24677-A Project - P4

Saeed Bai

TOTAL POINTS

48 / 50

QUESTION 1

1 Exercise 1.1 18 / 20

- **0 pts** Correct
- **1 pts** Wrong Kalman Filter initialization
- **1 pts** Wrong MSE definition

✓ **- 2 pts** **Unexpected MSE behavior**

- **3 pts** Wrong Kalman Filter process
- **5 pts** No graph on MSE
- **5 pts** No graph on state comparison

1 The error should be a decaying graph, this is increasing?

QUESTION 2

2 Execise 1.2 20 / 20

✓ **- 0 pts** **Correct**

- **2 pts** Unexpected pose graph behavior

QUESTION 3

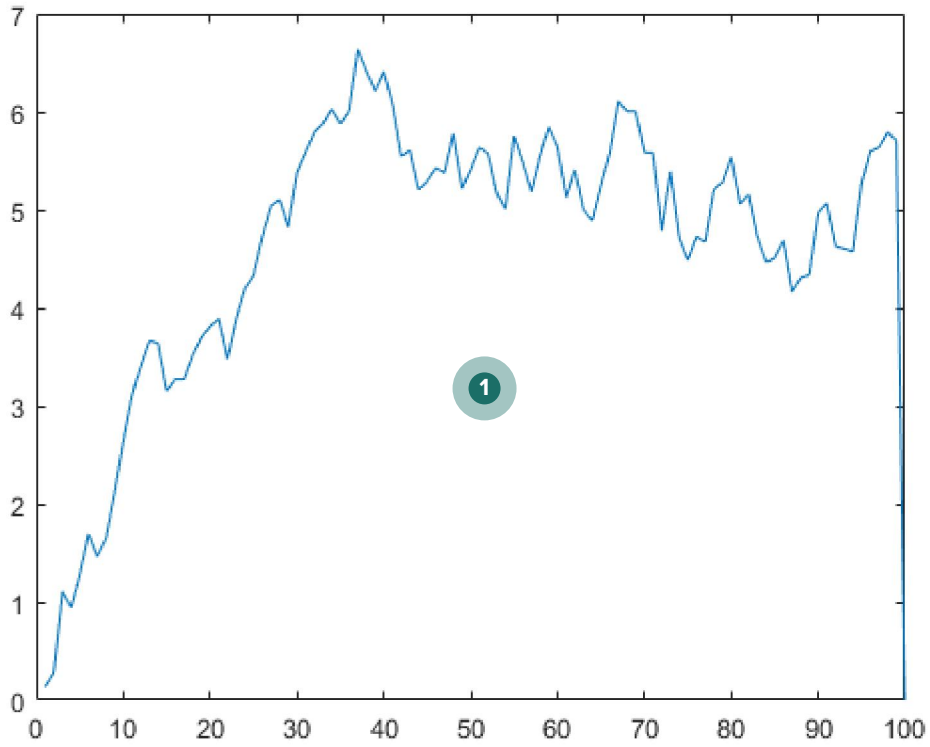
3 Result Graphs 10 / 10

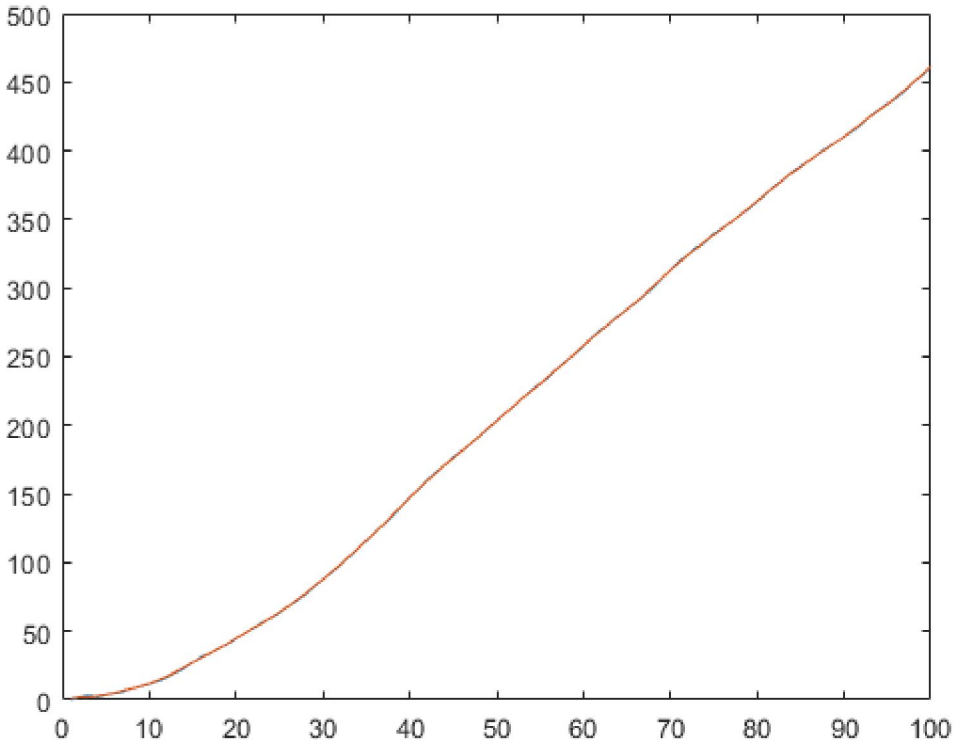
✓ **- 0 pts** **Correct**

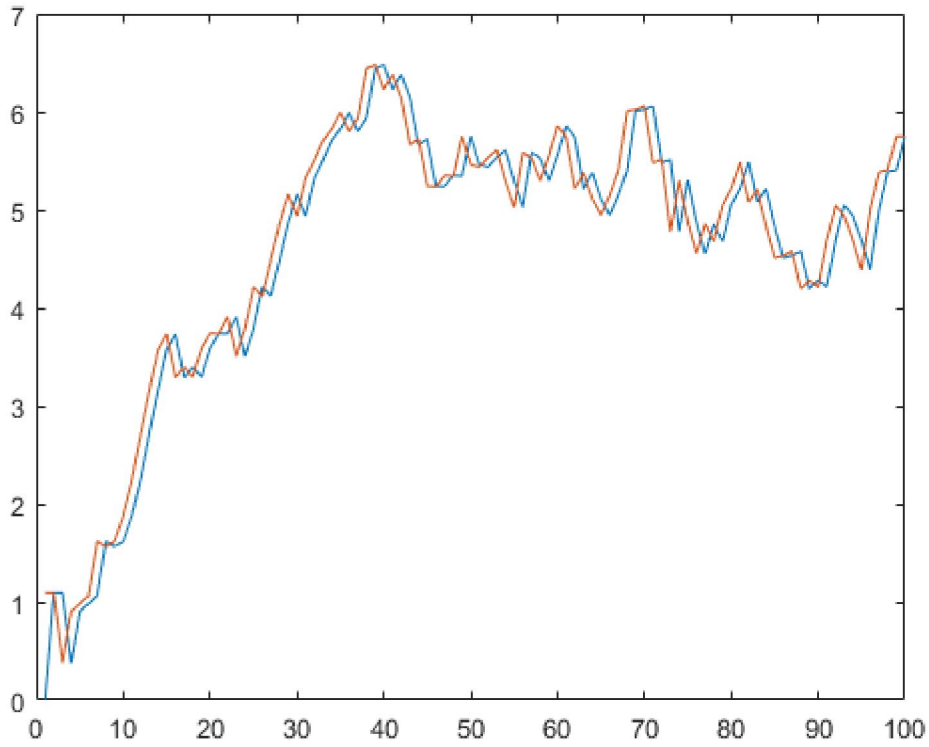
gradescope

```
% Project 4 exercise 1
clear all; clc
% State space
A = [1  1
     0  1];
B = [0  1]';
C = [1  0];
D = 0;
% White noise
g = [0  1]';
wk = [0     0
      0   0.1];

vk = 0.01;

% Random initialization
x_0 = [1    1]';

p_0 = [1    0
       0    1];
% Number of steps
N = 100;
% INITIALIZATIONS
x_est = zeros(2,N);
y_est = zeros(1,N);
x_bar = zeros(2,N);
x_hat = zeros(2,N);
y_hat = zeros(1,N);
y     = zeros(1,N);
p     = zeros(2,2,N);
p_hat = zeros(2,2,N);
meansqrerr = zeros(1,N);
% Set initial values
w  = normrnd(0,sqrt(0.1));
v  = normrnd(0,sqrt(0.01));
x_est(:,1) = x_0;
y_est(:,1) = C * x_est(:,1) + v;
x_bar(:,1) = [0 0]';
x_hat(:,1) = x_0;
p(:,:,1)   = p_0;
p_hat(:,:,1) = p_0;
% Assign initial observer value
L = [1  1]';

% Estimation
for i = 1 : N-1
    % get noise
    w  = normrnd(0,sqrt(0.1));
    v  = normrnd(0,sqrt(0.01));
    % next step
    x_est(:,i+1) = A * x_est(:,i) + B * w;
    y_est(:,i+1) = C * x_est(:,i) + D + v;
```

```
end

% Prediction
for i = 1 : N-1
    y(i) = y_est(i);
    x_hat(:,i) = x_bar(:,i) + L * (y(i) - C * x_bar(:,i));
    p(:,:,i) = (eye(length(L)) - L * C) * squeeze(p_hat(:,:,i));

    x_bar(:,i+1) = A * x_hat(:,i);
    p_hat(:,:,i+1) = A * squeeze(p(:,:,i)) * A' + wk;
    L = squeeze(p_hat(:,:,i+1)) * C' * inv(C * squeeze(p_hat(:,:,i+1))
 * C' + vk);
end

% Update mean-square error
for i = 1 : N - 1
    diff_1 = x_hat(1,i) - x_est(1,i);
    diff_2 = x_hat(2,i) - x_est(2,i);
    meansqrerr(1,i) = sqrt(diff_1^2 + diff_2^2);
end

% Get last prediction
x_hat(:,end) = x_bar(:,end) + L * (y_est(end) - C * x_bar(:,end));

% PLOTS
figure(1)
plot(1:N,meansqrerr);
figure(2)
plot(1:N,x_bar(1,:),1:N,x_hat(1,:));
figure(3)
plot(1:N,x_bar(2,:),1:N,x_hat(2,:));
```


7
6
5
4
3
2
1
0
0
10
20
30
40
50
60
70
80
90
100
1


500
450
400
350
300
250
200
150
100
50
0
0
10
20
30
40
50
60
70
80
90
100



Published with MATLAB® R2020b

1 Exercise 1.1 18 / 20

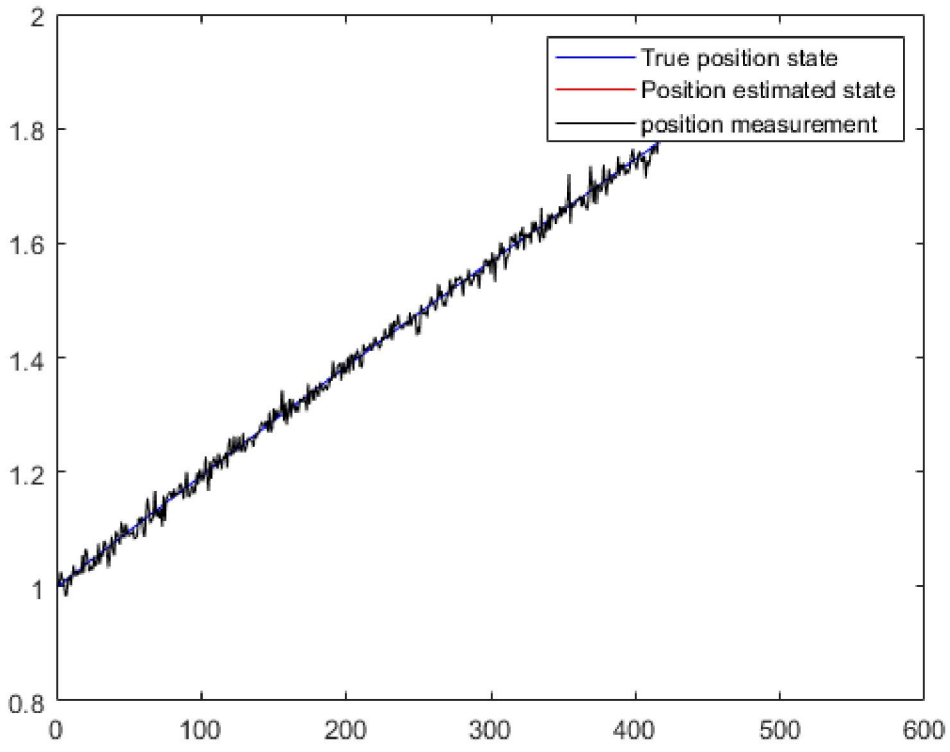
- **0 pts** Correct
- **1 pts** Wrong Kalman Filter initialization
- **1 pts** Wrong MSE definition
- ✓ **- 2 pts** **Unexpected MSE behavior**
- **3 pts** Wrong Kalman Filter process
- **5 pts** No graph on MSE
- **5 pts** No graph on state comparison

1 The error should be a decaying graph, this is increasing?

gradescope

```
% Project 4 exercise 1.2
clear all; clc

% INITIALIZATIONS
T = 0.002;
T_end = 1;
time_step = length(0:0.002:1);
x_est = zeros(2,time_step);
y_est = zeros(1,time_step);
y     = zeros(1,time_step);
x_bar = zeros(2,time_step);
x_hat = zeros(2,time_step);
p     = zeros(2,2,time_step);
p_hat = zeros(2,2,time_step);

anv   = 2.5;
env   = 2 * 10^-4;
u     = 0.1;

% Random Initialization
x_0 = [1    1]';

p_0 = [1    0
       0    1];
% State space
A = [1  T
     0  1];
B = [T^2 / 2    T]';
C = [1  0];
D = 0;

% Noise
wk = eye(2).* B + eye(2).*anv;
vk = env^2 / 12;

% First step

x_est(:,1) = x_0;
y_est(:,1) = C * x_est(:,1) + vk;
x_bar(:,1) = [0 0]';
x_hat(:,1) = x_0;
p(:,:,1)   = p_0;
p_hat(:,:,1) = p_0;

% Initial observer
L = [1  1]';
for i = 1 : time_step-1
    % get noise
    w  = normrnd(0,sqrt(anv));
    v  = normrnd(0,sqrt(env));
    % next step
    x_est(:,i+1) = A * x_est(:,i) + B * w;
```

```
    y_est(:,i+1) = C * x_est(:,i) + D + v;
end

for i = 1 : time_step-1
    y(i) = y_est(i);
    x_hat(:,i) = x_bar(:,i) + L * (y(i) - C * x_bar(:,i));
    p(:,:,i) = (eye(length(L)) - L * C) * squeeze(p_hat(:,:,i));
    x_bar(:,i+1) = A * x_hat(:,i) + B * u;
    p_hat(:,:,i+1) = A * squeeze(p(:,:,i)) * A' + wk;
    L = squeeze(p_hat(:,:,i+1)) * C' * inv(C * squeeze(p_hat(:,:,i+1))
 * C' + vk);
end
% Get last prediction
x_hat(:,end) = x_bar(:,end) + L * (y_est(end) - C * x_bar(:,end));

% PLOT
figure;
plot(1:time_step,x_est(1,:),'b',1:time_step,x_hat(1,:),'r',1:time_step,y_est(1,:),
legend('True position state','Position estimated state','position
 measurement')
```



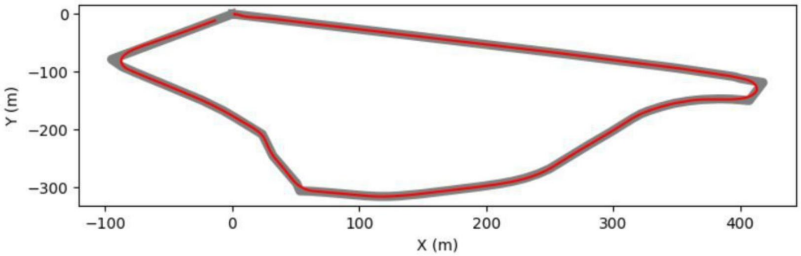
Published with MATLAB® R2020b

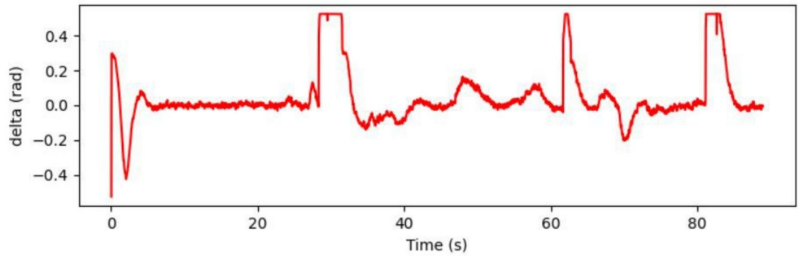
2 Execise 1.2 **20 / 20**

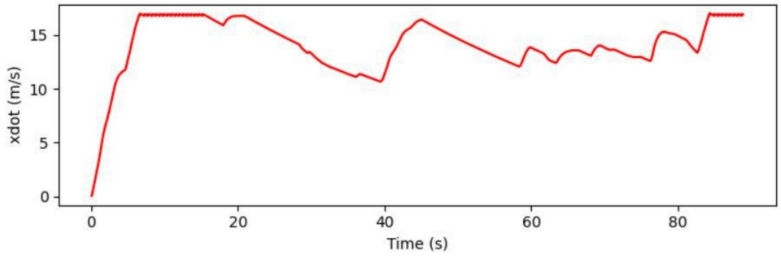
✓ **- 0 pts** **Correct**

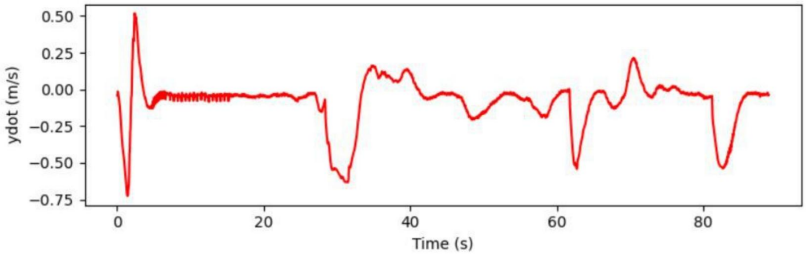
- 2 pts Unexpected pose graph behavior

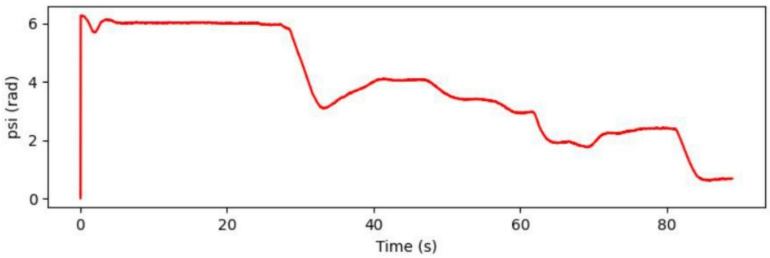
gradescope

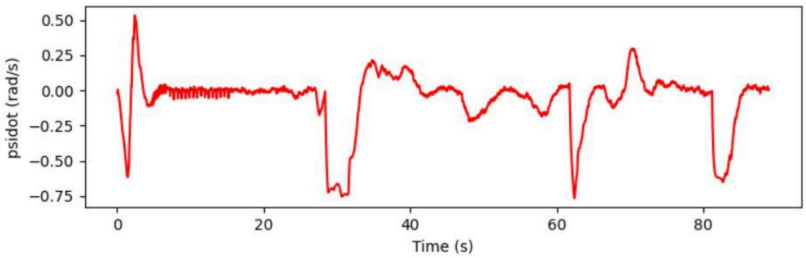

Y (m)
X (m)

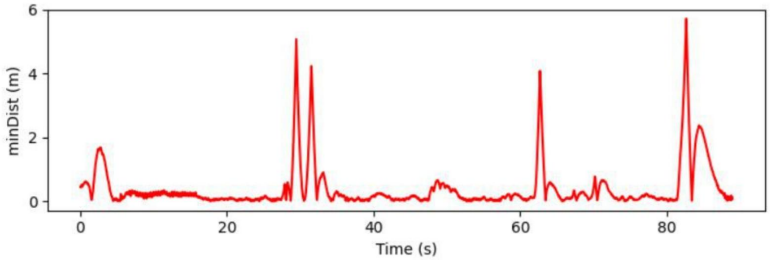

delta (rad)
Time (s)

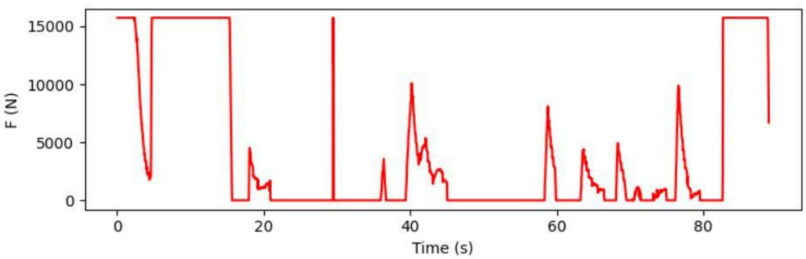

xdot (m/s)
Time (s)


ydot (m/s)
Time (s)


psi (rad)
Time (s)


psidot (rad/s)
Time (s)


minDist (m)
Time (s)


F (N)
Time (s)

3 Result Graphs 10 / 10

✓ - 0 pts Correct

gradescope