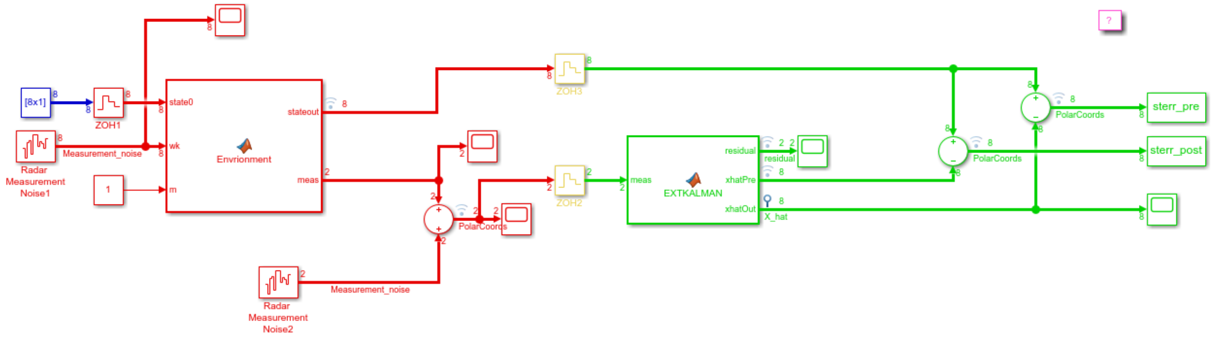

Figure 1.Simulink for ekf

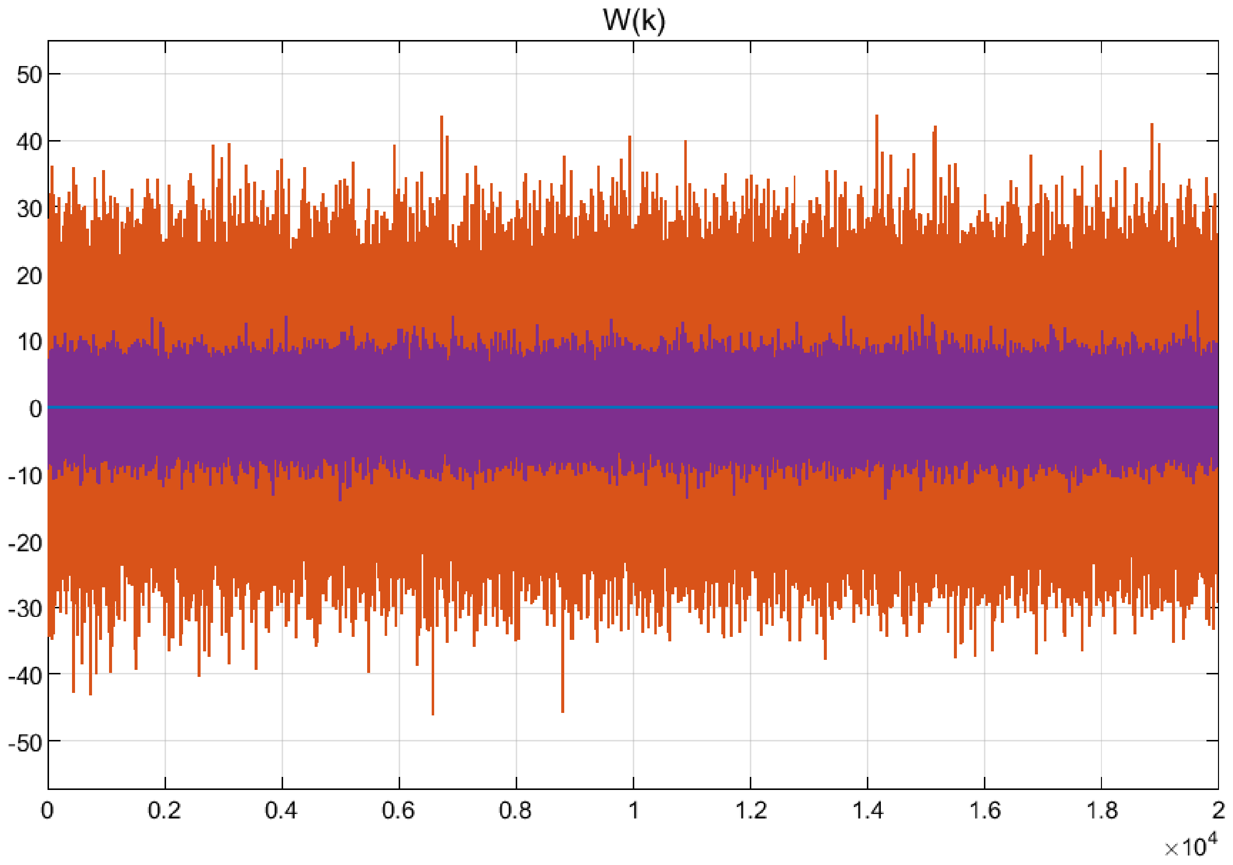

Fig 2.Process noise

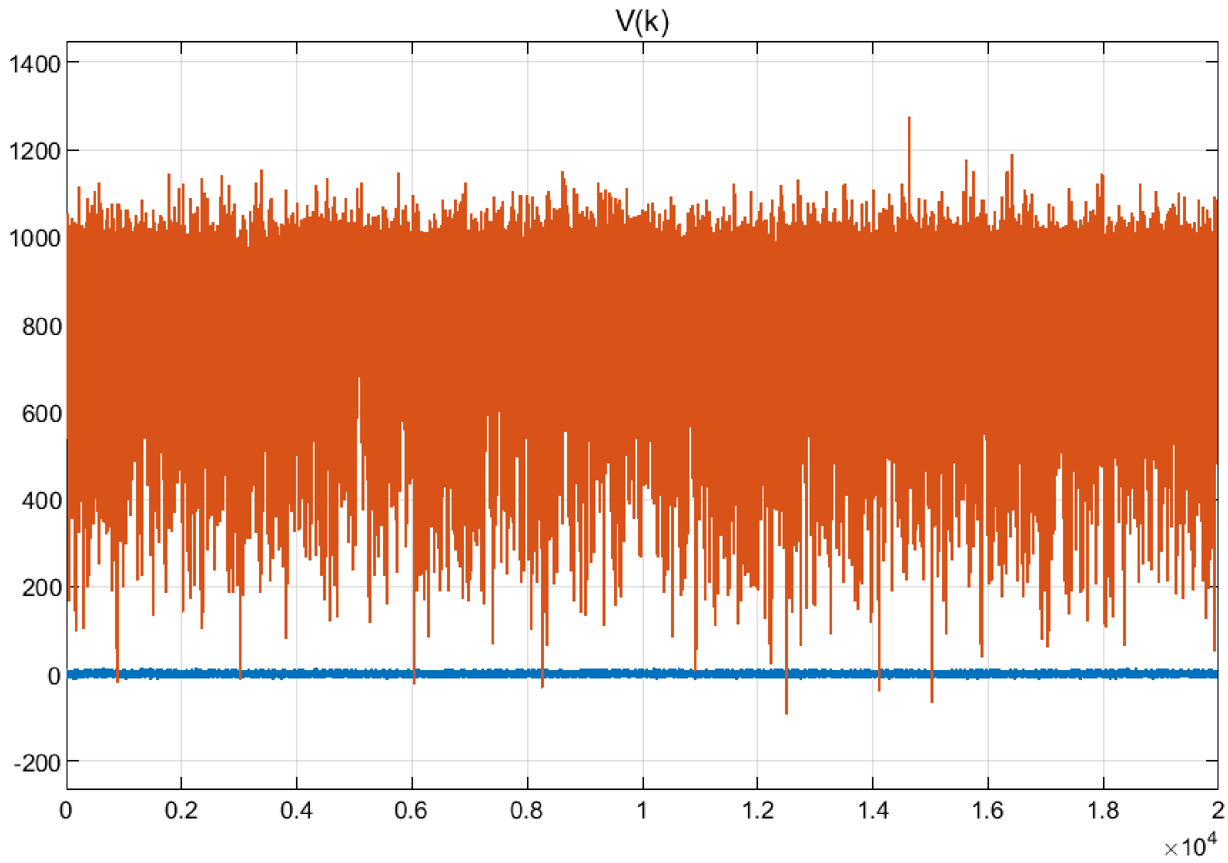

Fig3. Measurement noise

```
%initialize params
simlen = 40; %simulation length
stderr_pre = zeros(8,simlen,1000);  %state errors for pre state estimations
stderr_post = zeros(8,simlen,1000); %state errors for post state estimations
std_pre = zeros(8,simlen);          %standard deviation for state errors for
pre state estimations
std_post = zeros(8,simlen);         %standard deviation for state errors for
post state estimations
mean_pre = zeros(8,simlen);         %mean for state errors for pre state
estimations
mean_post = zeros(8,simlen);        %mean for state errors for post state
estimations

for i=1:8
    for j=2:simlen
        for k =1:1000
            stderr_pre(i,j,k)=sterr_pre(j+simlen*(k-1),i);      %get
stderr_pre from simulation result.
            stderr_post(i,j,k)=sterr_post(j+simlen*(k-1),i);    %get
stderr_post from simulation result.
        end
        std_pre(i,j)=std(stderr_pre(i,j,:));                         %get std for
pre
        std_post(i,j)=std(stderr_post(i,j,:));                       %get std for
post
        mean_pre(i,j)=mean(stderr_pre(i,j,:));                       %get mean for
pre
        mean_post(i,j)=mean(stderr_post(i,j,:));                     %get mean for
post
    end
end

%figures for standard deviation and mean values at each time step for pre and
post state estimations.
for i=1:8
    figure(i);
    subplot(4,1,1);
    plot(mean_pre(i,:));
    subplot(4,1,2);
    plot(mean_post(i,:));
    subplot(4,1,3);
    plot(std_pre(i,:));
    subplot(4,1,4);
    plot(std_post(i,:));
end
```

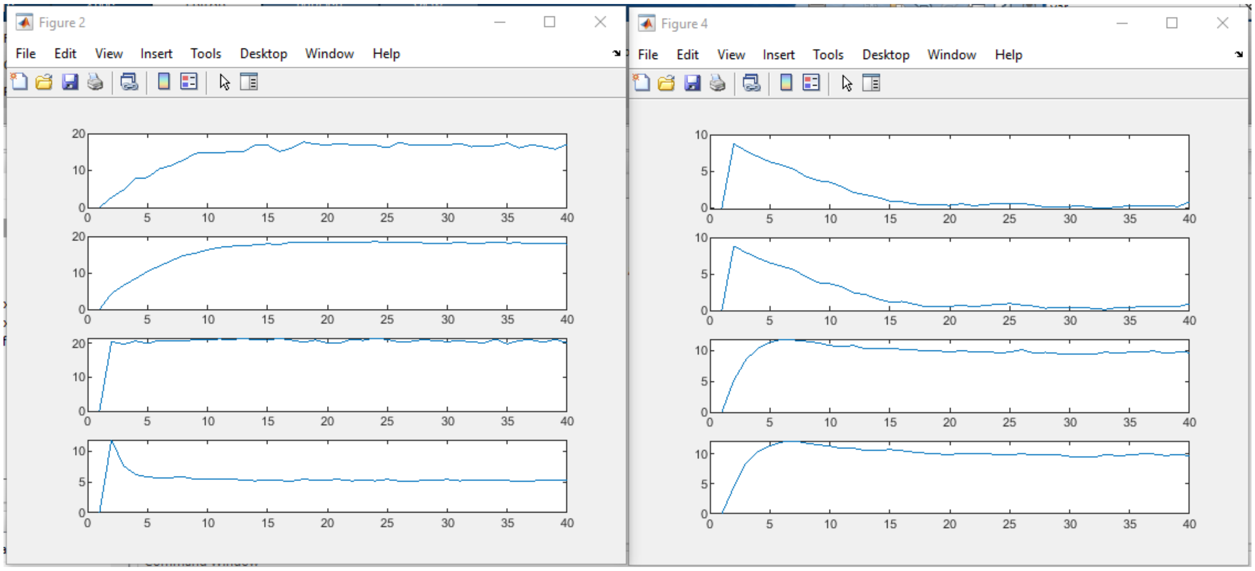
Figure 2
File Edit View Insert Tools Desktop Window Help
20
10
0
0 5 10 15 20 25 30 35 40
Figure 4
File Edit View Insert Tools Desktop Window Help
10
5
0
0 5 10 15 20 25 30 35 40

Fig 4. Mean and standard deviation for pre and post estimations.

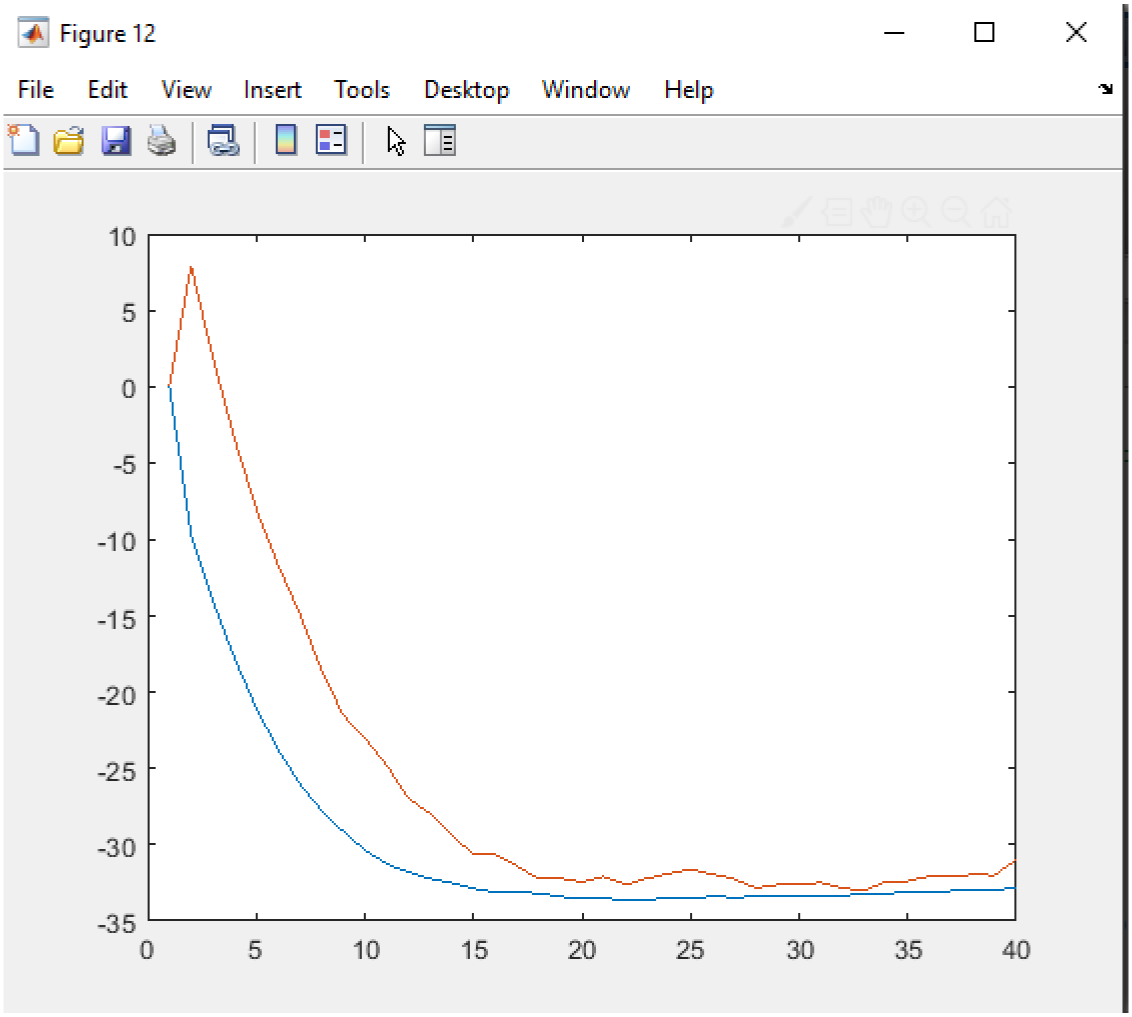

Figure 5.co-plot for standard deviation of vy

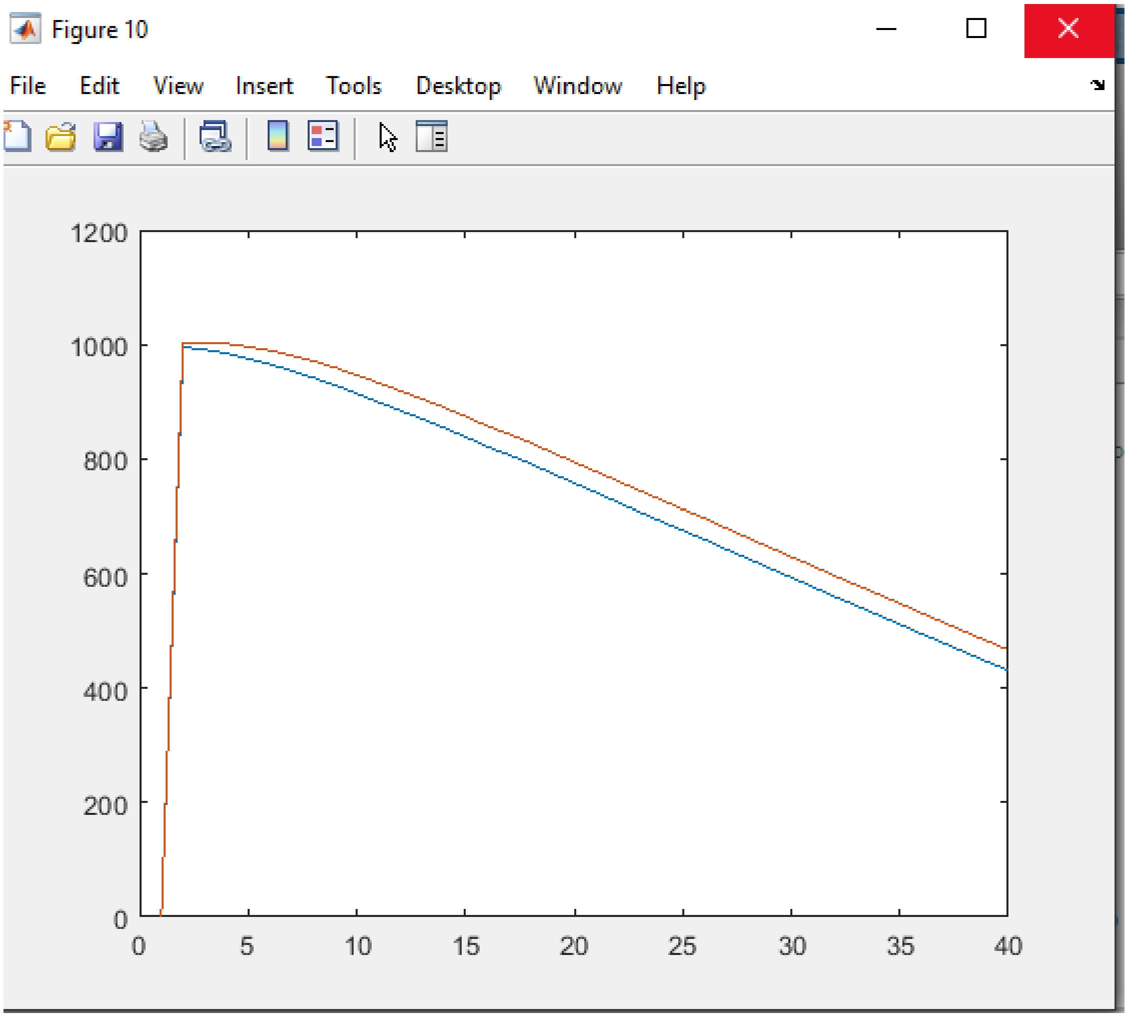

Figure 6.co-plot for standard deviation of ry