
Table of Contents

Laboratory work 8	1
Tracking and forecasting in conditions of measurement gaps	1
Group 5: Ruslan Agishev, Andrei Chemikhin, Valery Nevzorov	1
Skoltech, 2017	1
=====	1
=====	5
Conclusion	7

Laboratory work 8

Tracking and forecasting in conditions of measurement gaps

**Group 5: Ruslan Agishev, Andrei Chemikhin,
Valery Nevzorov**

Skoltech, 2017

=====

```
[N,M,T,v1,x1,sigmaA,sigmaN, F,G,H,P0,X0,R,Q, m,t] = init();
P = 0.2;
% =====
[x, z, ~] = trajgen_acc(x1, sigmaN, sigmaA, N, T, v1, P);

% gaps between measurements is given by P
figure(1)
plot(t,x, t,z, '.')
title('Trajectory and measurements')
legend('x(t)', 'z(t)');
xlabel('Time');
ylabel('Coordinate');
grid on;

% filtered values for obtained trajectory
% Kalman filter is developed to track moving object under conditions
% of
% loss of part of measurements determined by P-value.
[~,Ppr,Xf1,Pf11,K] = kalman_filter(X0,P0,F,Q,H,R,z);

figure(2)
plot(t,x, t,z, '.', t,Xf1(1,:));
legend('real', 'measure', 'filter');
ylabel('Coordinate')
```

```

xlabel('Time step')
grid on;

% backward smoothing algorithm for filtered trajectory
[Xsm, ~] = smoothing_back(Xfl, Pfl1, Ppr, F);
xsm = Xsm(1,:);
figure(3)
plot(t,xsm, t,Xfl(1,:), t,x, t,z, '.');
legend('smooth', 'filter', 'real', 'measure');
grid on
xlabel('Time')
ylabel('Traj')

% generation of forecasting trajectory
[Xprm,Pprm,Xflm,Pflm,Km] = kalman_filter_extra(X0,P0,F,Q,H,R,z,m);
[Xflmsm, ~] = smoothing_back(Xflm, Pfl1, Ppr, F);

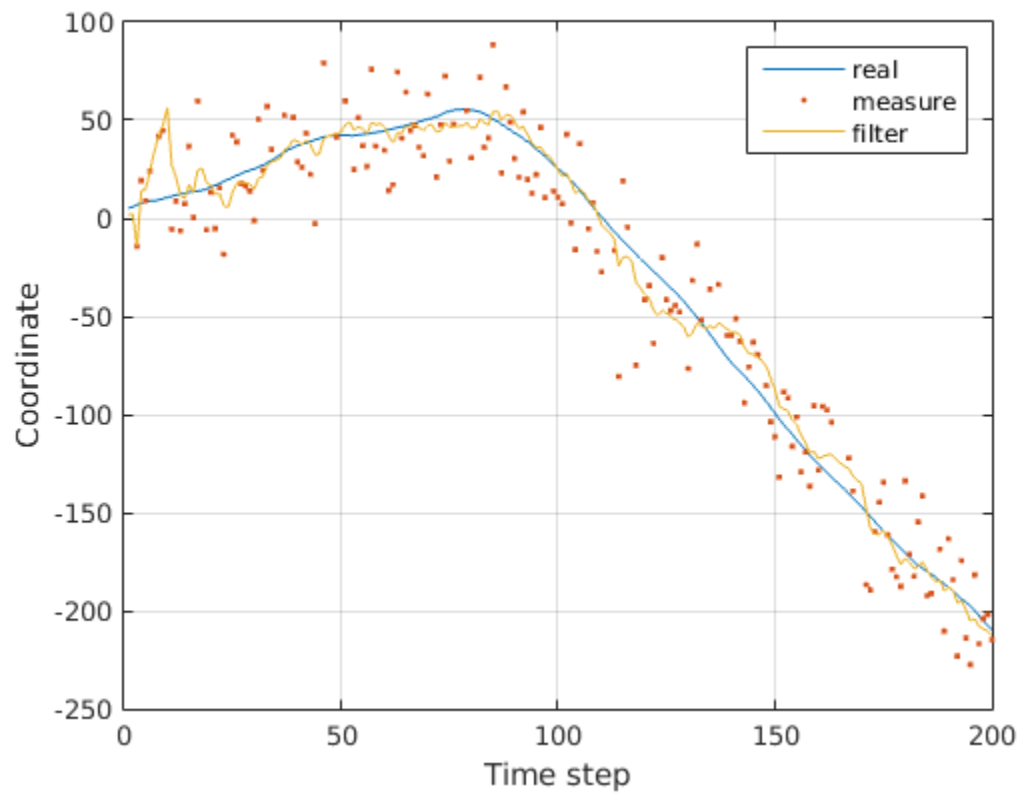
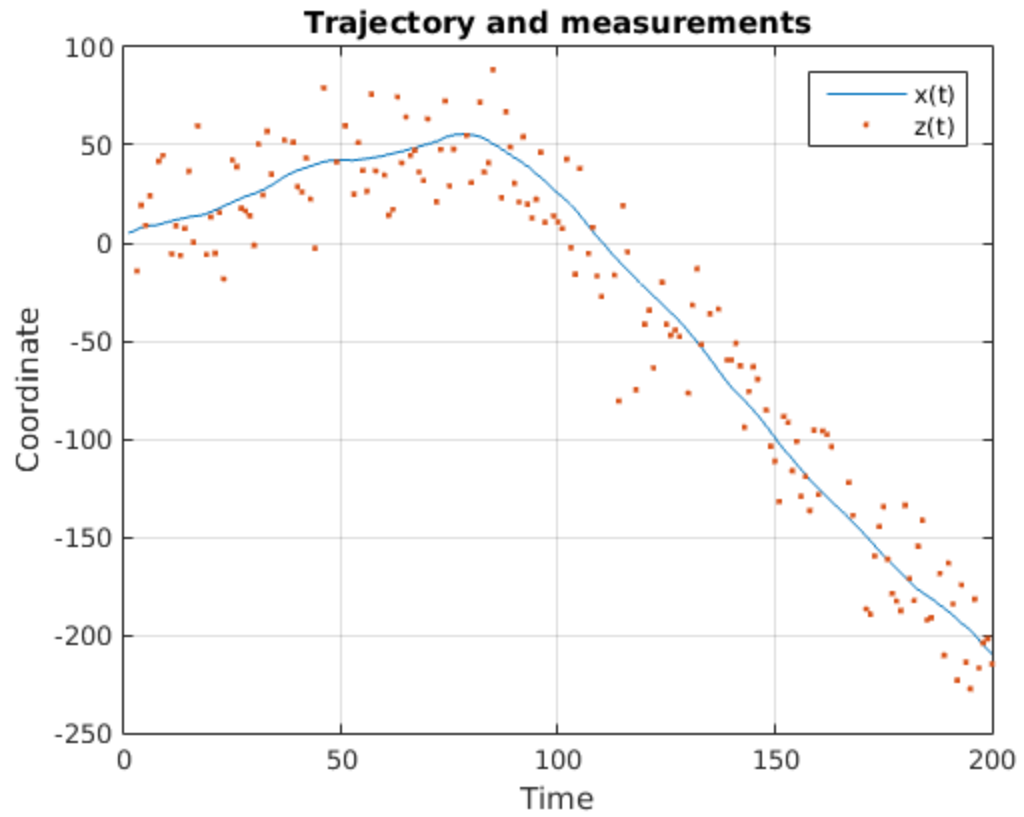
figure(4)
plot(t,x, t,z, '.', t,Xflmsm(1,:), t,Xsm(1,:));
legend('real', 'measure', '7 step', '1 step');
ylabel('Coordinate')
xlabel('Time step')
title('Extra filtration')
grid on;

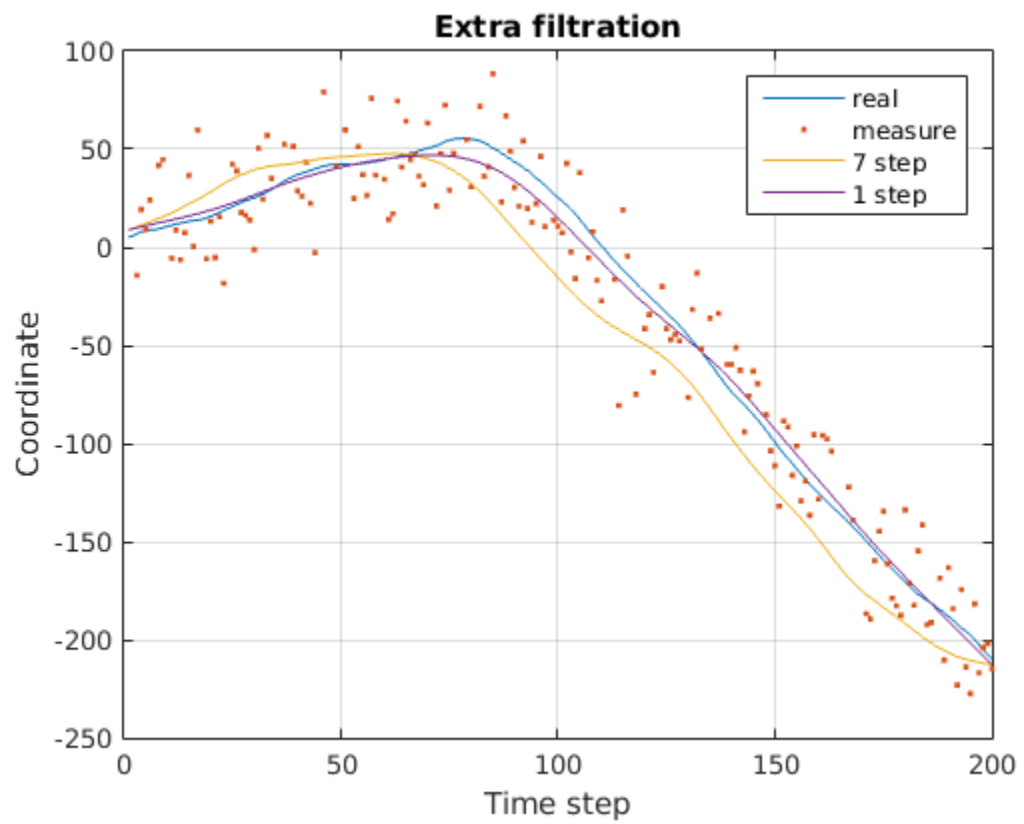
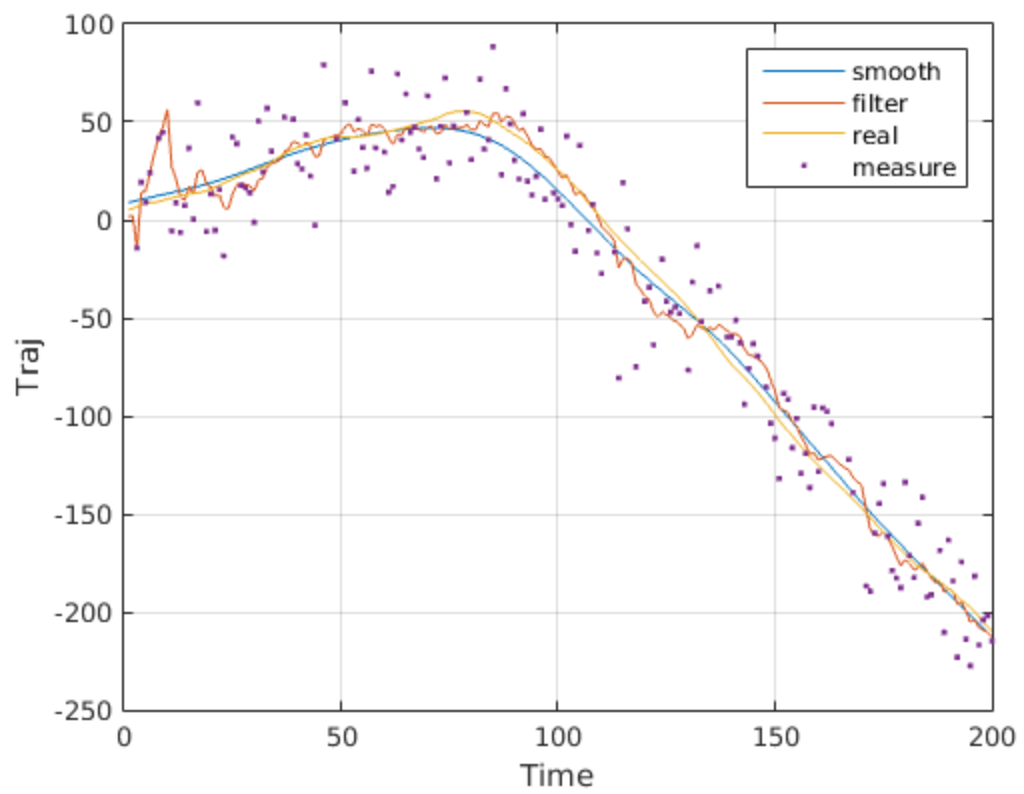
% generation of M=500 realizations of trajectories
X = cell(1,M);
Z = cell(1,M);
for i=1:M
    [X{i}, Z{i}] = trajgen_acc(x1, sigmaN, sigmaA, N, T, v1, P);
end
% Kalman-filtration of generated trajectories
Xfl_ = cell(1,M);
Xfl_ex = cell(1,M);
xfl = cell(1,M);
xfl_ex = cell(1,M);

for i=1:M
    [~,~,Xfl_{i},~,~] = kalman_filter(X0,P0,F,Q,H,R,Z{i});
    xfl{i} = Xfl_{i}(1,:);
    [~,~,Xfl_ex{i},~,~] = kalman_filter_extra(X0,P0,F,Q,H,R,Z{i},m);
    xfl_ex{i} = Xfl_ex{i}(1,:);
end

% final errors for filtered 1-step- and 7-step- forecasting
trajectories
% P = 0.2
fe2 = final_error(xfl, X);
fem2 = final_error(xfl_ex, X);

```





=====

```
[N,M,T,v1,x1,sigmaA,sigmaN, F,G,H,P0,X0,R,Q, m,t] = init();
P = 0.3;
% =====

for i=1:M
    [X{i}, Z{i}] = trajgen_acc(x1, sigmaN, sigmaA, N, T, v1, P);
end
% Kalman-filtration of generated trajectories
Xfl_ = cell(1,M);
Xfl_ex = cell(1,M);
xfl = cell(1,M);
xfl_ex = cell(1,M);

for i=1:M
    [~,~,Xfl_{i},~,~] = kalman_filter(X0,P0,F,Q,H,R,Z{i});
    xfl{i} = Xfl_{i}(1,:);
    [~,~,Xfl_ex{i},~,~] = kalman_filter_extra(X0,P0,F,Q,H,R,Z{i},m);
    xfl_ex{i} = Xfl_ex{i}(1,:);
end

% final errors for filtered 1-step- and 7-step- forecasting
trajectories
% P = 0.3
fe3 = final_error(xfl, X);
fem3 = final_error(xfl_ex, X);

% =====
[N,M,T,v1,x1,sigmaA,sigmaN, F,G,H,P0,X0,R,Q] = init();
P = 0.5;
% =====

for i=1:M
    [X{i}, Z{i}] = trajgen_acc(x1, sigmaN, sigmaA, N, T, v1, P);
end
% Kalman-filtration of generated trajectories
Xfl_ = cell(1,M);
Xfl_ex = cell(1,M);
xfl = cell(1,M);
xfl_ex = cell(1,M);

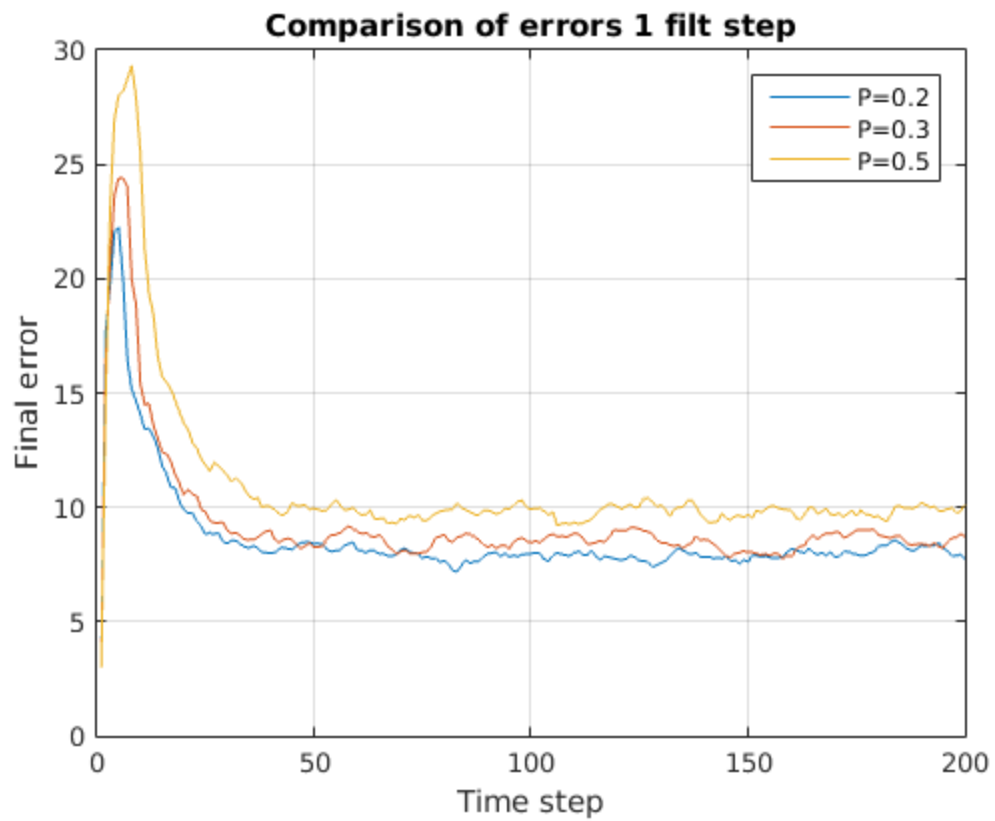
for i=1:M
    [~,~,Xfl_{i},~,~] = kalman_filter(X0,P0,F,Q,H,R,Z{i});
    xfl{i} = Xfl_{i}(1,:);
    [~,~,Xfl_ex{i},~,~] = kalman_filter_extra(X0,P0,F,Q,H,R,Z{i},m);
    xfl_ex{i} = Xfl_ex{i}(1,:);
end

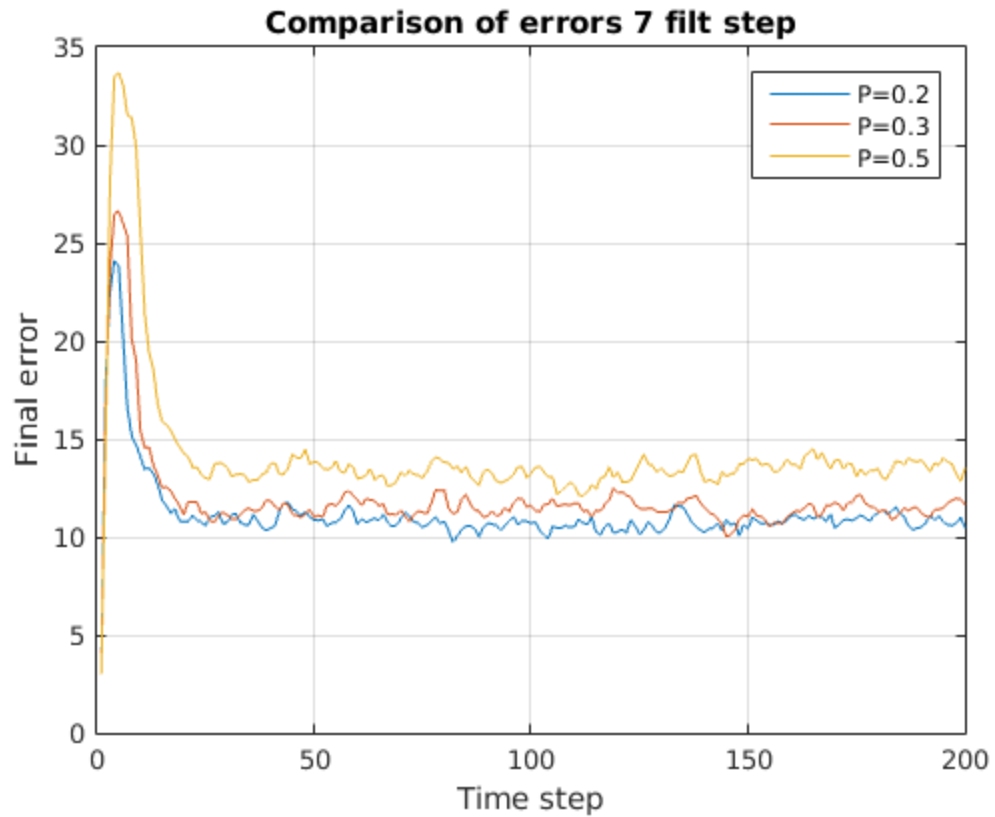
% final errors for filtered 1-step- and 7-step- forecasting
trajectories
% P = 0.5
```

```
fe5 = final_error(xfl, X);
fem5 = final_error(xfl_ex, X);

figure(5)
plot(t,fe2, t,fe3, t,fe5);
legend('P=0.2', 'P=0.3', 'P=0.5');
ylabel('Final error')
xlabel('Time step')
title('Comparison of errors 1 filt step')
grid on;
```

```
figure(6)
plot(t,fem2, t,fem3, t,fem5);
legend('P=0.2', 'P=0.3', 'P=0.5');
ylabel('Final error')
xlabel('Time step')
title('Comparison of errors 7 filt step')
grid on;
```





Conclusion

Less measurements are lost, higher the accuracy of estimation of the trajectory using Kalman filter. Graphs for 1-step and 7-step forecasting trajectory illustrate the statement.

Published with MATLAB® R2015a