Object Tracking using Kalman and Particle filtering Techniques

Kodali Sai Krishna
ROLL NO: 213EC6261

Department of Electronics and Communication Engineering

National Institute of Technology Rourkela

Rourkela, Odisha, 769008, India

2015

Object Tracking using Kalman and Particle filtering Techniques

A thesis submitted in partial fulfillment of the requirement for the degree of

Master of Technology
In
Electronics and Communication Engineering
Specialization: Signal and Image Processing

By

Kodali Sai Krishna
Roll No. 213EC6261

Under the Supervision of
Dr. A.K.SAHOO

Department of Electronics and Communication Engineering

National Institute of Technology Rourkela

Rourkela, Odisha, 769008, India

2015

DEPARTMENT OF ELECRTONICS AND COMMUNICATION
NATIONAL INSTITUTE OF TECHNOLOGY,
ROURKELA, ODISHA -769008.

DECLARATION

I certify that,

a. The work presented in this thesis is an original content of the research done by myself under the general supervision of my supervisor.
b. The work has not been submitted to any other institute for any degree or diploma.
c. The data used in this work is taken from free source and its credit has been cited in reference.
d. The materials (data, theoretical analysis and text) used for this work has been given credit by citing them in the text of thesis and their details in the references.
e. I have followed the thesis guidelines provided by the institution

KODALI SAI KRISHNA

DEPARTMENT OF ELECRTONICS AND COMMUNICATION
NATIONAL INSTITUTE OF TECHNOLOGY,
ROURKELA, ODISHA -769008.

CERTIFICATE

This is to certify that the work done in the report entitled "**Object Tracking using Kalman and Particle filtering Techniques"** by "**KODALI SAI KRISHNA"** is a record of research work carried out by him in National Institute of Technology, Rourkela under my supervision and guidance during 2014-15 in partial fulfillment of the requirement for the award of degree in **Master of Technology** in **Electronics and Communication Engineering (Signal and Image Processing),** National Institute of Technology, Rourkela. To the best of my knowledge, this thesis has not been submitted for any degree or diploma.

Date

Dr. A.K.SAHOO

ACKNOWLEDGEMENT

This research work is one of the significant achievements in my life and is made possible because of the unending encouragement and motivation given by so many in every part of my life. It is immense pleasure to have this opportunity to express my gratitude and regards to them.

Firstly, I would like to express my gratitude and sincere thanks to **Dr. A.K.SAHOO**, my supervisor, Department of Electronics and Communication Engineering for his esteemed supervision and guidance during the tenure of my project work. His valuable advices have motivated me a lot when I feel saturated in my work. His impartial feedback in every walk of the research has made me to approach a right way in excelling the work. It would also like to thank him for providing best facilities in the department.

I am very much indebted to Prof. S. K. Patra and Prof. K. K. Mohapatra for teaching me subjects that proved to be very helpful in my work. My special thanks go to Prof. S. Meher, Prof. L.P. Roy and Prof. S. Ari for contributing towards enhancing the quality of the work and eventually shaping my thesis.

Lastly, I would like to express my love and heartfelt respect to my parents, sister and brothers for their consistent support, encouragement in every walk of my life without whom I would be nothing.

Kodali Sai Krishna

ABSTRACT

Object tracking has been an active field of research in the past decade. There are many challenges in tracking the object understand in which kind of system model it is moving and which type of noise it is taking . System model can be either linear or nonlinear or coordinated turn model and such noise can be either Gaussian or non-Gaussian depending on the type of filter chosen, Extended Kalman filter (EKF) is widely used for tracking moving objects like missiles, aircrafts and robots. Here analyse the instance of a single sensor or observer bearing only tracking (BOT) problem for two different models. In model 1, the target is assumed to have a constant velocity and constant course. In model 2, the target is assumed to follow a coordinated turn model with constant velocity but varying course. Extended Kalman Filter is used to track the target in both cases.

For some application part, it is getting to be necessary to include components of nonlinearity and non-Gaussianity with a specific end goal to model exactly the essential dynamics of a system. The nonlinear extended Kalman filter (EKF) and the particle filter (PF) algorithms are used and compared the manoeuvring object tracking with bearing-only measurements. We also propose a proficient system for resampling particles to decrease the impact of degeneracy effect of particle propagation in the particle filter (PF) algorithm. One of the particle filter (PF) technique is sequential importance resampling (SIR). It is discussed and compared with the standard EKF through an illustrative example.

Keywords: Bearing Only Tracking (BOT), Extended Kalman filter (EKF), Particle Filter (PF), Sequential Importance Resampling (SIR).

Table of Contents

List of Figures

List of acronyms

BOT	Bearing Only Tracking
EKF	Extended Kalman Filter
PF	Particle Filter
SIR	Sequential Importance Resampling
IMM	Inter active Multiple Model
GHKF	Gauss-Hermite Kalman Filter
SIS	Sequential Importance Sampling
N_{eff}	Effective Sample Size

CHAPTER1
INTRODUCTION

Chapter1

Introduction

1.1 Tracking:

Tracking means finding the position of object with respect to time, in general while tracking the object first step is detection of the object, next tracking the object and finally observing its behaviour with respect to time. Object tracking is required in various tasks in real life, such as surveillance, video compression and video analysis. While tracking the object trajectory there exist certain problems those will be solved by different tracking techniques.

1.2 Filter

The term filter means it accepts certain type of data as input and processes it, transformed and then outputs the transformed data. The term filter comes in tracking the object trajectory is due to while tracking the object path there exists certain uncertainties comes into picture those are noise, occlusions, appearance changes. One of the technique to remove the existed noise is filtering. The term filtering fundamentally implies the methodology of separating out the noises in estimations and giving an ideal assessment to the state. Filter in generally removes those existed noises. Here filtering term is used while tracking the object so that it is called tracking with filters.

1.2.1 Adaptive Filter

The term Adaptive represents iterative in nature, in general adaptive filter means it models computation between two devices in iterative manner. In adaptive algorithm it portrays how the parameters are adjusted from one time moment to next time moment. In general in adaptive filter it takes input signal samples x(n) and process it and gives the output signal samples let us consider these output samples as y(n) in adaptive filter these output samples are varied from one time moment to next time moment by adjusting and taking certain

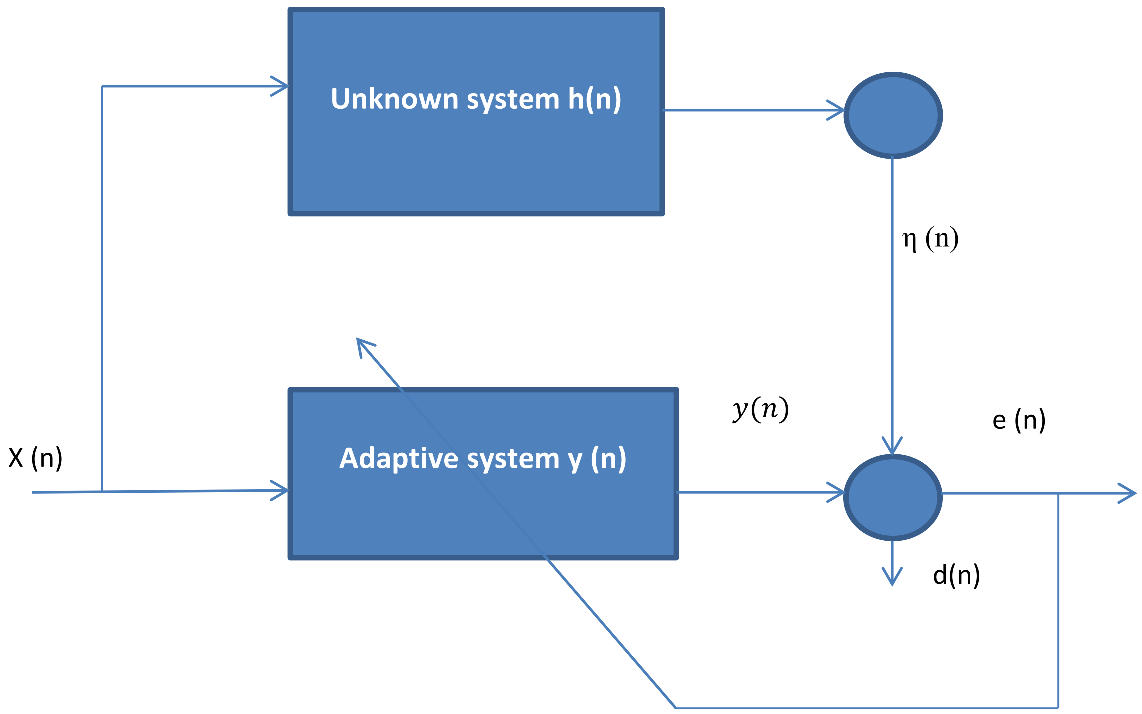

Fig1.1 GENERAL ADAPTIVE MODEL

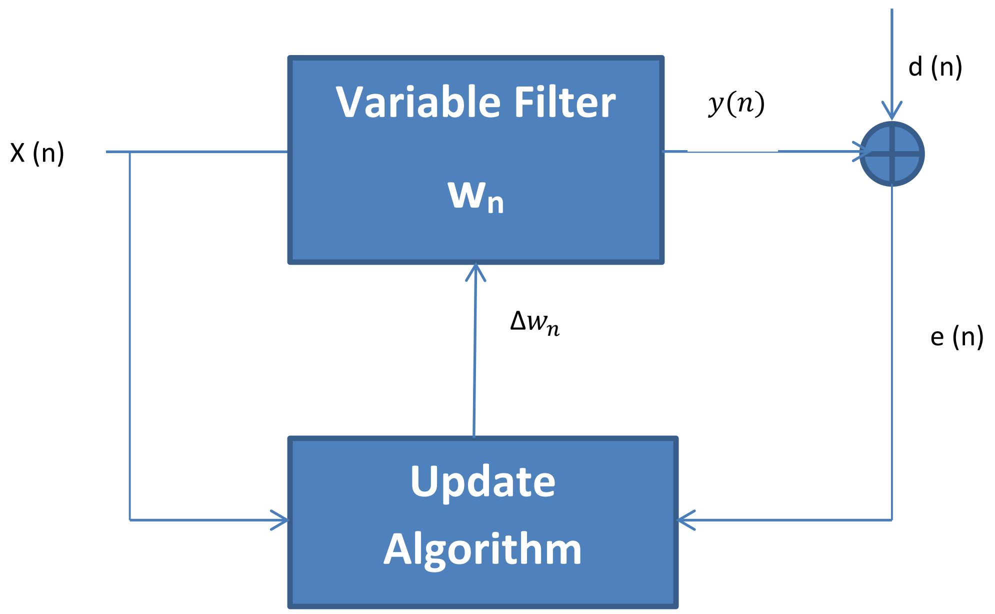

FIGURE 1.2 ADAPTIVE FILTER

Parameters in to consideration. Here d (n) is desired output signal and comparing of the output samples with desired signal samples gives the error signal e (n), here error signal is fed into procedure which adjusts the parameters of filter from one time step to another time step by the way as the time progresses the output signal sample values approached to the desired sample values and the error will be minimized and accuracy will be increased.

In adaptive filter there exists a closed loop adaptive filter and kernel adaptive filter .Closed loop adaptive filter is also called variable filter which means adjusting the certain parameters repeatedly up to error will be minimized. Kernel adaptive filter is a one of the nonlinear filtering technique.

With this there existed different adaptive filer tracking algorithms

Different algorithms which are using widely to estimate the object path are kalman filter, extended kalman filter, Unscented Kalman Filter, Particle Filter and Inter active Multiple Model (IMM) Filter, Gauss-Hermite Kalman Filter (GHKF).

From this using filtering techniques to remove the noise and estimate the optimal solution to unknown state or variables. In general state means position, velocity, acceleration and coordinated turn rate. Position means it may be either x coordinate or y coordinate, here tracking in two dimensions means how the object trajectory varies from one time instant to another time instant with respect to these two coordinates by taking into consideration certain control inputs, acceleration of the moving body with respect to these coordinate axes .

1.3 Different system models

There exists different models but some models which used widely are discussing here those are

- Constant velocity model
- Constant acceleration model
- Coordinated turn model

In constant velocity model, the object is moving with constant velocity it means rate of change in velocity which is nothing but acceleration is almost zero. It means acceleration terms are not taking into consideration with these kind of models for each and every time step.

The values x and y position for present time step are corresponding previous time step x and y positions plus velocity of the object in that particular direction. In constant acceleration model object is moving with constant acceleration here constant acceleration term is included for each and every time instant while updating the positions of x and y coordinates.

In coordinated turn model it includes one more new term that is coordinated turn rate which is also called as angular turn rate measured in radians per second here in this model, it takes velocity term but it can't take the acceleration term while updating the x and y coordinates, it takes the newly included term coordinated turn rate which is denoted by w in general, this term also has been included while updating the x and y coordinates.

In constant velocity model for present x or y position it wants previous x or y position and its corresponding x or y position velocity it means it is looking like a linear model so constant velocity model is also called linear model.

In constant acceleration model while measuring present x or y positions it wants previous x or y positions and corresponding velocities and corresponding acceleration terms are included. Here because of only one term this model seems to be not a linear model that term is acceleration term but any way this is not that much of nonlinear in nature. In coordinated turn model with that constant velocity term for each and every coordinate updating coordinated turn rate also has been included because of this term this model is seems to be nonlinear model.

1.4 Motivation:

The main theme of object tracking is based on the tracking the object path in the unknown environment. Tracking the object path within its environment is a crucial task and has been extensively explored in many research papers. Tracking the object with different nonlinear models has been studied in many other papers. There is need for highly accurate or optimal filtering techniques to track the object path. While tracking the object path it will be affected by different obstacles those are occlusions, noises and scene changes. These noise terms will be removed by using different adaptive filtering techniques. Hence to address these problems a robust and simple,yet efficient object tracking technique is necessary. The typical models discussed below as described in gives us an idea of the various steps involved in tracking the object trajectory process.

1.5 Problem Statement

In light of all the obstacles plaguing the object tracking process a need for an accurate and estimation system is required. Hence the problem at hand is to track the object trajectory as the object moves along the unknown environment. To track the object trajectory bearing measurements is also essential. Adaptive filter algorithms estimate the unknown position but there existed a lot of disturbances while estimating the states of the object. An advanced algorithm estimation approach is to be taken in which gradually update the particle positions with respect to its initial location and update the particle weights with respect to system dynamics.

1.6 Organization of Thesis:

This thesis consists of a total of six chapters organized as below

- Chapter 1. This chapter gives a brief introduction to the object tacking that is detecting tracking the object trajectory which is varying with respect to time, and the different system models in which object is moving and then little bit discussion about the filtering, and adaptive filtering algorithms.
- Chapter 2. This chapter will give description to the kalman filter, how the kalman filter works, and the models for which it works well and how it tracks the object trajectory in unknown environment and the steps which this kalman filter algorithm takes to give best solution for object path tracking.
- Chapter 3. This chapter will give detailed description to the extended kalman filter, how it over comes the disadvantages in kalman filter and for the models which it gives best estimation, and how the jacobian and hessian matrices approximates the nonlinearity and the algorithm steps which it takes to track the nonlinear object path effectively.
- Chapter 4. This chapter will discuss the complete description to the particle filter, here we also discussed the disadvantages existed in extended kalman filter, how these will be solved

in particle filter, compared it with discrete kalman filter and the basic steps to implement the particle filter algorithm.

- Chapter 5. The results of the trajectory paths obtained by applying the different filtering techniques to different system models are presented in this chapter. Evaluation and analysis of these results is also presented in this chapter.
- Chapter 6. This chapter will discuss Conclusion and also future improvements

CHAPTER2
KALMAN FILTER

Chapter 2

KALMAN FILTER

2.1 Introduction:

The Kalman filter was at first introduced by Rudolph E. Kalman in his innovative Paper (Kalman, 1960). Kalman filter is also called as linear quadratic estimator, it estimates the unknown variables by taking series of measurements which contains noise and other inaccuracies and estimates the unknown variables. Kalman filter is also called as linear state space estimator which uses linear state and measurement equations. Kalman filter operates recursively on stream of noisy input data to find the optimal estimate of unknown state.

2.2 Linear recursive estimator

Kalman filter operates by taking linear models and noise is Gaussian noise then it gives better results by minimizing the mean square error. One most praised apparatuses in straight minimum mean-squares estimation hypothesis, to be specific the Kalman channel. The channel has a personal connection with versatile channel hypothesis, to such an extent that a strong comprehension of its usefulness can recommend expansions of traditional versatile plans. After we have advanced adequately enough in our treatment of versatile channels. At that stage, we might tie up the Kalman channel with versatile slightest squares hypothesis and show how it can rouse helpful augmentations.

Kalman filter is a recursive estimator it means that while estimating the current state it requires previous state and its current measurements. These two are sufficient to estimate the current state.

Kalman filter averages the prediction of system state with new measurement by using weighted average phenomenon. The reason behind the weighted average is to estimate the state with lesser uncertainty that is for more accurate estimated values. The estimated value through weighted average lies between predicted and measurement values. With the new gauge and its covariance advising the expectation utilized as a part of the taking after cycle. The Kalman channel is a proficient strategy for deciding the advancements, when the perception process emerges from a limited dimensional direct state-space model, Kalman filter uses system model and control inputs to that corresponding model and it takes multiple series measurements in to consideration to estimate the unknown variables, which are varying continuously with time. This gives better estimate compared to the single measurement alone.

Kalman filter gives a recursive solution to discrete data linear filtering issues. Kalman filter is the subject of broad examination and application, especially in the zone of self-sufficient or aided route. It can estimate the state of the system very accurately when the system is unknown also and so it is very powerful linear recursive estimator of unknown state. Kalman filter is basically an ideal recursive information handling algorithm that mixes all accessible data measurement outputs and it has the knowledge about the system model, measurement sensors and then it can estimate the state of unknown variables by the way mean square error will be minimized.

Kalman filter is one of simplest linear state space model used widely in present days if the known system and measurement models are linear. While estimating the unknown state variables recursively with time there is certain uncertainty in measurement values. Uncertainty is that process noise and measurement noise included in the measured values and that noise included must be Gaussian in nature while kalman filter is using to estimate the unknown state variables. From this we can’t model the system entirely deterministically. Kalman filter uses linear equation systems with white Gaussian noises as standard model.

2.3 Basic Kalman filter model

Kalman filter uses two steps while estimating the unknown states.

- Prediction step
- Updating step

2.3.1 Prediction:

Forecast type of the Kalman channel depends on propagation of the one-stage expectation. One more implementation of kalman filter also known as time and measurement update form. In Prediction step current time state estimate is measured from previous time step. This estimate is also known as a priori state estimate.it does not require any measurement (observation) values.

Prediction step equations are written as

$$\hat{x}(t \mid t-1) = A(t-1).\hat{x}(t-1 \mid t-1) + B(t-1)u(t-1) + R_{vv}(t-1) \tag{2.1}$$

$$\tilde{p}(t \mid t-1) = A(t-1).\tilde{p}(t-1 \mid t-1)A'(t-1) + R_{ww}(t-1) \tag{2.2}$$

Here prediction measurements are time update equations that above two equations are a priori state estimate and a priori error covariance estimates respectively.

2.3.2 Update:

In update step, by taking measurement (observation) value in to consideration updating the apriori state estimate to improve accuracy of estimated state. The estimate after including observation value is called a posterior state estimate.

Update step equations are written as

$$e(t) = y(t) - \hat{y}(t \mid t-1) \tag{2.3}$$

$$R_{ee}(t) = C(t).\tilde{p}(t \mid t-1)C'(t) + R_{vv}(t) \tag{2.4}$$

$$K(t) = \tilde{p}(t \mid t-1)C'(t)R_{ee}^{-1}(t) \tag{2.5}$$

$$\hat{x}(t \mid t) = \hat{x}(t \mid t-1) + K(t).e(t) \tag{2.6}$$

$$\tilde{p}(t \mid t) = [I - K(t)C(t)]\tilde{p}(t \mid t-1) \tag{2.7}$$

Here $\hat{x}(t \mid t-1)$ and $\tilde{p}(t \mid t-1)$ are predicted state and covariance of the state without knowing the value of measurement value.

Whereas $\hat{x}(t \mid t)$ $\tilde{p}(t \mid t)$ are the estimated state and covariance values after knowing the measurement value.

$k(t)$ Is the Kalman gain at time step t.

$e(t)$ Is the residual measurement error on time step t.

$R_{ee}(t)$ Is prediction covariance by including measurement value at time step t.

From the above equations predicted, estimated state and covariance matrices, are independent of measurement values at any time. A, B, C and R known matrices at each and every time interval. It is known and understood that the Kalman filter has been generally utilized for parameter estimation in a state-space model. Since it is the ideally recursive linear filter in the Gaussian noise distribution environment.

CHAPTER3
EXTENDED KALMAN FILTER

Chapter3

EXTENDED KALMAN FILTER

3.1 Introduction

The extended Kalman filter stretches out the scope of Kalman filter to nonlinear ideal separating problems by shaping a Gaussian rough guess to the joint circulation of state x and estimations y utilizing a Taylor arrangement based transformation. Here First and second order extended kalman filters are introduced. To examine and make induction about an element state no less than two models are obliged first, a model portraying the advancement of the state with time what's more, second, a model relating the noisy estimations to the state (the estimation model). We will accept that these models are accessible in a probabilistic structure. The probabilistic state-space plan and the prerequisite for the upgrading of data on receipt of new estimations are in a perfect world suited for the Bayesian approach. This gives a thorough general system for element state estimation issues.

The Bayesian way to deal with element state estimation, one endeavours to build the back probability likelihood of the state based on all accessible data, including the arrangement of observed estimations. Since this work encapsulates all available measurable data, it might be said to be the complete solution to the estimation issue. On a fundamental level evaluation of the state may be obtained from the probability distribution function. A measure of the exactness of the estimate may likewise be acquired. For some issues, an assessment is required every time that estimation is obtained. For this situation, a recursive filter is an advantageous solution. A recursive sifting approach means that got information can be prepared sequentially rather than as a cluster, so it is not important to store the complete data set or to reprocess existing information if another measurement becomes available. Such a channel comprises of essentially two stages: forecast and upgrade. The forecast stage utilizes the system model to anticipate the state pdf forward starting with one measurement time then onto the next. Since the state is typically subject to unknown noise. Prediction generally interprets, twists, and spreads the state pdf. The update operation utilizes the most recent estimation to alter the prediction pdf. This is accomplished utilizing Bayes hypothesis, which is the mechanism for overhauling learning about the target state in the additional data from new information.

In many cases dynamic systems are not linear by nature, so the conventional kalman filter can't be used in evaluating the condition of such a system. In these kind of systems either measurement model or motion are nonlinear or any one of these motion or measurement model may be non-linear. So we portray some augmentations to the kalman filter those are extended kalman filter, unscented kalman filter, particle filter, gauss newton filter and Interactive Multi Model (IMM) filter. Out of these extended kalman filter can be evaluate the nonlinear dynamic systems by shaping Gaussian estimates to the joint distribution of the state x and estimation y. To start with the Extended Kalman filter (EKF), which is taking into account Taylor series arrangement estimate of the joint distribution, and after that the Unscented Kalman filter is take in to consideration in this unscented kalman filter, it takes sigma points into consideration while estimating the unknown variables. The extended Kalman filter broadens the extent of Kalman channel to nonlinear ideal separating issues by shaping a Gaussian estimate to the joint distribution of state x and estimations y utilizing a Taylor arrangement based change. If the non-linearity is up to certain extent extended kalman filter works fine. If the non-linearity is more and more and joint distribution for the state and measurement models is non Gaussian it can't work well or it can't track the object path correctly. It means that extended kalman filter is not the best estimate if the joint probability distribution for those state and measurement models is non Gaussian and system dynamics with more and more linearity, so we proceeded to one more filter to optimizing the filtering estimate that is to increase the accuracy and to get best estimation of object trajectory. One filter which satisfies above criteria very well and gives optimal solution to object path tracking is particle filter. Particle filter is one of the best solutions of object tracking in signal domain if the system dynamics that may be either system model or measurement models, are nonlinear or joint probability distribution for these two models is non Gaussian then gives results very accurately that is it tracks the object trajectory very well. One example of such nonlinearity case is bearing only object tracking it used is widely in lot of fields. In this there exists certain kind of target which contains more nonlinearity is maneuvering target means that it has abrupt change in its state by sudden using of acceleration parameter

3.2 Extended kalman filter

The filtering model used in the extended kalman filter is

$$X_{k+1} = f(X_k, k) + w_k \tag{3.1}$$

$$z_k = h(X_k, k) + v_k \tag{3.2}$$

Here X_k denotes the state vector at k^{th} time step

w_k is the process noise vector

v_k is measurement noise vector

$f(X_k, k)$ is nonlinear system state transition function which transits state from one time instant to another time instant

$h(X_k, k)$ is measurement function

z_k is measurement state vector at k^{th} time instant

3.2.1 Ekf algorithm

Similar to kalman filter extended kalman filter also has two steps to estimate the unknown state or variables. Those two steps are

- Prediction
- Updating

In prediction step prediction of new state vector and its corresponding covariance will be calculated for certain time interval by using its dynamic system and measurement models respectively. After prediction residual measurement error and residual error covariance matrices are calculated and these terms will be used in state and covariance updating, and to evaluate the kalman gain.

After predicting the state vector and covariance matrices, then next step is updating these state vector and covariance matrices, then calculating the kalman gain matrix and after getting this kalman gain final estimation of state vector and covariance matrices will be done by using this kalman gain term[8],[12].

Prediction:

$$X_k = f(X_{k-1}, u_{k-1,} 0) \tag{3.3}$$

$$P_k = A_k P_{k-1} A_k^T + W_k Q_{K-1} W_k^T \tag{3.4}$$

Kalman gain:

$$K_k = P_k H_k^T (H_k P_k H_k^T + v_k R_k v_k^T)^{-1} \tag{3.5}$$

Update

$$e(k) = (Z_k - h(X_k, 0) \tag{3.6}$$

$$e_c(k) = (H_k P_k H_k^T + v_k R_k v_k^T)^{-1} \tag{3.7}$$

$$P1_k = (I - K_k H_k)\, P_k \tag{3.8}$$

$$X1_k = X_k + K_k (Z_k - h(X_k, 0)) \tag{3.9}$$

Where $H_k = \frac{\partial}{\partial x} h(X_{k,} k)$ (3.10)

Here H_k is jacobian matrix

A_k Is jacobian for state transition matrix

$e(k)$ is measurement error vector

$e_c(k)$ is measurement error covariance

Q_{K-1} is process noise covariance matrix

R_k is measurement noise covariance matrix

3.3 EKF WITH SINGLE SENSOR CASE

For the extended kalman filter if any one of the system dynamic model or observation model is nonlinear, then kalman filter can't give good results or can't track the object trajectory correctly. While tracking a particular moving object which contains sensors around it in a particular area, sensors gives bearing angle to that object by taking reference of object positions with those sensor ordinates or coordinators and then gives the bearing angles to that object in general sensors has fixed ordinates then these sensors will give bearing angles depending on how the object is moving in particular area as time goes on increasing from one time step to next time step.

Here in bearing only tracking nonlinearity is existed in measurement model and considers for example sensor is located at origin and it is fixed throughout experiment then bearing angle is explored in [3],[1].

$$h_k = \tan^{-1}(\frac{y_k}{x_k}) \qquad (3.11)$$

But above sensor is stationary and it has its own ordinates in x and y directions so the bearing angle in single sensor case will be changed to is explored in [2].

$$h1_k = \tan^{-1}(\frac{y_k - y_s}{x_k - x_s}) \qquad (3.12)$$

But extended kalman filter is Taylor series approximation of joint distribution. In this using of Taylor series approximation to nonlinear observation model by calculating Jacobian and hessian matrices depending on order of the filter respectively if it is first order then evaluate the jacobian matrix, if it is second order extended kalman filter then evaluate the hessian matrix for it.

The major difference between extended kalman and traditional kalman filter is that A_K and H_K are replaced by jacobian matrices in extended kalman filter to approximate the nonlinearity by using Taylor series nonlinearity approximation method.

From the bearing angle $h1_k$ the measurement model is written as below

Observation model is

$$y_k = \tan^{-1}\left(\frac{y_k - y_s}{x_k - x_s}\right) + v_k \tag{3.13}$$

Jacobian of this model is evaluated by calculating the partial derivative of $h1_k$ with respect to state variables.

Here

$$\frac{\partial h1_k}{\partial x_t} = \frac{-(y_k - y_s)}{(y_k - y_s)^2 + (x_k - x_s)^2} \tag{3.14}$$

$$\frac{\partial h1_k}{\partial y_t} = \frac{(x_k - x_s)}{(y_k - y_s)^2 + (x_k - x_s)^2} \tag{3.15}$$

$$\frac{\partial h1_k}{\partial \dot{x}_t} = 0, \frac{\partial h1_k}{\partial \dot{y}_t} = 0 \tag{3.16}$$

Hence the jacobian matrix is written as like below for this single sensor case where sensor location is fixed.

$$H_K = \begin{bmatrix} \frac{-(y_k - y_s)}{(y_k - y_s)^2 + (x_k - x_s)^2} & \frac{(x_k - x_s)}{(y_k - y_s)^2 + (x_k - x_s)^2} & 0 & 0 \end{bmatrix} \tag{3.17}$$

3.4 EKF WITH MULTI SENSOR CASE

Above discussed case is for single sensor case. But in recent days multi sensor cases are using very widely to increase tracking accuracy. There is not that much of difference from single sensor to multi sensor case. In single sensor case only one bearing angle will be acquired. In multi sensor cases bearing angles will be equal to number of sensor it means each sensor is giving bearing angle by using its coordinates with respect to the object to be tracked [13].

For mathematical analysis and its understanding purpose bearing angle for multi sensor case is denoted as which is explored in [11].

$$h1_k^n = \tan^{-1}\left(\frac{y_k - y_s^n}{x_k - x_s^n}\right) \tag{3.18}$$

Where n denotes the number of sensors {n=1, 2, 3 ….N

Observation model for this multi sensor case is denoted in [1] as

$$y_k^n = \tan^{-1}\left(\frac{y_k - y_s^n}{x_k - x_s^n}\right) + v_k^n \tag{3.19}$$

Here (x_s^n, y_s^n) denotes x and y coordinates of sensor and v_k^n is measurement noise for corresponding sensor

3.4.1 First order Ekf

For better understanding purpose consider n=4

For this jacobian matrix will be 4×4 matrix.

Jacobian matrix for this 4 sensor case is

$$H_K = \begin{bmatrix} \frac{-(y_k - y_s^1)}{(y_k - y_s^1)^2 + (x_k - x_s^1)^2} & \frac{(x_k - x_s^1)}{(y_k - y_s^1)^2 + (x_k - x_s^1)^2} & 0 & 0 \\ \frac{-(y_k - y_s^2)}{(y_k - y_s^2)^2 + (x_k - x_s^2)^2} & \frac{(x_k - x_s^2)}{(y_k - y_s^2)^2 + (x_k - x_s^2)^2} & 0 & 0 \\ \frac{-(y_k - y_s^3)}{(y_k - y_s^3)^2 + (x_k - x_s^3)^2} & \frac{(x_k - x_s^3)}{(y_k - y_s^3)^2 + (x_k - x_s^3)^2} & 0 & 0 \\ \frac{-(y_k - y_s^4)}{(y_k - y_s^4)^2 + (x_k - x_s^4)^2} & \frac{(x_k - x_s^4)}{(y_k - y_s^4)^2 + (x_k - x_s^4)^2} & 0 & 0 \end{bmatrix} \tag{3.20}$$

3.4.1 Second order Ekf

Finding of second order partial derivatives to calculate the hessian matrix for second order extended kalman filter is explored in [13].

But here hessian matrix is giving for only two sensor case

Hence

$$\frac{\partial^2 h1_k^n}{\partial x_k \partial x_k} = \frac{-2(x_k - x_s^n)}{((y_k - y_s^n)^2 + (x_k - x_s^n)^2)^2} \tag{3.21}$$

$$\frac{\partial^2 h1_k^n}{\partial x_k \partial y_k} = \frac{(y_k - y_s^n)^2 - (x_k - x_s^n)^2}{((y_k - y_s^n)^2 + (x_k - x_s^n)^2)^2} \tag{3.22}$$

$$\frac{\partial^2 h1_k^n}{\partial y_k \partial y_k} = \frac{-2(y_k - y_s^n)}{((y_k - y_s^n)^2 + (x_k - x_s^n)^2)^2} \tag{3.23}$$

$$\frac{\partial^2 h1_k^n}{\partial y_k \partial x_k} = \frac{(y_k - y_s^n)^2 - (x_k - x_s^n)^2}{((y_k - y_s^n)^2 + (x_k - x_s^n)^2)^2} \tag{3.24}$$

$$\frac{\partial^2 h1_k^n}{\partial x_k \partial \dot{x}_k} = 0, \frac{\partial^2 h1_k^n}{\partial x_k \partial \dot{y}_k} = 0,\ \frac{\partial^2 h1_k^n}{\partial y_k \partial \dot{y}_k} = 0, \frac{\partial^2 h1_k^n}{\partial \dot{x}_k \partial x_k} = 0 \tag{3.25}$$

$$\frac{\partial^2 h1_k^n}{\partial \dot{x}_k \partial y_k} = 0, \frac{\partial^2 h1_k^n}{\partial \dot{x}_k \partial \dot{x}_k} = 0, \frac{\partial^2 h1_k^n}{\partial \dot{x}_k \partial \dot{y}_k} = 0, \frac{\partial^2 h1_k^n}{\partial \dot{y}_k \partial x_k} = 0 \tag{3.26}$$

$$\frac{\partial^2 h1_k^n}{\partial \dot{y}_k \partial y_k} = 0, \frac{\partial^2 h1_k^n}{\partial \dot{y}_k \partial \dot{x}_k} = 0, \frac{\partial^2 h1_k^n}{\partial \dot{y}_k \partial \dot{y}_k} = 0 \tag{3.27}$$

Hence from the above values hessian matrix for the second order extended kalman filter for multi sensor case is

$$H_{kk} = \begin{bmatrix} \frac{-2(x_k - x_s^n)}{((y_k - y_s^n)^2 + (x_k - x_s^n)^2)^2} & \frac{(y_k - y_s^n)^2 - (x_k - x_s^n)^2}{((y_k - y_s^n)^2 + (x_k - x_s^n)^2)^2} & 0 & 0 \\ \frac{(y_k - y_s^n)^2 - (x_k - x_s^n)^2}{((y_k - y_s^n)^2 + (x_k - x_s^n)^2)^2} & \frac{-2(y_k - y_s^n)}{((y_k - y_s^n)^2 + (x_k - x_s^n)^2)^2} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.28}$$

3.5 ILLUSTRATION

Let us consider certain signal for better understanding of extended kalman filter. Here consider cosine signal as an explanation with extended kalman filter to know how extended kalman filter estimates the unknown variables. Here also discussed how noise parameter takes into consideration, how jacobian and hessian matrices will be calculated for first and second order extended kalman filters respectively. In general nonlinearity is exist in whether in system dynamic model or measurement (observation) model. Then we propagate the state variables through these nonlinear models whether it may be any ono of those models or both of these models [13].

Consider cosine signal as like this

$$h_k = b_k \cos(\alpha_k) \tag{3.29}$$

Explanation

Let us consider measurement model as cosine signal denoted by adding some noise term to it

$$y_k = b_k \cos(\alpha_k) + v_k \tag{3.30}$$

From the above h_k is our cosine signal

In h_k there exist some terms b_k and α_k)

Here b_k is magnitude (amplitude) of the signal

α_k is angular velocity at a time step k

v_k is measurement noise term included in to this measurement model which has Gaussian nature

Here state variables are $b_k \;\; w_k \;\; \alpha_k$ respectively

Then state vector is represented like $X_K = [b_k \;\; w_k \;\; \alpha_k]$

To calculate jacobian matrices here first order derivatives of measurement model with respect to state variables will be calculated respectively.

Therefore first order derivatives are

$$\frac{\partial h_k}{\partial b_k} = \cos(\alpha_k), \frac{\partial h_k}{\partial w_k} = 0, \frac{\partial h_k}{\partial \alpha_k} = -b_k \sin(\alpha_k) \tag{3.31}$$

So from the above calculations for the first order extended kalman filter model jacobian matrix can be written as like this

$$H_k = [\cos(\alpha_k) \;\; 0 \; -b_k \sin(\alpha_k)] \tag{3.32}$$

For the second order kalman filter there is need to calculate hessian matrix also for the observation model. To calculate hessian there is need to evaluate second order derivative it means it represented by second order partial derivative which is shown below

Here

$$\frac{\partial^2 h_k}{\partial b_k \partial b_k}=0, \frac{\partial^2 h_k}{\partial b_k \partial w_k}=0, \frac{\partial^2 h_k}{\partial b_k \partial \alpha_k}=-\sin(\alpha_k) \tag{3.33}$$

$$\frac{\partial^2 h_k}{\partial w_k \partial b_k}=0, \frac{\partial^2 h_k}{\partial w \partial w_k}=0, \frac{\partial^2 h_k}{\partial w_k \partial \alpha_k}=0 \tag{3.34}$$

$$\frac{\partial^2 h_k}{\partial \alpha_k \partial b_k}=-\sin(\alpha_k), \frac{\partial^2 h_k}{\partial \alpha_k \partial w_k}=0, \frac{\partial^2 h_k}{\partial \alpha_k \partial \alpha_k}=-b_k\cos(\alpha_k) \tag{3.35}$$

After calculating the second order partial derivatives of the observation (measurement) model The hessian matrix can be written as [13].

$$H_{KK} = \begin{bmatrix} 0 & 0 & -\sin(\alpha_k) \\ 0 & 0 & 0 \\ -\sin(\alpha_k) & 0 & -b_k\cos(\alpha_k) \end{bmatrix} \tag{3.36}$$

From above measurements are only single dimensional, so calculation of hessian matrix is sufficient and the expressions are looks easier. But in some cases expressions are not looks simpler and evaluation of hessian matrix is not that of much easy like above case. It prone to programming and measurement errors easily and difficult to observe where the error has occurred.

3.5.1 Illustration with different models

Target motion is generally observed in 3 different ways [3].

- Consistent velocity model
- Consistent acceleration model
- Coordinate turn model

In general two dimensional discrete system models is denoted as

$$X_{t+1} = FX_t + Bv_t \tag{3.37}$$

Consistent velocity model

In consistent velocity model object is moving with constant velocity for this model state transition

matrix can be written as

$$F=\begin{bmatrix}1 & 0 & t & 0\\ 0 & 1 & 0 & t\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}. \tag{3.38}$$

Let us consider in bearing only tracking the state vector for this model is X= $[x, y, \dot{x},\dot{y}]$

Here x, y represents positions of x and y axis's respectively.

$\dot{x}$,$\dot{y}$ represents velocity components corresponding to x and y directions respectively.

In bearing only tracking consider at particular point observer is present, observer nothing like sensor is there it has its corresponding x and y location and that observer position is remains fixed throughout our experiment and that observation point is denoted as (x^s, y^s)

Consistent acceleration model

In consistent acceleration model object is moving with constant acceleration throughout time interval of experiment and state variables number will be increased that is inclusion acceleration terms in x and y direction to the x, y position and velocity terms.

Then state vector will be changed as

$X_t = [x_t \quad y_t \quad \dot{x}_t \quad \dot{y}_t \quad \ddot{x}_t \quad \ddot{y}_t]$'

Here in X_t t represents each and every time instant.

For constant acceleration model state transition matrix is

$$F = \begin{bmatrix}1 & 0 & t & 0 & t^2/2 & 0\\ 0 & 1 & 0 & t & 0 & t^2/2\\ 0 & 0 & 1 & 0 & t & 0\\ 0 & 0 & 0 & 1 & 0 & t\\ 0 & 0 & 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 0 & 0 & 1\end{bmatrix} \tag{3.39}$$

Coordinate turn model

In coordinate turn model object is moving constant velocity and varying course (direction). Course is nothing but direction, as time progresses object changes it direction continuously that is object is turning object, in this model state vector will be augmented with one more new parameter that is coordinated turn rate (angular velocity) denoted by W_k and measured in radian per second coordinated turn rate is also called as angular turn rate.

Here in coordinated turn model, we assumed the constant velocity model for object which is moving in a two dimensional system. And acceleration terms are not included in this model because here object is taken constant velocity objects so rate of change of velocity is zero.

Then state vector for this model will be like this [3]

$X_t = [x_t \quad y_t \quad \dot{x}_t \quad \dot{y}_t \quad w_t]'$

Here t represents each and every discrete time step

For the above coordinated turn model state transition matrix is explored in [5].

$$F=\begin{bmatrix} 1 & 0 & \frac{\sin(w_t \Delta k)}{w_t} & \frac{\cos(w_t \Delta k)-1}{w_k} & 0 \\ 0 & 1 & \frac{1-\cos(w_t \Delta k)}{w_t} & \frac{\sin(w_t \Delta k)-1}{w_t} & 0 \\ 0 & 0 & \cos(w_t \Delta k) & -\sin(w_t \Delta k) & 0 \\ 0 & 0 & \sin(w_t \Delta k) & \cos(w_t \Delta k) & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.40}$$

Here Δk is the sampling time interval; B is the control input matrix

X_t is the state vector at time step t.

v_t is the Gaussian noise with variance σ_ω^2

From the above model

$$x_{t+1} = x_t + \frac{\sin(w_t \Delta k)}{w_t}\dot{x}_t + \frac{\cos(w_t \Delta k)-1}{w_k}\dot{y}_t \tag{3.41}$$

$$y_{t+1} = y_t + \frac{\sin(w_t\Delta k)}{w_t}\dot{y}_t + \frac{1-\cos(w_t\Delta k)}{w_k}\dot{x}_t \tag{3.42}$$

$$\dot{x}_{t+1} = (\cos(w_t\Delta k)\ \dot{x}_t) - \sin(w_t\Delta k)\ \dot{y}_t \tag{3.43}$$

$$\dot{y}_{t+1} = (\cos(w_t\Delta k)\ \dot{y}_t) + \sin(w_t\Delta k)\ \dot{x}_t \tag{3.44}$$

$$w_{t+1} = w_t + v_t \tag{3.45}$$

The above equations are just the expansion of system model with the given state transition matrix. To estimate the unknown variables with extended kalman filter calculation of jacobians for both state transition matrix and measurement matrix s necessary.

While calculating jacobian matrix for the above state transition matrix just take the partial derivatives of the above equations with respect to state variables.

For the above state transition matrix jacobian matrix is

$$F=\begin{bmatrix} 1 & 0 & \frac{\sin(w_t\Delta k)}{w_t} & \frac{\cos(w_t\Delta k)-1}{w_t} & \frac{\partial x_{t+1}}{\partial w_t} \\ 0 & 1 & \frac{1-\cos(w_t\Delta k)}{w_t} & \frac{\sin(w_t\Delta k)-1}{w_t} & \frac{\partial y_{t+1}}{\partial w_t} \\ 0 & 0 & \cos(w_t\Delta k) & -\sin(w_t\Delta k) & \frac{\partial \dot{x}_{t+1}}{\partial w_t} \\ 0 & 0 & \sin(w_t\Delta k) & \cos(w_t\Delta k) & \frac{\partial \dot{y}_{t+1}}{\partial w_t} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.46}$$

From the above jacobians matrix it looks like same as state transition matrix except the last column. In the last column here taken the partial derivatives of all the state variables with respect to the coordinate turn rate parameter, here after evaluating these last column partial derivatives our required jacobian will be obtained [12].

$$\frac{\partial x_{t+1}}{\partial w_t} = \frac{w_t\Delta k\cos(w_t\Delta k)-\sin(w_t\Delta k)}{{w_t}^2}\dot{x}_t - \frac{w_t\Delta k\sin(w_t\Delta k)+\cos(w_t\Delta k)-1}{{w_t}^2}\dot{y}_t \tag{3.47}$$

$$\frac{\partial y_{t+1}}{\partial w_t} = \frac{w_t\Delta k\sin(w_t\Delta k)+\cos(w_t\Delta k)-1}{{w_t}^2}\dot{x}_t - \frac{w_t\Delta k\cos(w_t\Delta k)-\sin(w_t\Delta k)}{{w_t}^2}\dot{y}_t \tag{3.48}$$

$$\frac{\partial \dot{x}_{t+1}}{\partial w_t} = -\Delta k\sin(w_t\Delta k)\dot{x}_t - \Delta k\cos(w_t\Delta k)\,\dot{y}_t \tag{3.49}$$

$$\frac{\partial \dot{y}_{t+1}}{\partial w_t} = -\Delta k \cos(w_t \Delta k)\, \dot{x}_t - \Delta k \sin(\dot{w_t} \Delta k)\, y_t \quad (3.50)$$

3.6 Disadvantages of extended kalman filter:

1. While calculating the Jacobian matrices for first order filter and hessian matrices for second order filter these matrices need to exist then only any transformations and further calculations will be possible. If these matrices are not exist then transformation can't exist and there are some cases where these matrices are unable to exist [13].

2. Much of the time, the count of Jacobian and Hessian frameworks can be an exceptionally troublesome procedure, and it's likewise inclined to human errors. Furthermore programming errors are also included, these errors are normally difficult to investigate, as it's difficult to see which parts of the system delivers the errors by taking look at the evaluations, particularly as generally we don't know which sort of execution ought to anticipate.

3. The linear and quadratic changes produces reliable results only when the mistake engendering can be very much approximated by a linear or a quadratic capacity. In the event that this condition is not met, the execution of the filter can gives very poor results.

CHAPTER 4
PARTICLE FILTER

Chapter4

PARTICLE FILTER

4.1 Introduction

From the perspective of useful model on states, the EKF does not ensure an ideal arrangement in a complicated situation of nonlinearity and non-Gaussian distribution. The particle filter (PF) is an alternative with competitive performance contrasted with the extended kalman filter.

A few calculations don't adapt to nonlinear state or estimation models and non-Gaussian state or measured noises, under such sort of circumstances Particle filter is especially adjusted.

Customary routines are taking into account linearized models and Gaussian noise approximate estimations, so that the Kalman filter can be used. Here major analysis is based on how different state directions or numerous models can be used to bound the estimations.

In divergence to this, the particle filter estimates the optimal solution based on the system dynamic model. A well-understood issue with the particle filter is that its performance degrades immediately when the dimension of the state vector increases. Key hypothetical commitment here is to apply under estimation procedures prompting that the Kalman filter can be used to evaluate or take out all position derivatives, and the particle filter is connected with the part of the state vector containing just the position. Thus, in the particle filters dimensions just depending on the application, and this is the main measure to get real-time better superior algorithms. When applying optimal filtering tracking techniques are not tracking the object path there is need to shift to approximation methods one such kind of approximation method is Monte Carlo methods [15].

Sequential importance sampling uses Monte Carlo methods it approximates the full posteriori distribution instead of filtering distribution. Sequential importance sampling is the one of major step in particle filter when optimal filter technique with its corresponding prediction and updating steps are not sufficient to track the object path. Sequential importance sampling uses some random particles each particle contains its appropriate weights.

4.2 Sequential importance sampling

Essential thought of this design is to illustrate the posterior density for an arrangement of weighted arbitrary samples and to evaluate the parameters of interest depends upon these specimens and weights [4].

The filtering issue includes the estimation of the state vector at time k, given all the estimations up to and including time k, which we mean by $z_{1:k}$. at the point when the estimate and upgrade stage of the ideal filtering are not tractable accurately, then one needs to turn to estimated systems, one such technique is Monte Carlo evaluation. Sequential Importance sampling (SIS) is the most essential Monte Carlo technique utilized for this purpose.

The basic thought in sequential importance sampling is estimating the posterior distribution at time k that is P $(x_{0:k}/z_{1:k})$.

In sequential importance sampling it is helpful to consider the full posterior distribution at a time, instead of the filtering distribution, which is simply the marginal of the full posterior distribution. In filtering distribution while prediction the state at some time k that is for x_k it required measurement values from 1: k which is represented like this [13].

P $(x_k/z_{1:k})$ where as in full posterior distribution to predict the state $x_{0:k}$ it requires measurement values from 1: k which is represented with P $(x_{0:k}/z_{1:k})$ with weighted arrangement of particles $x_{0:k}^i$ with its corresponding weights w_k^i where i ranges from 1 to N, and recursively upgrade these particles to get a close estimation to this posterior distribution.

As the number of samples becomes very large, the SIS filter approaches optimal estimate.

The primary particle set is made by drawing N autonomous acknowledge from p(X0) and allotting uniform weight 1/N to each of them.

It additionally has two stages.

The expectation comprises of spreading every particle as per the advancement comparison.

The heaviness of every particle is will be changed during the rectification step.

Particle filter tracking algorithm requires five basic steps.

4.3 Basic Particle filter model

The particle filter gives an approximate solution to an exact model, rather than the optimal solution to an approximate model. Particle filtering is a general Monte Carlo (examining) technique for performing derivation in state-space models. Where the state of a system develops in time and data about the state is acquired by means of noisy estimations measured at each and every time step. In a general discrete-time state-space, the state of a system model is evolved like this [12].

$$x_t = f(x_{t-1}, w_{t-1}) \tag{4.1}$$

Here x_t represents the state vector of a system at time t

w_{t-1} is noise vector of system model.

f (.) represents nonlinear state function and it varies with respect to time and illustrates the reconstruction of new state vector for each and every time instant. State vector x_t is unobservable. Observation model is represented with h_t .

Then its measurement vector $z_t = h_t(x_t, v_t)$ (4.2)

Here $h_t(.)$ is nonlinear function and it is varies with respect to time which illustrating the measurement vector and v_t is measurement noise vector.

The major step in particle filtering algorithm is sequential importance sampling are given below.

4.4 Basic algorithm steps

- Initialization
- Time update (move particles)
- Measurement update (change weights)
- Degeneracy problem
- Resampling

Initialization

Here initially we take N random particles assign each particle with equal weights. In this step chose N random samples (particles) around the initial positons. It means suppose a particular object is moved in a two dimensional plane chose N random particles in that 2-D plane, here each particle contains x coordinate and y coordinate and some other parameters depend on the chosen model [2].

Time update

After initializing the particle set, Propagate the particle set according to the process model of target dynamics.

One-step prediction of each particle

$$x_{t+1}^{i}=f_t\left(x_t^{i}\right)+w_t\left(x_t^{i}\right) \tag{4.3}$$

The approximation error decreases as the number of particles grows on increasing.

Measurement update

Updating the probability density function based on received measurement value with that update particle weight [2].

The weights are adjusted using the measurement

$$w_{t+1}^{i}=w_t^{i}p(y_t/x_t^{i}) \tag{4.4}$$

After adjusting the weights all weights will be normalized

$$w_{t+1}^{i}=w_{t+1}^{i}/\alpha \tag{4.5}$$

Here $\alpha=\sum_{i=1}^{N}w_{t+1}^{i}$ (4.6)

Here particles can explain measurement gain weight, in this sensor gives bearing angles by taking N particles and the particles which are far from the true object state will lose weight and the particles which are nearer to true object state will gain the weight.

Degeneracy problem

As number of iterations increases, it leads degeneracy problem. Degeneracy problem is that after some iterations some particles with very less weight an remaining particles carries more weight, the particles which carries more weight has useful Information and rest of particles are unnecessary ones.one best solution to the eliminate degeneracy problem is resampling. Degeneracy is generally measured by effective sample size, it is represented by Neff. [4], [7].

$$N_{eff} = \frac{1}{\sum_{i=1}^{N} (w_k{}^i)^2} \tag{4.7}$$

Generally degeneracy is depends on effective sample size N_{eff} ,if the N_{eff} value is small that means it has larger variance in the weights so it has more degeneracy. If N_{eff} value is high then it has smaller variance in the weights and so it has less degeneracy problem.

Resampling

The resampling can't perform for each case, resampling will be performed if N_{eff} falls below certain threshold value then only we perform resampling in particle filter to increase the accuracy of estimated state [13]. Threshold values are 2N/3 or 3N/4

There are lot of approaches to perform resampling , out of those systematic resampling is one approach, in this technique after updating the particles weights with measurement equations we normalize the particle weights, after that we calculate the cumulative sum of normalized particle weights, here we generate a some random number in which interval that random number occurred we use that index value to regenerate the new samples, here we repeat the process until the regenerated sample number would be equal to that initially generated random particles after generating the new samples we estimate the new state of the object which we want to track ,and we reinitialize particles weight to 1/N before going to the second iteration this we repeat over 100 times to find trajectory of object. In this resampling technique in particle filter while resampling some times while generating random number, if it can't fall in any cumulative summed interval or only single particle will be picked sometimes causes sub optimal solution and after some iterations all particles will be divided in to subsets, after some more iterations more particles form a group but we don't want all particles to estimate state, here we are using all particles by computing mean of all particles to estimate new state of object currently also lot of research is going on to concentrate only on certain number of particles and to neglect remaining unwanted particles, resampling is great importance in particle filter technique while tracking the nonlinear path of object [4].

4.5 Sample impoverishment:

Resampling will be performed once it will meet effective sample size threshold criteria, while performing resampling major intension is removal of particles which carries lesser weight and major concentration is choose the particles which carrying higher weights i.e. particles which carrying higher weights will be chosen multiple times and get rid of smaller weight particles that is not picked these particles. While doing like this after some time steps diversity of particles tends to be decreased and at one particular exceptional case all particles may crumple into a single particle. This phenomenon is called sample impoverishment problem [4], [13], this will give contrarily affect to the nature of the estimate.

4.5.1 Regularized particle filter:

Regularized particle filter provide the solution to this sample impoverishment problem by approximating the filtering and posterior distribution with choosing the kernel density estimation of particles. The most used kernel while density estimation particles is Epanechnikov kernel [4] to solve the sample impoverishment problem.

4.3 Comparison to the discrete Kalman filter

	Kalman filter	Particle filter
State equation	$x_{t+1} = F_t x_t + G_t w_t$ $y_t = H_t x_t + v_t$	$x_{t+1} = f_t(x_t) + w_t(x_t)$ $y_t = h_t(x_t) + v_t(x_t)$
Noise type	Gaussian, unimodal	Any distribution and it is either uni or multi modal
output	$\widehat{x_t}, p_t$	$p(x_t)$
solution	Exact, optimal	Approximate
Computational speed	Fast	slow

Fig 4.1.Comparison between kalman and particle filters

It chiefly comprises in engendering weighted arrangement of particles which approximates likelihood appropriation of the state restrictively to the observations. It is very easy to implement. In particular when measurement equation is nonlinear, dynamic and observation noises is non Gaussian particle filter is used widely.

An established issue in signal processing is tracking the object path, here required object's position is to be evaluated by taking into account estimations of relative range and inclinations in to consideration to one's own position. Whereas sensors measures relative inclination to the object, or a radar measures relative angle, for the instance of direction measuring sensor, either the state model or estimation model is non-linear relying upon the decision of state directions, at these situations the particle filter is especially encouraging.

CHAPTER5
EXPERIMENTAL RESULTS AND DISCUSSION

Chapter5

EXPERIMENTAL RESULTS AND DISCUSSION

Constant Velocity and Course Target

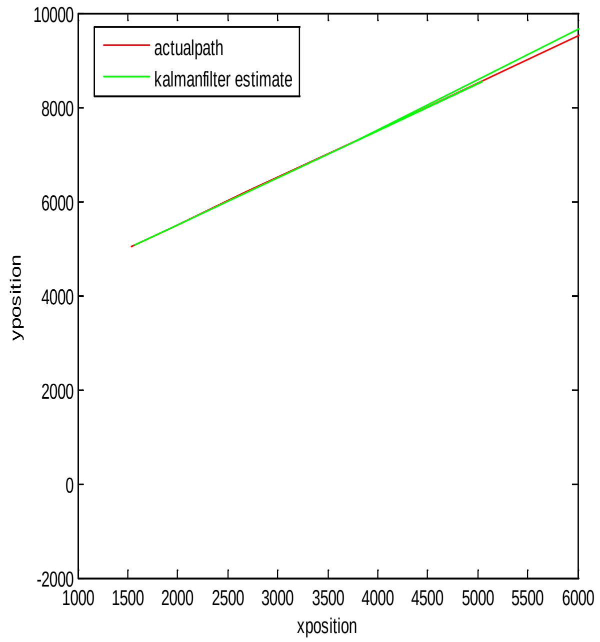

Fig 5.1 tracking with kalman filter case1

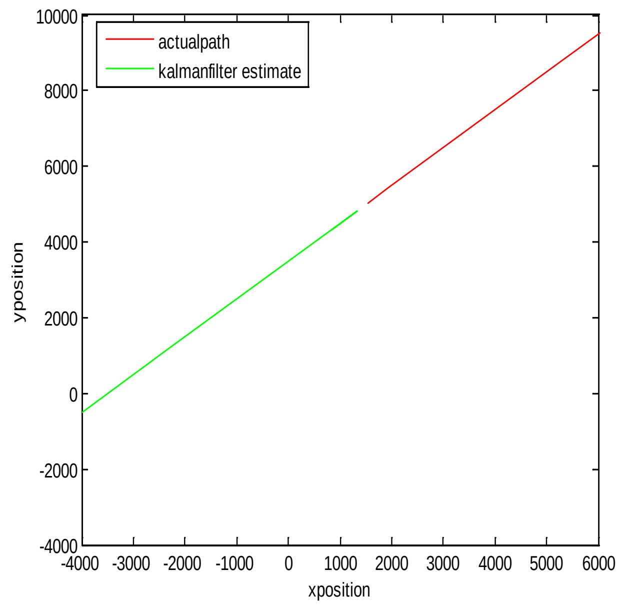

Fig 5.2 tracking with kalman filter case2

From above figure 5.1 in case 1 if the process model is linear and noise distribution is gaussian distribution with that initial assumption is good, then kalman filter optimally track the object path, here object starts at location(1500,5500) target moves constantly with 5km/hr,in fig 5.2 in case 2 process model is linear and noise probability distribution is gaussian and initial assumption is very poor then kalman filter can't track the object path in second case initial location is very poor and covariance error for above two cases is large.

Constant Velocity and Varying Course Target

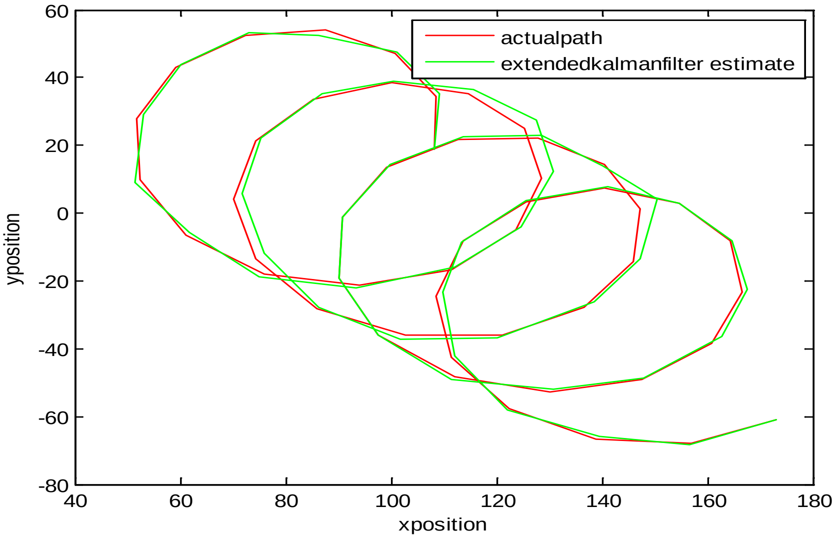

Fig 5.3 tracking the object path for a manoeuvuring targert case1

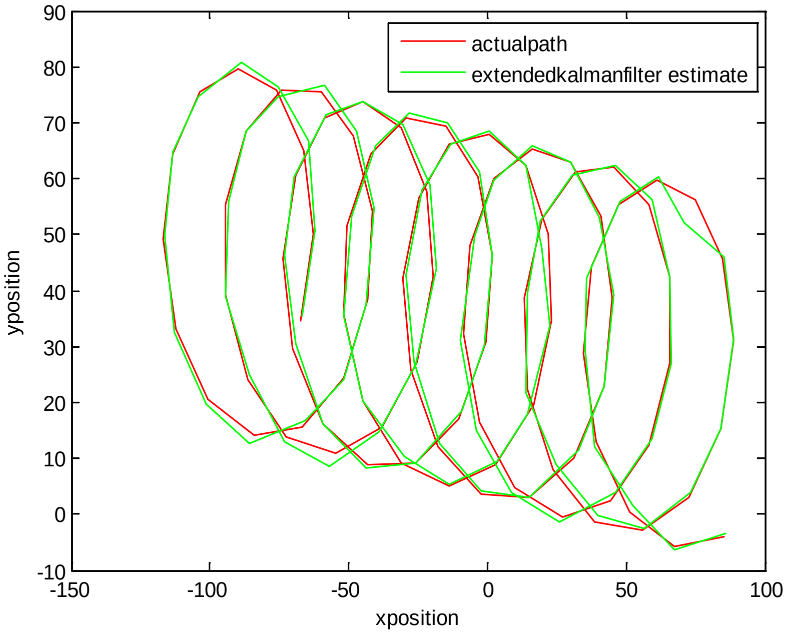

Fig 5.4 tracking the object path for a manoeuvuring targert case2

From figure 5.3, it represents tracking the object trajectory of a manoeuvuring targert with extended kalman filter with wk=0.1 radian per second[5],its initial location starts at (100,5) moving with consatant velocity 12m/sec and is simulated for 100 times here model aasumed is coordinate turn model which is a nonlinear model and assumed noise is gaussianly distributed extended kalman filter track the object path and gives the optimal estimation, where as in fig 5.4, it represents tracking the object trajectory of a manoeuvuring targert with extended kalman filter with wk=-0.1 radian per second[5], its initial location starts at (100,5) moving with consatant velocity 12m/sec and is simulated for 100 times here model asumed is coordinate turn model which is a nonlinear model and assumed noise is gaussianly distributed extended kalman filter track the object path and gives the optimal estimation.

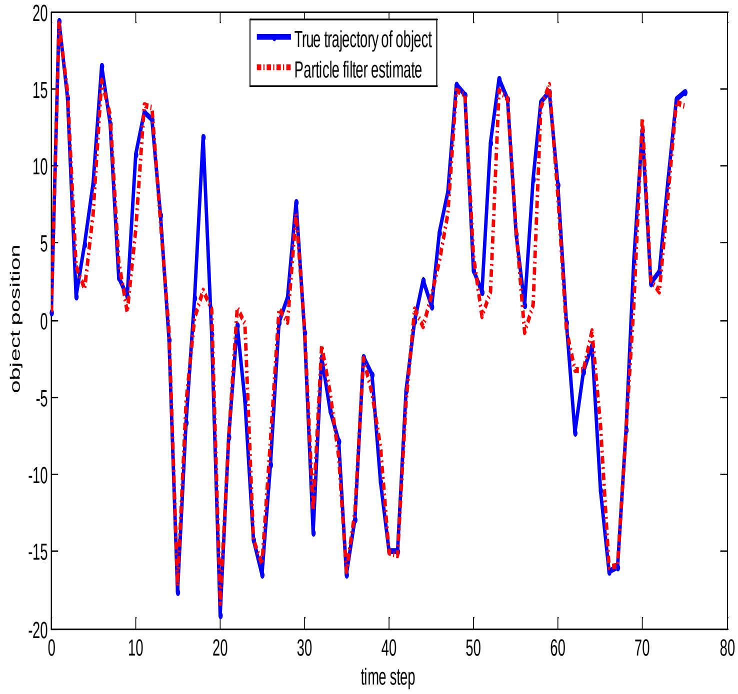

Fig 5.5 tracking one dimensional nonlinear example of particle filter

Fig 5.5,represents one dimensional nonlinear example[6] of particle filter here both process and measurement models are nonlinear and initial is object starts at location 0.5 and simulation is carried out particle filter tracking algorithm works good for this nonlinear model.

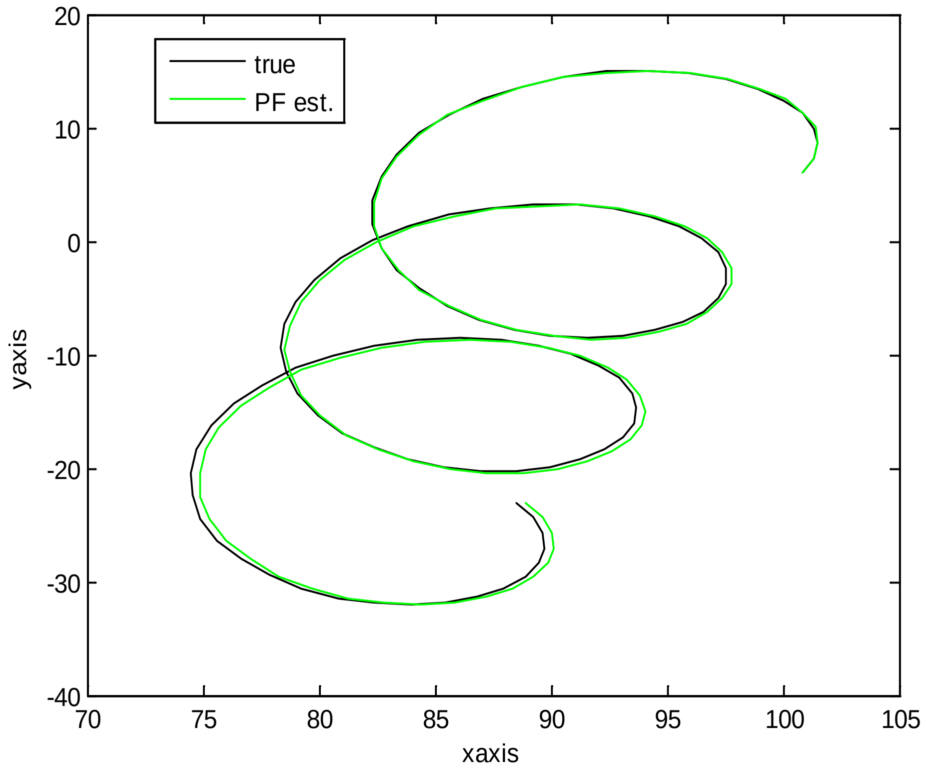

Fig 5.6 tracking two dimensional nonlinear example of particle filter

Fig 5.6 represents two dimensional nonlinear example of particle filter here both process and measurement models are nonlinear, it represents tracking the object trajectory of a manoeuvuring targert with extended kalman filter with wk=0.1 radian per second, its initial location starts at (100,5) moving with consatant velocity 12m/sec and is simulated for 100 times here model aasumed is coordinate turn model which is a nonlinear model and assumed noise is gaussianly distributed particle filter track the object path and gives the optimal estimation, particle filter tracking algorithm works good for this nonlinear models.

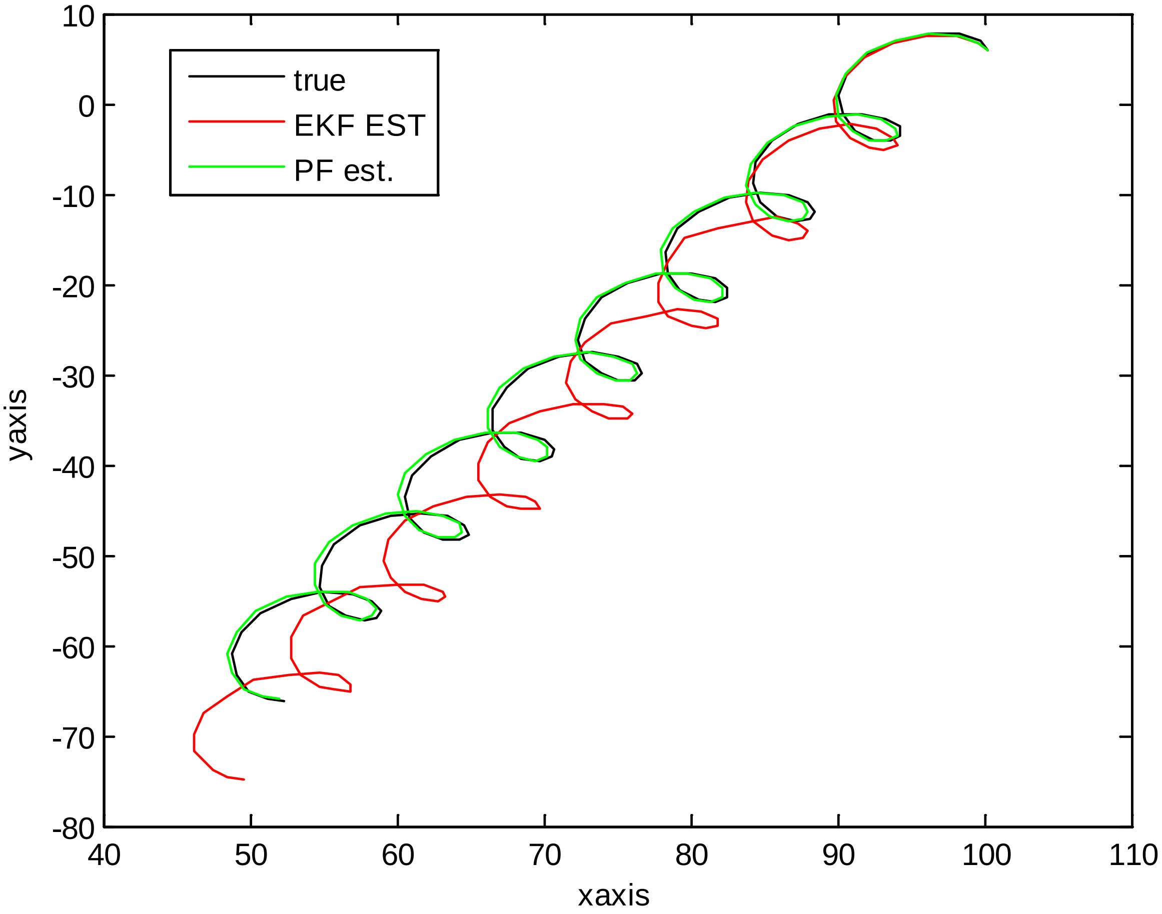

Fig 5.7 tracking comparision between particle filter and extende kalman filter

Fig 5.7 represents comparision between particle filter and extende kalman filter here both process and measurement models are nonlinear, it represents tracking the object trajectory of a manoeuvuring targert with wk=0.1 radian per second, its initial location starts at (100,5) moving with consatant velocity 12m/sec and is simulated for 100 times here model asumed is coordinate turn model which is a nonlinear model, from the figure extended kalman filter is not tracking the object path that much of accurately where as particle filter is giving the optimal estimation.

CHAPTER6
Conclusion and Future Work

Chapter6

Conclusion and Future Work

We have analyzed various cases of target tracking using extended kalman filter. Based on the simulation results obtained we have arrived at some major conclusions. Kalman filter gives better results if the chosen system model is linear model, assumed noise distribution is Gaussian distribution. Extended kalman filter works well even if chosen system model is nonlinear and noise distribution is Gaussian distribution and as nonlinearity is more and for non-Gaussian distribution case it can't give better results for these kind of cases particle filter gives optimal tracking. The use of particle filter in the estimation of the object tracking for any kind of system model and noise seems to have an advantage over the traditional kalman filter based tracking approaches. This is mainly due to the solution to problem of nonlinearity and non Gaussianity which has been addressed through the use of the random particles and resampling the use of these object detection algorithms may further increase the scope of the object path estimation by using some good resampling techniques as it increases the accuracy of the trajectory.

REFERENCES

1. Hue, Carine, J-P. Le Cadre and Patrick Pérez. "Tracking multiple objects with particle filtering." *Aerospace and Electronic Systems, IEEE Transactions on* 38, no. 3 (2002): 791-812.
2. Gustafsson, Fredrik, Fredrik Gunnarsson, Niclas Bergman, Urban Forssell, Jonas Jansson, Rickard Karlsson, and P-J. Nordlund. "Particle filters for positioning, navigation, and tracking." *Signal Processing, IEEE Transactions on* 50, no. 2 (2002): 425-437.
3. Chang, Dah-Chung, and Meng-Wei Fang. "Bearing-Only Maneuvering Mobile Tracking With Nonlinear Filtering Algorithms in Wireless Sensor Networks."*Systems Journal, IEEE Transactions on* 8, no. 1 (2014): 160-170.
4. Arulampalam, M. Sanjeev, Simon Maskell, Neil Gordon, and Tim Clapp. "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking."*Signal Processing, IEEE Transactions on* 50, no. 2 (2002): 174-188.
5. Radhakrishnan, K., A. Unnikrishnan, and K. G. Balakrishnan. "Bearing only tracking of maneuvering targets using a single coordinated turn model. *International Journal of Computer Applications on*1, no. 1 (2010): 25-33.
6. Gordon, Neil J., David J. Salmond, and Adrian FM Smith. "Novel approach to nonlinear/non-Gaussian Bayesian state estimation." In *IEEE Proceedings (Radar and Signal Processing), on* 140, no. 2 (1993): 107-113.
7. La Scala, Barbara, and Mark Morelande. "An analysis of the single sensor bearings-only tracking problem." In *Information Fusion, IEEE 11th International Conference*, (2008): 1-6.
8. Lin, Yuejin, Fasheng Wang, Yu Han, and Quan Guo. "Bearing-only target tracking with improved particle filter." In *Signal Processing Systems (ICSPS), IEEE 2nd International Conference on* 1, (2010): 331-333.

9. Yu, Jinxia, and Wenjing Liu. "Improved Particle Filter Algorithms Based on Partial Systematic Resampling." *IEEE International Conference on Intelligent Computing and Intelligent Systems on* 1, (2010):483-487.
10. Farina, Alfonso. "Target tracking with bearings–only measurements." *Signal processing on*78, no. 1 (1999): 61-78.
11. Crouse, David Frederic, R. W. Osborne, Krishna Pattipati, Peter Willett, and Yaakov Bar-Shalom. "2D Location estimation of angle-only sensor arrays using targets of opportunity." In *Information Fusion (FUSION), IEEE 13th Conference,* (2010): 1-8.
12. Panakkal, Viji Paul, and Rajbabu Velmurugan. "Bearings-only tracking using derived heading." In *IEEE Aerospace Conference, (2010*):1-11.
13. Hartikainen, Jouni, Arno Solin, and Simo Särkkä. "Optimal filtering with Kalman filters and smoothers." *Department of Biomedica Engineering and Computational Sciences, Aalto University School of Science: Greater Helsinki, Finland on*16 (2011).
14. Zhang, Jinghe, Greg Welch, Gary Bishop, and Zhenyu Huang. "A two-stage Kalman filter approach for robust and real-time power system state estimation. *"Sustainable Energy, IEEE Transactions on* 5, no. 2 (2014): 629-636.
15. Mallick, Mahendra, Mark Morelande, and Lyudmila Mihaylova. "Continuous-discrete filtering using EKF, UKF, and PF." In *Information Fusion (FUSION), IEEE 15th International Conference,* (2012):1087-1094.