

Univerzitet u Tuzli

Fakultet elektrotehnike

Automatika I robotika

ZADAĆA 4

Predmet: Stohastički sistemi i estimacije

Profesor: Prof. Dr. Lejla Banjanović-Mehmedović

Asistent: Azra Grudić Ribić

Mjesto i datum:

Tuzla, 14.01.2023.

Student:

Belma Nurkić

Zadatak 1 (4 boda)

Dizajnirati Kalmanov steady-state filter. Matrice A, B, C i D dobiti na osnovu podataka raspodijeljenih u excel fajlu Zadaća 4 – raspodjela podataka, kao i kovarijanse šuma procesa i mjerenja. Komentarisati dobijene rezultate.

Matlab kod:

```
data=xlsread('podaci4.xlsx');
data1=xlsread('zadaca4')
U2=data(602:701,3);
Y1=data(602:701,4);
set = iddata(Y1,U2,10);

Q=data1(32:32,6);
R=data1(32:32,7);

A = ss1.A;
B = ss1.B;
C = ss1.C;
D = ss1.D;
%Ispis matrica
A
B
C
D

Plant = ss(A,[B B],C,0,-1,'inputname',{'u' 'w'},'outputname','y');
[kalmf,L~,M,Z] = kalman(Plant,Q,R);
kalmf = kalmf(1,:);
M, % innovation gain
% Model sistema sa u,w,v ulazima i
% y i yv izlazima:
a = A;
b = [B B 0*B];
c = [C;C];
d = [0 0 0;0 0 1];
P = ss(a,b,c,d,-1,'inputname',{'u' 'w' 'v'},'outputname',{'y' 'yv'});
sys = parallel(P,kalmf,1,1,[],[]);
SimModel = feedback(sys,1,4,2,1);
SimModel = SimModel([1 3],[1 2 3]);
%Generisanje sinusoidalnog vektora
t = (0:100)';

u = sin(t/5);
%Procesnog i mjernog suma
rng(10,'twister');
w = sqrt(Q)*randn(length(t),1);
v = sqrt(R)*randn(length(t),1);
out = lsim(SimModel,[w,v,u]);
y = out(:,1); % pravi izlaz
ye = out(:,2); % filtrirani izlaz
yv = y + v; % mjereni izlaz
%Poredjenje na grafikonima
clf
subplot(211), plot(t,y,'b',t,ye,'r--'),
xlabel('Broj uzoraka'), ylabel('Izlaz')
title('Kalman filter odziv')
subplot(212), plot(t,y-yv,'g',t,y-ye,'r--'),
xlabel('Broj uzoraka'), ylabel('Greska')
% Kalmanov filter smanjuje gresku y-yv
% koja nastaje usljed mjernog suma. Za potvrdu, mozemo uporediti kovarijanse greske
```

```

MeasErr = y-yv;
MeasErrCov = sum(MeasErr.*MeasErr)/length(MeasErr);
EstErr = y-ye;
EstErrCov = sum(EstErr.*EstErr)/length(EstErr);
MeasErrCov
EstErrCov

```

Ispis:

A =

```

-0.0002    0.0122    0.0149    0.0145
-0.0302   -0.2849   -0.4550   -0.4548
-0.0441   -0.3375   -0.5625   -0.6827
-0.0261   -0.2488   -0.3048   -0.3657

```

B =

```

-0.6985
27.3001
43.9601
19.2221

```

C =

```

-0.4193    0.0221    0.0268    0.0261

```

D =

```

0

```

D =

```

0

```

M =

```

-0.2580
9.4051
14.5391
6.7022

```

MeasErrCov =

```

1.5637

```

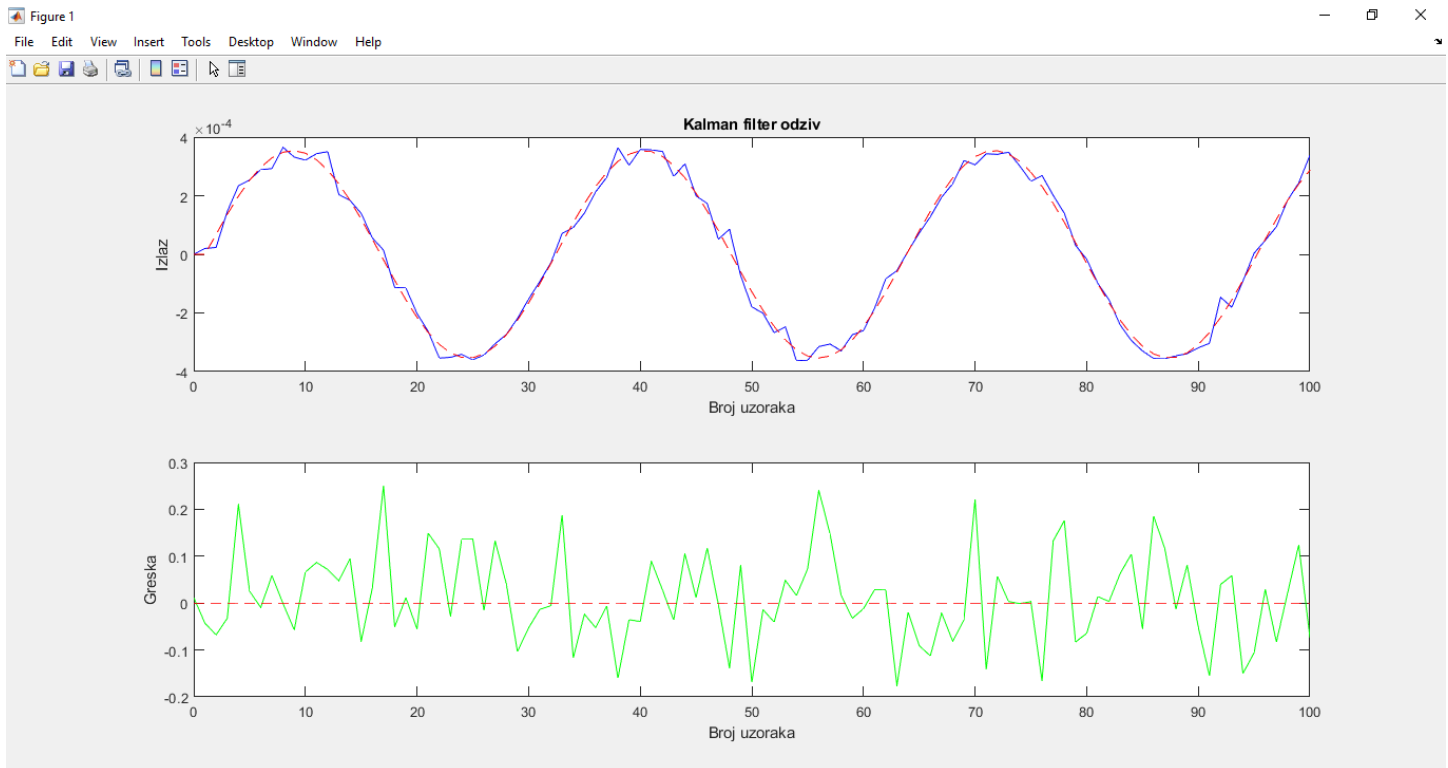
EstErrCov =

```

1.4439

```

Grafik:



Komentar:

U prvom zadatku smo dizajnirali Kalmanov stady-stay filter. Estimaciju smo izvršili preko System identification toolboxa. Dobili smo podatke za matrice A,B,C,D i kovarijansu greške, koja je dala vrijednosti oko nule, što znači da je sitem poprilično uklonio šum.

Zadatak 2 (4 boda)

Kao primjer se koristi matematičko klatno kao objekt koji se posmatra. Ugaona pozicija klatna se estimira preko Kalmanovog filtera i EKF koji su dodani u formi Simulink bloka. Kovarijansa šuma procesa i mjerenja je data excel fajlu u Zadaća 4 – kovarijanse Q i R. Kao inicijalne vrijednosti theta i theta_dot koristiti respektivno:

- a) $\pi/18$ i 0
- b) $\pi/4$ i 0
- c) $\pi/18$ i $\pi/9$
- d) $\pi/4$ i $\pi/9$

Komentarisati dobijene rezultate.

Šema matematičkog klatna je data na LV15.

Matlab kodovi:

```
% Copyright 2018 The MathWorks, Inc.
% Pendulum model
% Gravity
g = 9.81; % [m/s^2]
% Pendulum mass
m = 1; % [kg]
% Pendulum length
l = 0.5; % [m]

% State space representation
A = [0 1; -g/l 0];
B = [0; 1/(m*l^2)];
C = [1 0];
D = 0;

% Process noise covariance
Q = 1e-3;
% Measurement noise covariance
R = 1e-4;
% Sampling time
Ts = 0.01; % [s]

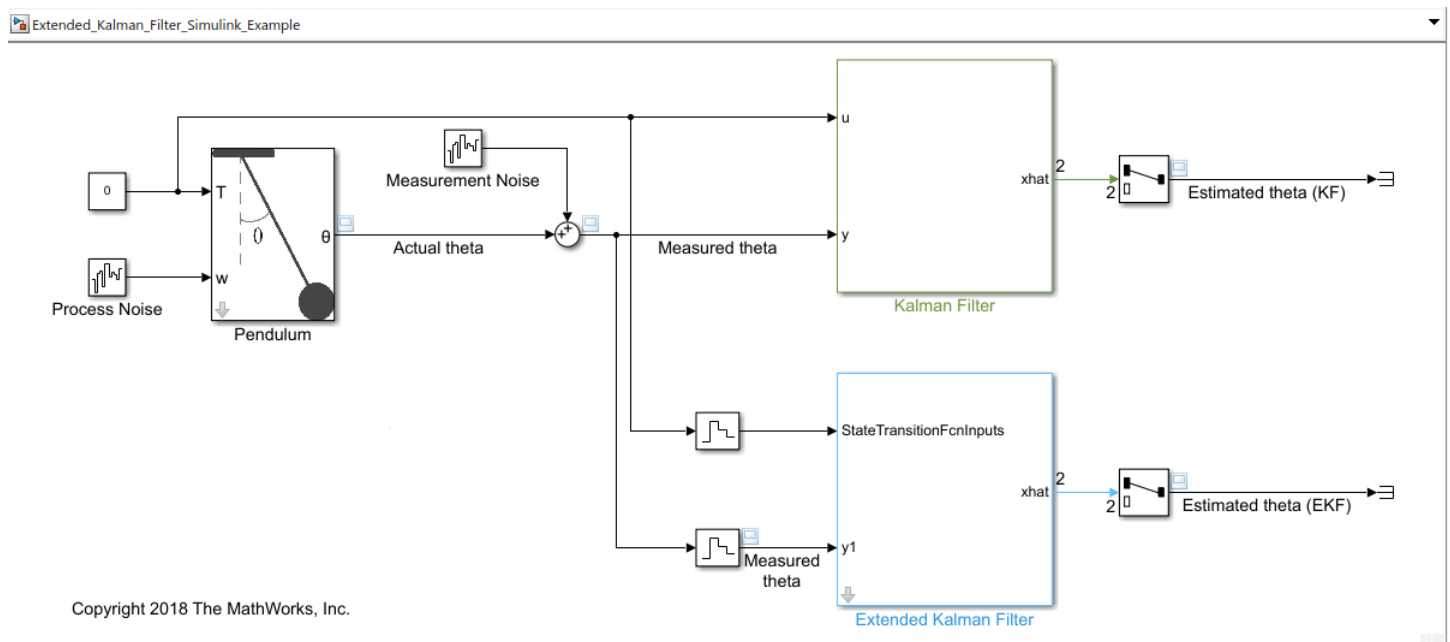
% Copyright 2018 The MathWorks, Inc.

function y = myMeasurementFcn(x)
% x1: Angular position (theta)
y = x(1);
end
% Copyright 2018 The MathWorks, Inc.

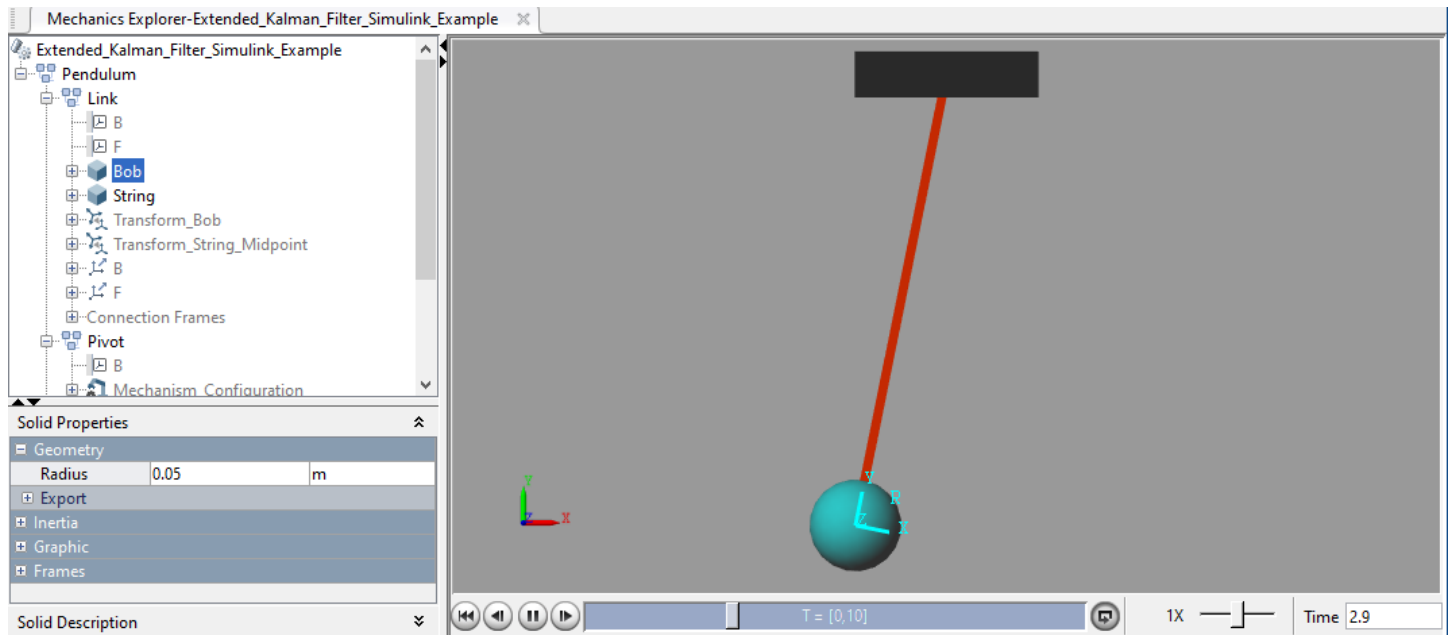
function x = myStateTransitionFcn(x,u)
% Sample time [s]
dt = 0.01;

% Using Euler discretization, next states
% can be calculated given the current
% states and input
x = x + [x(2); -9.81/0.5*sin(x(1)) + u]*dt;
end
```

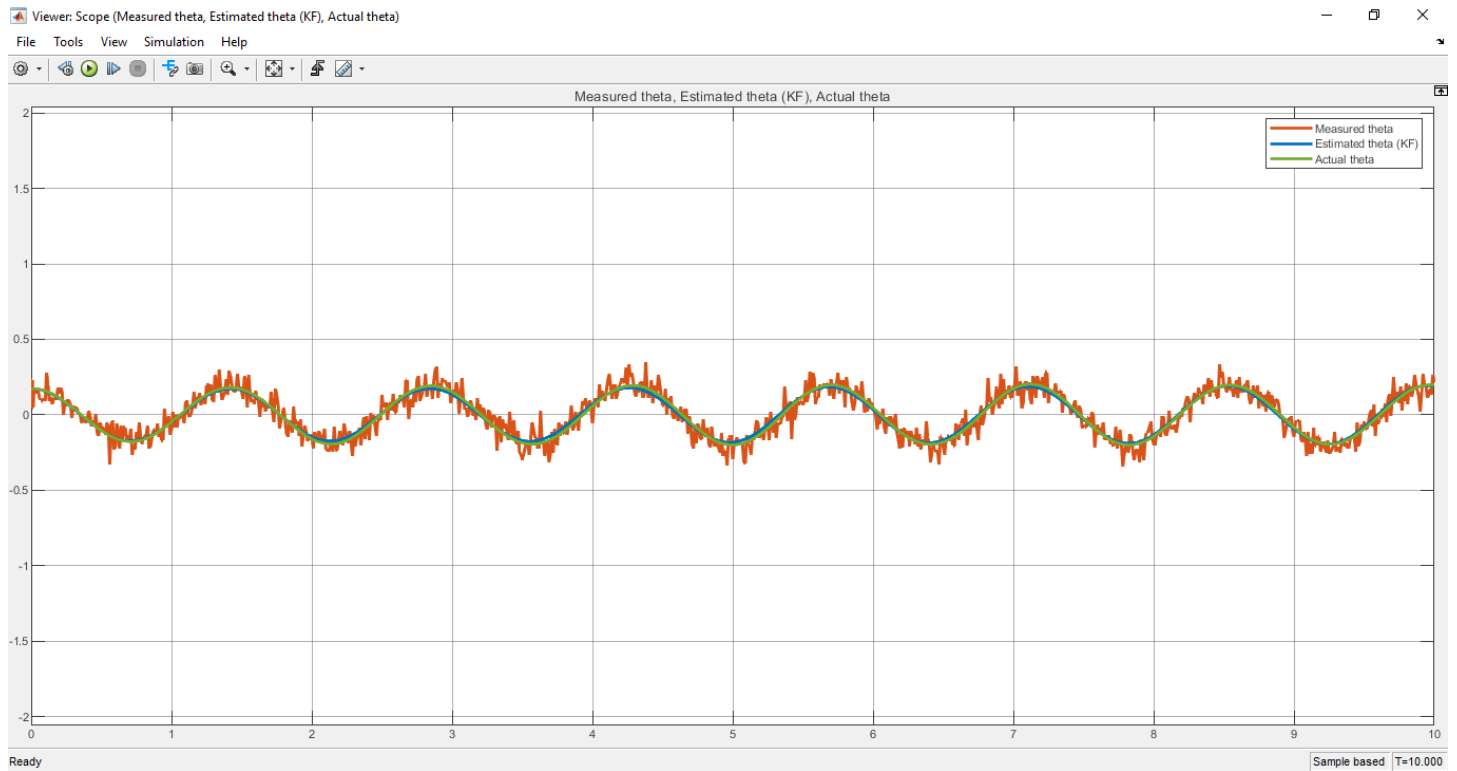
Simulnik blok:



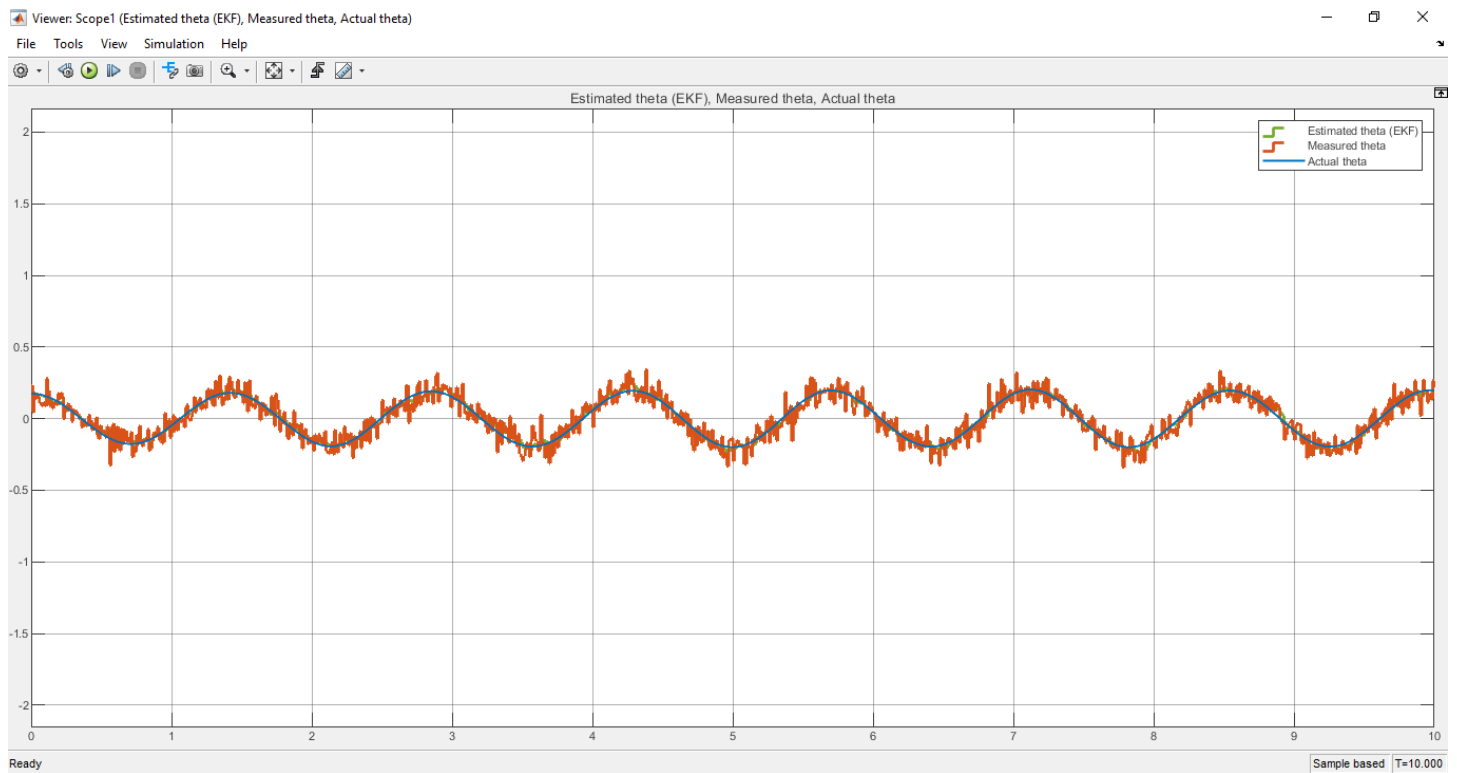
Simulacija:



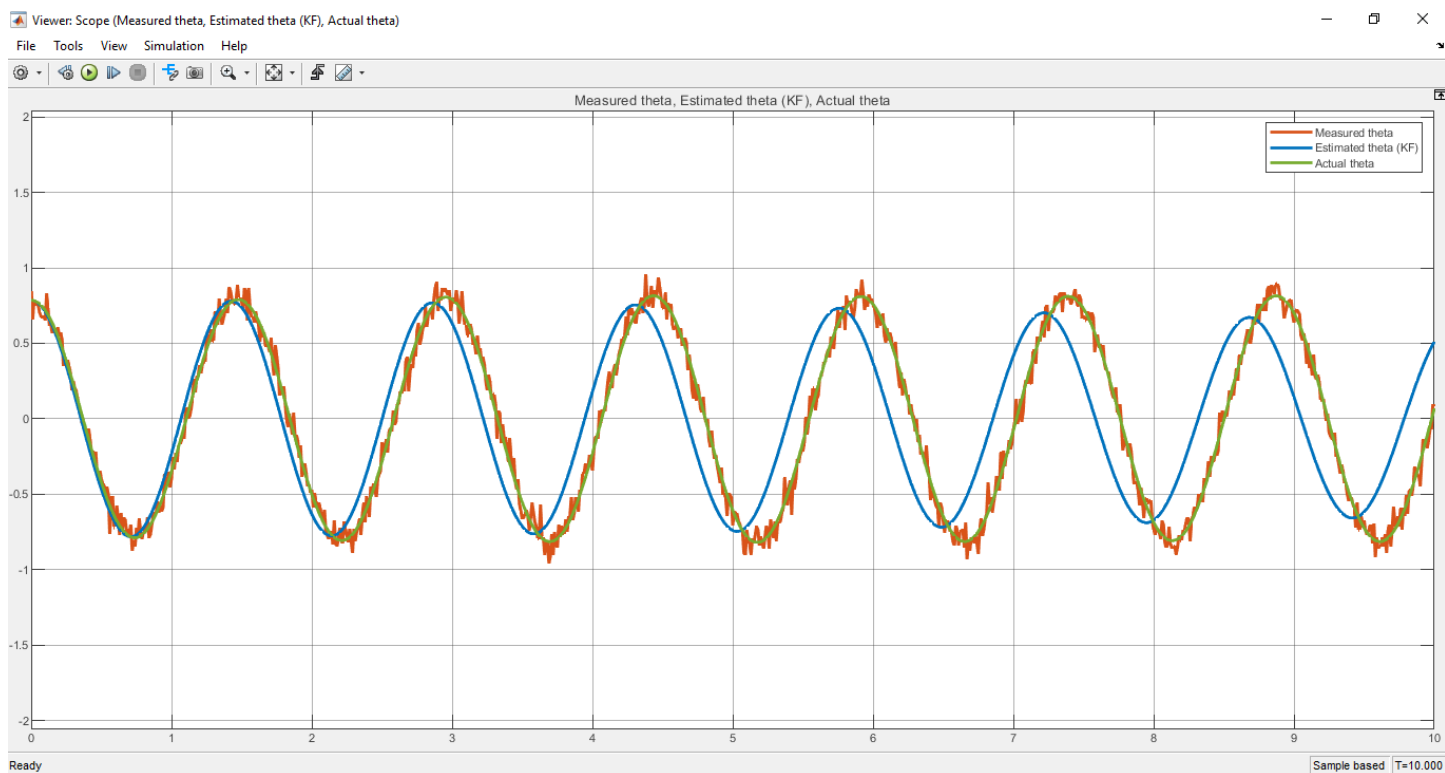
Posmatranja:



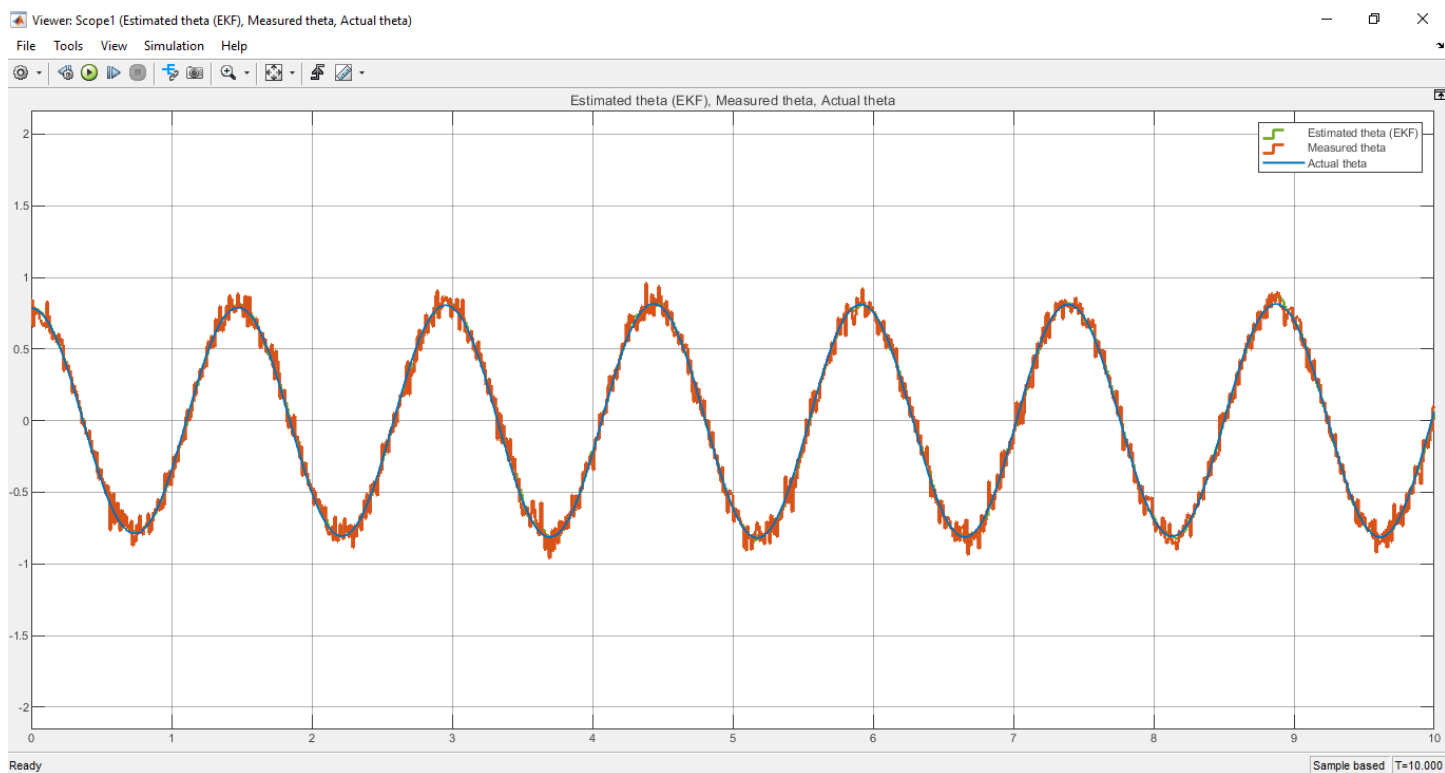
KF za uglove $\pi/18$ i 0



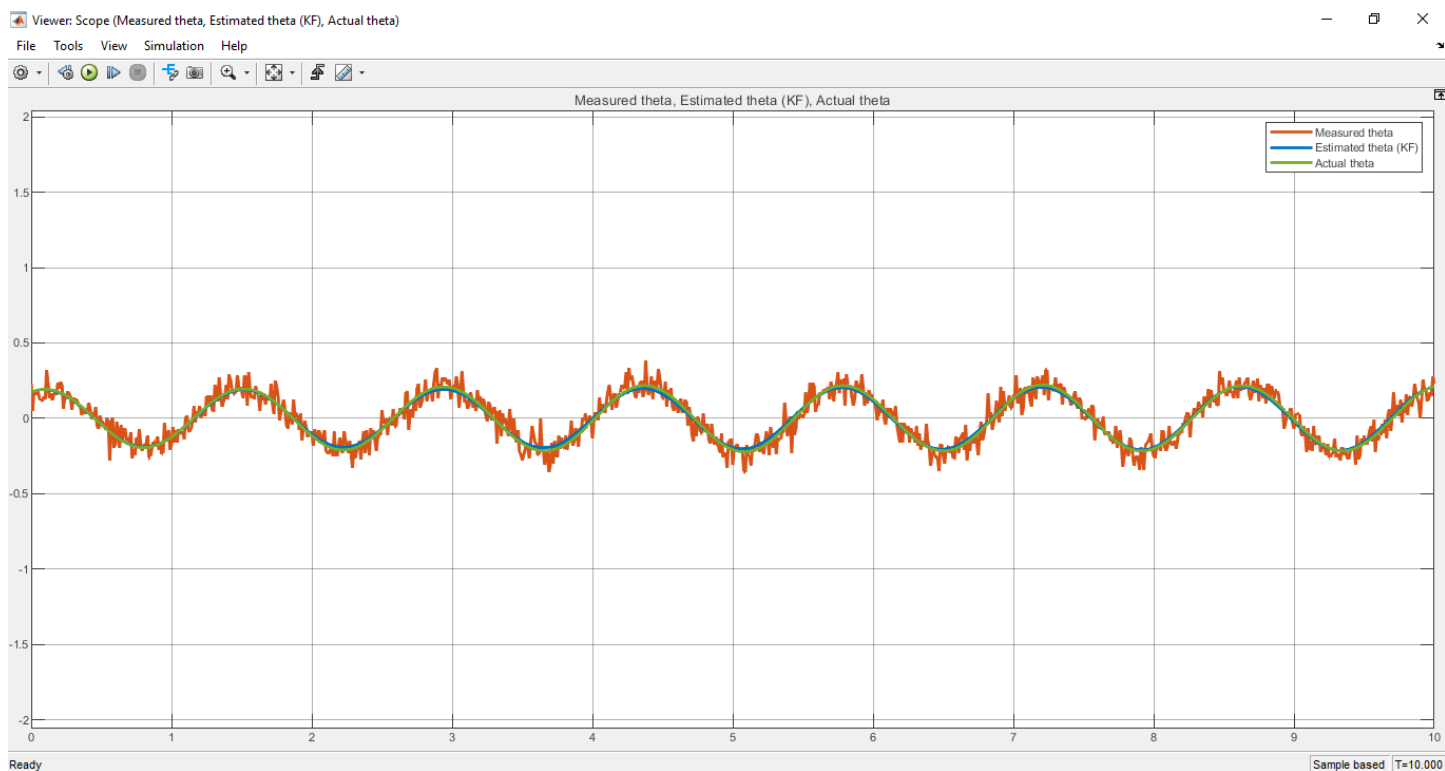
EKF za uglove $\pi/18$ i 0



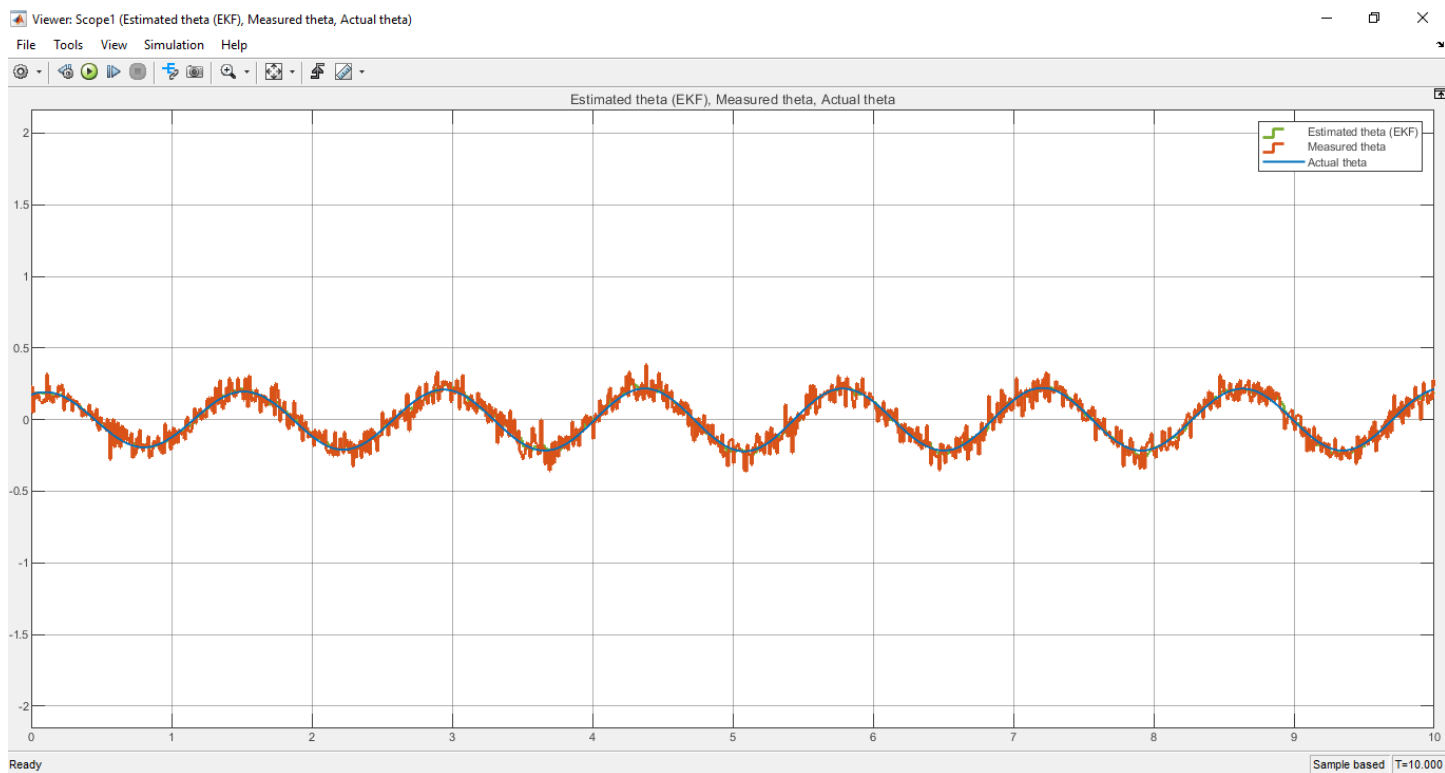
KF za uglove $\pi/4$ i 0



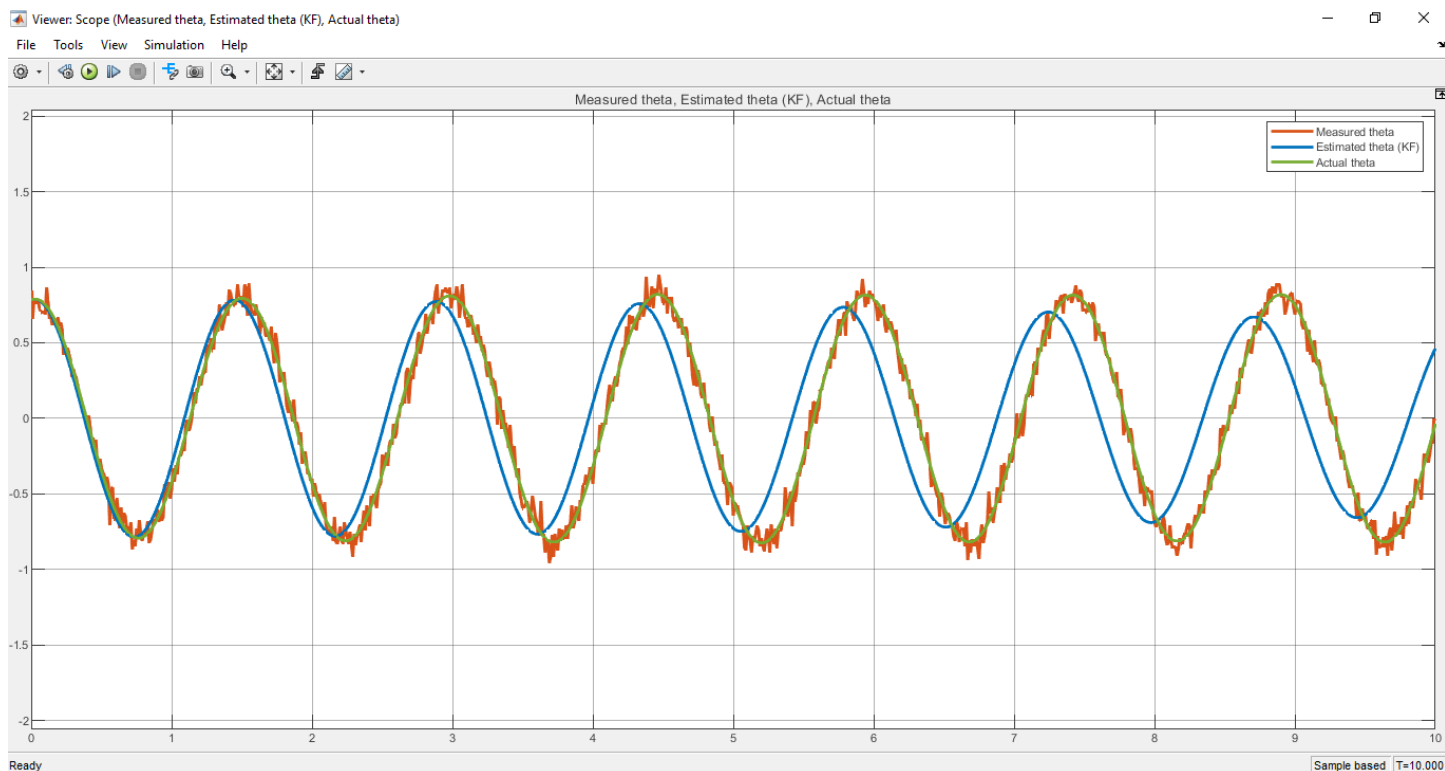
EKF za uglove $\pi/4$ i 0



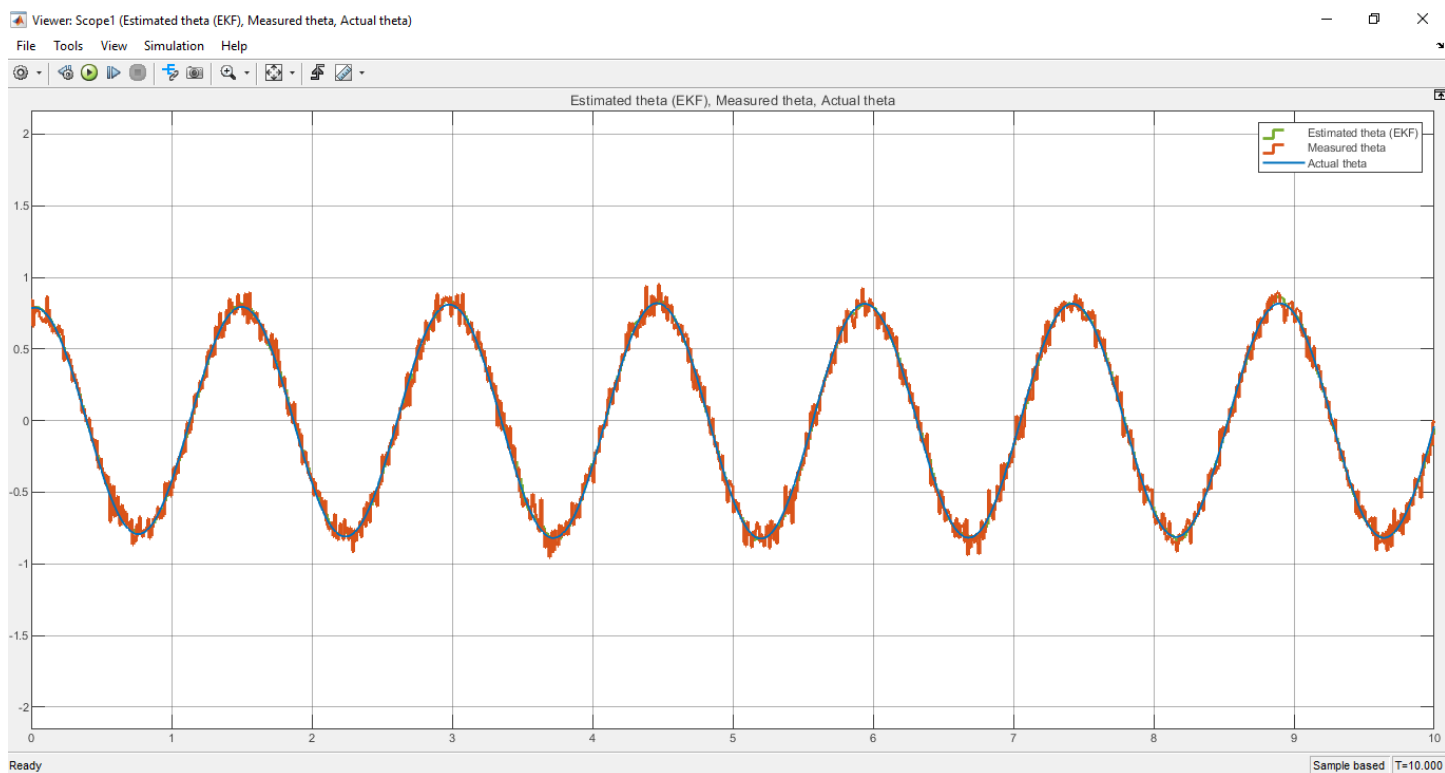
KF za uglove $\pi/18$ i $\pi/9$



EKF za uglove $\pi/18$ i $\pi/9$



KF za uglove $\pi/4$ i $\pi/9$



EKF za uglove $\pi/4$ i $\pi/9$

Komentar:

U drugom zadatku koristili smo matematičko klatno kao objekt koji se posmatra. Ugaona pozicija klatna se estimira preko KF i EKF, koji su dodani u formi simulnik bloka. U prvom slučaju smo posmatrali vrlo male uglove, $\pi/18$ i 0. Oba filtera su dali dobre rezultate, razlog tome je mali ugao pa se sistem ponašao linearno. Za male uglove KF i EKF daju jednako dobre rezultate pa je u ovom slučaju dovoljan je samo KF. U slučajevima sa većim uglovima EKF daje bolje rezultate a kod KF možemo jasno uočiti kašnjenje i grešku.