1) **Problem 1 (60 points) – Kalman Filter:** Consider a sensor that is tracking the motion of an object that is a known distance (Do = 100 m) from the sensor. Let θm be the angle (in radians) that is measured by the sensor.

Suppose you know that the position of the object (in meters) is governed by the following equation of motion:

$$X(t+1) = X(t) + 0.2\text{ m} + w(t)$$

where t is measured in seconds, the object is moving with a velocity of 0.2 m/s, and is subject to random impulses (e.g., wind gusts) characterize by w(t) that further change the position of the object (each second), where W(t) is assumed to be normally distributed around 0 with a standard deviation of 0.5 m.

The relationship between the sensor output and the actual position of the object may be characterized as follows:

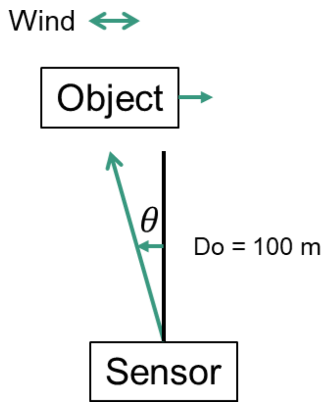
$$\theta m(t) = X(t)/Do + v(t)$$

where v(t) is a random uncertainty in the measurement of θ(t) that is assumed to be normally distributed around 0 with a standard deviation of 0.01 radians.

We want to build a Kalman filter that will best prectict the next location of the object and examine its characteristics with respect to the state estimation error covariance and sensor and process noise properties.

Suppose θm(0) = 0.000 rad.

a) (5) Draw a schematic diagram of the system.



b) (10) Write down the equations that you must use to track this object using a Kalman Filter. Your equations should be a single-state-variable version of the Kalman Filter equations that use all the information provided (i.e., variables and values). Simplify the equations as much as you can.

i) True System State:
   (1) x(k) = x(k-1)+ 0.3*dt + w(k-1)
ii) Measurement of system state:
   (1) θm(k) = x(k)/100 + v(R,t)

iii) Predicted State & Covariance:
   (1) Xe(k|k-1) = Xe(k-1|k-1)
   (2) Pe(k|k-1) = Pe(k-1|k-1) + 0.25
iv) Corrections Equations
   (1) y(k) = θm(k) – Xe(k|k-1)/100 m
   (2) S(k) = Pe(k|k-1)/10^4 + 0.0001
   (3) K(k) = 100*Pe(k|k-1)/(Pe(k|k-1) + 1)
v) Corrected Predicted State & Covariance
   (1) Xe(k|k) = Xe(k|k-1) + K(k) y(k)
   (2) Pe(k|k) = (1 – K(k)/100) Pe(k|k-1)

c) (5) Describe how the Kalman Filter algorithm works (hint: it is an example of a predictor-corrector algorithm).

i) One predicts the next (t+dt) state of the system (and its covariance) based on the previously updated corrected prediction (at t).
ii) One then takes a measurement of the state of the system at time t+dt.
iii) One uses that measurement to update the predicted state of the system at time t+dt.

d) (10) Write and submit a MATLAB program that:

i) Calculates and plots the the estimated a priori error covariances($P_{1|0}$, , $P_{2|1}$, $P_{3|2}$, …), the a posteriori error covariances(P_0, P_1, P_2, P_3, …), and the Kalman gains(K_1, K_2, K_3, …) as a function of time for t = 0 to 50 sec.

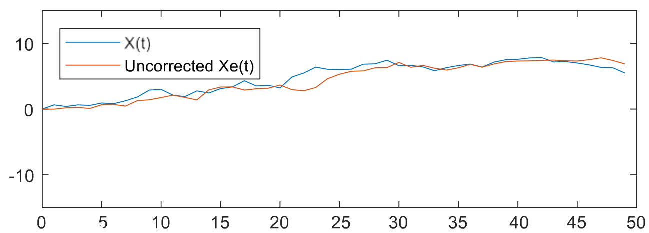
ii) Calculates and plots the actual position and the uncorrected Kalman Filter predicted positon as a function of time for t = 0 to 50 sec.

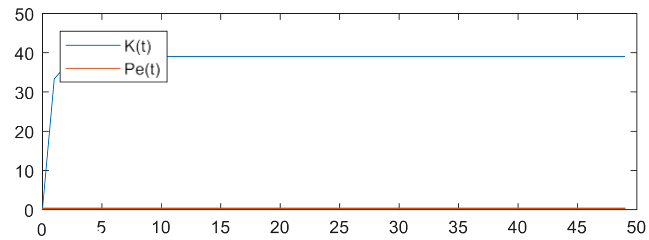
e) (10) Submit the output obtained from your program (pasted into your solutions).

i) Text Output:

X(50) = 5.4684
Xe(50) = 6.8844
K(50) = 39.0388
Pe(50) = 0.3904
>>

ii) Graphical Output:





f) (5) Interpret K(t) in terms of how it affects your (updated) estimate of x(t) as t goes to 50.

i) K indicates the degree to which one uses the most recent measurement to update the state estimate for the next time period. It rises very quickly from 0 to about 40 after about 4 time steps.

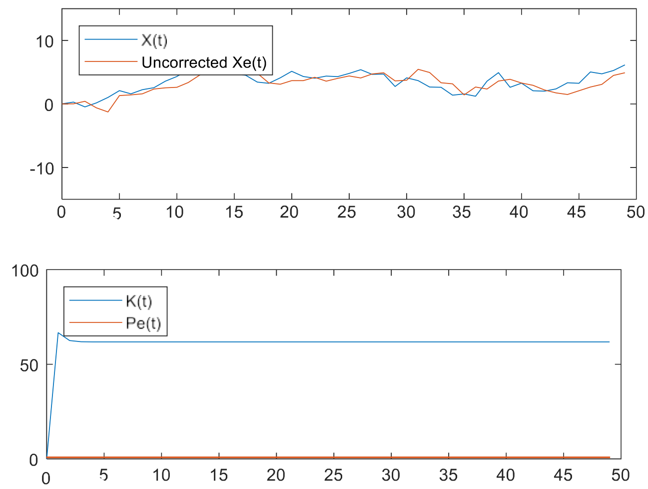
g) (10) Run your program for the following cases and provide the results below.

i) The standard deviation in the process noise is twice as large (1 m).

(1) Text Output:

X(50) = 6.123
Xe(50) = 4.884
K(50) = 61.80
Pe(50) = 0.618
>>

(2) Graphical Output:

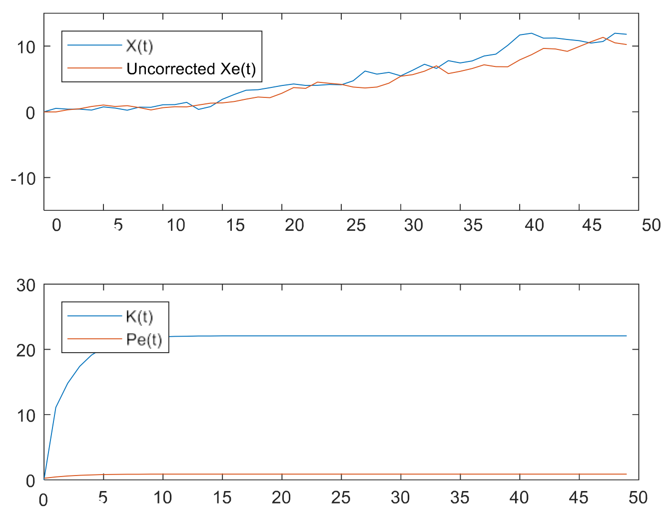


ii) The standard deviation in sensor noise is twice as large (0.02 rad).

(1) Text Output:

X(50) = 11.79
Xe(50) = 10.25
K(50) = 22.07
Pe(50) = 0.883
>>

(2) Graphical Output:



h) (5) Discuss how the process noise variance and the sensor noise variance affect the value of K(t).

i) Increasing process noise variance increases K.

ii) Increasing sensor noise variance decreases K.