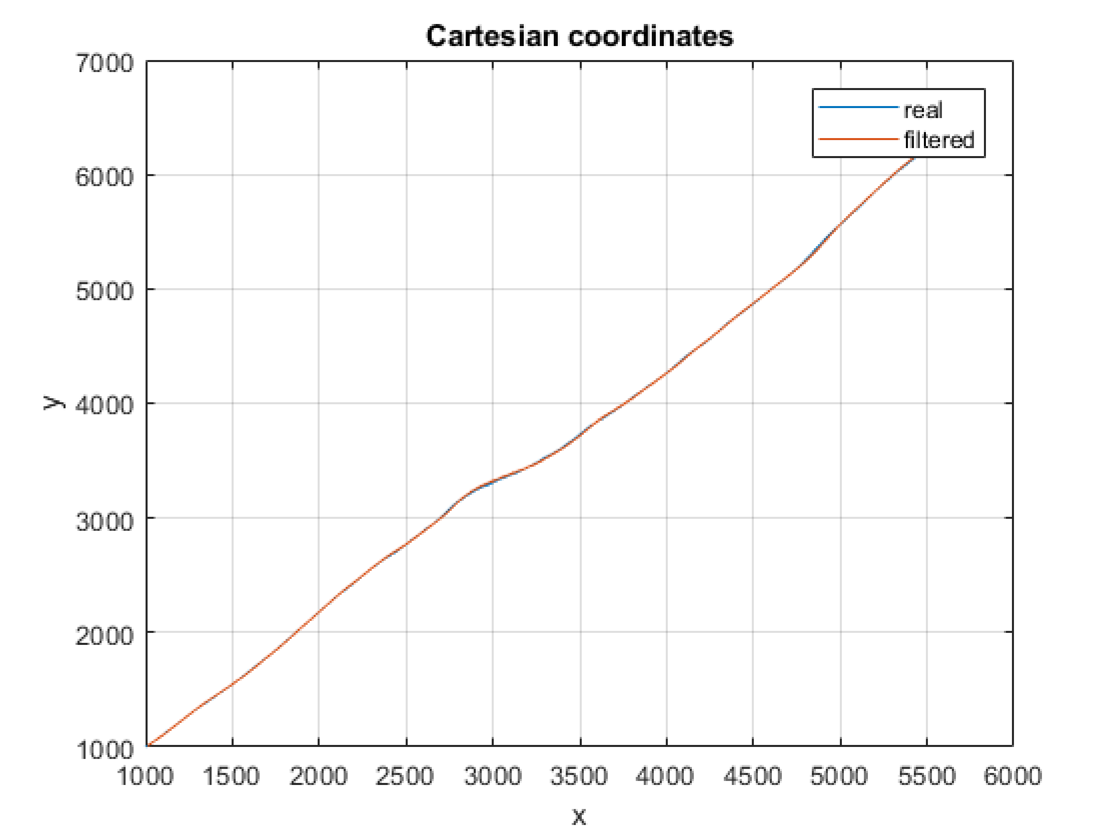
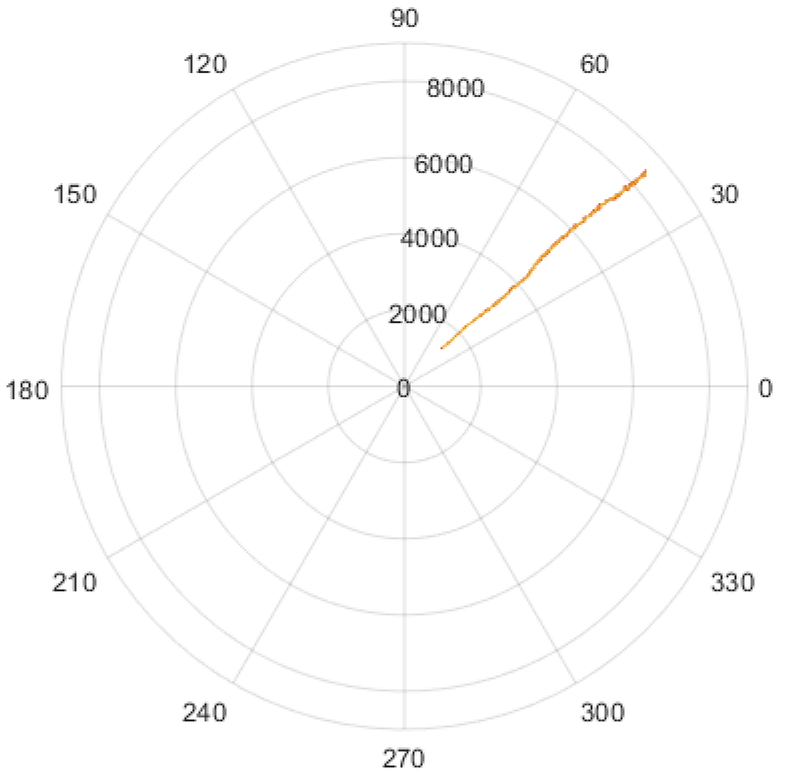
Laboratory work 12

Extended Kalman filter for navigation and tracking Group 5: Andrei Chemikhin, Ruslan Agishev, Valery Nevzorov Skoltech, 2017

```
% trajectory generation
close all;
clear;
N = 500;
T=1;
vx1=10;
vy1=10;
sigmaA=0.3;
sigmab = 0.004;
sigmaD = 50;
x1 = 1000;
y1 = 1000;
P0 = 10e10*eye(4);

[F,G] = state_space(T);
Q = G*G'*sigmaA^2;
R = diag([sigmaD^2 sigmab^2]);

t=1:N;
[x,y, b,D, bm,Dm] = trajgen_acc(sigmaA, N, T, x1, y1, vx1, vy1, sigmab,sigmaD);

[Xpr,Ppr,Xfl,Pfl,K] = extended_kalman_filter(P0,F,Q,R,b,D,bm,Dm);

xfl = Xfl(1,:);
yfl = Xfl(3,:);

figure
plot(x,y, xfl,yfl)
grid on
title('Cartesian coordinates')
xlabel('x')
ylabel('y')
legend('real', 'filtered')
```

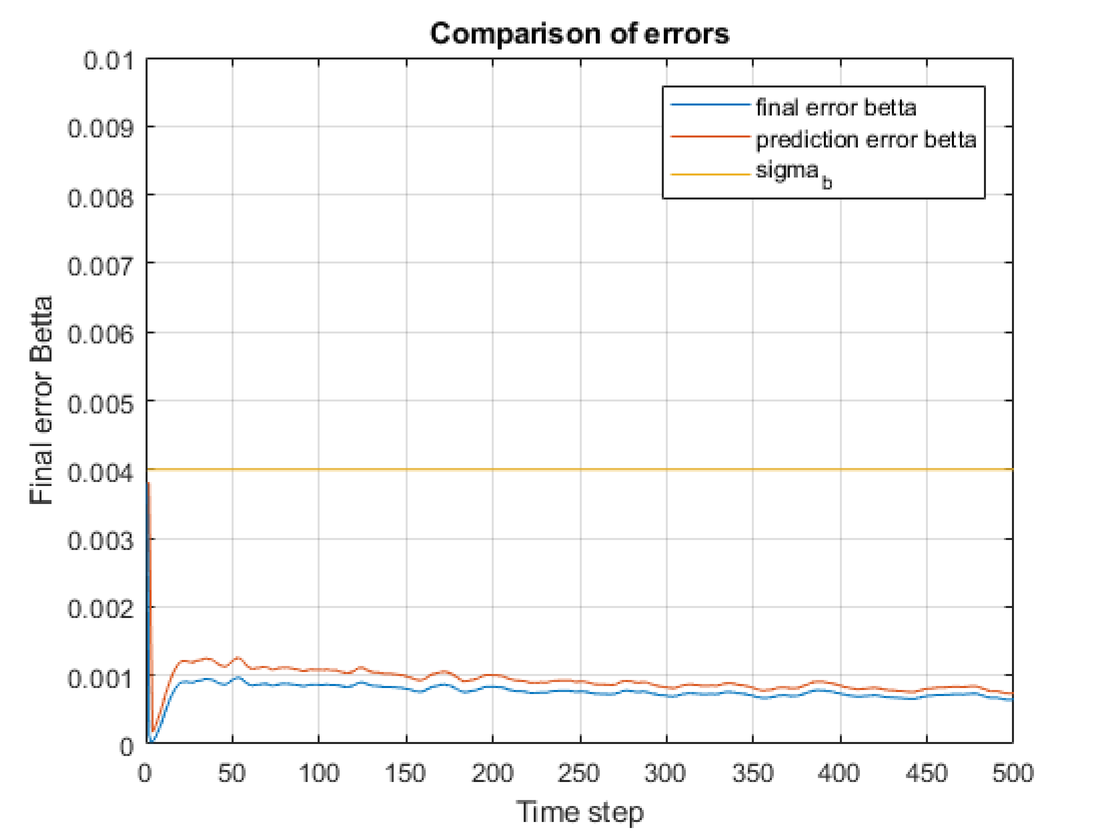


```
[Dfl, bfl] = cartesian2polar(xfl,yfl);

figure
polarplot(b,D, bm,Dm, bfl,Dfl);
```

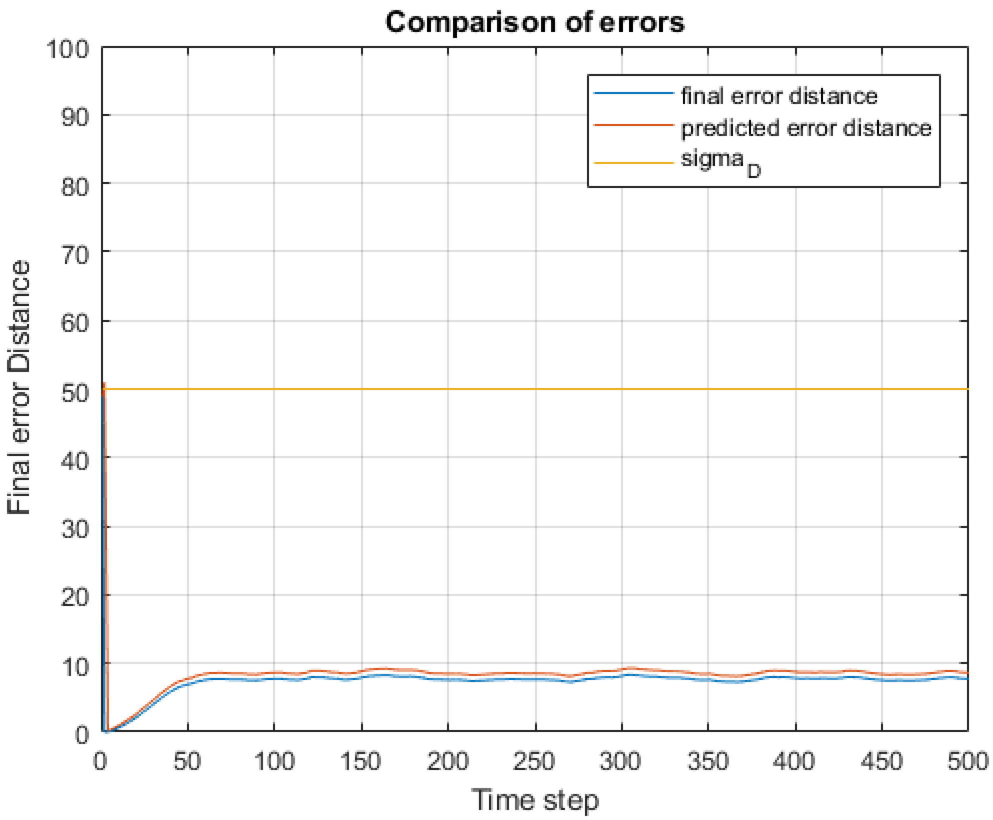


```
% final error
% generation of M=500 realiztions of trajectories

px = nan(1,N);
py = nan(1,N);

for i=1:(N-1)
    px(i) = sqrt(Pfl{i}(1,1));
    py(i) = sqrt(Pfl{i}(3,3));
end


M=500;
b = cell(1,M);
D = cell(1,M);
bm = cell(1,M);
Dm = cell(1,M);
for i=1:M
    [~,~, b{i},D{i}, bm{i},Dm{i}] = trajgen_acc(sigmaA, N, T, x1, y1, vx1, vy1, sigmab,sigmaD);
end

% Kalman-filtration of generated trajectories
bfl = cell(1,M);
Dfl = cell(1,M);
bpr = cell(1,M);
Dpr = cell(1,M);
```

```
Xfl_ = cell(1,M);
Xpr_ = cell(1,M);
xfl = cell(1,M);
yfl = cell(1,M);
xpr = cell(1,M);
ypr = cell(1,M);

for i=1:M
    [Xpr_{i},~,Xfl_{i},~,~] = extended_kalman_filter(P0,F,Q,R,b{i},D{i},bm{i},Dm{i});
    xfl{i} = Xfl_{i}(1,:);
    yfl{i} = Xfl_{i}(3,:);
    [Dfl{i}, bfl{i}] = cartesian2polar(xfl{i},yfl{i});

    xpr{i} = Xpr_{i}(1,:);
    ypr{i} = Xpr_{i}(3,:);
    [Dpr{i}, bpr{i}] = cartesian2polar(xpr{i},ypr{i});
end

fleb = final_error(bfl, b);
fleD = final_error(Dfl, D);

preb = final_error(bpr, b);
preD = final_error(Dpr, D);

figure
plot(t,fleb, t,preb, t,sigmab*ones(1,N));
legend('final error betta', 'prediction error betta', 'sigma_b');
ylabel('Final error Betta')
xlabel('Time step')
title('Comparison of errors')
ylim([0,0.01])
grid on;
```



```
figure
plot(t,fleD, t,preD, t,sigmaD*ones(1,N));
legend('final error distance', 'predicted error distance', 'sigma_D');
ylabel('Final error Distance')
xlabel('Time step')
title('Comparison of errors')
ylim([0,100])
grid on;
```



Published with MATLAB® R2017b