

Project Στοχαστικού Ελέγχου και Θεωρίας Εκτίμησης 2017-2018

ΟΝΟΜΑΤΕΠΩΝΥΜΟ : Λάμπης Παπακώστας

ΑΜ : 228467

ΤΟΜΕΑΣ : ΣΑΕ

ΕΤΟΣ : 5ο

ΕΠΙΛΟΓΗ ΕΡΓΑΣΙΑΣ : Έργο 3

ΠΕΡΙΕΧΟΜΕΝΑ

1. ΥΛΟΠΟΙΗΣΗ Α ΜΕΡΟΥΣ.....σελ. 2	
2. ΥΛΟΠΟΙΗΣΗ Β ΜΕΡΟΥΣ.....σελ. 19	
3. ΒΙΒΛΙΟΓΡΑΦΙΑ.....σελ. 31	

ΜΕΡΟΣ Α' :

1) Οι πόλοι που επιλέγουμε είναι: -500, -200, -100, -2, -1 ενώ το μηδενικό είναι το 20. Βλέπουμε ότι όλοι οι πόλοι είναι ευσταθείς ενώ οι δύο τελευταίοι είναι απομακρυσμένοι από τους υπόλοιπους κατά δύο τάξεις μεγέθους ενώ, καθώς βρίσκονται πιο δεξιά στο s-plane, η απόκρισή τους είναι αρκετά πιο αργή από τους υπόλοιπους.

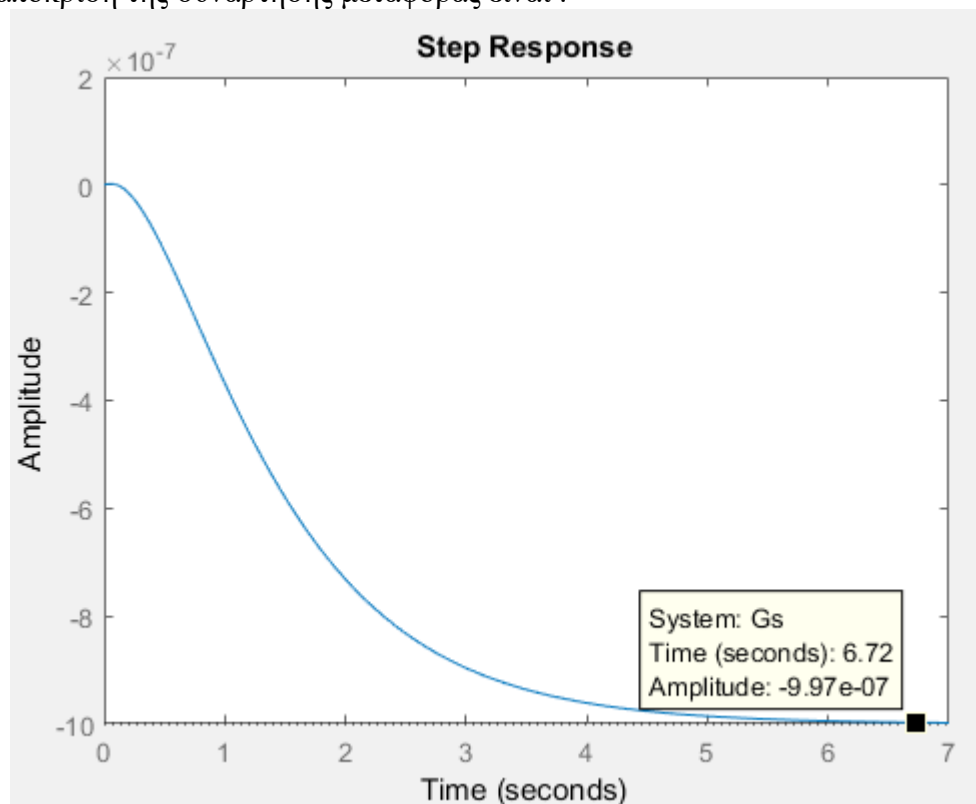
2) Τοποθετούμε το σύστημα με συμβολική γραφή στην MATLAB και κάνοντας τις παρακάτω εντολές :

```
% Question 1
syms s;
%Εύρεση της εξίσωσης της συνάρτησης μεταφοράς
S = (s-20) / ((s+500)*(s+200)*(s+100)*(s+2)*(s+1));
S = expand(S);
S = simplify(S);
pretty(S);
```

καταλήγουμε ότι η συνάρτηση μεταφοράς του συστήματος είναι :

$$\frac{s - 20}{s^5 + 803s^4 + 172402s^3 + 10511600s^2 + 30340000s + 20000000}$$

Η βηματική απόκριση της συνάρτησης μεταφοράς είναι :



Παρατηρούμε ότι το σύστημα είναι ευσταθές καθώς συγκλίνει σε μια τελική τιμή. Θέλουμε το σύστημα να έρθει σε μορφή Observable Canonical Form[1] ώστε να εγγυηθούμε την ιδιότητα της παρατηρησιμότητας. Ένα σύστημα είναι παρατηρήσιμο όταν ο πίνακας παρατηρησιμότητας έχει ίδια τάξη(rank) με τον πίνακα A του συστήματος. Εάν ένα σύστημα είναι παρατηρήσιμο, τότε όλες οι καταστάσεις του μπορούν να “μετρηθούν”, κάτι που θα μας βοηθήσει παρακάτω.

Ένα σύστημα της παρακάτω μορφής είναι Observable Canonical :

$$\mathbf{G}(s) = \frac{n_1 s^3 + n_2 s^2 + n_3 s + n_4}{s^4 + d_1 s^3 + d_2 s^2 + d_3 s + d_4}.$$

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 0 & 0 & -d_4 \\ 1 & 0 & 0 & -d_3 \\ 0 & 1 & 0 & -d_2 \\ 0 & 0 & 1 & -d_1 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} n_4 \\ n_3 \\ n_2 \\ n_1 \end{bmatrix} \mathbf{u}(t)$$

$$\mathbf{y}(t) = [0 \ 0 \ 0 \ 1] \mathbf{x}(t).$$

Σύμφωνα με την συνάρτηση μεταφοράς του συστήματος μας, ο χώρος κατάστασης είναι:

$$\dot{x} = \begin{bmatrix} 0 & 0 & 0 & 0 & -20000000 \\ 1 & 0 & 0 & 0 & -30340000 \\ 0 & 1 & 0 & 0 & -10511600 \\ 0 & 0 & 1 & 0 & -172402 \\ 0 & 0 & 0 & 1 & -803 \end{bmatrix} x + \begin{bmatrix} -20 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} u$$

$$y = [0 \ 0 \ 0 \ 0 \ 1] x$$

Ο κώδικας για την δημιουργία του χώρου κατάστασης είναι :

```
% Δημιουργία Observable Cannonical Form
% συστήματος ώστε να είναι παρατηρήσιμο το σύστημα
n = [0 0 0 1 -20];
d = [803 172402 10511600 30340000 20000000];
A = [0 0 0 0 -d(1,5);
     1 0 0 0 -d(1,4);
     0 1 0 0 -d(1,3);
     0 0 1 0 -d(1,2);
     0 0 0 1 -d(1,1)];
B = [n(1,5); n(1,4); n(1,3); n(1,2); n(1,1)];
C = [0 0 0 0 1];
D = 0;
sys = ss(A,B,C,D);
```

Επειδή θα χρειαστεί στα επόμενα ερωτήματα να βρούμε ποια κατάσταση αντιστοιχεί σε ποιον πόλο , θα μετασχηματίσουμε τον χώρο κατάστασης σε μορφή Jordan Canonical Form[2]. Οι μετατροπές που κάνουμε είναι :

$$\begin{aligned}A_J &= P^{-1}AP \\B_J &= P^{-1}B \\C_J &= CP \\D_J &= D\end{aligned}$$

όπου ο πίνακας P είναι ο πίνακας των ιδιοδιανυσμάτων του συστήματος.

Το σύστημα , στην τελική του μορφή , είναι :

$$\begin{aligned}\dot{x} &= \begin{bmatrix} -500 & 0 & 0 & 0 & 0 \\ 0 & -200 & 0 & 0 & 0 \\ 0 & 0 & -100 & 0 & 0 \\ 0 & 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix} x + \begin{bmatrix} -1.744e-8 \\ 1.861e-7 \\ -3.092e-7 \\ 2.277e-6 \\ -2.136e-6 \end{bmatrix} u \\ y &= [1 \ 1 \ 1 \ 1 \ 1] x\end{aligned}$$

Πρέπει να τονίσουμε ότι η απόκριση του συστήματος μεταξύ των δύο μορφών είναι η ίδια , απλά αλλάζει ο τρόπος που υπολογίζονται οι καταστάσεις.

Ο κώδικας για την παραγωγή του συστήματος είναι :

```
% Μετατροπή σε Jordan Cannonical Form;
[V,J] = jordan(A);
Bj = inv(V)*B;
Cj = C*V;
J = diag([-500 -200 -100 -2 -1]);
BJ = [Bj(2,1) ; Bj(3,1) ; Bj(1,1) ; Bj(4,1) ; Bj(5,1)];
sys = ss(J,BJ,Cj,D);
```

3- 4) Αφού δημιουργήσουμε το σύστημα, πρέπει να προσθέσουμε το διάνυσμα των διαταραχών σε 3 καταστάσεις συγκεκριμένα τις x1, x4 και x5. Ορίζουμε τον πίνακα διαταραχών $G = [1 \ 0 \ 0 \ 1 \ 1]^T$ και το σύστημα γίνεται:

$$\dot{x} = Ax + Bu + Gw$$

όπου w είναι μια τυχαία διαδικασία που μοντελοποιεί λευκό Gaussian θόρυβο με μηδενική μέση τιμή και συμμεταβλητότητα ίση με τον πίνακα Q μεγέθους 5x5.

Ορίζουμε το μοντέλο μετρήσεων για 3 καταστάσεις του συστήματος :

$$z = Hx + Wv$$

όπου z ο πίνακας των μετρήσεων μεγέθους 3×1 και v είναι μια τυχαία διαδικασία που μοντελοποιεί λευκό Gaussian θόρυβο με μηδενική μέση τιμή και συμμεταβλητότητα ίση με τον πίνακα R μεγέθους 3×3 .

Ο πίνακας H ονομάζεται πίνακας ευαισθησίας μετρήσεων και έχει μέγεθος 3×5 . Ορίζουμε τον πίνακα ίσο με :

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

καθώς θέλουμε να μετρήσουμε τις καταστάσεις x_1 , x_3 και x_5 .

Επίσης, ορίζουμε τον πίνακα μετρήσεων του θορύβου W ίσο με $[1; 1; 1]$ καθώς θέλουμε να προσθέσουμε θόρυβο και στις 3 καταστάσεις.

Επειδή πρέπει να διακριτοποιήσουμε το σύστημα μαζί με τον θόρυβο και τις διαταραχές και να υπολογίσουμε τους διακριτούς πίνακες μέσω MATLAB, θα δημιουργήσουμε ένα καινούργιο χώρο κατάστασης το οποίο θα έχει ως εισόδους την είσοδο του κανονικού συστήματος, την διαταραχή και τους θορύβους των μετρήσεων και ως εξόδους την έξοδο του κανονικού συστήματος και τις μετρήσεις των καταστάσεων x_1 , x_3 και x_5 . Εν ολίγοις, ο νέος χώρος κατάστασης θα είναι της μορφής :

$$\begin{aligned} \dot{x}_{5 \times 1} &= A_{5 \times 5} x_{5 \times 1} + \begin{bmatrix} B_{5 \times 1} & G_{5 \times 1} & 0_{5 \times 1} \end{bmatrix} \begin{bmatrix} u \\ w \\ v \end{bmatrix}_{3 \times 1} \\ \begin{bmatrix} y_{1 \times 1} \\ z_{3 \times 1} \end{bmatrix}_{4 \times 1} &= \begin{bmatrix} C_{1 \times 5} \\ H_{3 \times 5} \end{bmatrix}_{4 \times 5} x_{5 \times 1} + \begin{bmatrix} D_{1 \times 1} & 0_{1 \times 2} \\ 0_{3 \times 2} & W_{3 \times 1} \end{bmatrix}_{4 \times 3} \begin{bmatrix} u \\ w \\ v \end{bmatrix}_{3 \times 1} \end{aligned}$$

Ο νέος χώρος κατάστασης τελικά είναι :(αλλαγή)

$$\begin{aligned} \dot{x} &= \begin{bmatrix} -500 & 0 & 0 & 0 & 0 \\ 0 & -200 & 0 & 0 & 0 \\ 0 & 0 & -100 & 0 & 0 \\ 0 & 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix} x + \begin{bmatrix} -1.744e-8 & 1 & 0 \\ 1.861e-7 & 0 & 0 \\ -3.092e-7 & 0 & 0 \\ 2.277e-6 & 1 & 0 \\ -2.136e-6 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ w \\ v \end{bmatrix} \\ \begin{bmatrix} y \\ z \end{bmatrix} &= \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} x + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ w \\ v \end{bmatrix} \end{aligned}$$

Η διαδικασία έγινε με τον παραπάνω κώδικα :

```
% Πίνακας Θορυβών
G = [1 ; 0 ; 0 ; 1 ; 1];
W = [1 ; 1 ; 1];
% Πίνακας ευαισθησίας μετρήσεων
H = [ 1 0 0 0 0 ;
      0 0 1 0 0 ;
      0 0 0 0 1];
% Νέος χώρος κατάστασης
sysA = ss(J,[Bj G 0*Bj] , [Cj ; H] , [D zeros(1,2); zeros(3,2) W]);
```

Για να διακριτοποιήσουμε το σύστημα , θα πρέπει να βρούμε την κατάλληλη περίοδο δειγματοληψίας ώστε να διατηρήσει το σύστημα τις ιδιότητες της ελέγξιμότητας και ειδικά της παρατηρησιμότητας. Σύμφωνα με το [3] , πρέπει να ισχύει η παρακάτω σχέση :

$$\omega_s \geq 35\omega_{BW}$$

όπου ω_s είναι η συχνότητα δειγματοληψίας και ω_{BW} το εύρος ζώνης του συστήματος , δηλαδή η συχνότητα εκείνη όπου έχει πλάτος -3dB στο διάγραμμα Bode.

Για να βρούμε το Bandwidth του συστήματος πρέπει να το εκφράσουμε υπό την μορφή συναρτήσεων μεταφοράς και ύστερα με την εντολή bandwidth() βρίσκουμε τις επιθυμητές ποσότητες και επιλέγουμε την μικρότερη. Το σύστημα έχει 12 συναρτήσεις μεταφοράς και εμείς θα επιλέξουμε το μικρότερο bandwidth.

Ο πίνακας με τα bandwidth είναι :

BW				
3x4 double				
	1	2	3	4
1	0.8391	498.8142	99.7628	0.9976
2	1.1821	498.8142	Inf	0.9976
3	Inf	Inf	Inf	Inf

όπου επιλέγουμε τα 0.8391 rad/s.

Η μέγιστη περίοδος δειγματοληψίας υπολογίζεται τότε :

$$T_{smax} = 0.2139$$

Ο κώδικας για την παραπάνω διαδικασία είναι :

```
% Σχεδιασμός συναρτήσεων μεταφοράς
[n1,d1] = ss2tf(sysA.A,sysA.B,sysA.C,sysA.D,1);
[n2,d2] = ss2tf(sysA.A,sysA.B,sysA.C,sysA.D,2);
[n3,d3] = ss2tf(sysA.A,sysA.B,sysA.C,sysA.D,3);
tf11 = tf(n1(1,:),d1); tf12 = tf(n1(2,:),d1);
tf13 = tf(n1(3,:),d1); tf14 = tf(n1(4,:),d1);
tf21 = tf(n2(1,:),d2); tf22 = tf(n2(2,:),d2);
tf23 = tf(n2(3,:),d2); tf24 = tf(n2(4,:),d2);
tf31 = tf(n3(1,:),d3); tf32 = tf(n3(2,:),d3);
tf33 = tf(n3(3,:),d3); tf34 = tf(n3(4,:),d3);
% Εύρεση Bandwidth των δυο συναρτήσεων μεταφοράς
BW = [bandwidth(tf11) bandwidth(tf12) bandwidth(tf13) bandwidth(tf14);
      bandwidth(tf21) bandwidth(tf22) bandwidth(tf23) bandwidth(tf24);
      bandwidth(tf31) bandwidth(tf32) bandwidth(tf33) bandwidth(tf34) ];
% Επιλογή ελαχίστου Bandwidth
wbw = 0.8391;
% Μεγιστη περίοδος δειγματοληψίας
Tsmax = (2*pi)/(35*wbw);
```

Επιλέγοντας περίοδο δειγματοληψίας $T_s = 0.1$, το διακριτό σύστημα είναι :

$$x[n+1] = \begin{bmatrix} 1.929e-22 & 0 & 0 & 0 & 0 \\ 0 & 2.061e-09 & 0 & 0 & 0 \\ 0 & 0 & 4.54e-05 & 0 & 0 \\ 0 & 0 & 0 & 0.8187 & 0 \\ 0 & 0 & 0 & 0 & 0.9048 \end{bmatrix} x[n] + \begin{bmatrix} -6.184e-10 & 0.002 & 0 \\ -8.719e-11 & 0 & 0 \\ 1.861e-09 & 0 & 0 \\ 2.063e-07 & 0.09063 & 0 \\ -2.033e-07 & 0.09516 & 0 \end{bmatrix} \begin{bmatrix} u[n] \\ w[n] \\ v[n] \end{bmatrix}$$

$$\begin{bmatrix} y[n] \\ z[n] \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} x[n] + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u[n] \\ w[n] \\ v[n] \end{bmatrix}$$

Πρέπει να τονίσουμε ότι η ιδιότητα της παρατηρητικότητας παραμένει ίδια :

The discrete system is Observable

όπως και η ευστάθεια του συστήματός καθώς όλες οι ιδιοτιμές έχουν πραγματικό μέρος μικρότερο της μονάδας :

```
>> eig(sysAd)

ans =
|
0.0000
0.0000
0.0000
0.8187
0.9048
```

Ο κώδικας για την παραπάνω διαδικασία είναι :

```
% Επιλογή περιόδου δειγματοληψίας
Ts = 0.1;
% Διακριτοποίηση συστήματος
sysAd = c2d(sysA , Ts , 'zoh');
% Παρατηρησιμότητα
if rank(observ(sysAd.A,sysAd.C)) == 5
    fprintf('The discrete system is Observable \n');
end
% Ιδιοτιμές συστήματος
eig(sysAd);
```

(Το e-8 σημαίνει 10^{-8} κλπ).

5) Καλούμαστε να επιλέξουμε τις παραμέτρους ώστε να υλοποιήσουμε αργότερα το διακριτό φίλτρο Kalman. Οι πίνακες που εκφράζουν τις συμμεταβλητότητες των θορύβων , Q και R , θα πρέπει να είναι συμμετρικοί , τετραγωνικοί και θετικώς ορισμένοι. Επιλέγουμε τους πίνακες ως εξής :

$$Q = \begin{bmatrix} 2.5 & 0 & 0 & 0 & 0 \\ 0 & 2.5 & 0 & 0 & 0 \\ 0 & 0 & 2.5 & 0 & 0 \\ 0 & 0 & 0 & 2.5 & 0 \\ 0 & 0 & 0 & 0 & 2.5 \end{bmatrix} \quad \text{και} \quad R = \begin{bmatrix} 1.3 & 0 & 0 \\ 0 & 1.3 & 0 \\ 0 & 0 & 1.3 \end{bmatrix}$$

Το αρχικό διάνυσμα κατάστασης είναι $x = [0 \ 0 \ 0 \ 0 \ 0]^T$ και ο αρχικός πίνακας συμμεταβλητότητας σφάλματος P , μεγέθους 5x5 , επιλέγεται ίσος με τον μοναδιαίο.

Ο κώδικας που ορίζουμε τα παραπάνω μεγέθη είναι :

```
% Συνδιασπορές θορύβων
q = 2.5; r = 1.3;
% Πίνακες συνδιασπορών
Q = q*eye(5); R = r*eye(3);
% Αρχικοποίηση παραμέτρων
x0 = [0 ; 0 ; 0 ; 0 ; 0];
P0 = eye(5);
```

Επίσης , για να υλοποιήσουμε το φίλτρο Kalman , θα πρέπει να έχουμε διαθέσιμο το διάνυσμα των μετρήσεων z , άρα θα πρέπει να προσομοιώσουμε την λειτουργία του σήματος. Αρχικά , ορίζουμε την είσοδο u ως μια ημιτονοειδής συνάρτηση ως προς τον χρόνο. Για τους λευκούς θορύβους , δεν χρησιμοποιούμε την εντολή της MATLAB wgn καθώς η μέση τιμή της δεν προσεγγίζει την μηδενική σε σχέση με τις παρακάτω συναρτήσεις :(αλλαγή)


```

% Χρονικό διάστημα εκτέλεσης της προσομοίωσης
% του συστήματος
t = (0:Ts:50);
t = t';
% Είσοδος του συστήματος
u = sin(2*t);
% Οι διαταραχές και ο θόρυβος των μετρήσεων
w = sqrt(q)*randn(length(t),1);
v = sqrt(r)*randn(length(t),1);
% Προσομοίωση του συστήματος
[out,~,states] = lsim(sysAd,[u w v],t,x0);
out = out' ;

```

Πολλαπλασιάζουμε επίσης τις συναρτήσεις με την ρίζα της τιμής της συµμεταβλητότητας ώστε να εκφράζουν Gaussian µεταβλητή.

Τέλος , με την εντολή lsim , προσομοιώνουμε το σύστημα για 50 δευτερόλεπτα. Αποθηκεύουμε επίσης τις εξόδους του συστήματος (πίνακας out) όπως και τις καταστάσεις για µελλοντική χρήση (πίνακας states). Πρέπει να τονίσουμε ότι οι 3 τελευταίες γραµµές του πίνακα out είναι τα διανύσµατα µετρήσεων z.

Ο κώδικας για την παραπάνω διαδικασία είναι :

```

% Προσομοίωση του συστήματος
[out,~,states] = lsim(sysAd,[u w v],t,x0);
out = out' ;
% Διάνυσμα μετρήσεων
z = [ out(2,:) ; out(3,:) ; out(4,:) ];

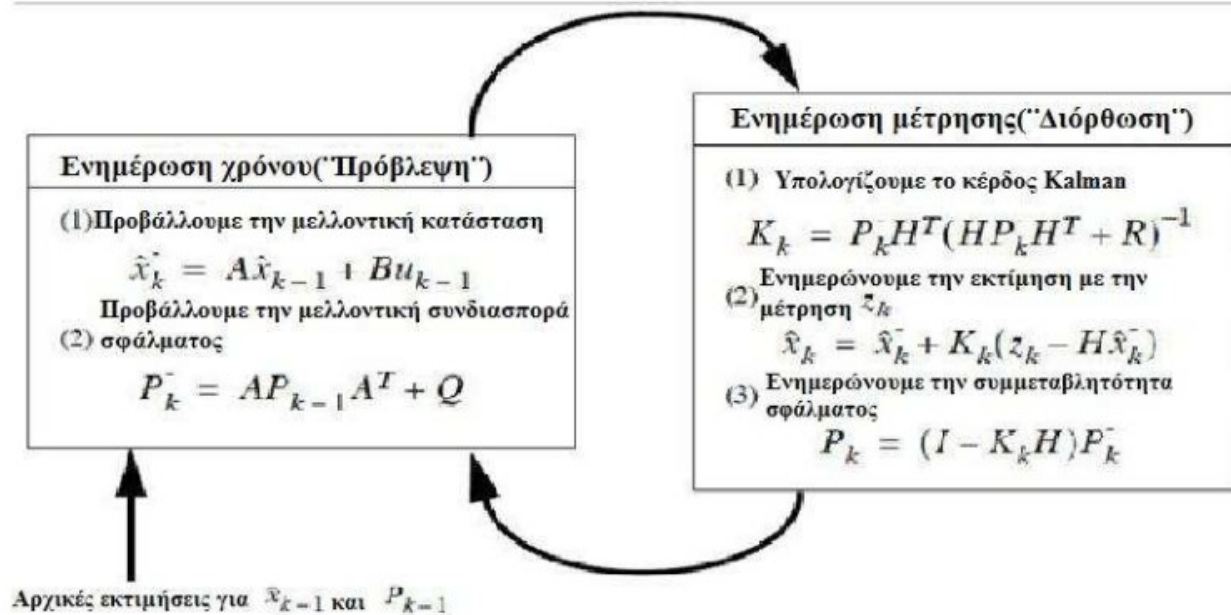
```

6) Ο αλγόριθμος του διακριτού φίλτρου Kalman περιγράφεται ως εξής :

1. Προσδιορισμός αρχικού διανύσματος καταστάσεων και αρχικού πίνακα συµμεταβλητότητας σφάλματος
2. Στάδιο πρόβλεψης όπου προσδιορίζουμε την µελλοντική a priori κατάσταση και τον µελλοντικό a priori πίνακα συµμεταβλητότητας σφαλμάτων
3. Στάδιο μέτρησης όπου i) υπολογίζουμε το κέρδος Kalman , ii) ενημερώνουμε την a posteriori εκτίμηση της κατάστασης χρησιμοποιώντας το διάνυσμα μέτρησης και iii) ενημερώνουμε τον a posteriori πίνακα συµμεταβλητότητας σφάλματος

Με τον όρο a priori αναφερόμαστε στο διάνυσμα της σάρωσης κ , το οποίο γνωρίζει την εκάστοτε διαδικασία μέχρι εκείνη την σάρωση ενώ με τον όρο a posteriori εκείνο το διάνυσμα που γνωρίζει και την μέτρηση της σάρωσης κ.

Η διαδικασία φαίνεται με το παρακάτω διάγραμμα :



Ο κώδικας που υλοποιεί τον αλγόριθμο φαίνεται παρακάτω :

```

% Question 6
% Δέσμευση αποθηκευτικού χώρου για μεγέθη που
% θέλουμε να υπολογίσουμε
innovationA = zeros(3,length(t));
x_e = zeros(5,length(t));
% Πίνακας B του αρχικού συστήματος
Bd = sysAd.B(:,1);
% Αρχικές Εκτιμήσεις για την κατάσταση και για την
% συµμεταβλητότητα σφάλματος
x = x0;
P = P0;
% Υλοποίηση Αλγορίθμου του φίλτρου Kalman
for i=1:length(t)
    %Πρόβλεψη
    %x[k] (a priori) = Ax[k-1] + Bu[k-1]
    x = sysAd.A*x+Bd*u(i,:);
    % P[k] (a priori) = A*P[k-1]*A' + Q
    P = A*P*A'+Q;
    %Διόρθωση
    % K = P[k] (a priori)*H'/(H*P[k] (a priori)*H')
    K = P*H'/(H*P*H'+R);
    % Υπολογισμός καινοτομιών
    innovationA(:,i) = z(:,i) - H*x;
    % x[k] (a posteriori) = x[k] (a priori) + K*(z[k]-H*x[k] (a priori))
    x = x + K*innovationA(:,i);
    % P[k] (a posteriori) = (I-K*H)*P(a priori)
    P = (eye(5)-K*H)*P;
    % Διάνυσμα εκτίμησης των καταστάσεων
    x_e(:,i) = x;
end
  
```

Ο πίνακας κέρδους Kalman είναι :

```
K =  
  
    0.9775    0.0428    0.0000  
    1.3576    0.3032   -0.0007  
    0.0424    0.9193   -0.0000  
   -0.0055    0.0268    0.0045  
    0.0003   -0.0005    0.8260
```

Ο πίνακας συμεταβλητότητας σφάλματος P είναι :

```
P =  
  
    1.3276    1.8526    0.0821   -0.0067    0.0000  
    1.8750   11.3316    0.3899   -0.0178   -0.0009  
    0.0571    0.3914    1.1928    0.0347   -0.0000  
   -0.0066   -0.0142    0.0353    3.6836    0.0058  
    0.0004    0.0021   -0.0007    0.0049    1.0738
```

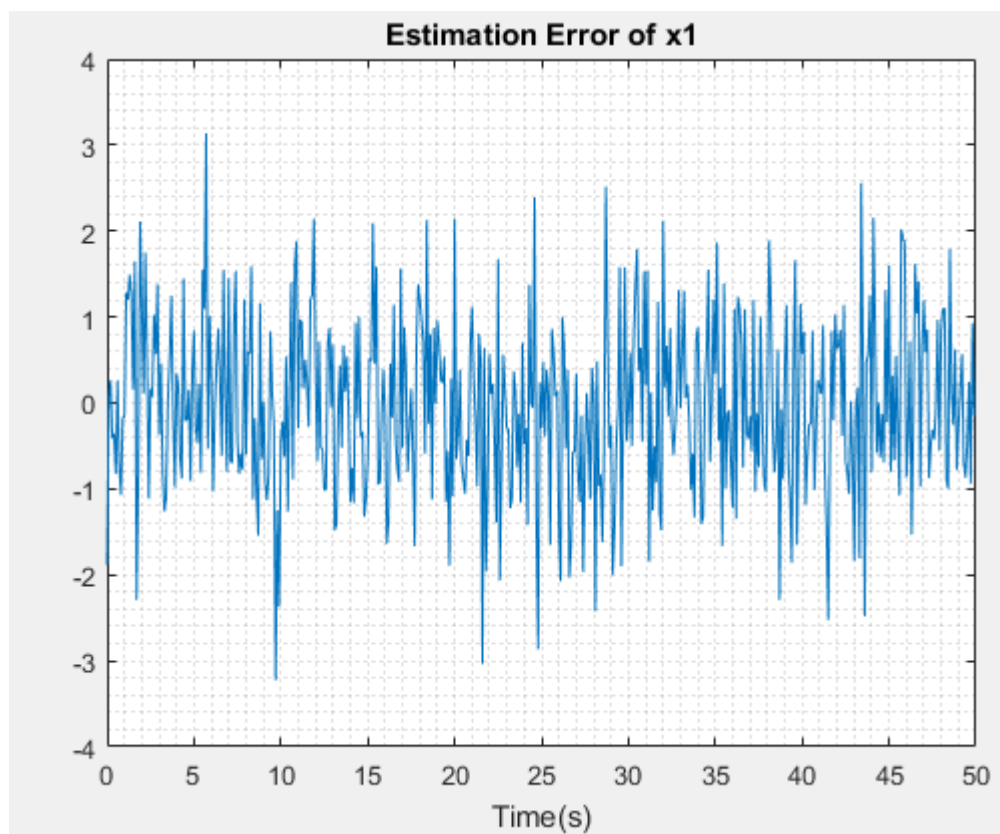
Για την εύρεση της απόδοσης των εκτιμήσεων θα υλοποιήσουμε τον παρακάτω κώδικα :

```
% Δέσμευση χώρου για μεγαλύτερη ταχύτητα  
% εκτέλεσης  
error = zeros(5,length(t));  
N=length(t);  
% Απόδοση της εκτίμησης του φίλτρου Kalman  
for j = 1:5  
    % Εύρεση σφάλματος μεταξύ των καταστάσεων  
    % και της εκτίμησης αυτών  
    error(j,:) = statesA(:,j) - (x_eA(j,:))';  
end
```

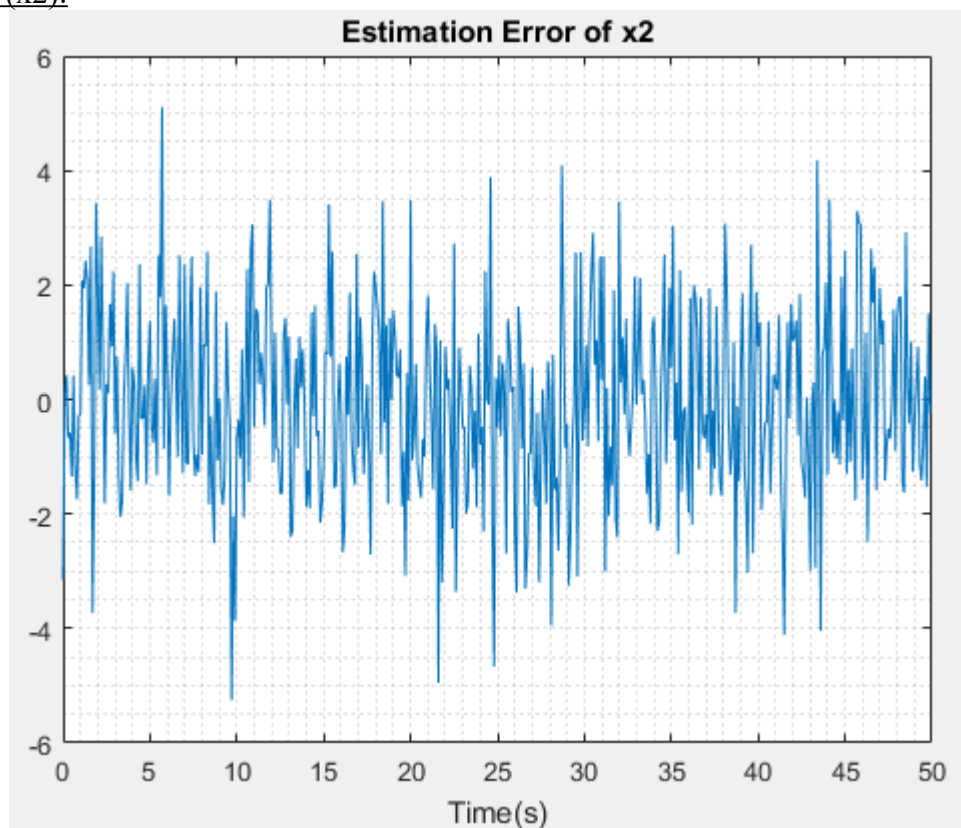
Με άλλα λόγια , βρίσκουμε την διαφορά μεταξύ κατάστασης και εκτιμώμενης κατάστασης , κανονικοποιούμε ως προς την εκάστοτε μέγιστη κατά απόλυτη τιμή ώστε να εκφράσουμε την ποσοστιαία διαφορά μεταξύ τους. Όσο πιο κοντά είναι η εκτίμηση με την κατάσταση , τόσο πιο μικρή είναι η ποσοστιαία διαφορά άρα το φίλτρο Kalman έχει εκτιμήσει αρκετά ικανοποιητικά την κατάσταση.

Παρατηρούμε ότι χρησιμοποιώντας μικρότερη περίοδο δειγματοληψίας , το φίλτρο Kalman έχει μεγαλύτερη απόδοση άρα και λιγότερα σφάλματα μεταξύ εκτιμήσεων και καταστάσεων.

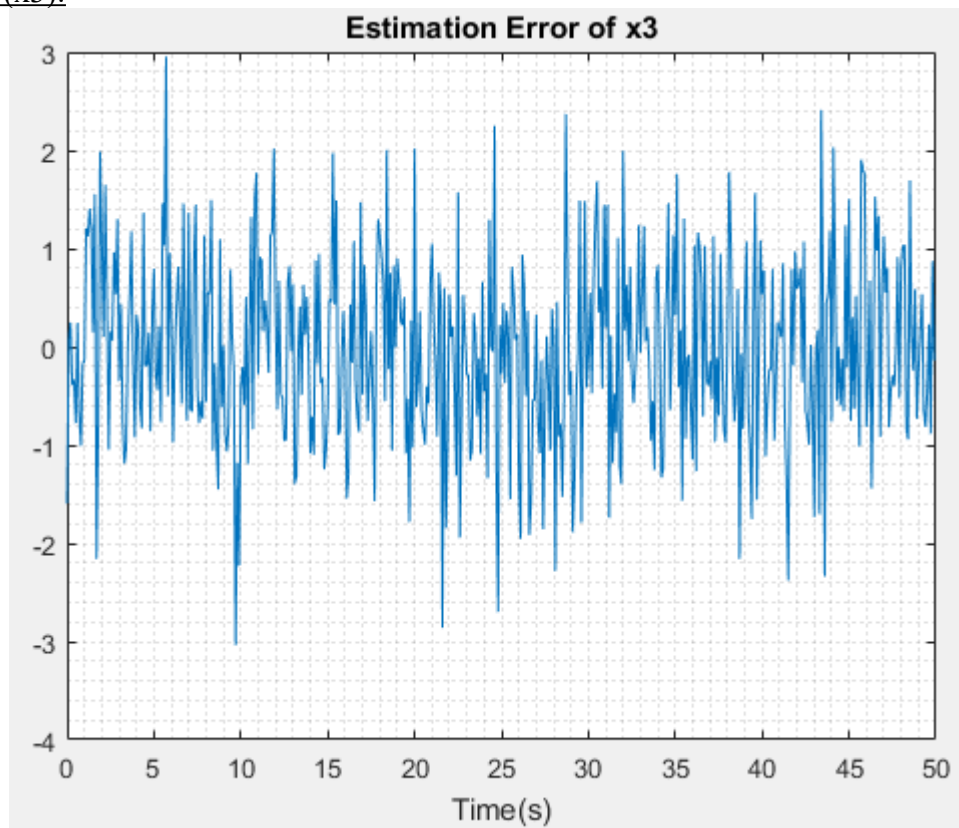
Οι γραφικές παραστάσεις των σφαλμάτων για όλες τις καταστάσεις φαίνονται παρακάτω :
Κατάσταση 1 (x_1):



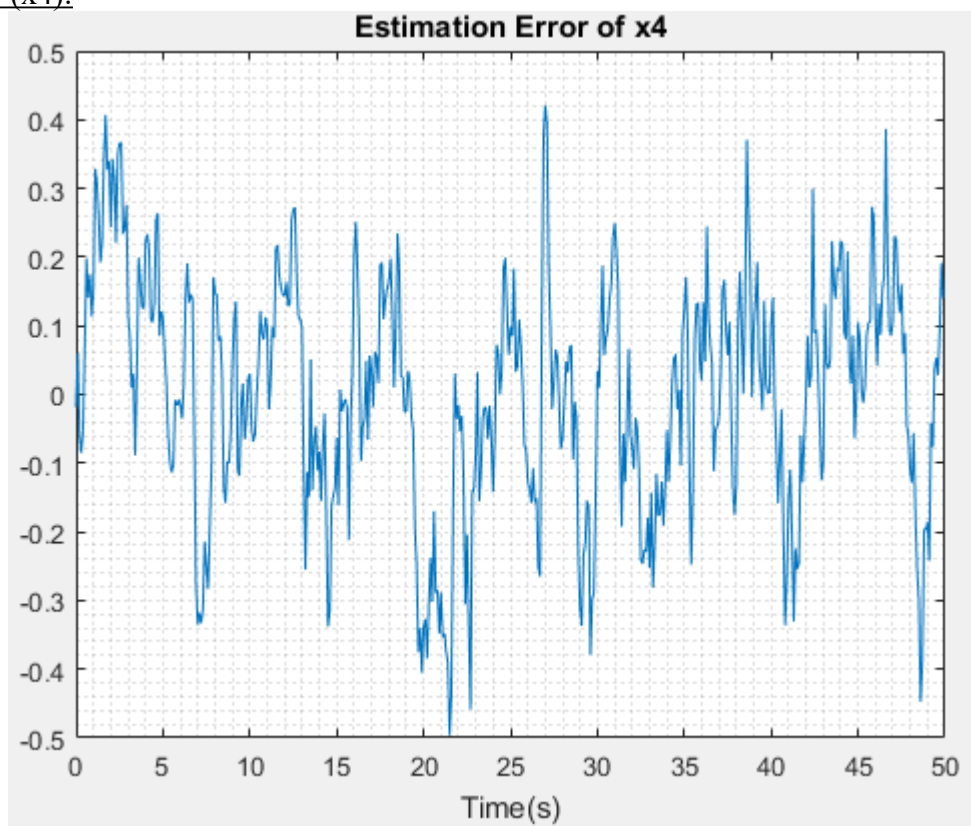
Κατάσταση 2 (x_2):



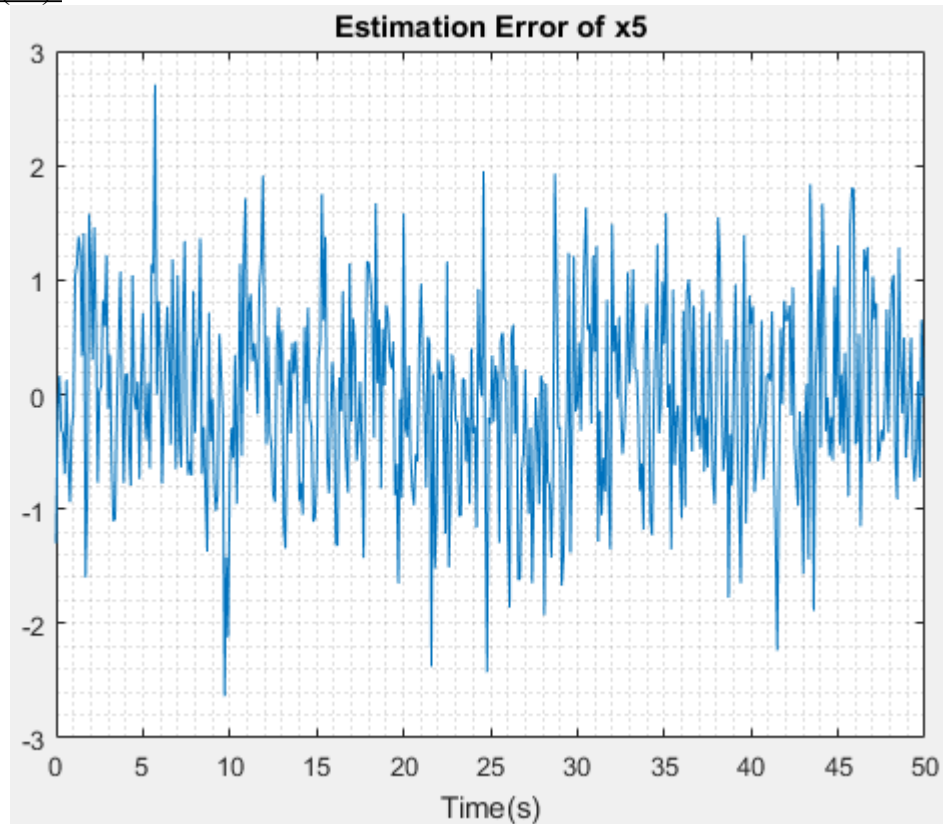
Κατάσταση 3 (x_3):



Κατάσταση 4 (x_4):



Κατάσταση 5 (x5):



Παρατηρούμε ότι για μεγάλες συµµεταβλητότητες θορύβου και διαταραχών το φίλτρο µπορεί να εκτιµήσει σε ικανοποιητικό βαθµό τις καταστάσεις , ειδικότερα αυτές που έχουµε µετρήσει. Για να υπολογίσουµε την απόδοση του φίλτρου , θα χρησιµοποιήσουµε το µέσο τετραγωνικό σφάλµα όπου εκφράζεται από τον τύπο:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N \|x_i - \hat{x}_i\|^2}$$

όπου N είναι ο αριθµός των δειγµάτων και οι κάθετες γραµµές εκφράζουν την ευκλείδεια νόρµα µεταξύ του διανύσµατος της πραγµατικής κατάστασης και της εκτίµησης αυτής. Για να υπολογίσουµε την απόδοση αφαιρούµε την παραπάνω ποσότητα µε το 1. Η απόδοση του φίλτρου είναι :

`performance =`

`0.7325`

Η παραπάνω διαδικασία έγινε με τον παρακάτω κώδικα : (αλλαγή)

```
% Μέσος όρος απόδοσης φίλτρου Kalman
% για κάθε κατάσταση
p = zeros(5,1);
for j = 1:5
    % Κανονικοποίηση ως προς την μέγιστη απόκλιση
    error(j,:) = error(j,:)/max(abs(error(j,:)));
    % Εύρεση νόρμας των διανυσμάτων σφάλματος
    p(j) = norm(error(j,:));
end
% Τύπος RMSE
performance = 1- sqrt(sum(p)/N)
```

Οι καινοτομίες ή innovation αναπαριστά την διαφορά ανάμεσα στην πρόβλεψη μέτρησης και την πραγματική μέτρηση , δηλαδή εισάγει στον εκτιμητή νέες πληροφορίες. Όταν η καινοτομία είναι ίσο με το μηδέν , τότε η πρόβλεψη συμπίπτει με την πραγματική μέτρηση. Ο τύπος που υπολογίζουμε την καινοτομία είναι :

$$v_k \stackrel{\text{def}}{=} z_k - H_k \hat{x}_{k(-)}.$$

όπου z είναι το διάνυσμα μετρήσεων και x το εκτιμώμενο a priori διάνυσμα καταστάσεων.

Οι καινοτομίες αποτελούν επίσης ένα μέτρο για το αν το φίλτρο Kalman έχει μοντελοποιηθεί κατάλληλα. Όταν οι καινοτομίες έχουν μηδενική μέση τιμή και παρουσιάζουν μια Gaussian συμπεριφορά , δηλαδή έχουν ιδιότητες λευκού θορύβου , τότε το φίλτρο Kalman έχει μοντελοποιηθεί κατάλληλα.

Αρχικά , κανονικοποιούμε τα διανύσματα των καινοτομιών διαιρώντας με την μέγιστη τιμή ώστε να περιορίζονται μεταξύ [-1 , 1] μέσω του παρακάτω κώδικα :

```
% Απεικονίσεις Καινοτομιών
% Κανονικοποίηση των καινοτομιών ώστε
% να έχουν εύρος απο [-2 ,2] και να φανεί καλύτερα
% η μέση τιμή
innovationA1_n = innovationA(1,:)/max(abs(innovationA(1,:)));
innovationA2_n = innovationA(2,:)/max(abs(innovationA(2,:)));
innovationA3_n = innovationA(3,:)/max(abs(innovationA(3,:)));
```

Οι μέσες τιμές των καινοτομιών είναι :

```
0.0098    0.0098    0.0013
```

Παρατηρούμε ότι βρίσκονται αρκετά κοντά στο 0 αλλά και είναι ίδιας τάξης με τις μέσες τιμές των λευκών θορύβων που υλοποιήσαμε παραπάνω:

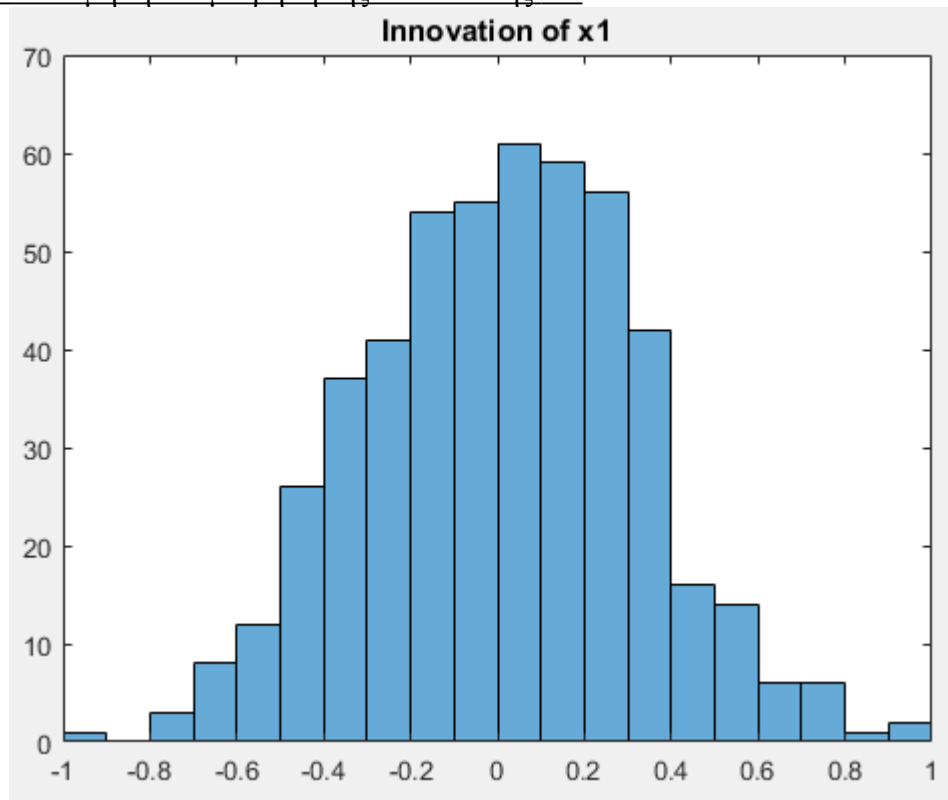
```
mean_noise =
0.0117    0.0309
```

Ο κώδικας για την παραπάνω διαδικασία είναι :

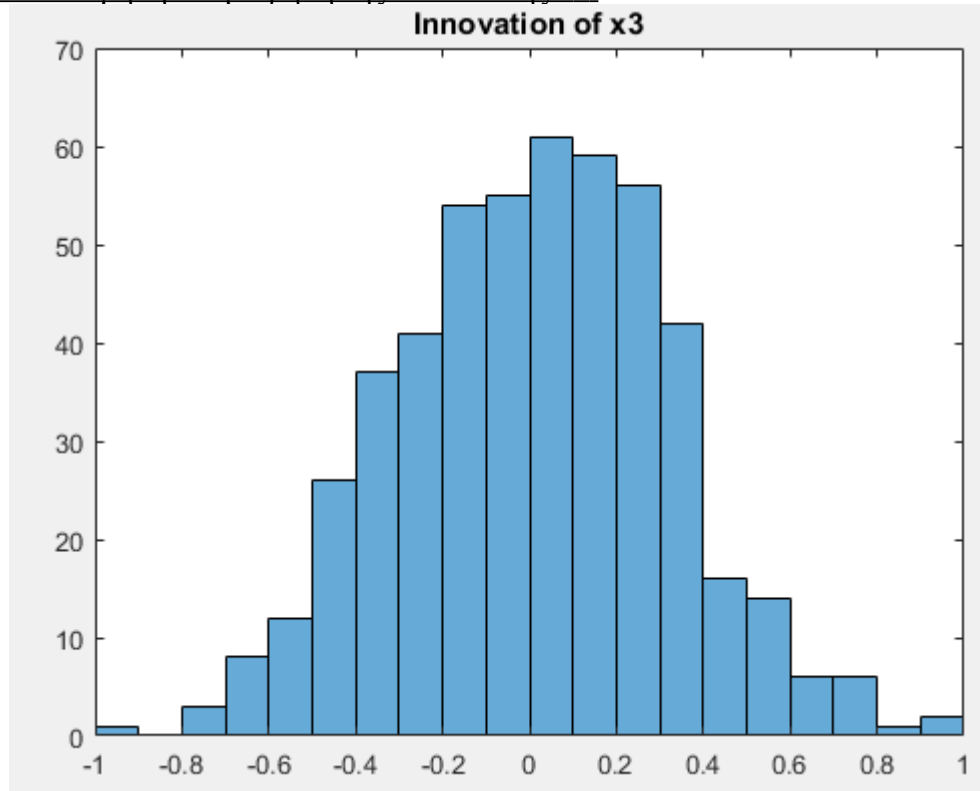
```
% Μέσες τιμές καινοτομιών  
N = length(t);  
means_in = [ sum(innovationA1_n)/N ;  
             sum(innovationA2_n)/N ;  
             sum(innovationA3_n)/N ];
```

Παρακάτω φαίνονται τα ιστογράμματα των καινοτομιών ώστε να δούμε αν έχουν Gaussian συμπεριφορά :

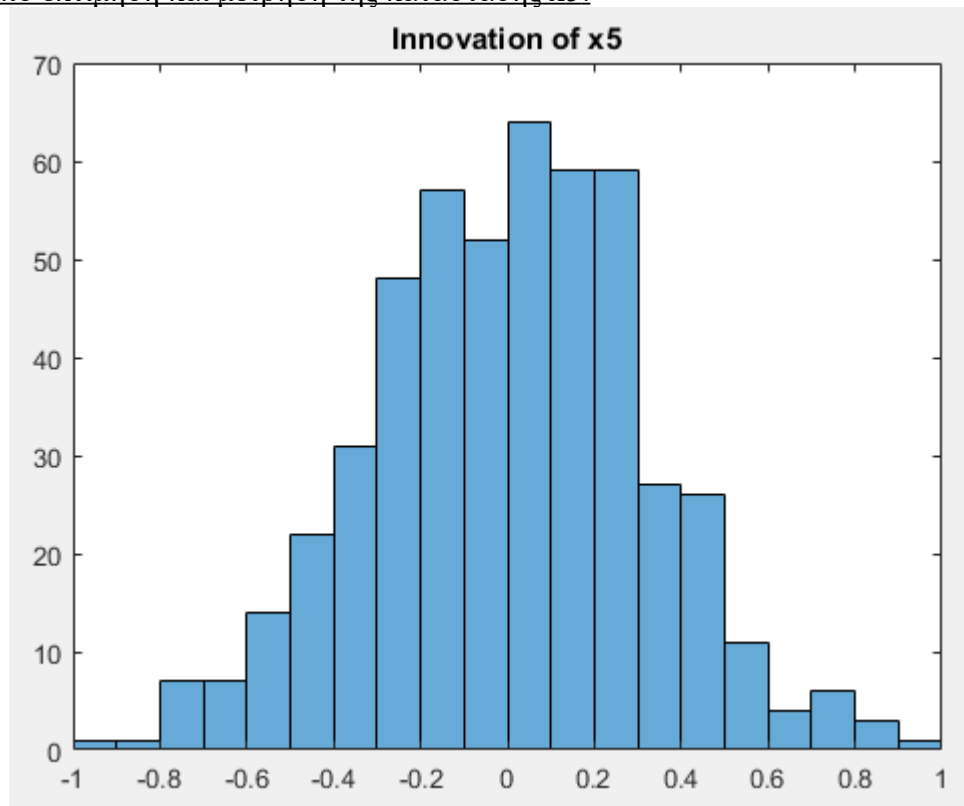
Καινοτομία από εκτίμηση και μέτρηση της κατάστασης x1:



Καινοτομία από εκτίμηση και μέτρηση της κατάστασης x_3 :



Καινοτομία από εκτίμηση και μέτρηση της κατάστασης x_5 :



Παρατηρούμε ότι και οι 3 καινοτομίες έχουν συμπεριφορά κανονικής κατανομής. Άρα συμπεραίνουμε ότι οι καινοτομίες έχουν συμπεριφορά λευκού θορύβου.

Πρέπει να τονίσουμε ότι για την συγγραφή της αναφοράς , επειδή χρειάστηκε να τρέξουμε πολλές φορές τον κώδικα , χρησιμοποιήσαμε τα αρχεία w.mat και v.mat ώστε να έχουμε τα ίδια αποτελέσματα σε κάθε εκτέλεση.

Η δημιουργία των παραπάνω έγινε με το αρχείο whiteNoise.m που φαίνεται παρακάτω:

```
clc; clear all; close all;

% Δημιουργία Λευκών Θορύβων για να έχουμε
% τα ίδια αποτελέσματα στην
% προσομοίωση
q = 2.5; r = 1.3;
N = 500;
w = sqrt(q)*randn(N,1);
v = sqrt(r)*randn(N,1);
save('w.mat','w');
save('v.mat','v');
```

Πρέπει να τονίσουμε ότι τα διάφορα σφάλματα που έχουμε οφείλονται στην μη-ιδανικότητα των λευκών θορύβων που ορίζουμε ως εισόδους.

Ο κώδικας για την εκτέλεση του Μέρους Α είναι στο αρχείο Project3A.m .

ΜΕΡΟΣ Β' :

1) Καλούμαστε τώρα να υλοποιήσουμε τον αλγόριθμο του φίλτρου Kallman, για το σύστημα που έχει μόνο τους δυο κυρίαρχους πόλους. Οι κυρίαρχοι πόλοι, δηλαδή οι πόλοι που βρίσκονται πιο δεξιά στο s-plane και έχουν πιο αργή απόκριση, είναι ο -2 και ο 1.

Από το σύστημα του μέρους Α , επιλέγουμε τις δυο τελευταίες καταστάσεις καθώς αφού έχουμε γράψει τον χώρο κατάστασης σε Jordan Canonical Form άρα το σύστημα που επιλέγουμε είναι :

$$\dot{x} = \begin{bmatrix} -2 & 0 \\ 0 & -1 \end{bmatrix} x + \begin{bmatrix} 2.277e-6 \\ -2.136e-6 \end{bmatrix} u$$

Οι πίνακες των θορύβων επιλέγονται ως $G = [1 \ 1]^T$ και $W = [1 \ 1]^T$ με τους αντίστοιχους πίνακες συμμεταβλητότητας Q και R που είναι ίδιοι με το μέρος Α.

Ο πίνακας H των μετρήσεων των καταστάσεων είναι :

$$H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Έπειτα , διακριτοποιούμε το σύστημα με την ίδια περίοδο δειγματοληψίας όπως στο Α δηλαδή $T_s = 0.1$ και το διακριτό σύστημα που θα εξετάσουμε είναι :

$$x[n+1] = \begin{bmatrix} 0.8187 & 0 \\ 0 & 0.9048 \end{bmatrix} x[n] + \begin{bmatrix} 2.064e-07 & 0.09063 & 0 \\ -2.033e-07 & 0.09516 & 0 \end{bmatrix} \begin{bmatrix} u[n] \\ w[n] \\ v[n] \end{bmatrix}$$
$$z[n] = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} x[n] + \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u[n] \\ w[n] \\ v[n] \end{bmatrix}$$

Η διαδικασία για την παραγωγή του παραπάνω συστήματος έγινε ως εξής :

Αρχικά , δημιουργήσαμε το πλήρες σύστημα στον συνεχή χρόνο , τοποθετώντας τους νέους πίνακες θορύβων και τον καινούργιο πίνακα ευαισθησίας μετρήσεων :

```
%Question 1
% Ορισμός του πλήρες συστήματος
A = diag([-500 -200 -100 -2 -1]);
B = [-1.744e-8 ; 1.861e-7 ; -3.092e-7 ; 2.277e-6 ; -2.136e-6];
% Νέος πίνακας ευαισθησίας των μετρήσεων
H = [0 0 0 1 0 ; 0 0 0 0 1];
% Νέοι πίνακες θορύβων
G = [0 ; 0 ; 0 ; 1 ; 1];
W = [1 ; 1 ];
% Νέος χώρος κατάστασης
sysB = ss(A,[B G 0*B] , H , [zeros(2,1) zeros(2,1) W]);
```

Έπειτα , διακριτοποιούμε το σύστημα με περίοδο δειγματοληψίας $T_s = 0.1$:

```
% Επιλογή περιόδου δειγματοληψίας
Ts = 0.1;
% Διακριτοποίηση συστήματος
sysBd = c2d(sysB , Ts , 'zoh');
```

Ύστερα, κρατάμε τα στοιχεία των πινάκων που μας ενδιαφέρουν(δύο τελευταίες καταστάσεις) από το συνεχές σύστημα , δημιουργούμε το σύστημα που θα εξετάσουμε και το διακριτοποιούμε με την ίδια περίοδο δειγματοληψίας:

```
% Κρατάμε μόνο το υποσύστημα με τις 2 τελευταίες καταστάσεις
A1 = [sysB.A(4,4) sysB.A(4,5);
      sysB.A(5,4) sysB.A(5,5)];
B1 = [sysB.B(4,1) ; sysB.B(5,1)];
G1 = [sysB.B(4,2) ; sysB.B(5,2)];
H1 = [sysB.C(1,4) sysB.C(1,5);
      sysB.C(2,4) sysB.C(2,5)];
W1 = sysB.D(:,3);
% Μειωμένο Σύστημα
sysm = ss(A1,[B1 G1 0*B1],H1,[0*W1 0*W1 W1]);
% Διακριτοποίηση μειωμένου συστήματος
sysmd = c2d(sysm , Ts , 'zoh');
```

Πρέπει να τονίσουμε ότι το σύστημα μας είναι παρατηρήσιμο :

The discrete subsystem is Observable

```
% Παρατηρησιμότητα
if rank(observ(sysmd.A,sysmd.C)) == 2
    fprintf('The discrete subsystem is Observable \n');
end
```

2) Επιλέγουμε όπως αναφέραμε και παραπάνω τους πίνακες συμμεταβλητότητας των θορύβων όπως παραπάνω με μόνη διαφορά τις διαστάσεις των πινάκων. Ο πίνακας Q έχει διάσταση 2×2 όπως και ο πίνακας R:

```
% Συνδιασπορές θορύβων
q1 = 2.5; r = 1.3;
% Πίνακες Συνδιασπορών
Q1 = q1*eye(2);
R = r*eye(2);
```

Με την εντολή `lsim`, προσομοιώνουμε το σύστημα με τις ίδιες εισόδους όπως το μέρος Α για 50s και με αρχικές συνθήκες $[0 \ 0]^T$ όπου αποθηκεύουμε τις μετρήσεις στον πίνακα `z`:

```
% Χρονικό διάστημα εκτέλεσης της προσομοίωσης
% του συστήματος
t = (0:Ts:50-Ts);
t = t';
% Είσοδος του συστήματος
u = sin(2*t);
% Οι διαταραχές και ο θόρυβος των μετρήσεων
load w.mat;
load v.mat;
x0 = [0 ; 0];
[out,~,statesB] = lsim(sysmd,[u w v],t,x0);
out = out' ;
% Διάνυσμα μετρήσεων
z = [ out(1,:) ; out(2,:) ];
```

Για να προσομοιώσουμε το πλήρες σύστημα με το φίλτρο Kalman, θα πρέπει να επιλέξουμε τον πίνακα `P` ως μη διαγώνιο. Πρέπει να τονίσουμε ότι ο πίνακας `P` πρέπει να είναι συμμετρικός και θετικά ορισμένος. Για να είναι ένας πίνακας συμμετρικός και θετικά ορισμένος θα πρέπει όλες οι ιδιοτιμές του να είναι θετικές και όλες οι ορίζουσες D_i όπου $i = 1, \dots, n$ να είναι θετικές. Επειδή η επιλογή κατάλληλων τιμών και ταυτόχρονα η ικανοποίηση των παραπάνω κριτηρίων, εργαστήκαμε ως εξής :

α) Ορίζουμε έναν πίνακα `L` διαστάσεων 5×5 που περιέχει διανύσματα ορθοκανονικών βάσεων ώστε να εξασφαλίσουμε ότι ο πίνακάς έχει τάξη ίση με την διάστασή του.

β) Ορίζουμε έναν διαγώνιο πίνακα `M` διαστάσεων 5×5 ο οποίος περιέχει μόνο θετικά στοιχεία

γ) Κάνουμε την πράξη L^*M*L^T ώστε ο πίνακας να είναι συμμετρικός .

Ο πίνακας αρχικοποίησης `P0` που θα χρησιμοποιήσουμε είναι :

```
P0 =

    0.9634    0.2744    0.1823    0.1280   -0.0005
    0.2744    0.9746    0.0496    0.0045    0.1796
    0.1823    0.0496    0.9355    0.1397    0.1230
    0.1280    0.0045    0.1397    1.0320    0.0482
   -0.0005    0.1796    0.1230    0.0482    1.1363
```

Βλέπουμε ότι όλες οι ιδιοτιμές του πίνακα `P0` είναι θετικές :

```
>> eig_P'

ans =

    1.4686    0.6056    1.1110    1.0313    0.8251
```

όπως και οι ορίζουσες D_i :

```
Det_i =

    0.9634    0.8636    0.7781    0.7774    0.8409
```

Ο κώδικας για την παραπάνω διαδικασία είναι ο εξής :

```
% Ορισμός αρχικού πίνακα συνδιασπορας σφαλματος
% Επιλογή ορθοκανονικού πίνακα ώστε να υπάρχουν
% όλες οι ορίζουσες Di
L = orth(rand(5));
% Ο πίνακας αυτός εγγυάται ότι θα είναι θετικά
% ορισμένος
M = diag(abs(rand(5, 1)) + 0.5);
% Δημιουργία Συμμετρικού πίνακα
P0 = L*M*L';
% Έλεγχος ιδιοτιμών
eig_P = eig(P0);
% Έλεγχος οριζουσών Di
Det_i = [P0(1,1) det(P0(1:2,1:2)) det(P0(1:3,1:3)) ...
         det(P0(1:4,1:4)) det(P0)];
```

3) Ο στόχος του φίλτρου Kalman του συγκεκριμένου συστήματος είναι να περιγράψει όσο το δυνατόν καλύτερα το πλήρες σύστημα χρησιμοποιώντας κάποιες μόνο καταστάσεις. Σε διάφορες βιβλιογραφίες αυτό περιγράφεται ως Reduced Order Kalman Filter. Συγκεκριμένα , στην εργασία αυτή χρησιμοποιήσαμε το Reduced Schmidt-Kalman Filter[3] όπου θα περιγράψουμε την διαδικασία του παρακάτω.

Θεωρούμε ότι το σύστημά μας είναι της παρακάτω μορφής :

$$\begin{bmatrix} \tilde{x}[n+1] \\ x[n+1] \end{bmatrix} = \begin{bmatrix} A1 & 0 \\ 0 & A2 \end{bmatrix} \begin{bmatrix} \tilde{x}[n] \\ x[n] \end{bmatrix} + \begin{bmatrix} B1 \\ B2 \end{bmatrix} u[n] + \begin{bmatrix} G1 \\ G2 \end{bmatrix} \begin{bmatrix} \tilde{w}[n] \\ w[n] \end{bmatrix}$$

$$z = [H1 \quad H2] \begin{bmatrix} \tilde{x}[n] \\ x[n] \end{bmatrix} + \begin{bmatrix} W1 \\ W2 \end{bmatrix} v[n]$$

όπου τα x με την περισπωμένη είναι οι καταστάσεις που θέλουμε να παρατηρήσουμε και για τους θορύβους ισχύουν :

$$\tilde{w}[n] \sim N(0, Q1) \quad [n] \sim N(0, Q1) \quad v[n] \sim N(0, R)$$

και πίνακας συμμεταβλητότητας σφάλματος είναι της μορφής :

$$P = \begin{bmatrix} P1 & S \\ S^T & P2 \end{bmatrix}$$

Ο αλγόριθμος που θα υλοποιήσει το φίλτρο Kalman τότε θα αποτελείται από το πρώτο σκέλος που υλοποιεί το στάδιο της “πρόβλεψης” :

$$\begin{aligned}\tilde{x}^{-}[n] &= A_1 \tilde{x}^{+}[n-1] + B_1 u[n-1] \\ \tilde{P}_1^{-}[n] &= A_1 \tilde{P}_1^{+}[n-1] A_1^T + Q_1 \\ S^{-}[n] &= A_1 S^{+}[n-1] A_2^T \\ P_2^{-}[n] &= A_2 P_2^{+}[n-1] A_2^T + Q_2\end{aligned}$$

και το δεύτερο σκέλος που αποτελεί την “διόρθωση” της παραπάνω πρόβλεψης :

$$a[n] = H_1 P_1^{-}[n] H_1^T + H_1 S^{-}[n] H_2^T + H_2 (S^{-}[n])^T H_1^T + H_2 P_2^{-}[n] H_2^T + R$$

$$K[n] = (P_1^{-}[n] H_1^T + S^{-}[n] H_2^T) (a[n])^{-1}$$

$$\tilde{x}^{+}[n] = \tilde{x}^{-}[n] + K[n] (z[n] - H_1 \tilde{x}^{-}[n])$$

$$P_1^{+}[n] = (I - K[n] H_1) P_1^{-}[n] - K[n] H_2 (S^{-}[n])^T$$

$$S^{+}[n] = (I - K[n] H_1) S^{-}[n] - K[n] H_2 P_2^{-}$$

$$P_2^{+}[n] = P_2^{-}[n]$$

Ο αλγόριθμος τρέχει για όσο χρονικό διάστημα τρέχει και η προσομοίωση του συστήματος.

Για την παραπάνω διαδικασία εργαστήκαμε ως εξής :

Αρχικά , προσδιορίσαμε τις αρχικές συνθήκες για τους πίνακες P1 διάστασης 2x2 , P2 διάστασης 3x3 και S διάστασης 2x3.

```
% ~~~~~
% Πίνακας συμμεταβλητότητας σφαλμάτων για το
% υποσύστημα που μας ενδιαφέρει
P10 = [P0(4,4) P0(4,5);
       P0(5,4) P0(5,5)];
% Αντίστοιχα για τις καταστάσεις που θεωρούμε
% αδιάφορες
P20 = [ P0(1,1:3); P0(2,1:3); P0(3,1:3)];
%
S0 = [ P0(4,1:3); P0(5,1:3)];
```

Οι πίνακες είναι :

```
P10 =
    1.0320    0.0482
    0.0482    1.1363

P20 =
    0.9634    0.2744    0.1823
    0.2744    0.9746    0.0496
    0.1823    0.0496    0.9355

S0 =
    0.1280    0.0045    0.1397
   -0.0005    0.1796    0.1230
```

Ύστερα , ορίζουμε τους πίνακες A2 διάστασης 3x3 που περιέχει την “μετάβαση” των καταστάσεων που δεν μας ενδιαφέρουν, H2 διάστασης 2x3 που περιέχει την ευαισθησία των μετρήσεων των ίδιων καταστάσεων και Q2 διάστασης 3x3 συµμεταβλητότητας διαταραχών των ίδιων πάλι καταστάσεων.

```
% Προσδιορισμός παραμέτρων για την υλοποίηση
% του φίλτρου
A2 = [sysBd.A(1,1:3); sysBd.A(2,1:3); sysBd.A(1,1:3)];
H2 = [sysBd.C(1,1:3) ; sysBd.C(2,1:3)];
q2 = 2.5;
Q2 = q2*eye(3);
```

Έπειτα , αρχικοποιούμε τις συνθήκες για το φίλτρο Schmidt-Kalman .

```
% Δέσμευση αποθηκευτικού χώρου για μεγέθη που
% θέλουμε να υπολογίσουμε
innovationB = zeros(2,length(t));
x_eB = zeros(2,length(t));
% Αρχικές Εκτιμήσεις για την κατάσταση
x = x0;
% Αρχικές τιμές πινάκων συµμεταβλητότητας σφάλματος
P1 = P10; P2 = P20; S = S0;
```

Τέλος , υλοποιούμε τον αλγόριθμο Schmidt-Kalman για το χρονικό διάστημα των 50s με βήμα όσο η περίοδος δειγματοληψίας ξεκινώντας από την διαδικασία πρόβλεψης

```
] for i=1:length(t)
    % Πρόβλεψη
    % x[n] (a priori) = A1*x[n-1]+B1*u[n]
    x = A1*x+B1*u(i,:);
    % P1[n] (a priori) = A1*P1[n-1]*A1'+Q1
    P1 = A1*P1*A1'+Q1;
    % S[n] (a priori) = A1*S[n-1]*A2'
    S = A1*S*A2';
    % P2[n] (a priori) = A2*P2[n-1]*A2'+Q2
    P2 = A2*P2*A2'+Q2;
```


και αργότερα κάνουμε την διαδικασία διόρθωσης

```
% Διόρθωση
% an = H1*P1[n] (a priori)*H1' + H1*S[n] (a priori)*H2'
%      + H2*S'[n] (a priori)*H1' + H2*P2[n] (a priori)*H2' + R
an = H1*P1*H1' + H1*S*H2' + H2*S'*H1' + H2*P2*H2' + R;
% K[n] = (P1[n] (a priori)*H1' + S[n] (a priori)*H2')*inv(an)
K = (P1*H1'+S*H2')/an;
% Υπολογισμός καινοτομιών
innovationB(:,i) = z(:,i) - H1*x ;
% x[n] (a posteriori) = x[n] (a priori)+K[n]*(z[n]-H1*x[n] (a priori)
x = x + K*innovationB(:,i);
% Αποθήκευση εκτίμησης
x_eB(:,i) = x;
% P1[n] (a posteriori) = (I-K[n]*H1)*P1[n] (a priori)
%                      -K[n]*H2*S'[n] (a priori)
P1 = (eye(2)-K*H1)*P1-K*H2*S';
% S[n] (a posteriori) = (I-K[n]*H1)*S[n] (a priori)
%                      -K[n]*H2*P2[n] (a priori)
S = (eye(2)-K*H1)*S - K*H2*P2;
% P2[n] (a posteriori) = P2[n] (a priori)
P2= P2;
end
```

Ο πίνακας κέρδους Kalman είναι :

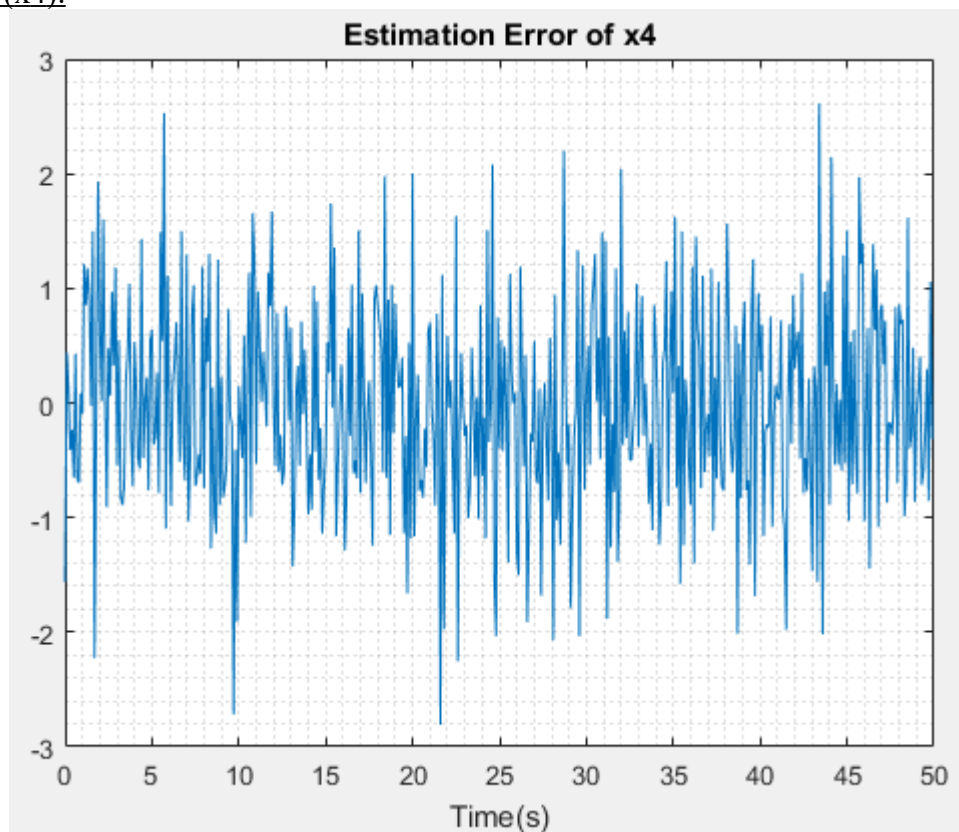
```
K =
    0.8805    0
    0    0.8236
```

Ο πίνακας συµμεταβλητότητας σφάλματος P είναι :

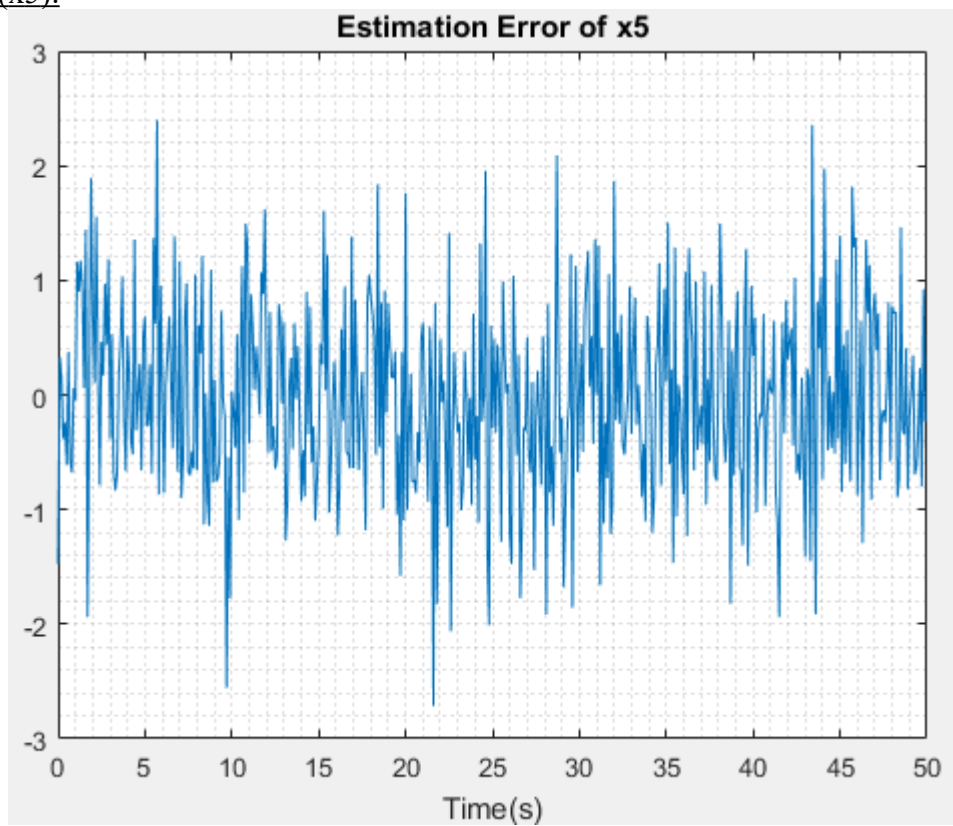
```
P1 =
    1.1446    0
    0    1.0707
```

Για την απόδοση του φίλτρου κάνουμε την ίδια διαδικασία με το μέρος A.

Οι γραφικές παραστάσεις των σφαλμάτων για τις μειωμένες καταστάσεις φαίνονται παρακάτω :
Κατάσταση 4 (x4):



Κατάσταση 5 (x5):

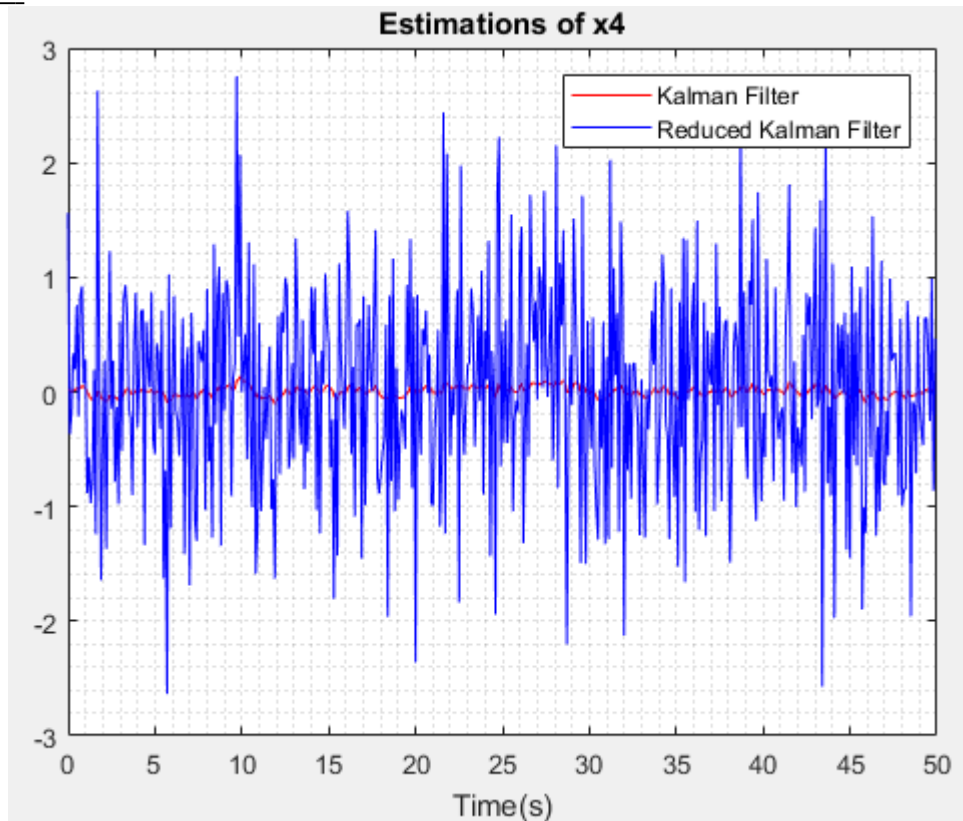


Η απόδοση του φίλτρου Schmidt-Kalman είναι :

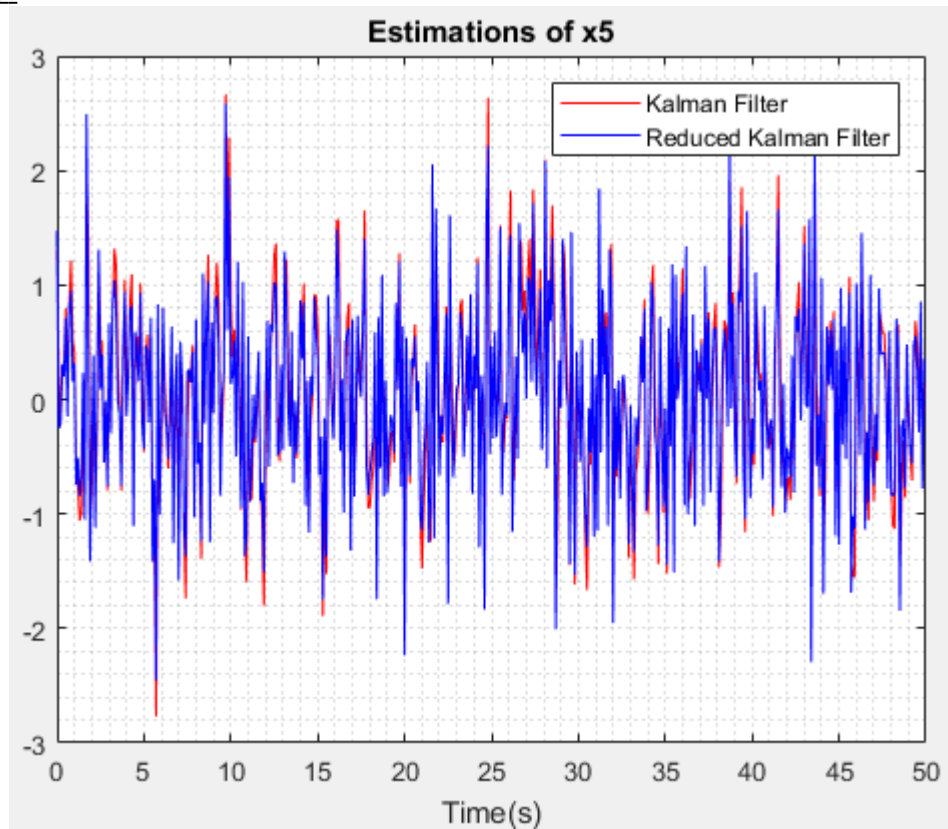
```
performanceB =  
  
0.8323
```

4) Αφού φορτώσουμε τον πίνακα των εκτιμήσεων της πρώτης προσομοίωσης, απεικονίζουμε σε κοινό διάγραμμα τις εκτιμήσεις των καταστάσεων που είναι ίδιες μεταξύ τους

Κατάσταση x4:



Κατάσταση x5:



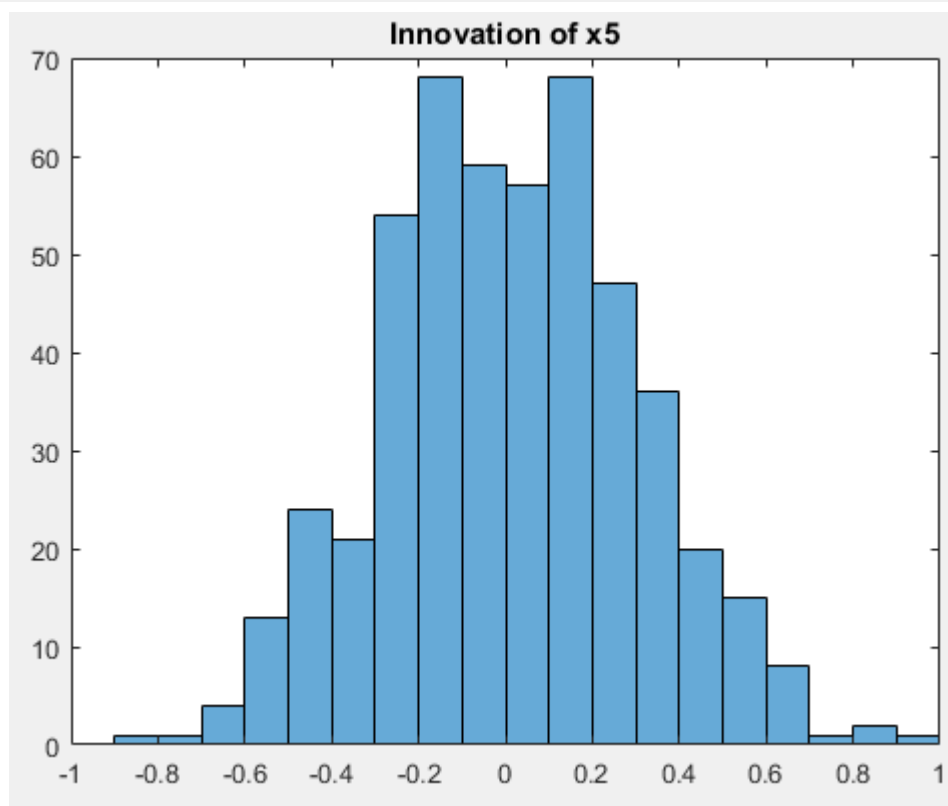
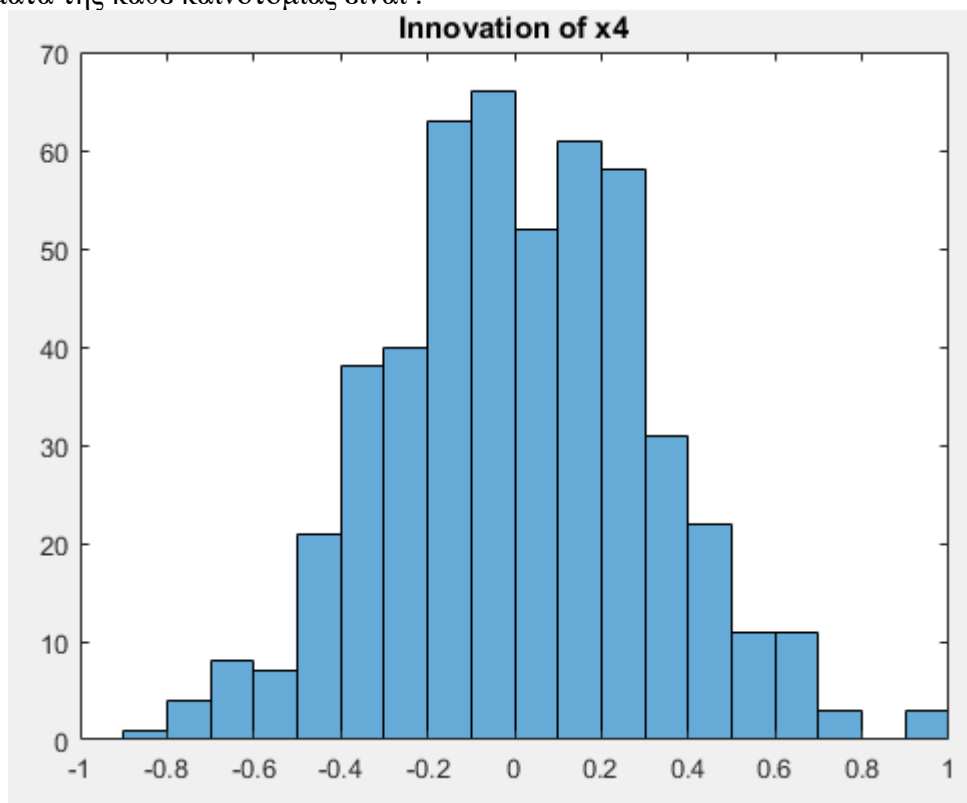
Παρατηρούμε ότι το δεύτερο φίλτρο εκτιμά την κατάσταση x5 αρκετά ικανοποιητικά ενώ βλέπουμε μεγάλες διαφορές στην εκτίμηση της κατάστασης x4. Αυτό το γεγονός πιθανόν οφείλεται στην αδυναμία επιλογής κατάλληλων αρχικών συνθηκών του πίνακα συμμεταβλητότητας σφάλματος. Πρέπει επίσης να τονίσουμε ότι επιλέγοντας θόρυβο με μεγαλύτερη συμμεταβλητότητα, οι εκτιμήσεις μεταξύ των δύο φίλτρων είναι πολύ κοντά μεταξύ τους.

5) Θα εξετάσουμε κατά πόσο οι καινοτομίες συμπεριφέρονται ως λευκοί θόρυβοι εξετάζοντας την μέση τιμή τους και το ιστόγραμμα της κάθε μίας. Η παραγωγή των παραπάνω στοιχείων γίνεται με τον ίδιο τρόπο όπως το πρώτο μέρος.

Οι μέσες τιμές είναι :

```
mean_noise =  
  
0.0117    0.0309
```

Τα ιστογράμματα της κάθε καινοτομίας είναι :



Παρατηρούμε ότι οι καινοτομίες έχουν μέση τιμή ίδιας τάξης με τους θορύβους αλλά δεν παρατηρούμε μια Gaussian συμπεριφορά από τα ιστογράμματα τους. Παρόλο όμως που οι καινοτομίες δεν έχουν κανονική κατανομή, βλέπουμε ότι το φίλτρο συμπεριφέρεται ικανοποιητικά.

Ο κώδικας για την εκτέλεση του Μέρους Β είναι στο αρχείο Project3B.m .

BIBΛΙΟΓΡΑΦΙΑ :

- [1] <https://www.engr.mun.ca/~millan/Eng6825/canonicals.pdf>
- [2] https://en.wikipedia.org/wiki/Jordan_normal_form
- [3] M. Sami Fadali , Antonio Visioli – Digital Control Engineering Analysis And Design (2nd Edition)
- [4] Mohinder S. Grewall , Angus P. Andrews – Kalman Filtering-Theory and Practice Using MATLAB (4th Edition)
- [5] Dan Simon – Optimal State Estimation -Kalman, H infinity, and Nonlinear Approaches (11th Edition)
- [6] Robert H. Bishop , Richard C. Dorf – Σύγχρονα Συστήματα Αυτομάτου Ελέγχου (11 η Έκδοση)