Middle East Technical University
Department of Computer Engineering

CENG 382

ANALYSIS OF DYNAMIC SYSTEMS WITH FEEDBACK

Spring 2011

Homework 3

General Rules

- Due date is **30 May 2011, 23:55 (sharp)**.
- Late submission is **not** allowed.
- There is **no** teaming up. The homework has to be done/turned in individually.
- In case of cheating, all parts involved (source(s) and receiver(s)) get zero.
- **Use MATLAB installed on department machines** for programming. A tutorial for MATLAB is available at http://www.mathworks.com/help/pdf_doc/matlab/getstart.pdf.
- Your report must be prepared with a word processor (latex is recommended). Presentation of your reports will affect your grade.
- You are supposed to submit your MATLAB codes and reports (pdf) electronically as one zip file which is named as "studentid.zip" through COW (https://cow.ceng.metu.edu.tr). (The files name should be main.m, particle.m, k_predict.m, k_update.m and hw3.pdf (for report)).
- Also submit hard copy of your report.
- **Warning: Hard copy of report will also be graded.**

Homework

Tracking an object is difficult because of noise. Noise can arise from the movement of the object, called process noise, and from sensors, called measurement noise. Kalman filtering is a well-known method to estimate true state values from noisy data. Kalman filter can be seen as a recursive estimator as it only uses the estimated state from the previous time step and the current measurement to estimate the current state.

Suppose you have a linear system model as described in homework 2. This object (particle) is tracked with a sensor, such as a radar, that only measures the position of the object. By using the measurements, you are asked to estimate the true state of the object with a Kalman filter.

The `particle` function in homework 2 is to be used to create the trajectory data with process noise. The system matrix in homework 2 should also be used in the Kalman filter.

Create two files to be named `k_predict.m` and `k_update.m`. These files must hold the functions whose prototypes are given below:

```
function [Zpredict,Ppredict] = k_predict(Zestimate,A,P,Q)
….
end

function [Zupdate,Pupdate] = k_update(Zpredict,Ppredict,ZN,H,R)
```

```
….
end
```

For the `k_predict` function, $Zestimate$ is the estimated state at time $k-1$, A is the system matrix, P is error covariance matrix at time $k-1$, Q is the covariance matrix of the process noise. The forces in homework 2 should be process noise obtained from the Gaussian distribution with zero mean and covariance Q. $Zpredict$ is predicted state estimate at time k and $Ppredict$ is predicted estimate error covariance at time k.

For the `k_update` function, ZN is the measurement data at time k, H is the measurement model, R is the covariance matrix of measurement noise, $Zupdate$ is updated state estimate at time k, $Pupdate$ is updated estimate error covariance at time k.

Measurement data (at time k) can be obtained as follows:

$$\vec{ZN}(k)_{2\times 1} = H_{2\times 4} * \vec{z}(k)_{4\times 1} + \vec{V}(k)_{2\times 1}$$

where z is state vector, H is the measurement model, V is the measurement noise obtained from the Gaussian distribution with zero mean and covariance R.

Evaluate this system over 100 time steps and estimate the position of the object from the (noisy) measurement data at each time step.

Modify the `main.m` in homework 2 so that it now uses the `particle.m`, `k_predict.m` and `k_update.m`.

Assume that the covariance matrix Q of process noise is the diagonal matrix with 0.02 along the diagonal, the covariance matrix R of measurement noise is the diagonal matrix with 5, T (time interval) is 1, $m = 0.5$. The initial position is given by $x = 0$ and $y = 0$, and initial velocities in both directions are 1. We are assuming proper units.

Plot the x positions of the true state (z), estimated state ($Zestimate$) and the measurement of true state (ZN) at the same figure with different colors over the whole duration. For example:

```
plot(z(1,:),'b')
hold on
plot(Zestimate (1,:),'r')
hold on
plot(ZN (1,:),'g')
```

Plot the y positions of the true state (z), estimated state (Zestimate) and the observation of true state (ZN) at the same figure with different colors over the whole duration. For example:

```
figure
plot(z(2,:),'b')
```

```
hold on
plot(Zestimate (2,:),'r')
hold on
plot(ZN (2,:),'g')
```

In the report describe your system matrix, distribution matrix, input vector, state vector briefly and explain the steps of the Kalman filter clearly. Produce the plots with the given parameters: Two different plots, the first one is for x position and the second one is for y position.