijom:

$$y(k) = -a_1 y(k-1) - a_2 y(k-2) + b_0 u(k-1) + b_1 u(k-2), \quad (2-2)$$

iz čega slijedi regresijski vektor:

$$\varphi^T = [-y(k-1) \ -y(k-2) \ u(k-1) \ u(k-2)]. \quad (2-3)$$

Vektor estimiranih parametara je oblika:

$$\hat{\Theta} = [\hat{a}_1 \ \hat{a}_2 \ \hat{b}_0 \ \hat{b}_1], \quad (2-4)$$

dok je stvarni vektor parametara zadan kao:

$$\Theta = [a_1 \ a_2 \ b_0 \ b_1] = [0.7 \ 0.1 \ 2.0 \ 1.1]. \quad (2-5)$$

U nastavku su navedeni rekurzivni izrazi koji predstavljaju RLS metodu estimacije parametara (za ARX model):

$$\begin{aligned} \hat{\varepsilon}(k+1) &= y(k+1) - \varphi^T(k+1)\hat{\Theta}(k), \\ q(k+1) &= P(k)\varphi(k+1) [\lambda(k+1) + \varphi^T(k+1)P(k)\varphi(k+1)]^{-1}, \\ P(k+1) &= [I - q(k+1)\varphi^T(k+1)] P(k)\lambda(k+1)^{-1}, \\ \hat{\Theta}(k+1) &= \hat{\Theta}(k) + q(k+1)\hat{\varepsilon}(k+1), \end{aligned} \quad (2-6)$$

gdje su $\hat{\varepsilon}$ predikcijska pogreška, q Kalmanov vektor pojačanja, P matrica kovarijanci, λ faktor zaboravljanja i $\hat{\Theta}$ vektor estimiranih parametara.

U sklopu seminara korišten je library za Simulink u kojem je implementiran blok RLS estimatora (slika 2.1). Opis ulazno-izlaznih signala bloka dan je u tablici 2.1.



Slika 2.1: Blok za RLS estimaciju parametara.

Tablica 2.1: Objašnjenja ulazno-izlaznih signala bloka RLS estimacije.

ULAZI	OBJAŠNJENJE
u	trenutna vrijednost ulaznog signala u proces
y	trenutna vrijednost izlaznog signala iz procesa
λ	faktor zaboravljanja
IZLAZI	OBJAŠNJENJE
trag(P)	trag matrice kovarijanci P
$\hat{\Theta}$	vektor estimiranih parametara procesa

Gore navedeni RLS blok koristi *m-funkciju* *RLS_student* u koju je bilo potrebno dopisati kod kako bi ona bila u potpunosti funkcionalna. U nastavku je dan Matlab kod funkcije *RLS_student*:

Matlab kod 1: RLS_student.m - *m-funkcija* koja implementira RLS algoritam

```

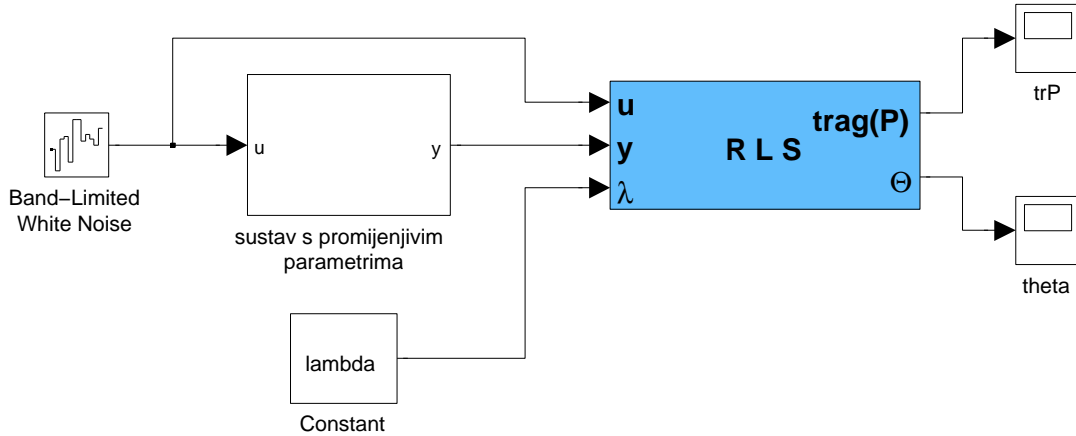
1 function [izlaz] = RLS_student(u, y, na, nb, lambda,t,...
2                               P_init, theta_init)
3
4 persistent P theta u_prosli y_prosli phi_y phi_u
5 if t==0
6     new_P=P_init; new_theta=theta_init;
7     phi_y=zeros(1,na); phi_u=zeros(1,nb+1);
8     phi=[phi_y phi_u]';
9 else
10    phi_y=[-y_prosli phi_y(1:(length(phi_y)-1))];
11    phi_u=[u_prosli phi_u(1:(length(phi_u)-1))];
12    phi=[phi_y phi_u]';
13    % dopisani izrazi za osvježavanje
14    % matrice P i parametara theta
15    new_P=(eye(na+nb+1)-(P*(phi*phi')))/...
16           (lambda+phi'*P*phi))*P/lambda;
17    new_theta=theta+new_P*phi*(y-phi'*theta);
18 end
19 u_prosli=u;
20 y_prosli=y;
21 P=new_P;
22 theta=new_theta;
23 izlaz=[trace(P) theta'];

```

Izrazi za osvježavanje matrice kovarijanci i vektora parametara napisani su prema 2-6. Jedan od problema RLS metode je inicijalizacija parametara. Pokazuje se da odabir početnih vrijednosti matrice kovarijanci i vektora estimiranih parametara može drastično utjecati na rad algoritma.

2.2.1 Utjecaj početne vrijednosti matrice P

U Simulinku je složena simulacijska shema prema slici 2.2. Kao ulazni signal korišten je *Band Limited White Noise* (Noise Power = 0.1). U blok RLS estimatora potrebno je unijeti početne vrijednosti matrice kovarijanci P_{init} i početni vektor parametara Θ_{init} . U ovom potpoglavlju promatra se utjecaj početne vrijednosti matrice kovarijanci. Početna vrijednost vektora parametara je nul-vektor, a iznos faktora zaboravljanja λ je 1.



Slika 2.2: Blok shema simulacije rada RLS estimatora parametara.

Simulacija je obavljena za sljedeće početne vrijednosti matrice kovarijanci:

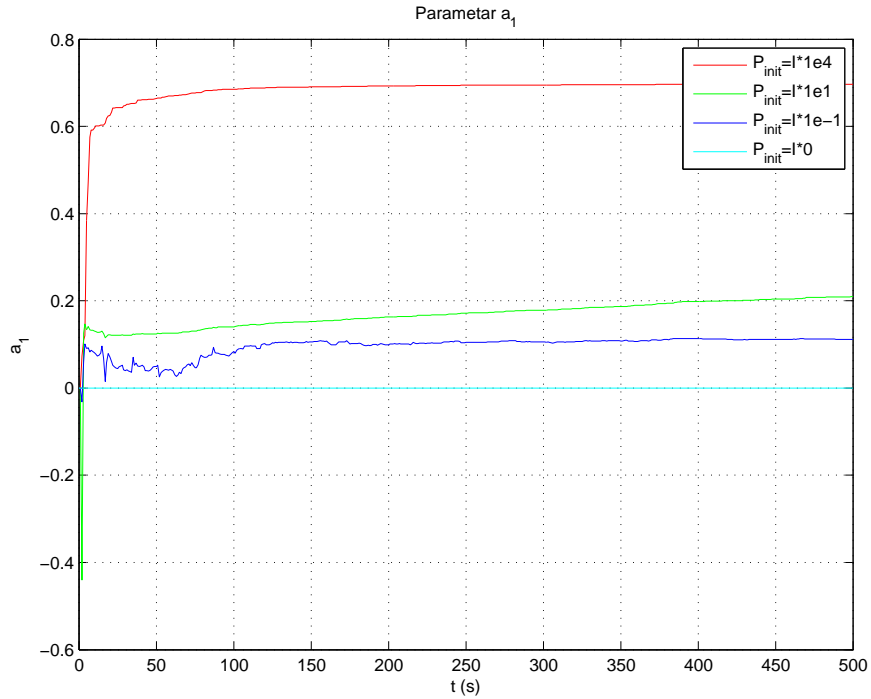
- $P = I_{4 \times 4} \cdot 10000$
- $P = I_{4 \times 4} \cdot 10$
- $P = I_{4 \times 4} \cdot 0.1$
- $P = I_{4 \times 4} \cdot 0$

Na slikama 2.3 – 2.6 prikazani su odzivi estimiranih parametara u vremenu. Kao što se može vidjeti, najbrža estimacija dobivena je uz matricu kovarijanci s najvećim početnim vrijednostima na dijagonali. Najlošija estimacija dobivena je za nul-matricu kovarijanci (tu zapravo ni nema estimacije). Razlog tome je što matrica kovarijanci u vremenu teži prema nul-matrici. Kada se to dogodi, kažemo da se matrica P "isključila". Po izrazima 2-6 se vidi da u tom slučaju RLS algoritam više ne uzima u obzir nova mjerenja i vektor parametara se više ne mijenja, tj. estimacija više ne radi.

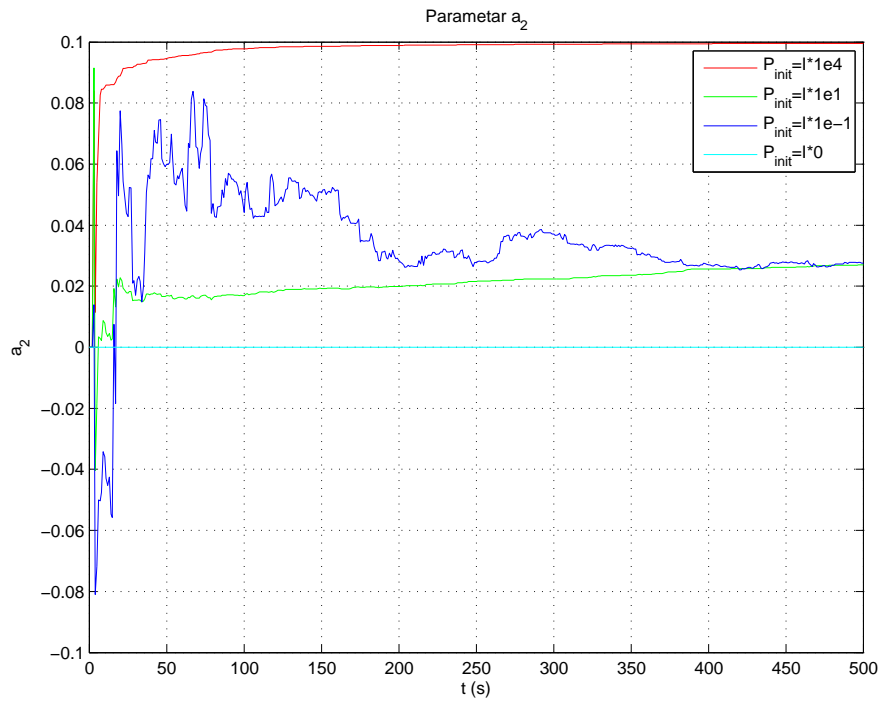
Na slici 2.7 prikazan je iznos traga matrice P tijekom simulacije. Iz te slike se može vidjeti da trag matrice kovarijanci tijekom vremena teži prema nuli ($\lim_{k \rightarrow \infty} \text{tr}(P) = 0$). Trag matrice P u nekom koraku označava kolika je nesigurnost procijenjenih parametara u tom koraku. Veći trag označava veću nesigurnost, što u algoritmu RLS metode daje veći iznos tzv. Kalmanovog pojačanja (faktor korekcije) preko kojeg se onda u izračunu novih parametara daje veći značaj mjernim podacima u odnosu na prethodno izračunate parametre. Manji iznos traga matrice u nekom koraku označava veću sigurnost u procijenjene parametre u tom koraku, što onda rezultira u manjem iznosu Kalmanovog pojačanja i manjem doprinosu novih mjernih podataka u novom izračunu parametara. U skladu s dosad rečenim, veliki početni

trag matrice P znači da će identifikacijski algoritam biti brži, no to opet znači da će postojati veće fluktuacije upravljačkog signala, a samim time i veće forsiranje aktuatora u upravljačkom krugu. S druge strane, mali početni trag matrice P rezultirat će sporijom identifikacijom, ali i manjim oscilacijama upravljačkog signala i manjim forsiranjem aktuatora.

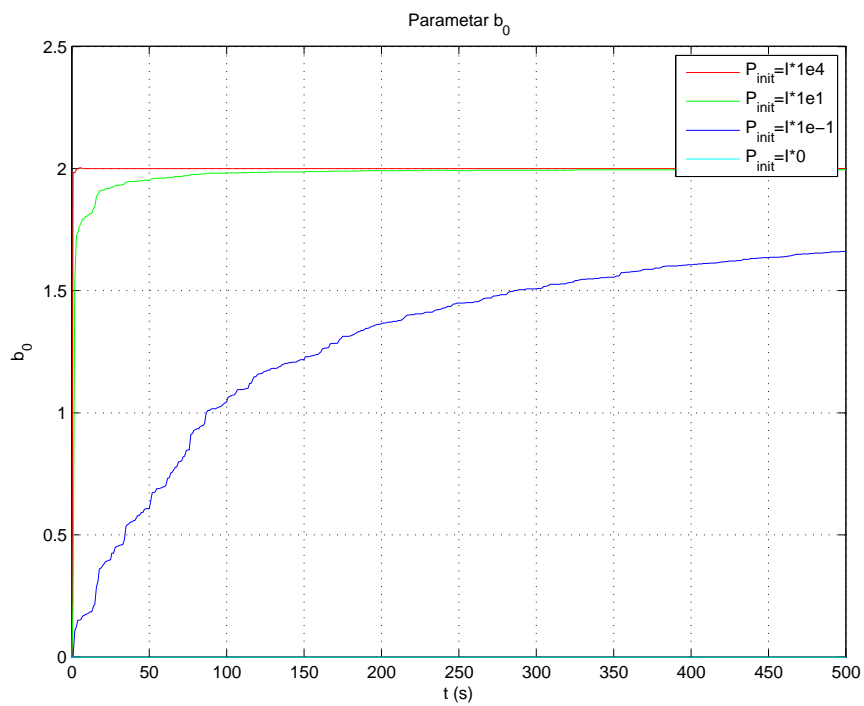
U daljnjim razmatranjima biti će korištena početna vrijednost matrice kovarijanci koja daje najbržu estimaciju parametara - $P_{init} = I_{4 \times 4} \cdot 10000$.



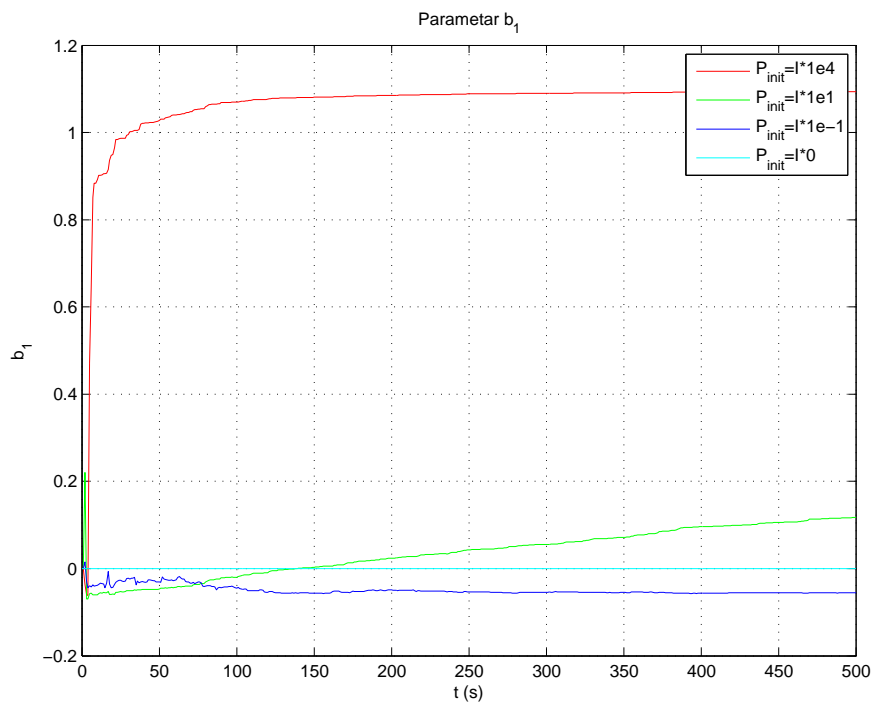
Slika 2.3: Estimacija parametra a_1 za razne vrijednosti početne matrice kovarijanci P .



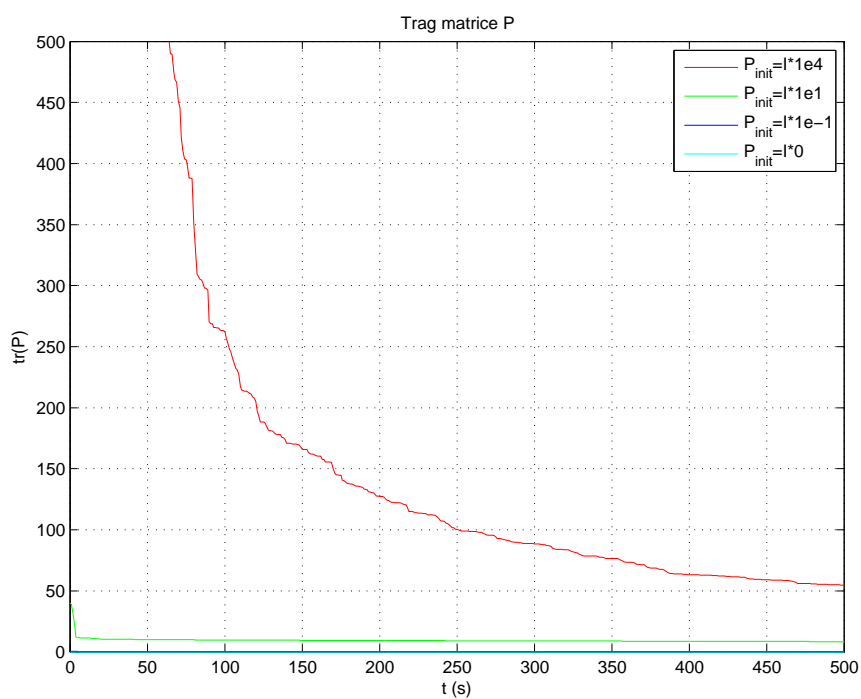
Slika 2.4: Estimacija parametra a_2 za razne vrijednosti početne matrice kovarijanci P .



Slika 2.5: Estimacija parametra b_0 za razne vrijednosti početne matrice kovarijanci P .



Slika 2.6: Estimacija parametra b_1 za razne vrijednosti početne matrice kovarijanci P .



Slika 2.7: Iznos traga matrice kovarijanci P tijekom simulacije za njezine razne početne vrijednosti (povećan prikaz po y-osi da se bolje vidi).

2.2.2 Utjecaj početne vrijednosti vektora estimiranih parametara $\hat{\Theta}$

U ovom potpoglavlju promatra se utjecaj početnih vrijednosti vektora estimiranih parametara $\hat{\Theta}$. Korištena je simulacijska shema sa slike 2.2. U ovom slučaju je $\lambda = 1$ i $P_{init} = I_{4 \times 4} \cdot 10000$.

Korištene su sljedeće početne vrijednosti vektora parametara:

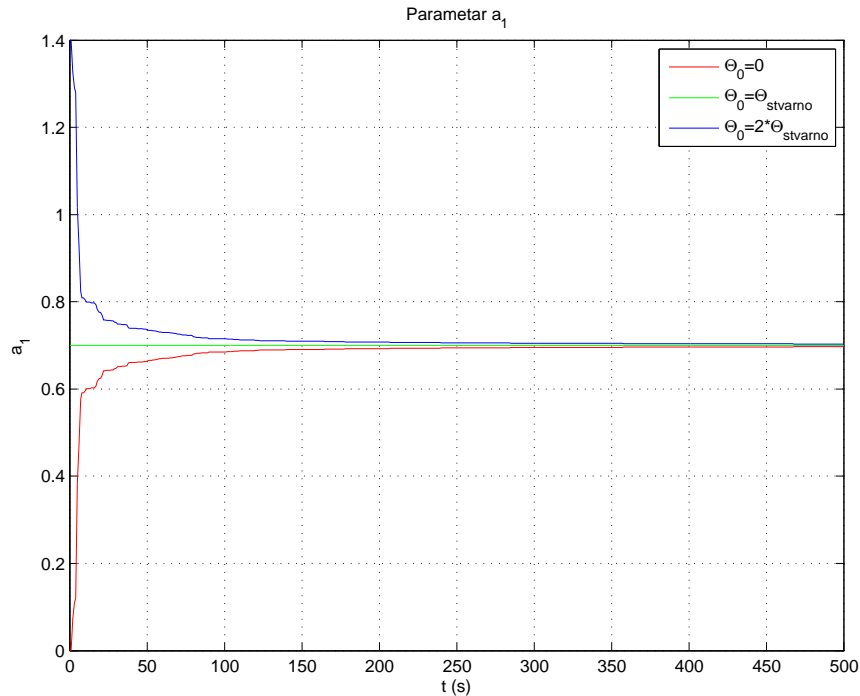
- nul-vektor,
- stvarne vrijednosti parametara,
- dvostruki iznos od stvarnih vrijednosti.

Kao što je i očekivano, ako nam je početni vektor estimiranih parametara jednak stvarnim parametrima, estimacija je gotova u nultom koraku i dalje se vektor parametara više ne mijenja. Brzina estimacije je podjednako brza u ostalim slučajevima.

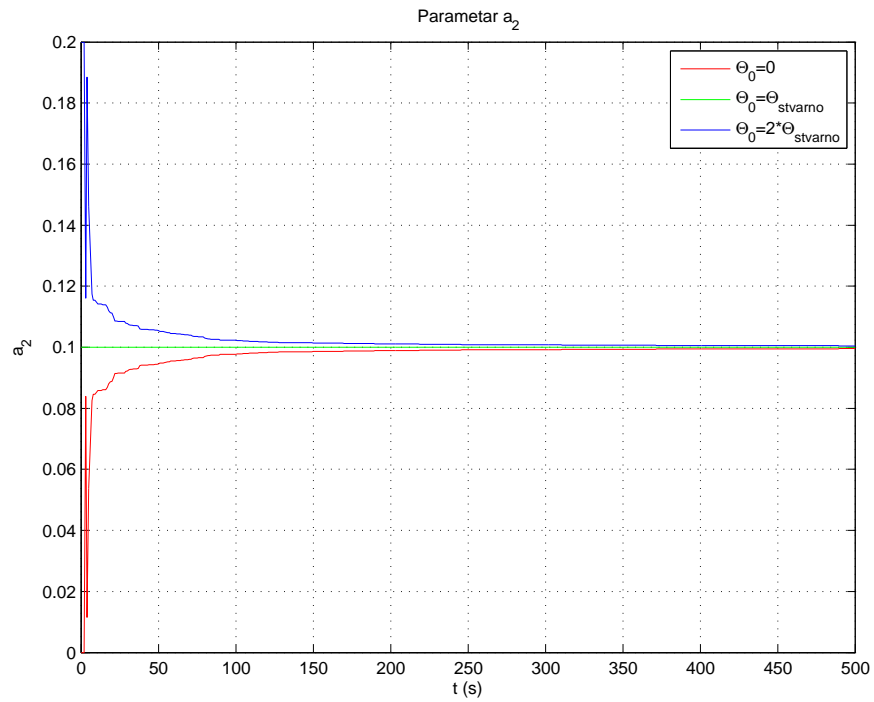
U realnom slučaju može se dogoditi da već na neki način poznajemo određeni broj parametara procesa kojeg želimo identificirati (npr. teoretskom analizom). Prirodno je u takvom slučaju za početne vrijednosti parametara koje poznajemo postaviti upravo te poznate vrijednosti. Ako pak ne znamo ni jedan parametar onda je često korisna tehnika zamisliti da je proces običan integrator s jediničnim pojačanjem. *ZOH* diskretizacijom integratora dobivamo $\frac{h}{z-1}$ pa su inicijalni parametri dani sa:

$$a_1 = -1 \text{ i } a_i = 0 \text{ za } i \neq 1 \text{ te } b_0 = h \text{ i } b_j = 0 \text{ za } j \neq 0.$$

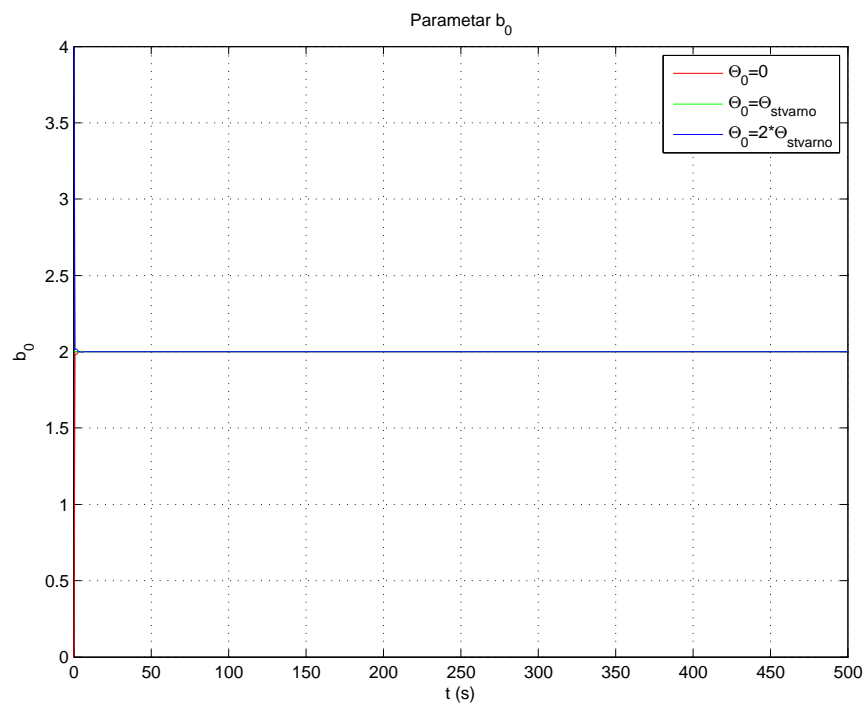
U daljnjim razmatranjima biti će korišten nul-vektor za početni vektor estimiranih parametara.



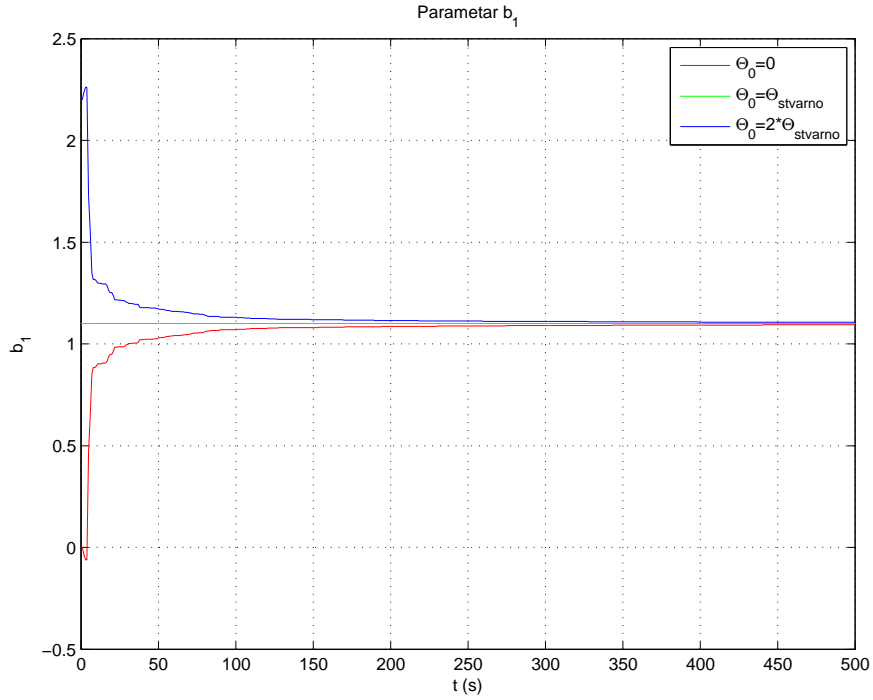
Slika 2.8: Estimacija parametra a_1 za razne vrijednosti početnog vektora $\hat{\Theta}$.



Slika 2.9: Estimacija parametra a_2 za razne vrijednosti početnog vektora $\hat{\Theta}$.



Slika 2.10: Estimacija parametra b_0 za razne vrijednosti početnog vektora $\hat{\Theta}$.



Slika 2.11: Estimacija parametra b_1 za razne vrijednosti početnog vektora $\hat{\Theta}$.

2.2.3 Utjecaj faktora zaboravljanja λ

U ovom potpoglavlju promatrat će se utjecaj faktora zaboravljanja λ . U ovom slučaju faktor zaboravljanja je konstantan. Opet je korištena simulacijska shema sa slike 2.2 uz $P_{init} = I_{4 \times 4} \cdot 10000$ i $\hat{\Theta} = \underline{0}$. Parametri procesa se mijenjaju u $t = 500s$ (b_0 za faktor $K = 1.5$) i u $t = 1500s$ (a_1 za isti faktor kao i b_0).

Kao što je već ranije rečeno, kada trag matrice padne u nulu, RLS algoritam se gasi, tj. nije više sposoban estimirati parametre. To se jasno vidi iz izraza 2-7:

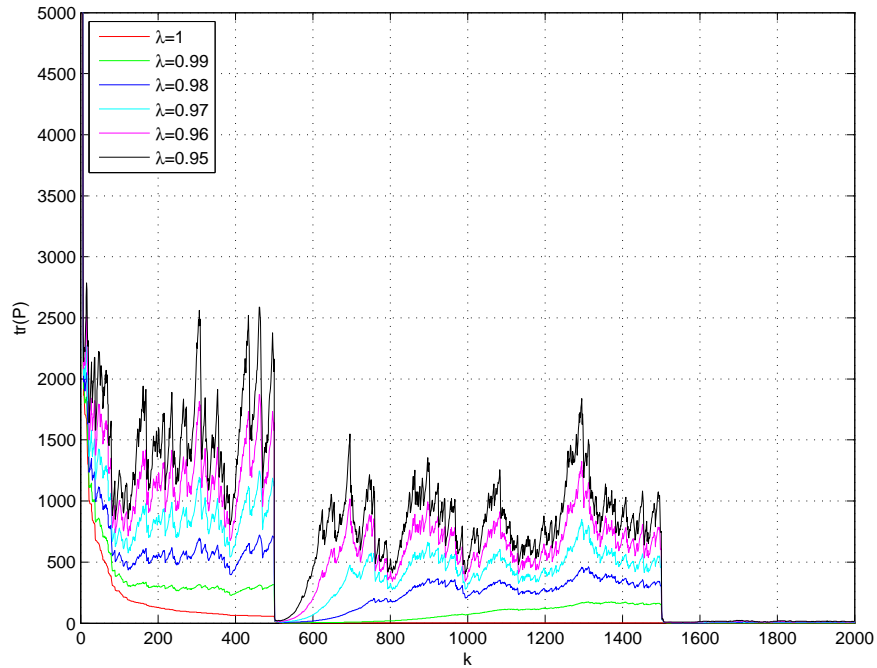
$$\hat{\Theta}(k+1) = \hat{\Theta}(k) + P(k+1)\varphi(k+1)\varepsilon(k+1), \quad (2-7)$$

koji povezuje vektor parametara i matricu kovarijance.

To nije problem u slučajevima kada se estimiraju parametri vremenski invarijantnog procesa (procesa kojem se parametri ne mijenjaju u vremenu). No, RLS algoritam se jako često koristi za estimaciju parametara vremenski promijenjivih procesa (npr. u algoritmima adaptivnog upravljanja). U tom slučaju nije bitna samo konvergencija prema stvarnim vrijednostima parametara, već i mogućnost slijeđenja vremenski promijenjivih parametara u stvarnom vremenu. Dobro slijeđenje promjenjivih parametara postiže se kada se osigura da matrica kovarijanci P ne postane premala, a to se osigurava faktorom zaboravljanja $\lambda \leq 1$.

Na slici 2.12 prikazan je iznos traga matrice P kroz simulaciju za različite faktore zaboravljanja. Kao što se vidi, smanjivanjem faktora zaboravljanja trag matrice P raste, tj. kako simulacija ide prema kraju, trag matrice se održava na nekoj pozitivnoj vrijednosti različitoj od nule, za razliku od slučaja kada je $\lambda = 1$ gdje trag matrice P teži prema nuli.

To znači da će uz faktor zaboravljanja $\lambda \leq 1$ estimacija biti brža i postojati će mogućnost estimacije promijenjivih parametara procesa. No, što je iznos faktora zaboravljanja manji, postojati će veće oscilacije estimiranih parametara procesa, što će posljedično utjecati na velike oscilacije parametara adaptivnog regulatora i forsi-



Slika 2.12: Iznos traga matrice kovarijanci P kroz simulaciju za različite faktore zaboravljanja.

ranje aktuatora. Stoga je potrebno naći kompromis između dva oprečna zahtjeva:

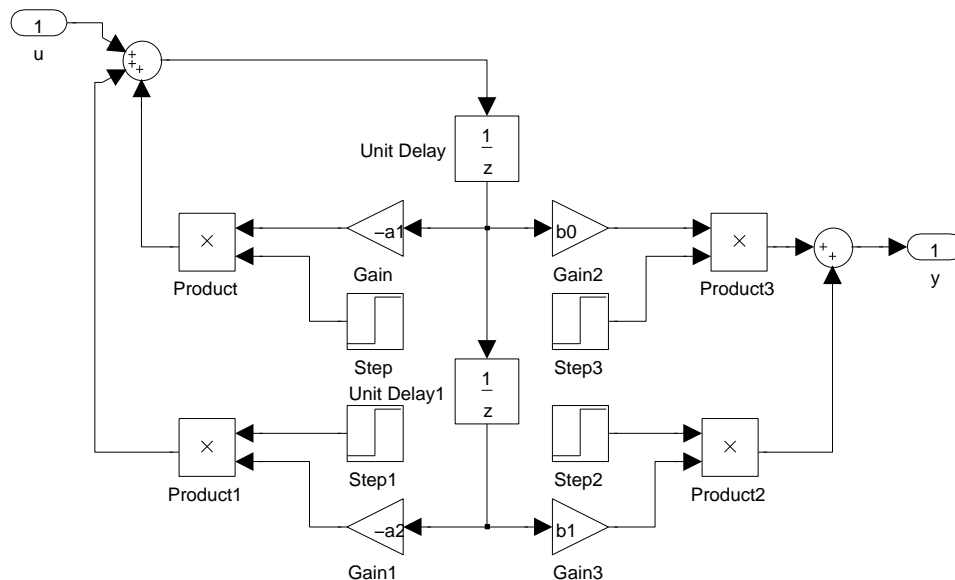
- brzo slijeđenje promjenjivih parametara uz velike oscilacije parametara (manji λ),
- male oscilacije parametara uz inerniji algoritam estimacije parametara (veći λ).

U praksi se pokazalo da je najbolje isprobati vrijednosti faktora zaboravljanja iz intervala $[0.95, 0.99]$, uz preporučenu vrijednost $\lambda = 0.98$. U nastavku će rad algoritma estimacije parametara biti provjeren na vremenski promjenjivom procesu uz step funkciju kao pobudni signal.

Simulacija procesa s promjenjivim parametrima uz step funkciju kao pobudni signal Na slici je prikazana blok shema zadanog procesa kojemu se pojedini parametri mogu skokovito mijenjati u vremenu.

Neka se parametar b_0 skokovito promijeni u trenutku $t = 150s$, uz koeficijent $K_{b_0} = 1.5$ ($b_{0_{novi}} = K_{b_0} b_{0_{stari}}$). Na ulazu procesa je skokovita funkcija s početkom u 100.

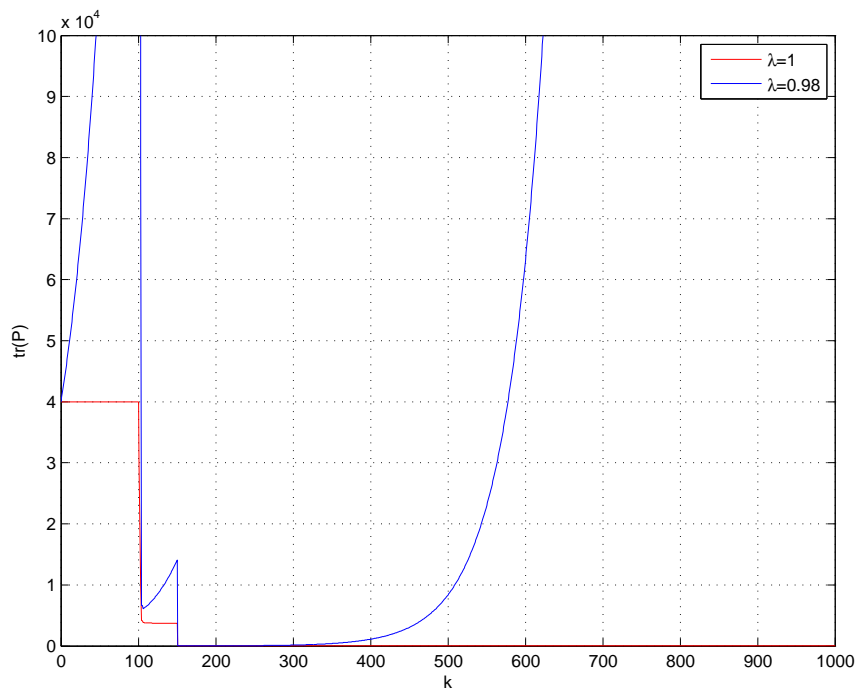
Na slici 2.14 prikazano je ponašanje traga matrice P za dva iznosa faktora zaboravljanja - $\lambda = 1$ i $\lambda = 0.98$ - pri estimaciji promjenjivih parametara procesa sa slike 2.13. Smanjenjem faktora zaboravljanja ispod 1 dolazi do tzv. *blow-up* učinka u matrici kovarijance P (trag matrice počne neograničeno rasti). To se događa zbog konstantnog faktora zaboravljanja i nedovoljno perzistentnog ulaznog signala. Skokovita funkcija koja je na ulazu procesa ne pobuđuje proces dovoljno dobro, tj. red njezine perzistentnosti je 1, a da bi RLS estimator dobro radio potrebna je perzistentnost ulaznog signala reda koji je barem jednak $n_a + n_b + 1$ (gdje su n_a i n_b redovi polinoma nazivnika i brojnika procesa). Naime, ako pobudni signal nije dovoljno



Slika 2.13: Blok shema zadanog procesa s vremenski promijenjivim parametrima.

perzistentan $P(k)\varphi(k+1) \rightarrow 0$ pa prema 2-6 $P(k+1) \approx \frac{P(k)}{\lambda}$. Kako je $\lambda < 1$, elementi matrice P rastu neograničeno.

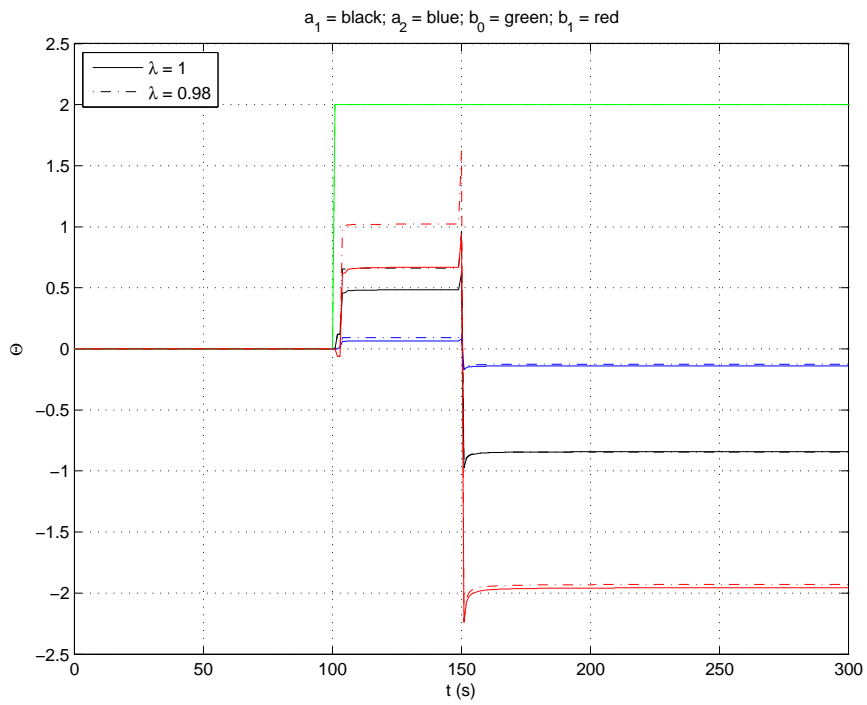
Na slici 2.15 prikazani su odzivi estimiranih parametara za obje vrijednosti faktora zaboravljanja. RLS estimator je slabo estimirao parametre, upravo zbog toga što je pobuda nedovoljno perzistentna.



Slika 2.14: Iznos traga matrice kovarijanci P tijekom simulacije za različite vrijednosti faktora zaboravljanja λ uz pobudu tipa $S(t-100)$.

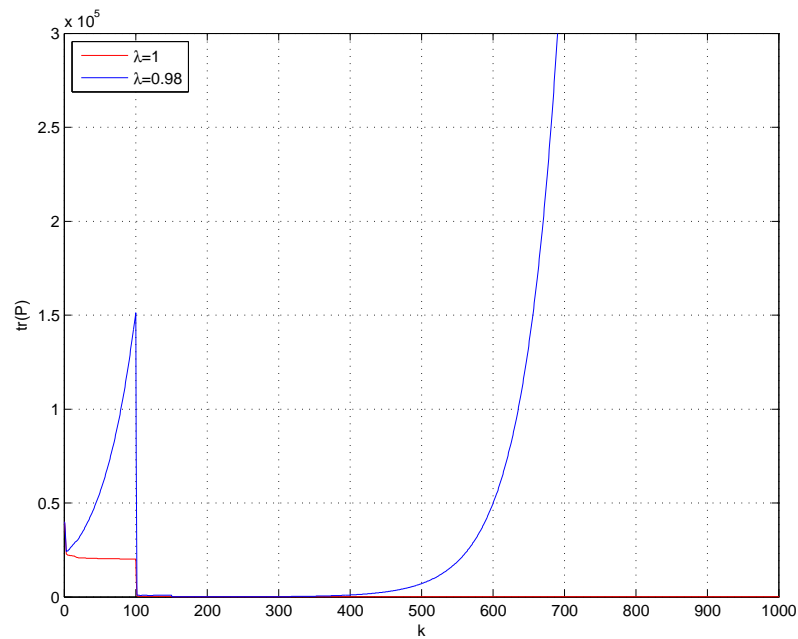
Ako na izlaz procesa superponiramo bijeli šum ($Noise\ power = 1e-4$) dobivamo rezultate prikazane na slikama 2.16 i 2.17. Vidimo da je estimacija i dalje prilično loša zbog nedovoljne perzistentnosti ulaznog signala.

No, ako bijeli šum superponiramo na ulazni signal, dobit ćemo ulazni signal koji

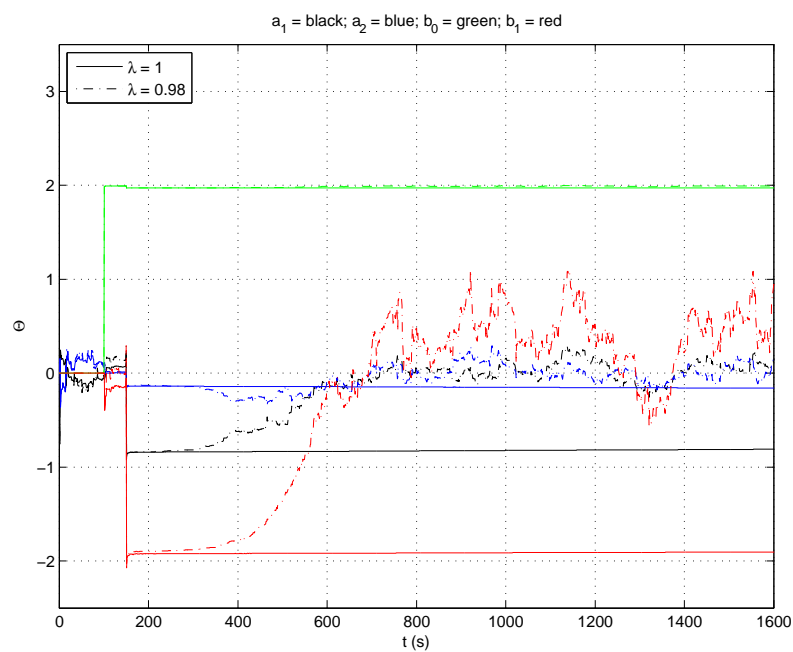


Slika 2.15: Odziv estimiranih parametara procesa $\hat{\Theta}$ za različite vrijednosti faktora zaboravljanja λ uz pobudu tipa $S(t - 100)$.

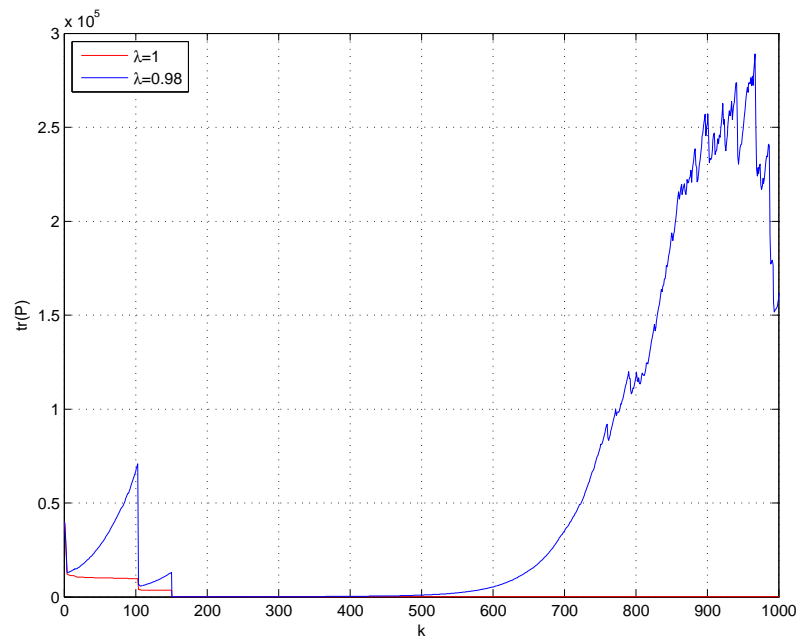
će perzistentno pobuđivati proces i time će se estimacija poboljšati. Rezultati su prikazani na slikama 2.18 i 2.19.



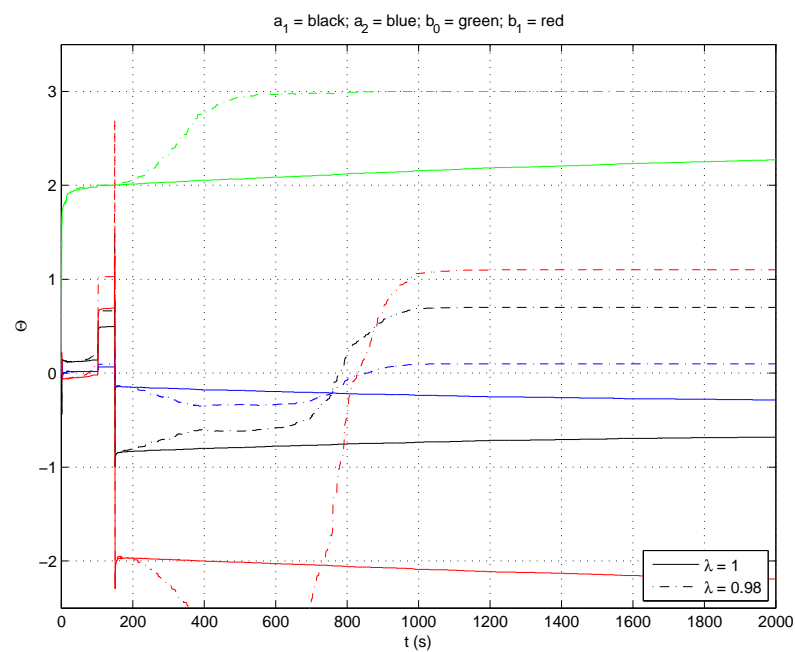
Slika 2.16: Iznos traga matrice kovarijanci P tijekom simulacije za različite vrijednosti faktora zaboravljanja λ uz pobudu tipa $S(t - 100)$ te superponiran bijeli šum na izlaz procesa.



Slika 2.17: Odziv estimiranih parametara procesa $\hat{\theta}$ za različite vrijednosti faktora zaboravljanja λ uz pobudu tipa $S(t - 100)$ te superponiran bijeli šum na izlaz procesa.



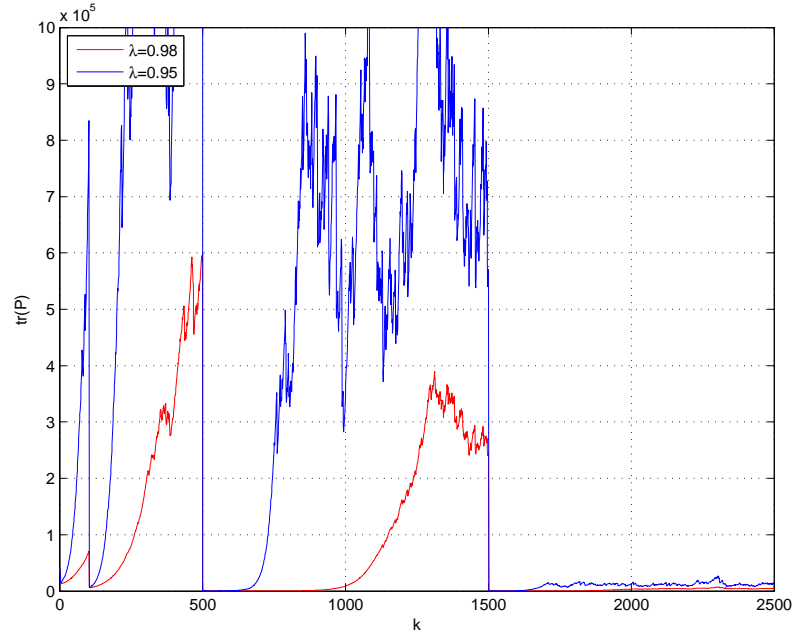
Slika 2.18: Iznos traga matrice kovarijanci P tijekom simulacije za različite vrijednosti faktora zaboravljanja λ uz pobudu tipa $S(t - 100)$ te superponiran bijeli šum na ulaz procesa.



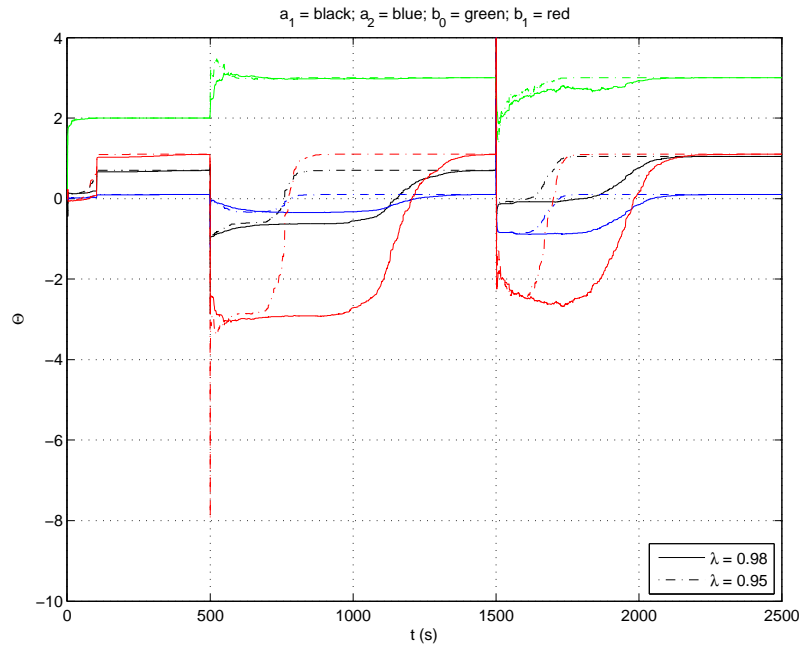
Slika 2.19: Odziv estimiranih parametara procesa $\hat{\theta}$ za različite vrijednosti faktora zaboravljanja λ uz pobudu tipa $S(t - 100)$ te superponiran bijeli šum na ulaz procesa.

2.2.4 Promjena parametara u brojniku i nazivniku

Neka se sada u trenutku $t = 500s$ promijeni parametar b_0 za faktor $K_{b_0} = 1.5$, te u trenutku $t = 1500s$ parametar a_1 za isti faktor. Na ulazni signal je superponiran bijeli šum kako bi imali perzistentan ulazni signal ($Noise\ power = 1e-4$). Korištene su dvije vrijednosti faktora zaboravljanja - $\lambda = 0.98$ i $\lambda = 0.95$. Rezultat estimacije prikazan je na slikama 2.20 i 2.21.



Slika 2.20: Iznos traga matrice kovarijanci P tijekom simulacije za različite vrijednosti faktora zaboravljanja λ uz pobudu tipa $S(t - 100)$ te superponiran bijeli šum na ulaz procesa.



Slika 2.21: Odziv estimiranih parametara procesa $\hat{\theta}$ za različite vrijednosti faktora zaboravljanja λ uz pobudu tipa $S(t - 100)$ te superponiran bijeli šum na ulaz procesa.

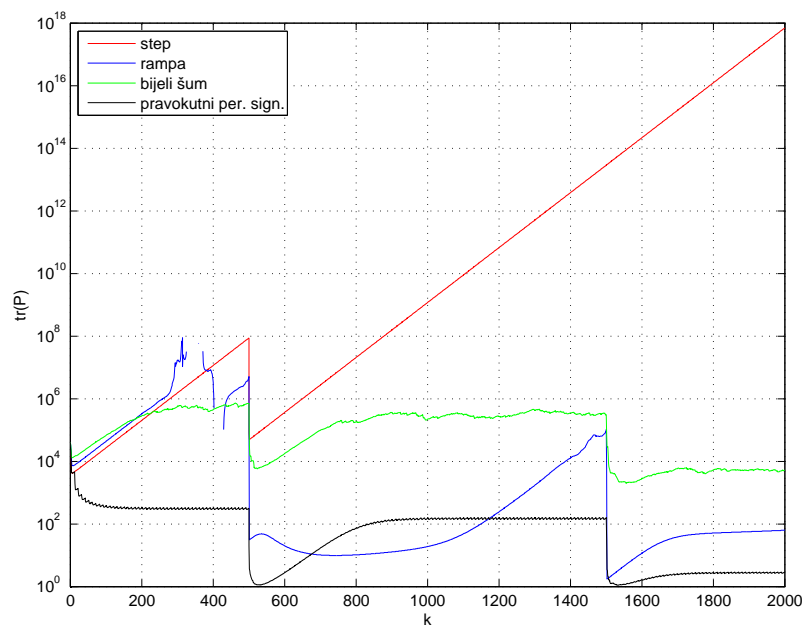
Iz odziva se može vidjeti da estimacija u oba slučaja radi dobro - RLS može pratiti promjene parametara i u brojniku i u nazivniku. Uz manji faktor zaboravljanja estimacija je brža, no u skladu s time su i veće oscilacije parametara (najbolje se vidi na primjeru estimiranog parametra b_1 koji ima velike skokove kada dođe do promjene parametara procesa). Također, brža je estimacija pri promjeni parametra nazivnika nego pri promjeni parametra brojnika.

2.2.5 Perzistentnost signala

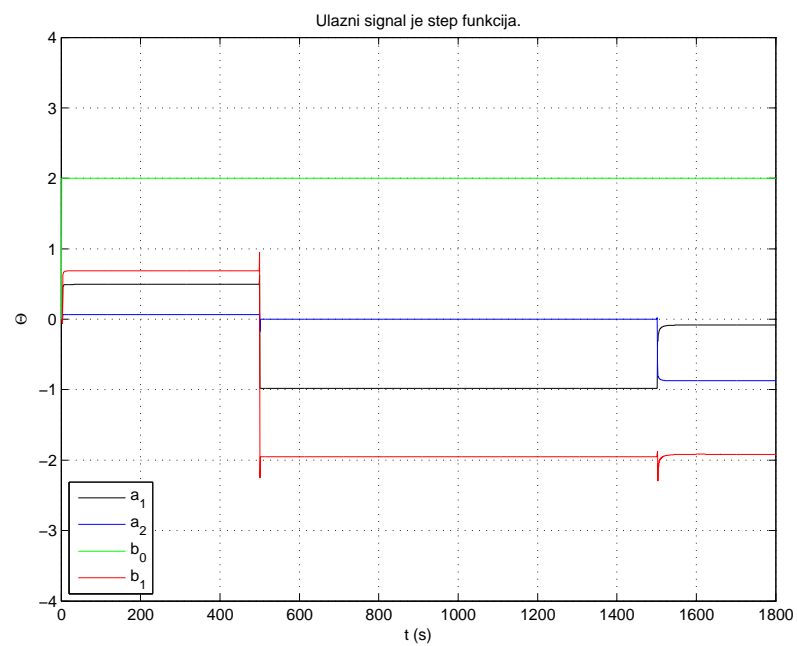
U ovom potpoglavlju imamo istu situaciju kao u prošlom, samo što će se sada promatrati utjecaj korištenja pojedinih tipova ulaznih signala na kvalitetu estimacije. Korišteni ulazni signali su:

- step funkcija,
- rampa,
- bijeli šum ($NP = 0.0001$),
- pravokutni periodički signal ($A = 1$, $T_p = 20s$).

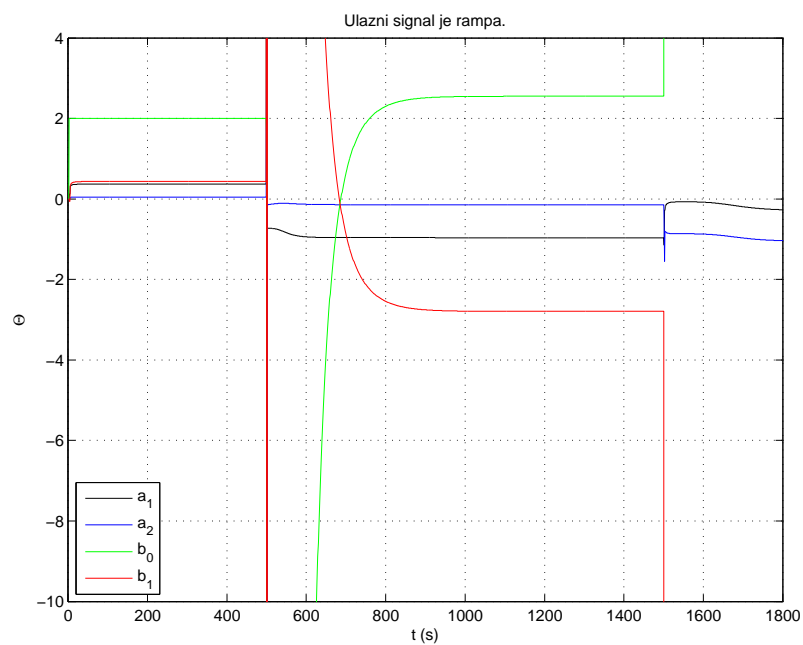
Korišteni faktor zaboravljanja je $\lambda = 0.98$. Na slici 2.22 prikazan je odziv traga matrice P . Za signale step i rampa trag matrice kovarijanci brzo poraste, što znači da su ti signali niskog reda perzistencije. Ovo je već objašnjeno ranije - uz pobudni signal koji nije perzistentan i konstantni faktor zaboravljanja $\lambda < 1$ vrijedi $P(k+1) \approx \frac{P(k)}{\lambda}$ pa elementi matrice P rastu neograničeno u vremenu. Estimacija pomoću rampe i step funkcija će biti loša, što pokazuju i odzivi na slikama 2.23 i 2.24.



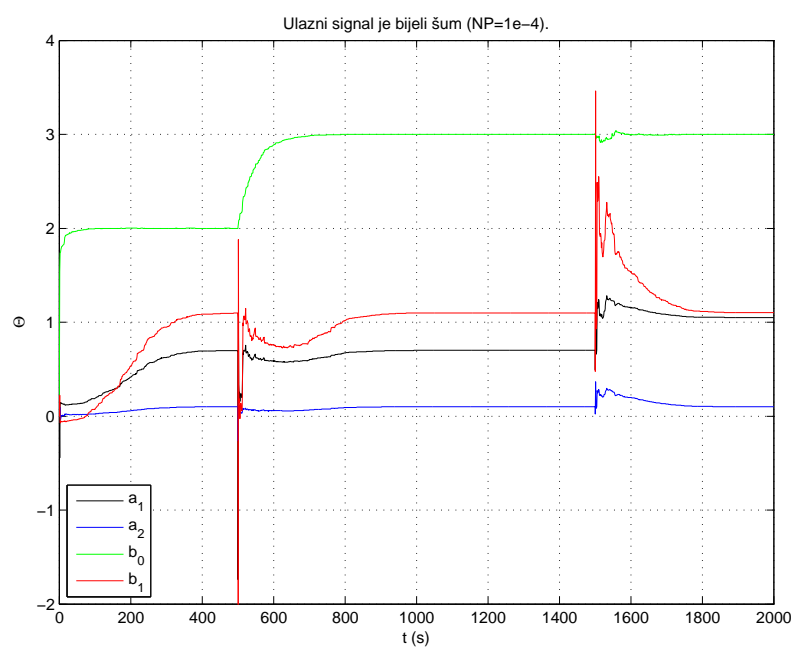
Slika 2.22: Iznos traga matrice kovarijanci P uz različite pobude.



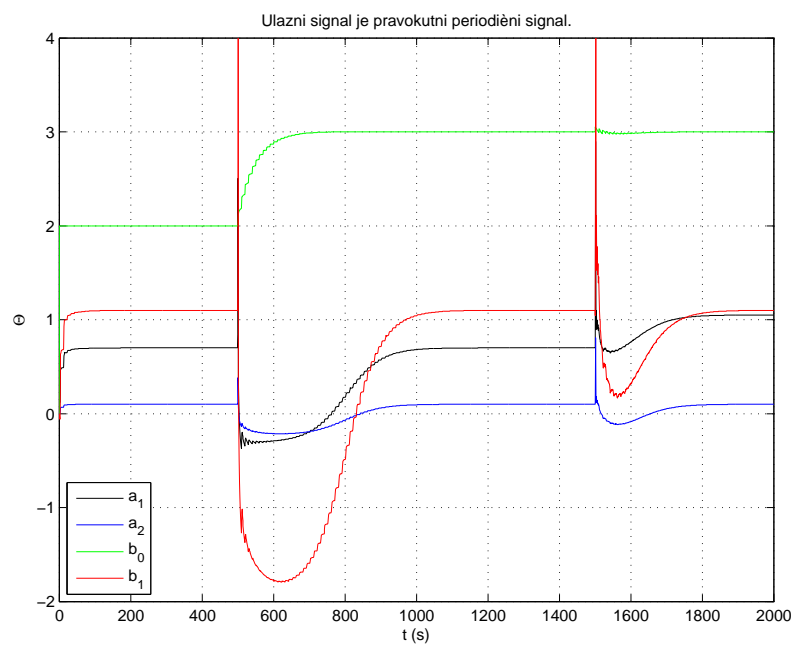
Slika 2.23: Odzivi estimiranih parametara uz pobudu tipa step funkcije.



Slika 2.24: Odzivi estimiranih parametara uz pobudu tipa rampa.



Slika 2.25: Odzivi estimiranih parametara uz pobudu tipa bijeli šum.



Slika 2.26: Odzivi estimiranih parametara uz pobudu tipa pravokutni periodični signal.

Estimacija uz pobudne signale tipa bijeli šum i periodični pravokutni signal daje puno bolje rezultate, jer su ti signali puno bogatiji u frekvencijskom smislu od step funkcije i rampe (pobuđuju više frekvencija). Na slikama 2.25 i 2.26 prikazani su odzivi estimiranih parametara za te pobudne signale. Nešto bolju estimaciju ipak daje bijeli šum (brža estimacija i manje oscilacije parametara).

Trag matrice kovarijanci može poslužiti kao mjera perzistencije ulaznog signala. Za signale koji nisu perzistentni i konstantni faktor zaboravljanja manji od 1 će trag matrice kovarijanci neograničeno rasti. Za perzistentni pobudni signal će trag matrice eventualno oscilirati oko neke ravnotežne vrijednosti.

Najveću perzistenciju pobude ima bijeli šum, jer on pobuđuje sve frekvencije. U praksi se taj signal ne može ostvariti - bijeli šum je, slično kao i impulsni signal $\delta(t)$ samo teoretski, tj. matematički konstrukt. Zato se u praksi koriste ispitni signali koji se više ili manje približavaju svojstvima idealnog bijelog šuma. Neki primjeri su telegrafski signal, DRBS, PRBS i sl.

2.2.6 Red pretpostavljenog modela

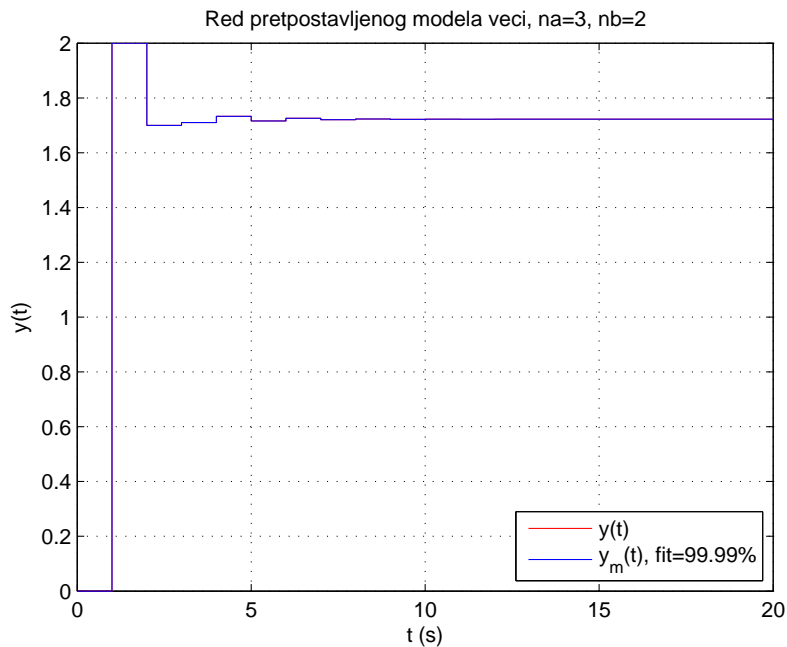
Promotrimo slučaj u kojem smo pretpostavili da je red modela zadan s $n_a = 3$ i $n_b = 2$:

$$G_m(z) = \frac{b_0 z^{-1} + b_1 z^{-2} + b_2 z^{-3}}{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}} \quad (2-8)$$

Estimacijom s ovakvom pretpostavkom dobivamo model s vektorom estimiranih parametara:

$$\hat{\Theta} = [0.2564 \quad -0.2082 \quad -0.0440 \quad 1.9999 \quad 0.2127 \quad -0.4833]. \quad (2-9)$$

Usporedbom odziva koje daje ovaj model i stvarni proces (slika 2.27) vidimo skoro savršeno poklapanje, što znači da je RLS estimator mogao sa više polova/nula postići jednako vladanje kao proces. Također, valja primjetiti da estimacijom parametri koji su višak u odnosu na stvarni proces (b_2 i a_3) nisu postavljeni u nulu, već su dobiveni neki sasvim drugi parametri koji uz ovaj red modela daju isto vladanje kao stvarni proces.



Slika 2.27: Usporedba odziva modela većeg reda i odziva procesa na step pobudu.

Uzmimo sada da je pretpostavljeni red modela dan sa $n_a = 3$ i $n_b = 1$ (znači jedan parametar u nazivniku je viška). Estimacijom dobivamo sljedeći vektor parametara:

$$\hat{\Theta} = [0.7000 \quad 0.1000 \quad 0.0000 \quad 2.0000 \quad 1.1000]. \quad (2-10)$$

U ovom slučaju je RLS estimator pogodio točne parametre procesa pa je parametar koji je bio viška postavio u nulu. Time je naravno dobiven model procesa koji u potpunosti odgovara stvarnom procesu.

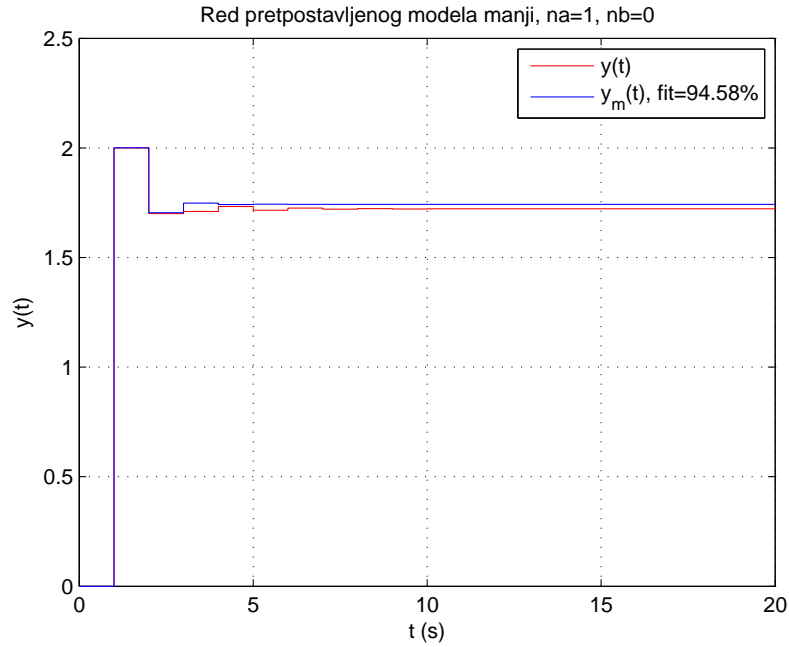
Ako pretpostavimo manji red modela od stvarnog procesa uz $n_a = 1$ i $n_b = 0$ dobivamo model:

$$G_m(z) = \frac{b_0 z^{-1}}{1 + a_1 z^{-1}}, \quad (2-11)$$

čiji su estimirani parametri:

$$\hat{\Theta} = [0.1478 \ 2.0005] . \quad (2-12)$$

Uz pretpostavljeni manji red modela od stvarnog procesa dobivamo model koji nije u stanju u potpunosti rekonstruirati vladanje stvarnog procesa, ali mu je dosta blizu (slika 2.28).



Slika 2.28: Usporedba odziva modela manjeg reda i odziva procesa na step pobudu.

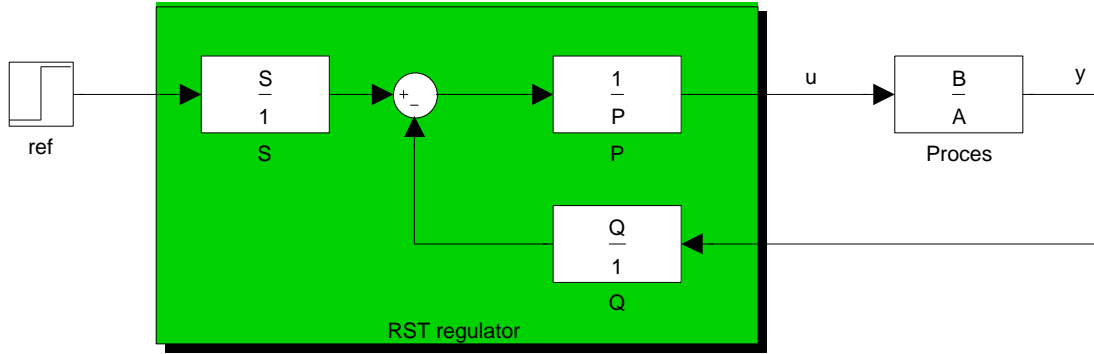
Kao mjera poklapanja odziva modela i stvarnog procesa na step u gornjim primjerima je korištena sljedeća funkcija:

$$fit [\%] = \left(1 - \frac{\|y_p - y_m\|}{\|y_p - \bar{y}_p\|} \right) \cdot 100\%, \quad (2-13)$$

gdje je y_p snimljeni odziv stvarnog procesa na step, \bar{y}_p srednja vrijednost tog odziva, y_m odziv modela na step, a operator $\| \cdot \|$ standardna euklidska norma vektora.

2.3 Adaptivni RST regulator

Opći oblik RST regulatora (polinomskog regulatora) prikazan je na slici 2.29.



Slika 2.29: Blok shema zatvorenog kruga upravljanja s RST regulatorom.

Proces je zadan prijenosnom funkcijom (uz vrijeme diskretizacije $T_s = 1s$):

$$G(z) = \frac{B(z)}{A(z)} = \frac{b_0 z^{-1} + b_1 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}, \quad (2-14)$$

dok se željeno vladanje zatvorenog kruga zadaje modelskom prijenosnom funkcijom:

$$G_M(z) = \frac{B_M(z)}{A_M(z)}. \quad (2-15)$$

Pri tome vrijedi $\deg B_M = \deg B$ i $\deg A_M = \deg A$. Diskretni polinom $A_M(z)$ može se dobiti preslikavanjem željenih polova iz kontinuirane domene. Odabran je željeni modelski polinom $A_M(s)$ drugog reda koji je oblika:

$$A_M(s) = s^2 + 2\zeta\omega_n s + \omega_n^2. \quad (2-16)$$

Polove modelskog zatvorenog kruga upravljanja možemo namjestiti preko parametara ζ i ω . Uzmimo da su željeni pokazatelji kvalitete odziva zatvorenog kruga upravljanja maksimalno nadvišenje $\sigma = 10\%$ i vrijeme smirivanja $t_{1\%} = 5s$. Pomoću izraza 2-17 možemo izračunati odgovarajuće parametre ζ i ω pa zatim rješavanjem jednadžbe $A_M(s) = 0$ dolazimo do polova $s_{1,2}$ u kontinuiranoj domeni, koje jednostavnim preslikavanjem $z_{m1,m2} = e^{T_s s_{1,2}}$ prebacujemo u diskretnu domenu.

$$\begin{aligned} \sigma &= 100e^{-\frac{\zeta\pi}{\sqrt{1-\zeta^2}}}, \\ t_{1\%} &= \frac{4.6}{\zeta\omega_n}. \end{aligned} \quad (2-17)$$

Tako dobijemo polinom $A_M(z)$ kao $A_M(z) = (z - z_{m1})(z - z_{m2})$. Polinom $B_M(z)$ odabire se tako da bude:

$$B_M(z) = B(z) \frac{A_M(1)}{B(1)}, \quad (2-18)$$

kako bi u stacionarnom stanju bilo jedinično pojačanje.

Iz sheme na slici 2.29 možemo odrediti prijenosnu funkciju zatvorenog kruga upravljanja:

$$G_{cl} = \frac{B(z)S(z)}{A(z)P(z) + B(z)Q(z)} = \frac{A_o(z)B_M(z)}{A_o(z)A_M(z)}, \quad (2-19)$$

koja je izjednačena s modelskom prijenosnom funkcijom 2-15 proširenom u brojniku i nazivniku s tzv. observerskim polinomom $A_o(z)$. Uzima se da su polinomi $A_M(z)$, $A_o(z)$ i $P(z)$ monici. Iz 2-19, uz uvrštenje 2-18 dobivamo:

$$S(z) = \frac{A_M(1)}{B(1)} A_o(z) \quad (2-20)$$

$$A(z)P(z) + B(z)Q(z) = A_o(z)A_M(z). \quad (2-21)$$

Regulator je kauzalan ako su zadovoljeni uvjeti:

$$\begin{aligned} \deg A_o &\geq 2\deg A - \deg A_M - 1 + i, \\ \deg P &= \deg A_o + \deg A_M - \deg A = \deg A_o, \\ \deg Q &< i + \deg A, \end{aligned} \quad (2-22)$$

gdje je i broj integratora uključenih u regulator. Regulator najnižeg reda dobije se uz izbor $\deg P = \deg Q = \deg S = \deg A_o = \deg A = \deg A_M = 2$.

Polinom $S(z)$ određen je u 2-20 dok se polinomi $P(z)$ i $Q(z)$ dobiju kao rješenje polinomske Diophantove jednadžbe 2-21, koja izjednačavanjem koeficijenata uz iste potencije operatora z prelazi u sustav linearnih jednadžbi.

Za naš primjer izabran je regulator najnižeg reda te jedan integrator uz dead-beat observerski polinom ($A_o = z^2$) pa Diophantovu jednadžbu možemo pisati ovako:

$$(z^2 + a_1z + a_2)(z - 1)(z + p_0) + (b_0z + b_1)(q_2z^2 + q_1z + q_0) = (z - z_{m_1})(z - z_{m_2})z^2, \quad (2-23)$$

gdje su a_1 , a_2 , b_0 , b_1 , z_{m_1} i z_{m_2} konstante, a p_0 , q_0 , q_1 i q_2 parametri koje treba izračunati. Sljedećim naredbama u Matlabu lako dolazimo do izraza za parametre u ovisnosti o navedenim konstantama:

Matlab kod 2: diofant.m - skripta koja rješava Diopfantovu jednadžbi

```
1 clear; clc;
2
3 syms z a1 a2 b0 b1 p0 q2 q1 q0 zm1 zm2
4
5 left=collect((z^2+a1*z+a2)*(z-1)*(z+p0)+...
6             (b0*z+b1)*(q2*z^2+q1*z+q0));
7 right=collect((z-zm1)*(z-zm2)*z^2);
8 lc=coeffs(left,z);
9 rc=coeffs(right,z);
10
11 rj=solve(lc(end-1)-rc(end-1),lc(end-2)-rc(end-2),...
12         lc(1),lc(2),'p0,q0,q1,q2');
13
14 p0=simplify(simple(rj.p0));
15 q0=simplify(simple(rj.q0));
16 q1=simplify(simple(rj.q1));
17 q2=simplify(simple(rj.q2));
18
19 fprintf('p0=%s;\n',char(p0));
20 fprintf('q0=%s;\n',char(q0));
21 fprintf('q1=%s;\n',char(q1));
22 fprintf('q2=%s;\n',char(q2));
```

Time su dobiveni polinomi $P(z)$ i $Q(z)$, a $S(z)$ znamo već od prije:

$$\begin{aligned} P(z) &= z^2 + z(p_0 - 1) - p_0 \\ Q(z) &= q_2 z^2 + q_1 z + q_0 \\ S(z) &= s_2 z^2 = \frac{1 - z_{m1} - z_{m2} + z_{m1} z_{m2}}{b_0 + b_1} z^2. \end{aligned} \quad (2-24)$$

Iz sheme na slici 2.29 dobivamo izraz za upravljački signal:

$$u = \frac{1}{P}(Sr - Qy). \quad (2-25)$$

Uvrštavanjem polinoma 2-24 u izraz 2-25 dobivamo rekurzivnu relaciju za upravljački signal:

$$u(k) = (1 - p_0)u(k-1) + p_0 u(k-2) + s_2 r(k) - q_2 y(k) - q_1 y(k-1) - q_0 y(k-2). \quad (2-26)$$

U sklopu korištenog libraryja za Simulink, implementiran je i blok RST regulatora (slika 2.30). Objašnjenja pojedinih ulazno-izlaznih signala dana su u tablici 2.2.



Slika 2.30: Blok za RST estimaciju parametara.

Tablica 2.2: Objašnjenja ulazno-izlaznih signala bloka RLS estimacije.

ULAZI	OBJAŠNJENJE
Θ	vektor trenutnih vrijednosti parametara procesa
ref	trenutna vrijednost referentnog signala
y	trenutna vrijednost povratne petlje
IZLAZI	OBJAŠNJENJE
u	trenutna vrijednost upravljačkog signala

RST blok koristi *m-funkciju* *RST_student* koju je trebalo dopuniti korisnim naredbama. Na osnovu rezultata dobivenih u Matlab kod 2 te rekurzivne relacije 2-26 napisana je navedena *m-funkcija*:

Matlab kod 3: RST_student.m - *m-funkcija* u kojoj je implementiran dizajnirani RST regulator.

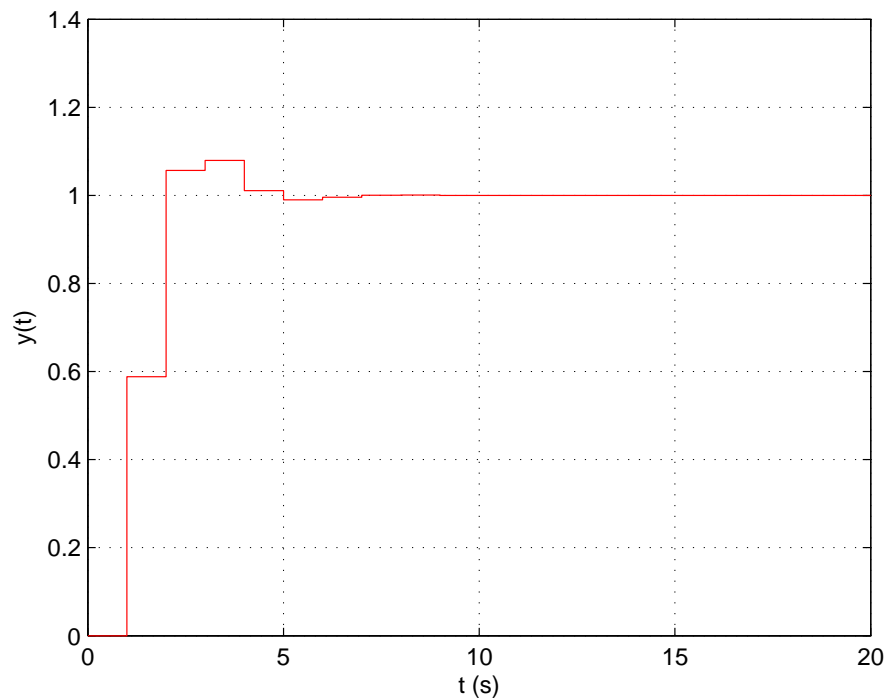
```
1 function uc = RST.student(ref, y, theta, na, nb)
2 a1=theta(1); a2=theta(2); b0=theta(3); b1=theta(4);
3 Ts=1; sigma=10; tlpusto=5;
4 zeta=abs(log(sigma/100))/sqrt(pi^2+(log(sigma/100))^2);
5 wn=4.6/tlpusto/zeta;
6 s=roots([1 2*zeta*wn wn^2]);
7 zm1=exp(s(1)*Ts); zm2=exp(s(2)*Ts);
8 persistent u_k1 y_k1 u_k2 y_k2
9 if isempty(u_k1), u_k1=0; y_k1=0; u_k2=0; y_k2=0; end
```



```

10 p0=real(b1/b0 - (b1^2*(b1 + b0*zml)*(b1 + b0*zm2))/...
11     (b0*(b0 + b1)*(a2*b0^2 - a1*b0*b1 + b1^2)));
12 q0=real(-(a2*b1^2*(a1 + zm1 + zm2 - 1) - a2^2*b0^2 ...
13     + a2*b0*b1*(a1 - a2 + zm1*zm2))/((b0 + b1)...
14     *(a2*b0^2 - a1*b0*b1 + b1^2)));
15 q1=real((a1*b0^2 - a2*b0^2 + b1^2 + b0^2*zm1*zm2 + ...
16     b0*b1*zm1 + b0*b1*zm2)/b0^3 - (b1^2*(b1 + ...
17     b0*zm1)*(b1 + b0*zm2)*(b0 + b1 - a1*b0))/...
18     (b0^3*(b0 + b1)*(a2*b0^2 - a1*b0*b1 + b1^2)));
19 q2=real(-(a1*a2 - a2 + a2*zm1 + a2*zm2)*b0^2 + (a1 + ...
20     a1*a2 - a1*zm1 - a1*zm2 + a2*zm1 + a2*zm2 - ...
21     a1^2)*b0*b1 + (a1 + a2 + zm1 + zm2 - a1*zm1 - ...
22     a1*zm2 - zm1*zm2 - a1^2 - 1)*b1^2)/((b0 + b1)*...
23     (a2*b0^2 - a1*b0*b1 + b1^2)));
24 s2=real((1-zml-zm2+zm1*zm2)/(b0+b1));
25 uc=(1-p0)*u_k1+p0*u_k2+s2*ref-q2*y-q1*y_k1-q0*y_k2;
26 if isnan(uc), uc=0.1; end
27 if abs(uc)>0.7, uc=sign(uc)*0.7; end
28 u_k2=u_k1;
29 u_k1=uc;
30 y_k2=y_k1;
31 y_k1=y;
32 end

```

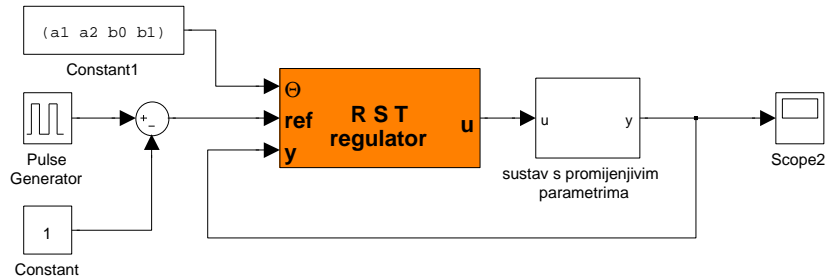


Slika 2.31: Odziv zatvorenog kruga upravljanja na step.

Na slici 2.31 je prikazan odziv zatvorenog kruga upravljanja na step funkciju. Iz odziva vidimo da je postignuo željeno vladanje - nadvišenje nije veće od 10% i vrijeme smirivanja je oko 5s.

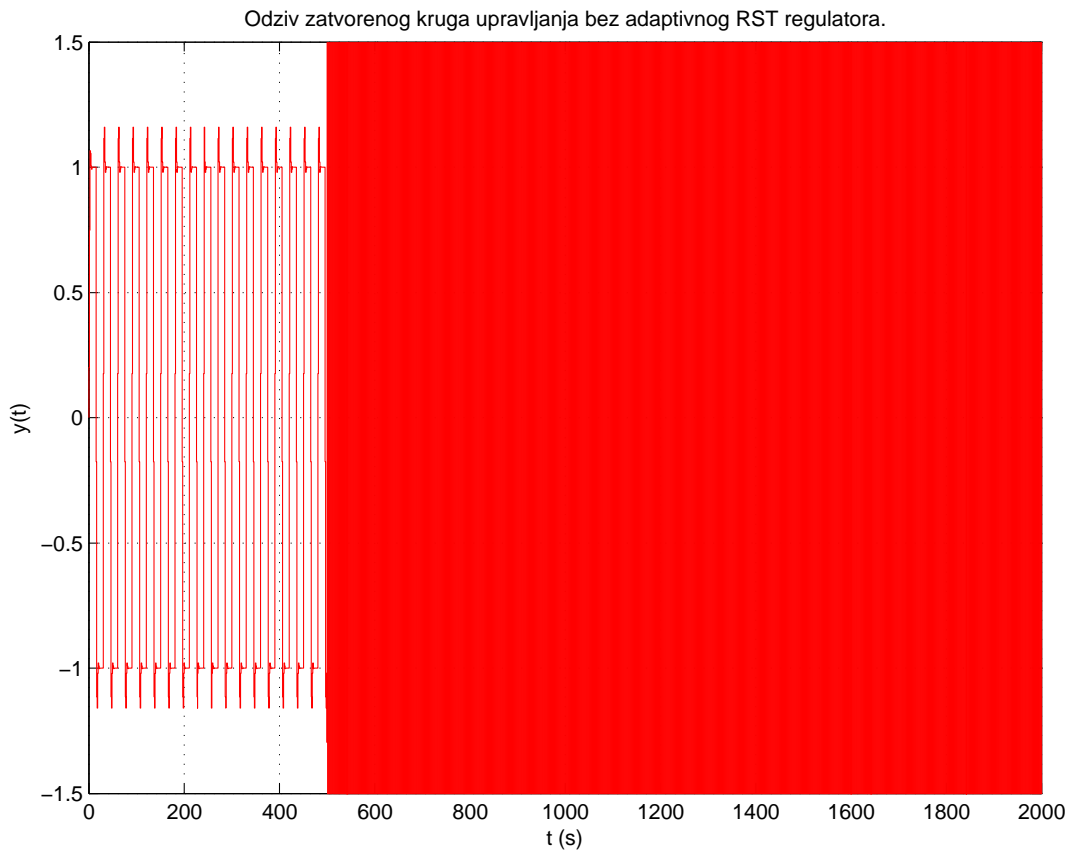
2.3.1 Odziv zatvorenog kruga upravljanja s nominalnim RST regulatorom

U ovom potpoglavlju promatramo ponašanje zatvorenog kruga upravljanja kada u procesu dođe do skokovitih promjena parametara (jednako kao i do sada - b_0 u $t = 500s$ s faktorom $K_{b_0} = 1.5$ te a_1 u $t = 1500s$ s istim faktorom). Korišten je nominalni RST regulator (bez adaptacije, što znači da ne koristimo RLS estimator parametara procesa). Simulacijska shema prikazana je na slici 2.32



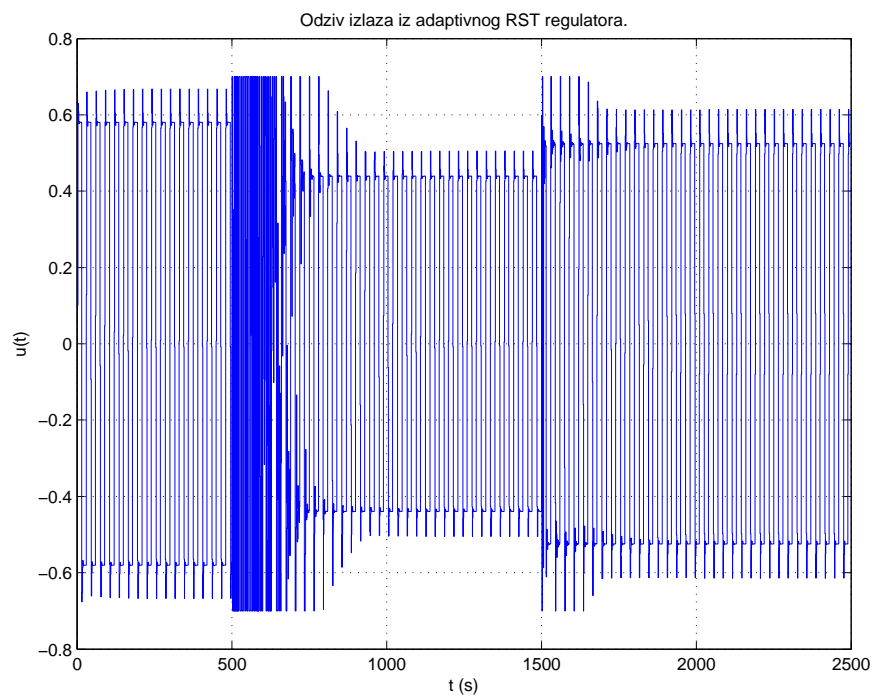
Slika 2.32: Simulacijska shema zatvorenog kruga upravljanja s nominalnim RST regulatorom.

Odziv zatvorenog kruga upravljanja prikazan je na slici 2.33

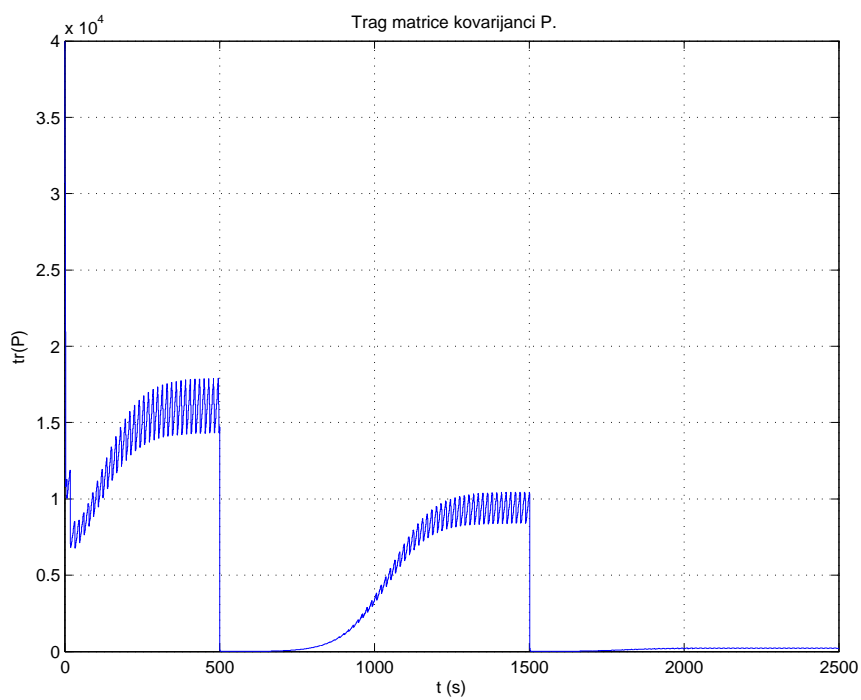


Slika 2.33: Odziv zatvorenog kruga upravljanja.

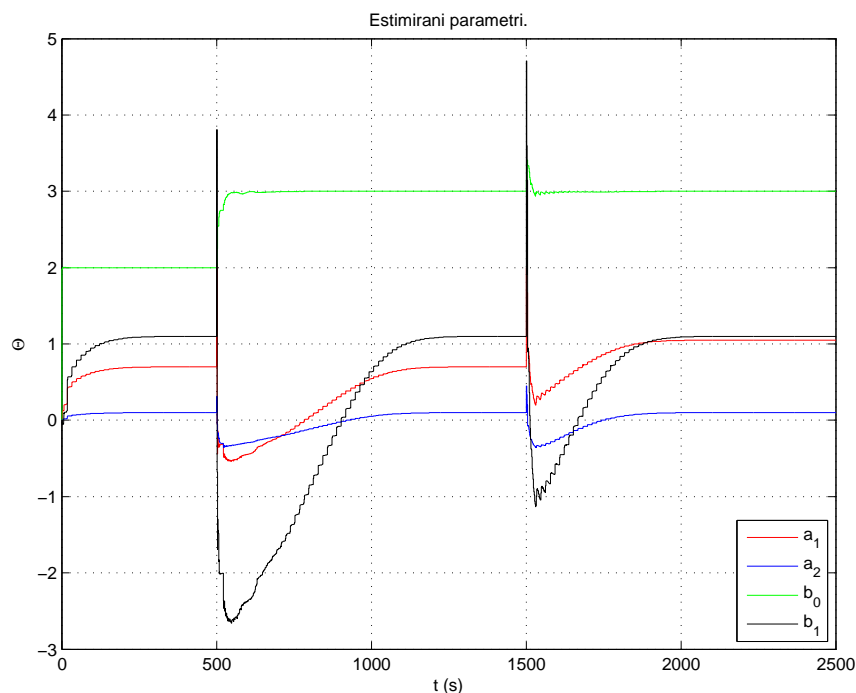
Kao što se vidi iz odziva, RST regulator radi ispravno sve do trenutka $t = 500s$, kada se promijeni prvi parametar procesa. Kako koristimo nominalni RST regulator, njegovi polinomi su određeni za nominalne parametre procesa, a budući da su se parametri procesa promijenili RST regulator više ne može raditi zadovoljavajuće.



Slika 2.36: Odziv upravljačkog signala adaptivnog RST regulatora.



Slika 2.37: Trag matrice kovarijanci P tijekom simulacije zatvorenog kruga upravljanja.



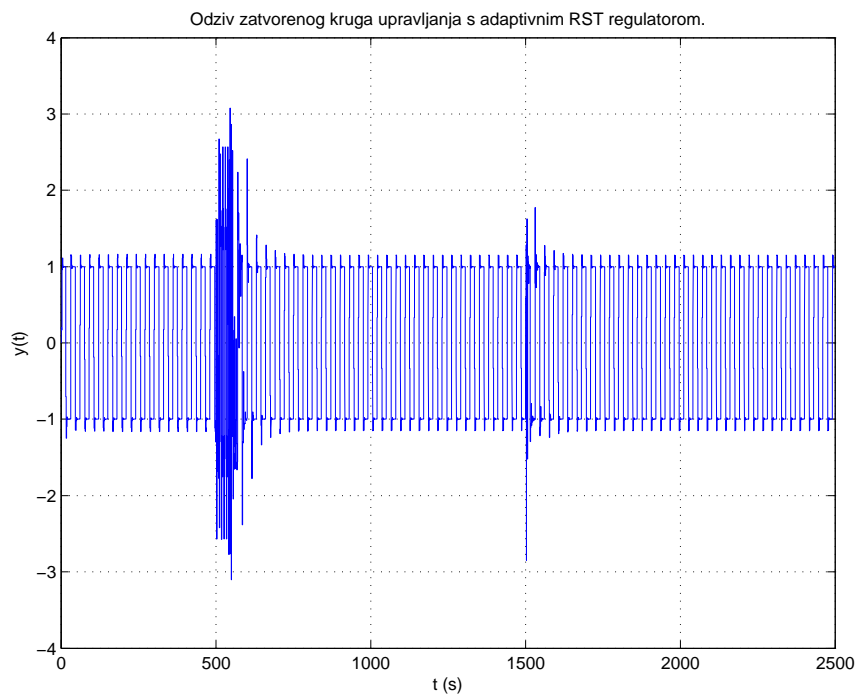
Slika 2.38: Estimirani parametri tijekom simulacije zatvorenog kruga upravljanja.

Iz odziva vidimo da je RST regulator dobro radio sve do promjene parametara procesa (prvo promjena u brojniku u $t = 500s$ pa onda promjena u nazivniku u $t = 1500s$). Nakon promjene parametra u brojniku procesa RLS estimatoru je trebalo dosta vremena da opet iskonvergira prema pravim vrijednostima parametara. U tom prijelaznom razdoblju je RST regulator računao koeficijente svojih polinoma s krivim parametrima procesa pa je upravljački signal počeo divljati (vidimo da stalno udara u ograničenje). No, to je u ovom slučaju dobro, jer je na taj način na ulaz procesa dovedena frekvencijski vrlo bogata pobuda (perzistentan ulazni signal) što je poboljšalo estimaciju parametara u RLS estimatoru. Na taj način je poboljšana konvergencija RLS estimatora pa su nakon nekog vremena opet estimirani ispravni parametri procesa. Zbog toga je RST regulator opet ustabilio odziv zatvorenog kruga upravljanja. Slična stvar se dogodi kod promjene parametra u nazivniku, no ovdje je konvergencija RLS estimatora brža pa su i manje perturbacije upravljačkog signala i izlaza iz procesa.

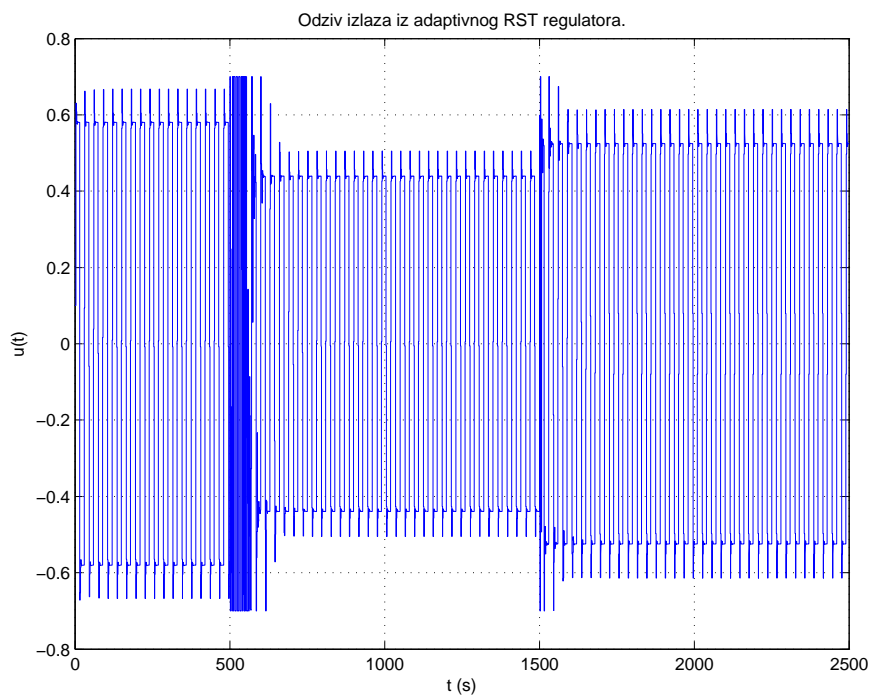
2.3.3 Utjecaj faktora zaboravljanja

U ovom potpoglavlju promatrat ćemo utjecaj smanjenja faktora zaboravljanja - u ovom slučaju ćemo koristiti $\lambda = 0.95$.

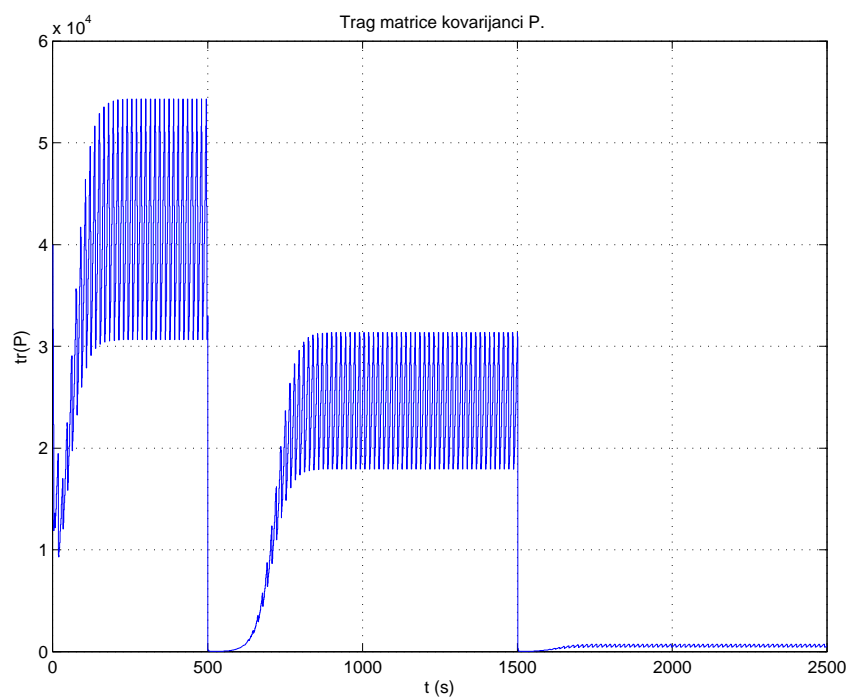
Smanjenje faktora zaboravljanja je ubrzalo konvergenciju RLS estimatora pa je i RST regulatoru potrebno manje vremena da osigura nominalno vladanje zatvorenog upravljačkog kruga nakon promjene parametara procesa (manje su perturbacije izlaznog signala i upravljačkog signala, slike 2.39 i 2.40). S druge strane, postoje veći skokovi estimiranih parametara pri skokovitoj promjeni stvarnih parametara procesa (slika 2.42). Na odzivu traga matrice P (slika 2.41) vide se veće oscilacije.



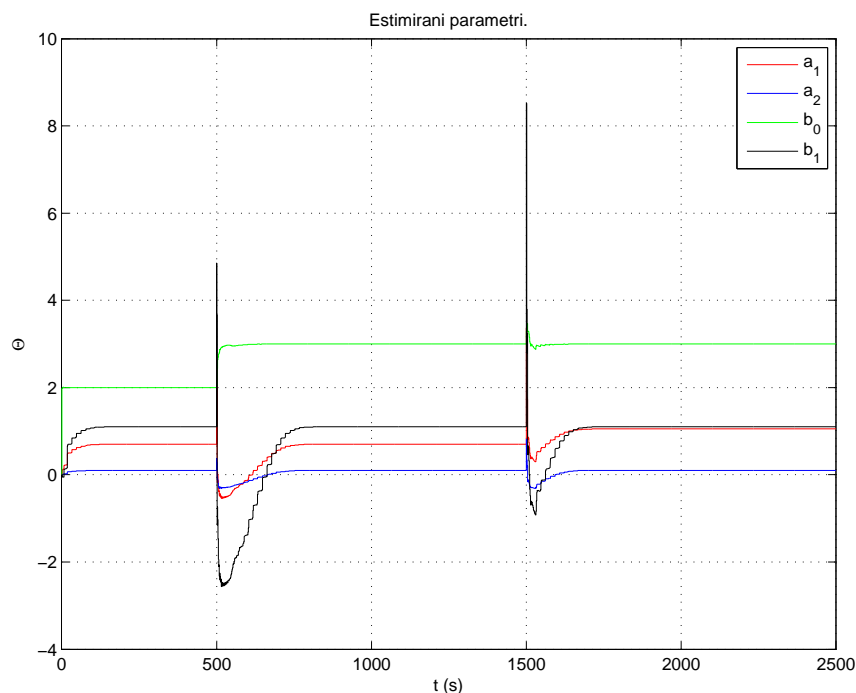
Slika 2.39: Odziv zatvorenog kruga upravljanja s adaptivnim RST regulatorom.



Slika 2.40: Odziv upravljačkog signala adaptivnog RST regulatora.



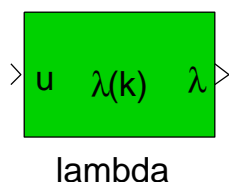
Slika 2.41: Trag matrice kovarijanci P tijekom simulacije zatvorenog kruga upravljanja.



Slika 2.42: Estimirani parametri tijekom simulacije zatvorenog kruga upravljanja.

2.3.4 Promjenjivi faktor zaboravljanja

U ovom potpoglavlju ćemo provjeriti što se događa ako se uvede promjenjivi faktor zaboravljanja. Ova funkcionalnost je implementirana u bloku prikazanom na slici 2.43. Objašnjenja ulaza, izlaza i parametara ovog bloka dana su u tablici 2.3.



Slika 2.43: Blok koji implementira dinamičku promjenu faktora zaboravljanja.

Tablica 2.3: Objašnjenja ulaza, izlaza i parametara bloka za promjenu faktora zaboravljanja.

ULAZI	OBJAŠNJENJE
u	trenutna vrijednost upravljačkog signala RST regulatora
IZLAZI	OBJAŠNJENJE
λ	faktor zaboravljanja u k-tom koraku
PARAMETRI	OBJAŠNJENJE
λ_0	početna vrijednost faktora zaboravljanja
λ_K	konačna vrijednost faktora zaboravljanja
K	pojačanje

Rekurzivni zakon po kojem se mijenja vrijednost faktora zaboravljanja je:

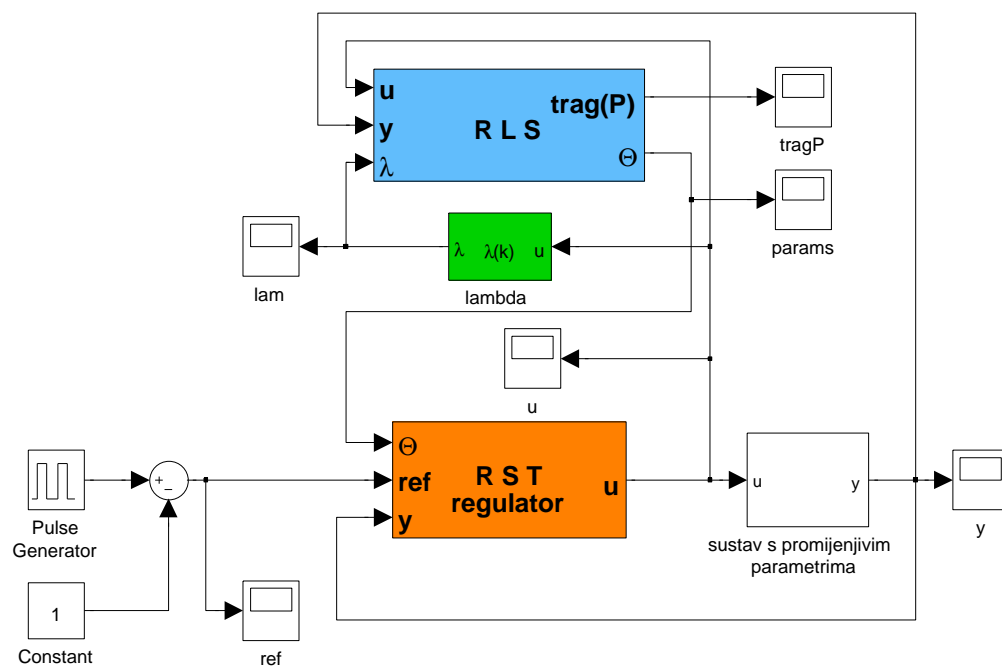
$$\begin{aligned}\lambda(k) &= K\lambda(k-1) + (1-K)\lambda_K, \\ \lambda(0) &= \lambda_0,\end{aligned}\tag{2-27}$$

gdje su λ_K , λ_0 i K parametri navedeni u tablici 2.3. Dodatno, u bloku je implementirano resetiranje trenutne vrijednosti faktora zaboravljanja na vrijednost λ_0 u slučaju da se detektira zasićenje upravljačkog signala na izlazu RST regulatora (time je posredno detektirana promjena parametara procesa). Matlab funkcija u kojoj je implementirana ova funkcionalnost je dana u nastavku.

Matlab kod 4: lambda.m - *m-funkcija* koja implementira dinamičku promjenu faktora zaboravljanja

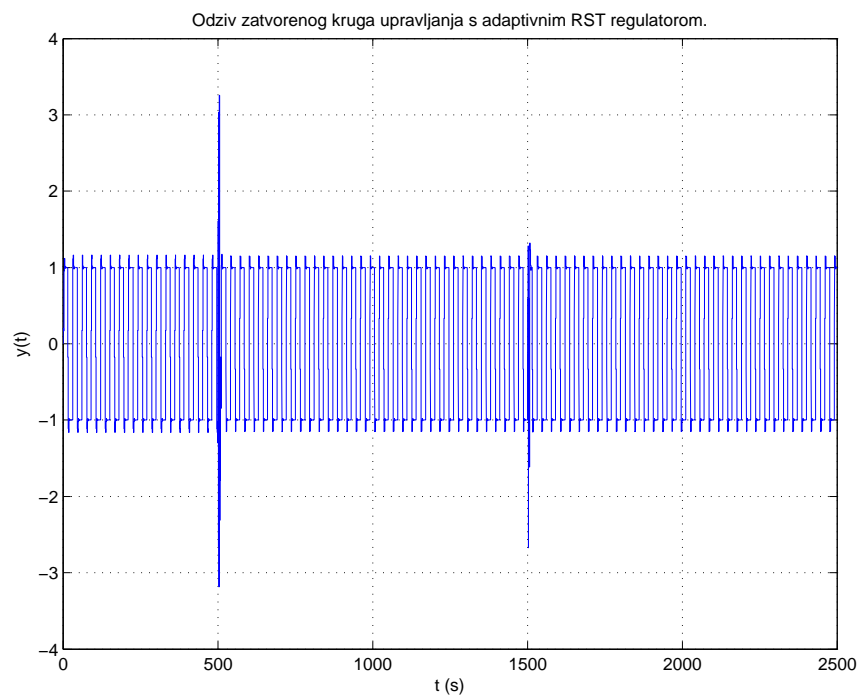
```
1 function lam=lambda(lambda0, lambdaK, K, upr)
2 persistent lambda
3 if isempty(lambda) || abs(upr) ≥ 0.7
4     new_lambda=lambda0;
5 else
6     new_lambda=K*lambda+(1-K)*lambdaK;
7 end
8 lambda=abs(new_lambda);
9 lam=abs(new_lambda);
```

Simulacijska shema koja je korištena u ovom slučaju prikazana je na slici 2.44.

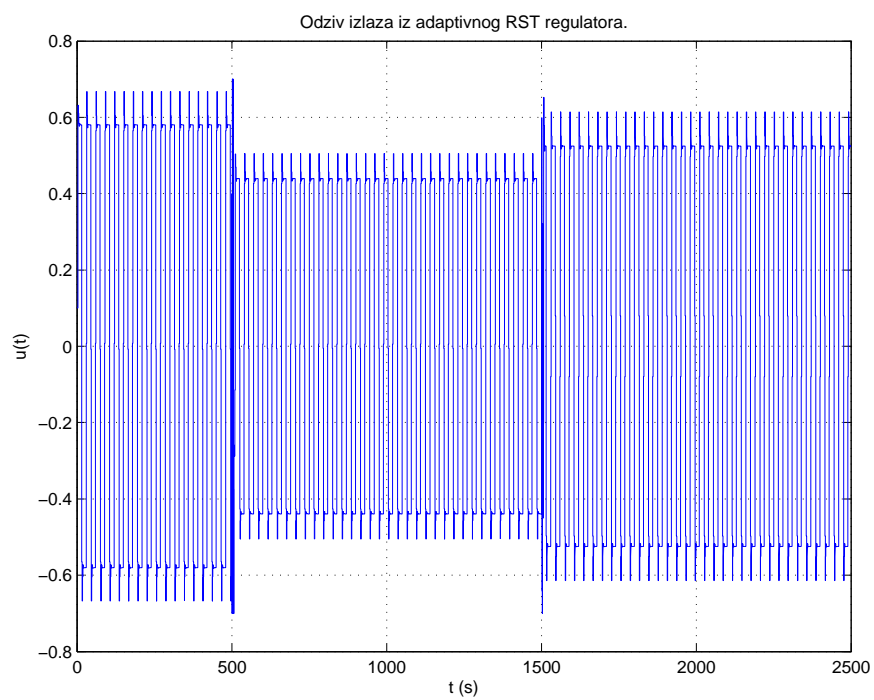


Slika 2.44: Simulacijska shema zatvorenog kruga upravljanja s adaptivnim RST regulatorom uz promjenjivi faktor zaboravljanja.

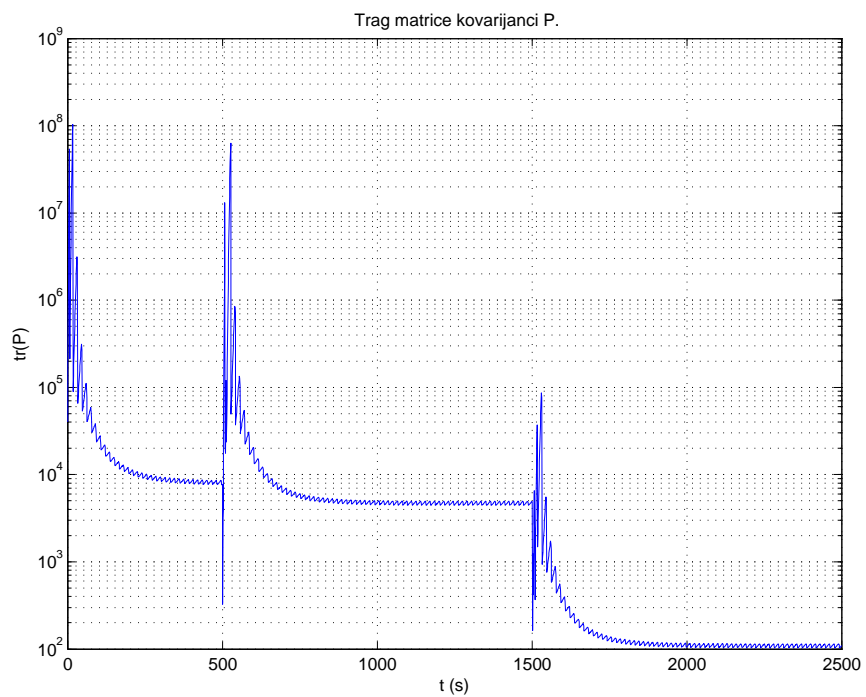
Parametri koji su korišteni u bloku za promjenu faktora zaboravljanja su: $K = 0.95$, $\lambda_0 = 0.1$ i $\lambda_K = 0.99$. Iz odziva na slikama 2.45 – 2.48 vidimo veliko poboljšanje vladanja zatvorenog kruga upravljanja uvođenjem promjenjivog faktora zaboravljanja. U odnosu na slučajeve kada je faktor zaboravljanja bio konstantan ($\lambda = 0.98$ i $\lambda = 0.95$) uočava se puno brža konvergencija RLS estimatora uz iste, odnosno manje skokove estimiranih parametara prilikom skokovite promjene pravih parametara procesa. To odmah znači i puno kvalitetnije ponašanje zatvorenog kruga upravljanja, jer je adaptacija na promjene parametara procesa brža, a samim time su i perturbacije upravljačkog i izlaznog signala manje.



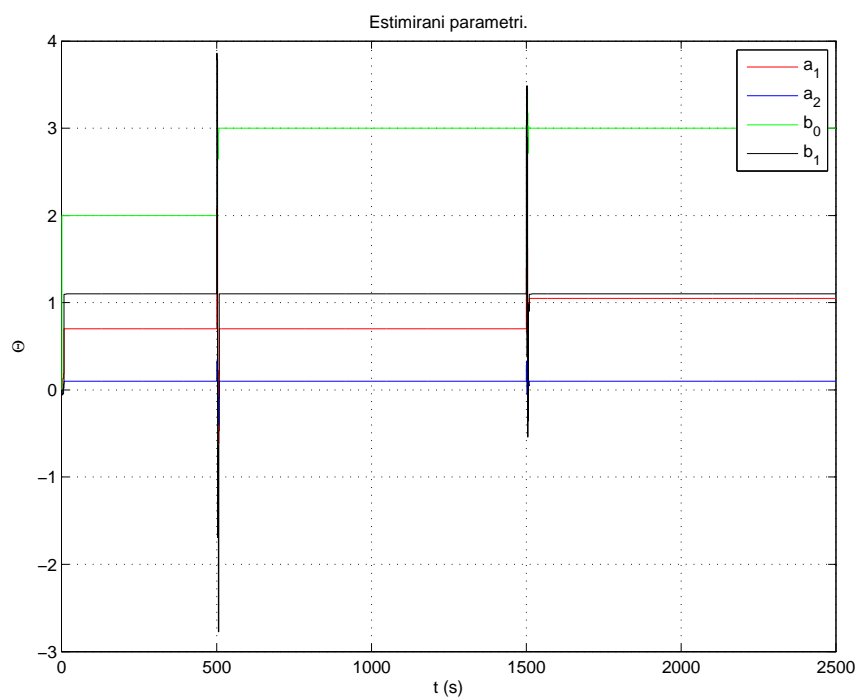
Slika 2.45: Odziv zatvorenog kruga upravljanja s adaptivnim RST regulatorom.



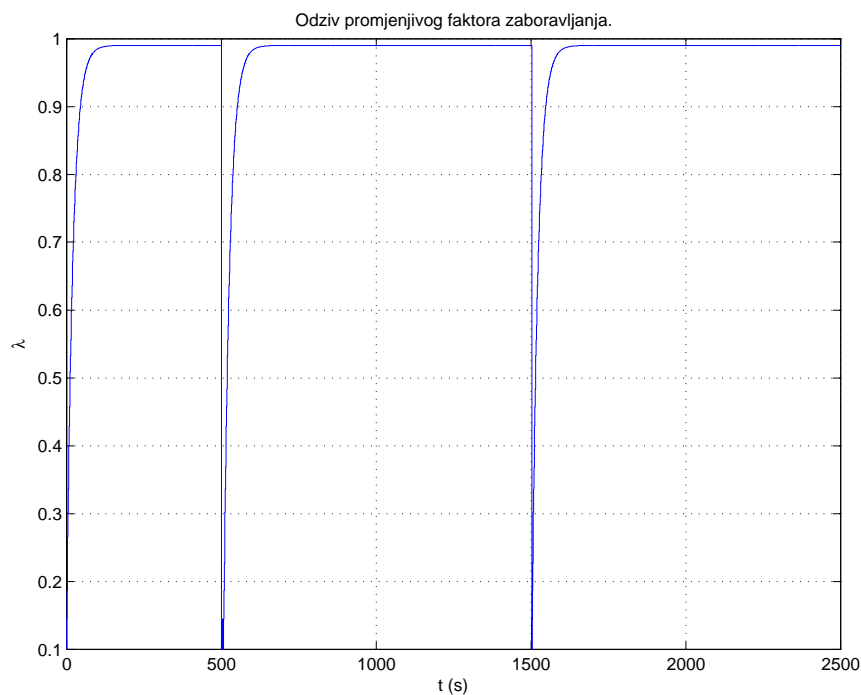
Slika 2.46: Odziv upravljačkog signala adaptivnog RST regulatora.



Slika 2.47: Trag matrice kovarijanci P tijekom simulacije zatvorenog kruga upravljanja.



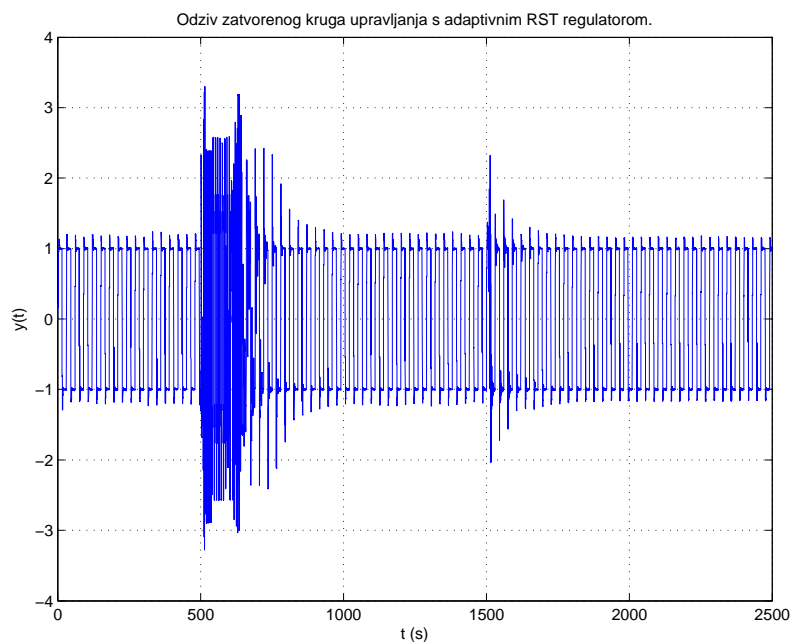
Slika 2.48: Estimirani parametri tijekom simulacije zatvorenog kruga upravljanja.



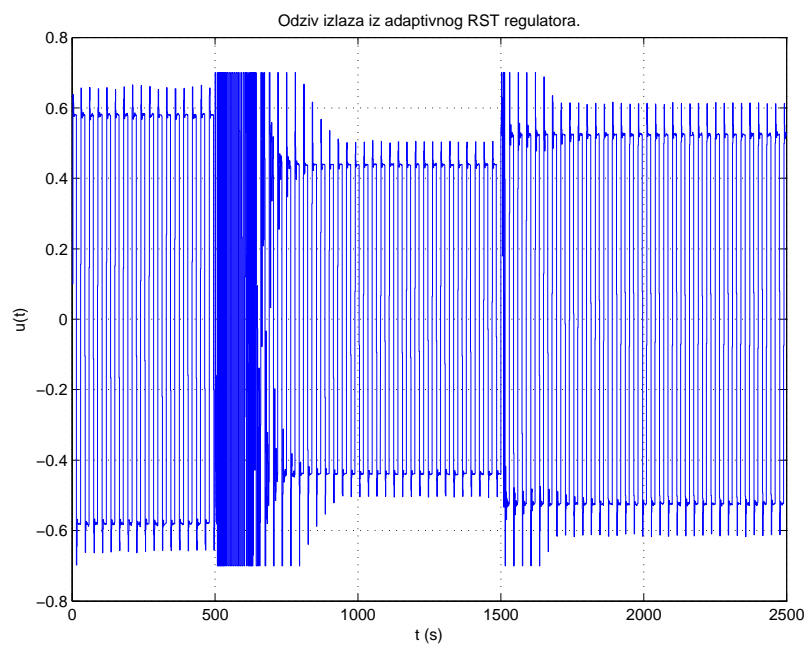
Slika 2.49: Odziv promjenjivog faktora zaboravljanja λ .

2.3.5 Šum na izlazu procesa

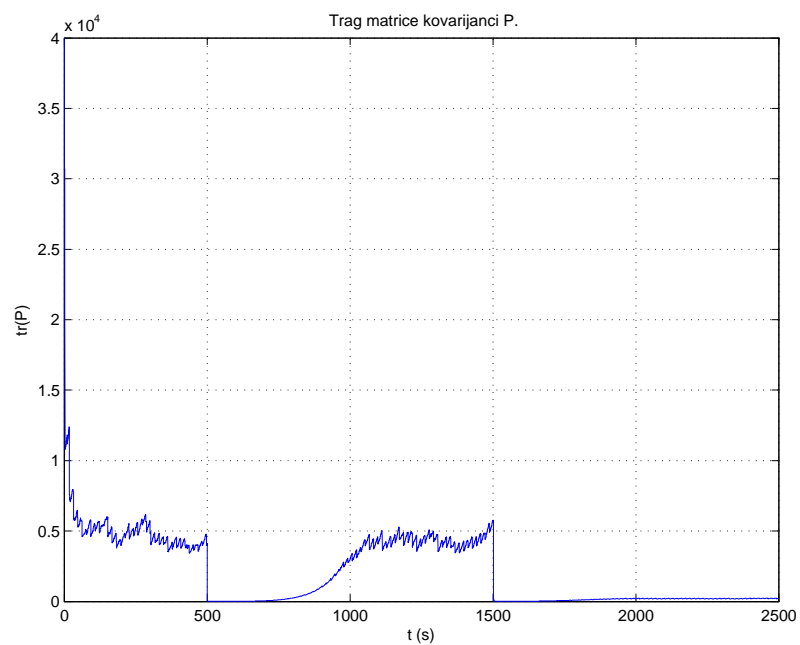
U ovom potpoglavlju opet koristimo konstantan faktor zaboravljanja ($\lambda = 0.98$) i promatramo što se događa ako dodamo šum na izlaz procesa. Kao što se vidi iz odziva na slikama 2.50 – 2.53, dodavanje šuma na izlaz procesa pogoršava estimaciju. Glavna pretpostavka RLS metode estimacije parametara je da poremećaj u procesu ne korelira ni sa ulaznim ni sa izlaznim signalom procesa. Kako u zatvorenom krugu upravljanja šum preko povratne veze propagira prema ulazu procesa, ta pretpostavka više nije zadovoljena, jer ulazni signal u proces i šum postaju korelirani.



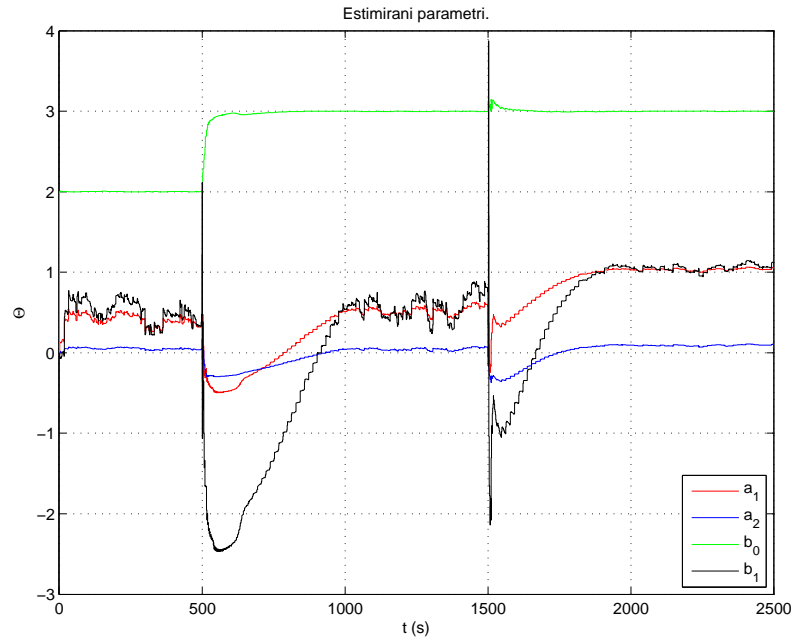
Slika 2.50: Odziv zatvorenog kruga upravljanja s adaptivnim RST regulatorom.



Slika 2.51: Odziv upravljačkog signala adaptivnog RST regulatora.



Slika 2.52: Trag matrice kovarijanci P tijekom simulacije zatvorenog kruga upravljanja.

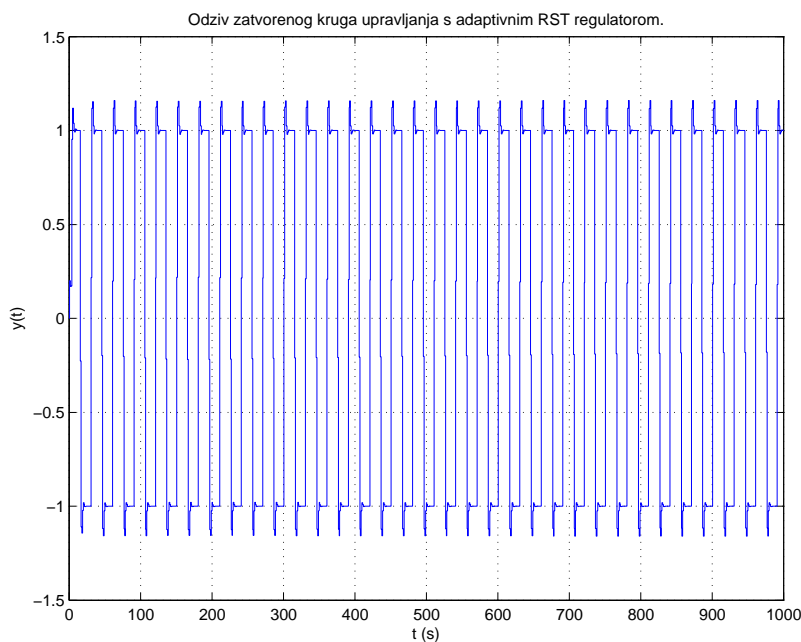


Slika 2.53: Estimirani parametri tijekom simulacije zatvorenog kruga upravljanja.

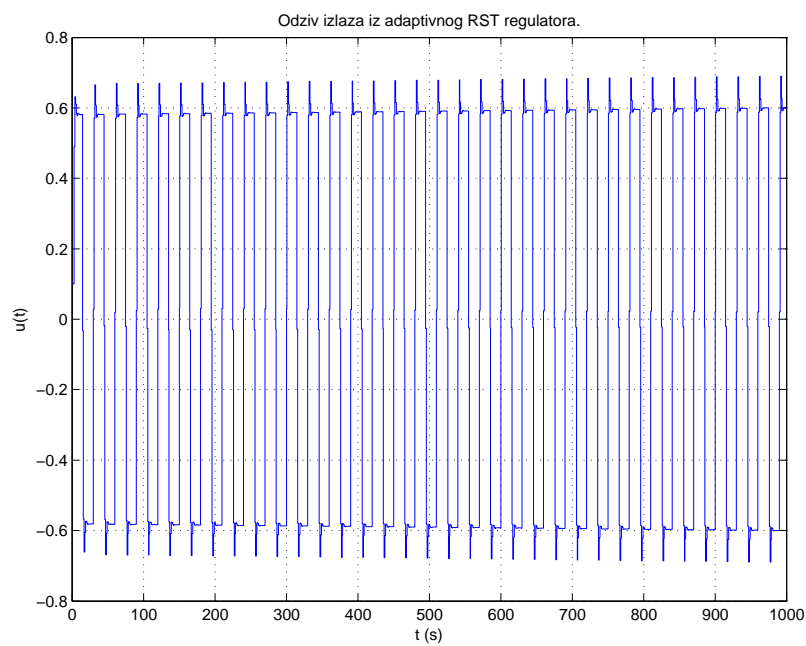
2.3.6 Linearna promjena parametra procesa

Pogledajmo sada što se događa s estimacijom kada se jedan parametar mijenja linearno u vremenu (do sada su promjene parametara bile skokovite). Mijenja se parametar b_0 od svoje nominalne vrijednosti na početku simulacije do 90% svoje nominalne vrijednosti na kraju simulacije. Korišten je promjenjivi faktor zaboravljanja ($K = 0.95$, $\lambda_0 = 0.1$, $\lambda_K = 0.92$).

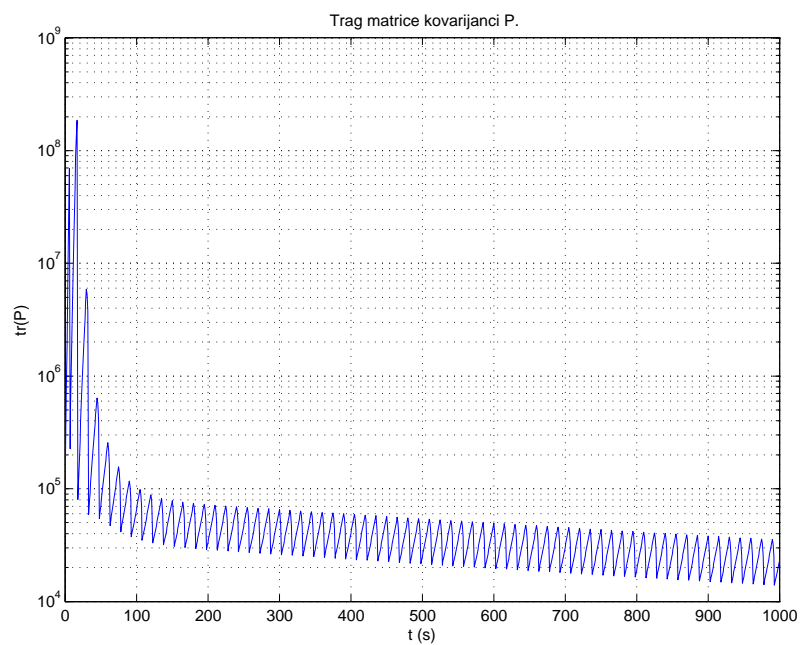
Iz odziva se vidi da je RLS estimator u stanju pratiti linearnu promjenu parametra b_0 , no isto tako se vide i oscilacije u estimaciji parametara a_1 te b_1 . Unatoč tome, vladanje zatvorenog kruga upravljanja je solidno.



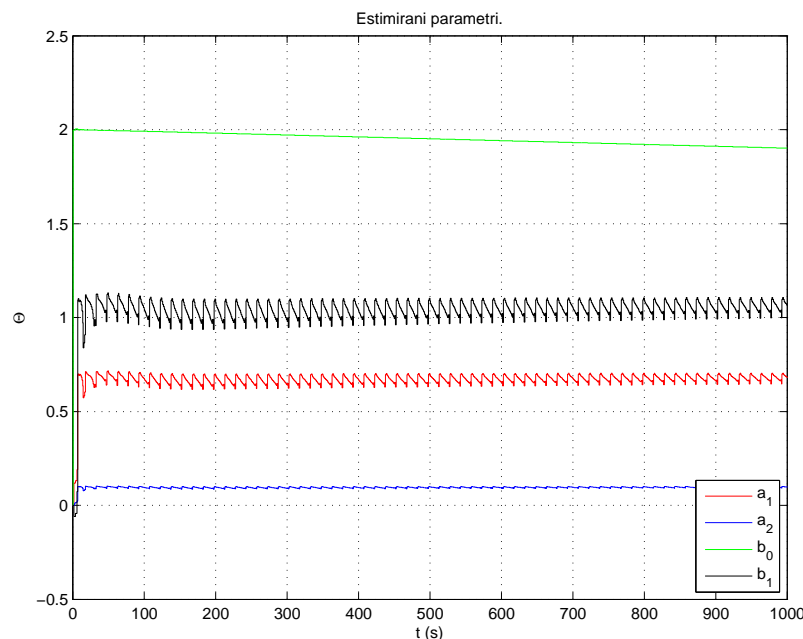
Slika 2.54: Odziv zatvorenog kruga upravljanja s adaptivnim RST regulatorom.



Slika 2.55: Odziv upravljačkog signala adaptivnog RST regulatora.



Slika 2.56: Trag matrice kovarijanci P tijekom simulacije zatvorenog kruga upravljanja.



Slika 2.57: Estimirani parametri tijekom simulacije zatvorenog kruga upravljanja.

2.4 Zaključak

U sklopu ovog seminara obrađena je rekurzivna metoda estimacije parametara procesa zasnovana na metodi najmanjih kvadrata (RLS metoda). Također je obrađen i algoritam upravljanja zatvorenim krugom koji se temelji na metodi postavljanja polova (RST regulator, polinomski regulator). Razvijeni RST regulator ima adaptivna svojstva, jer koristi RLS metodu koja estimira promjenjive parametre procesa on-line i na taj način omogućava RST regulatoru da osvježi koeficijente svojih polinoma u stvarnom vremenu i na taj se način prilagodi promjenjenim parametrima procesa.

U prvom dijelu seminara je obrađena RLS metoda estimacije parametara procesa. Analizirani su utjecaji raznih parametara na rad algoritma estimacije. Pokazalo se da je jedan od nedostataka RLS metode potreba za inicijalizacijom pojedinih varijabli (matrica kovarijanci te početni vektor estimiranih parametara), jer razne inicijalizacije daju drastično različite rezultate. Također je važno paziti da sustav čije parametre estimiramo pobuđujemo perzistentnom pobudom, jer inače estimacija neće raditi dobro. Kod estimacije vremenski varijantnih procesa nužno je koristiti neki oblik faktora zaboravljanja (bio on konstantan ili promjenjiv) kako bi se matrica kovarijanci održala na nekoj pozitivnoj vrijednosti, jer se inače estimacija nakon nekog vremena gasi (matrica kovarijanci teži prema nul-matrici). Ako je faktor zaboravljanja konstantan, njegovo smanjenje utječe na bržu estimaciju, ali i veće oscilacije estimiranih parametara. Uz pobudu koja nije perzistentna može doći i do tzv. *blow-up* učinka kod matrice kovarijanci (njezini elementi počnu neograničeno rasti). Na kraju, vidjeli smo da RLS metoda dosta dobro radi i ako pretpostavljeni red modela nije jednak redu stvarnog procesa. Ako nismo sigurni koji je stvarni red procesa, uvijek je bolje pretpostaviti veći red, jer ćemo na taj način sigurno obuhvatiti dinamiku stvarnog procesa. Ako pretpostavimo manji red od stvarnog, nećemo moći postići svu dinamiku stvarnog procesa, no to može biti opravdano ako su razlike u ponašanjima modela i procesa zanemarive. Na taj način smo dobili vrlo jednostavan model koji i dalje dosta dobro opisuje stvarni proces.

U drugom dijelu seminara bavili smo se adaptivnim RST regulatorom. Pokazali smo da metodom postavljanja polova možemo vrlo jednostavno postići da se zatvoreni krug upravljanja ponaša prema nekoj željenoj modelskoj funkciji. Adaptivni RST regulator koristi RLS estimaciju parametara procesa pa je opet bitno dobro inicijalizirati RLS algoritam. Pokazalo se da je bolje koristiti promjenjivi faktor zaboravljanja od konstantnog. U sklopu ovog seminara korištena je promjena faktora zaboravljanja temeljena na filtru prvog reda uz resetiranje filtra svaki put kad se detektira promjena parametara procesa. Detekcija promjene parametara procesa napravljena je posredno preko upravljačkog signala RST regulatora. Naime, kada se promjene parametri procesa, RLS estimator ne može odmah dati nove parametre, pa će neko vrijeme RST regulator računati koeficijente svojih polinoma pomoću krivih parametara procesa i time će postati nestabilan, tj. upravljački signal će udariti u ograničenje. Prema tome, detekcijom zasićenja upravljačkog signala možemo posredno detektirati da su se promijenili parametri procesa. Na ovaj način postignuto je vrlo kvalitetno vladanje zatvorenog kruga upravljanja. Konvergencija RLS estimatora je drastično ubrzana, bez povećanja naglih skokova estimiranih parametara uslijed skokovite promjene pravih parametara procesa. Zbog toga nema ni velikih perturbacija upravljačkog signala na izlazu RST regulatora, tj. izlaznog signala iz procesa.