International Journal of Advanced Technology and Engineering Exploration
ISSN (Print): 2394-5443 ISSN (Online): 2394-7454
Volume-2 Issue-6 May-2015

Estimation of System Parameters Using Kalman Filter and Extended Kalman Filter

Nousheen Fahmedha*, P Chaitanya Prakash, Pooja A and Rachana R
Sir M Visvesvaraya Institute of Technology, Bangalore, India

Abstract

In 1960, Sir R E Kalman published his famous paper describing a solution to discrete data linear filtering problem. Since then there has been advances in digital computing and Kalman filter has been subjected to extensive research and application, particularly in the area of autonomous or assisted navigation. This paper gives a brief introduction on Kalman filter, the equations that can be used for discrete stochastic systems which has additive white Gaussian noise present that models 'unpredictable disturbances' of the system. We then have simulation results of a system considered. However in real time scenarios we might also encounter non-linear systems and Kalman filter is not a good choice for such systems. So we go in for extended Kalman filter that linearizes the non-linear parameters about its mean and covariance using Jacobian matrices, and then using the same algorithm as of Kalman Filter. This algorithm is implemented on a target moving with a known trajectory with Gaussian noise introduced in the measurements. With the help of this algorithm we have tried to minimize the error in the measurements which is shown in the simulation results. We have also explained about various tuning factors that affect the estimation part and showed how these values stabilise after few iterations.

Keywords

Stochastic Systems, Least square method, Likelihood function, Normal Probability distribution.

1. Introduction

Estimation is the process of inferring the value of a quantity of interest from indirect, inaccurate and uncertain observations. The main purpose of estimation can be [1]:

- Determination of planet orbit parameters
- Determination of the position and velocity of an aircraft in an air traffic control system
- Determination of model parameters for predicting the state of a physical system or forecasting economic or other variables

Estimation can be defined as "The process of selecting a point from continuous space – the best estimate." Filtering is the estimation of the state of a dynamic system. The word 'filter' is used because the process of estimating or obtaining the best estimate from the noisy data involves the elimination of undesired signal, which in this case is noise.

An optimal estimator is an algorithm that processes observations to yield an estimate of a variable of interest, which optimizes a certain criterion. So a proper estimate of the system parameters is needed to extract the right information and then enhance it. Kalman filter is ideal for linear stochastic systems that is of our interest in this paper.

The Kalman filter is a set of mathematical equations that provides an efficient computational solution of the least-squares method. The main advantages of using Kalman filter are:

- Provides running measure of accuracy of predicted parameters.
- Permits optimum handling of measurements of accuracy.
- Allows optimum use of a priori information if available.
- Permits target dynamics to be used directly to optimize filter parameters.
- Addition of random-velocity variable, which forces Kalman filter to be always stable.

Kalman filter can also be used for non-linear systems though it is not the optimal algorithm for estimation. This is achieved by linearizing the non-linear parameters about its mean and covariance. This algorithm is called 'Extended Kalman filter'.

*Author for correspondence

2. Kalman Filter

Let us consider a linear stochastic system[3] whose state vector is given by $X \in \Re_n$. X holds all the parameters of interest to us. Let us try to estimate its position in Cartesian coordinate system and so our parameters of interest will be x, v_x, y, v_y that represents its x-coordinates, velocity along x axis, y-coordinates, velocity along y axis respectively.

Also let us consider process noise and measurement noise to be independent, white Gaussian noise with normal probability distribution. As already mentioned Kalman filter is optimal estimator for linear systems. The difference equations of a linear stochastic systems without any control inputs is given by [4],

$$X_\kappa = A * X_{\kappa-1} + W_{\kappa-1} \quad (2.1)$$

With measurement vector $Z \in \Re_m$ that is

$$Z_\kappa = H * X_\kappa + V_\kappa \quad (2.2)$$

In the above equations k is used as an index for time, W_κ and V_κ are process noise and measurement noise whose covariance matrix is given by Q and R respectively where,

$$E[V_\kappa V_\kappa^T] = Q_k \quad (2.3)$$

$$E[W_\kappa W_\kappa^T] = R_\kappa \quad (2.4)$$

Although Q and R matrices change and needs to be updated we have kept it constant and to simplify it even further we have used a diagonal matrix that holds the square of the error. The 'Q' matrix can be designed such that there would be a plant noise coefficient 'q' scaled out of the matrix, which can be used to tune the Kalman Filter. X_κ at K=0 is modelled as a random variable, with Gaussian distribution with known mean and covariance. We can also include the control input matrix say U, but for better understanding of the algorithm we neglect it.

The algorithm has three steps that are run in a recursive loop fashion. The first step is the prediction or time update step. At this stage, the equations given by (2.5) and (2.6) are used to project the current state and error covariance estimate to obtain a priori estimate for the next time step.

$$X_{predicted} = A * X_{\kappa-1} \quad (2.5)$$

$$P_{predicted} = A * P_{\kappa-1} * A^T + Q \quad (2.6)$$

Where 'A' is a state transition matrix that relates the state at previous time step 'K-1' to the current step 'K'.

The next stage of this algorithm involves calculation of innovation and Kalman gain. Innovation is a measure of discrepancy between the predicted measurement and actual measurements 'Z_κ' given by equation (2.7).Kalman gain is also referred to as blending factor that minimizes the posterior covariance and is given by equation (2.9). It can be inferred from equation (2.9), as the measurement error covariance 'R' approaches zero the actual measurement 'Z_κ' is accurate while the predicted value is less accurate and if priori estimate error covariance 'P' approaches zero, the actual measurement 'Z_κ' is less accurate while the predicted measurement '$H*X_{predicted}$' is more accurate.

$$Y = Z_\kappa - H * X_{predicted} \quad (2.7)$$

$$S = H * P * H^T + R \quad (2.8)$$

$$K = P * H^T * S \quad (2.9)$$

Where H is a constant matrix that relates the state to the measurement vector.

The last stage in this algorithm is to update the state vector using the predicted values, Kalman gain computed and innovation. Equation (2.10) and (2.11) gives the state update and covariance update equations respectively.

$$X_{updated} = X_{predicted} + K * Y \quad (2.10)$$

$$P_{updated} = (I - K * H) * P_{predicted} \quad (2.11)$$

As already mentioned, Q and R are assumed to be constant and so both error covariance P and Kalman gain K will stabilize quickly and then remain constant. Then these parameters can be pre-computed by running the filter off-line. It is frequently the case that the measurement error does not remain constant. For instance, when sighting beacons in an optoelectronic tracker ceiling panels, the noise in measurements of nearby beacons will be smaller than that in far-away beacons. Also, the process noise Q is sometimes changed dynamically during filter operation—becoming Q_κ —in order to adjust to different dynamics. For instance, if we are tracking the position of a user of a 3D virtual environment, we might reduce the magnitude of Q_κ if the user seems to be moving slowly, and increase the magnitude if the dynamics start changing rapidly. In such cases Q_κ

might be chosen to account for both uncertainty about the user's intentions and uncertainty in the model.

3. Algorithm Summary

Table 1 summarizes all the equations from 2.1 to 2.11 with each stage labelled.
Blue = inputs;
Orange = outputs;
Black = constants;
Grey = intermediary variables.

Table 1: Summarizes Kalman filter algorithm for every stage

State Prediction (Predict where we are going to be)	$X_{predicted} = A*X_{n-1} + B*U_n$
Covariance Prediction (Predict how much error)	$P_{predicted} = A*P_{n-1}*A^T + Q$
Innovation (Comapre reality against prediction)	$\bar{y} = Z_n - H* X_{predicted}$
Innovation Covariance (Compare real error against prediction)	$S = H* P_{predicted}*H^T + R$
Kalman Gain (Moderate the prediction)	$K = P_{predicted}* H^T*S^{-1}$
State Update (New estimate of where we are)	$X_n = X_{predicted} + K*\bar{y}$
Covariance Update (New estimate of error)	$P_n = (I - K*H)* P_{predicted}$

4. Extended Kalman Filter

As described in section 2, Kalman Filter algorithm is used to solve linear stochastic systems with white Gaussian noise. When the same algorithm is applied to non-linear systems it may fail. Instead we can linearize the non-linear parameters about its estimated mean and covariance and then apply Kalman filter algorithm to it. Such a filtering technique is referred to as Extended Kalman Filter or EKF [6][7].

We can linearize the estimate around the current estimate using the partial derivatives of the process and measurement function to compute estimate even in non-linear situations. There are few changes that needs to be brought in Kalman filter algorithm before we can apply it to nonlinear systems. First and foremost we must define our state vector $X \in \Re_n$ whose equation is given by,

$$X_\kappa = f(X_{\kappa-1}, U_\kappa, W_{\kappa-1}) \quad (4.1)$$

With measurement vector $Z \in \Re_m$ given by,

$$Z_\kappa = h(X_\kappa, V_\kappa) \quad (4.2)$$

Where the variables used are same as in section 2.0. 'A' is a Jacobian matrix of partial derivatives of 'f' with respect to 'x' and is given by,

$$A_{[i,j]} = \partial f[i] / \partial x[j]$$

'W' is a Jacobian matrix of partial derivatives of 'f' with respect to 'w' and is given by,

$$W_{[i,j]} = \partial f[i] / \partial w[j]$$

'H' is a Jacobian matrix of partial derivatives of 'h' with respect to 'x' and is given by,

$$H_{[i,j]} = \partial h[i] / \partial x[j]$$

'V' is a Jacobian matrix of partial derivatives of 'h' with respect to 'v' and is given by,

$$V_{[i,j]} = \partial h[i] / \partial v[j]$$

Here the Jacobian matrices A, W, H, V are not constant and are to be updated for every time step. Thus to sum up all the equations used in EKF, we have:

EKF time update equations:

$$X_{predicted} = f(X_{\kappa-1}, U_\kappa, 0) \quad (4.3)$$

$$P_{predicted} = A_\kappa * P_{\kappa-1} * A_\kappa^T + W_\kappa * Q_{\kappa-1} * W_\kappa^T \quad (4.4)$$

EKF measurement update equations:

$$K_\kappa = P_\kappa * H_\kappa^T / (H_\kappa * P_\kappa * H_\kappa^T + V_\kappa * R_\kappa * V_\kappa^T) \quad (4.5)$$

$$X_{updated} = X_\kappa = X_{predicted} + K_\kappa * (Z_\kappa - h(X_\kappa, 0)) \quad (4.6)$$

$$P_{updated} = P_\kappa = (I - K * H) * P_{predicted} \quad (4.7)$$

Note[8]: One important feature of EKF is that the Jacobian matrix 'H' used in equation (4.5) serves to correctly propagate or magnify only the relevant component of the measurement information. The main disadvantage of EKF is that if measurement vector 'Z' and measurement function 'h' are not mapped one to one then we might find that filter will quickly diverge and our main aim to track any target will fail. So for such cases where the systems are nonlinear in nature and have non-Gaussian noise, we have advanced filtering techniques like particle filter.

5. Simulation Results

Kalman filter:

To begin with we start tuning the filter by taking plant noise ‘q = 1’ and see the results. The best tuning of filter is achieved with ‘q = 0.1’ and time step t = 0.5. The results shown here are for these values only.

In figure 1 the blue ‘.-‘ graph gives the difference between the true measurement and the measurement with noise while the red ‘o-‘ graph gives the difference between the true measurement and the updated measurements we get from equation (2.10). As we can see the error has been reduced after the values have been processed using Kalman filter algorithm. In figure 2, we have the tuning factor ‘q’ versus covariance in both the direction. This graph will help us tune the filter. In figure 3 we have the tuning factor ‘q’ versus Kalman gain ‘K’ in both the directions [2].

Extended Kalman filter (EKF):

We have considered a target moving in a straight line and the measurement vector is in spherical coordinates, thus making the system non-linear in nature. We have estimated the position in spherical coordinates only. In figure 4 the blue ‘.-‘ graph gives the difference between the true measurement and the measurement with noise while the red ‘o-‘ graph gives the difference between the true measurement and the updated measurements all considered in spherical coordinates.

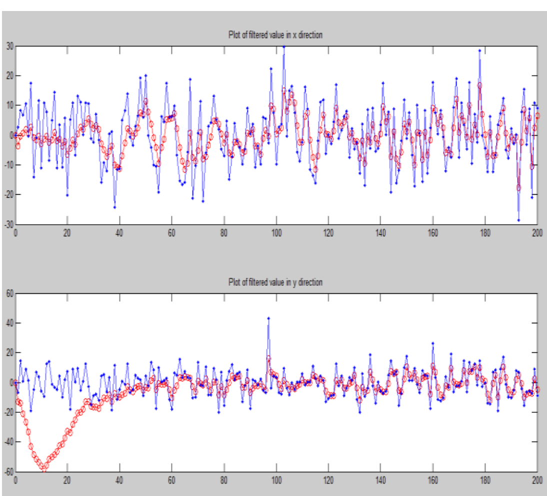

Figure 1: Comparison between error in measurement vector and the error in the updated measurement vector for Kalman Filter Algorithm

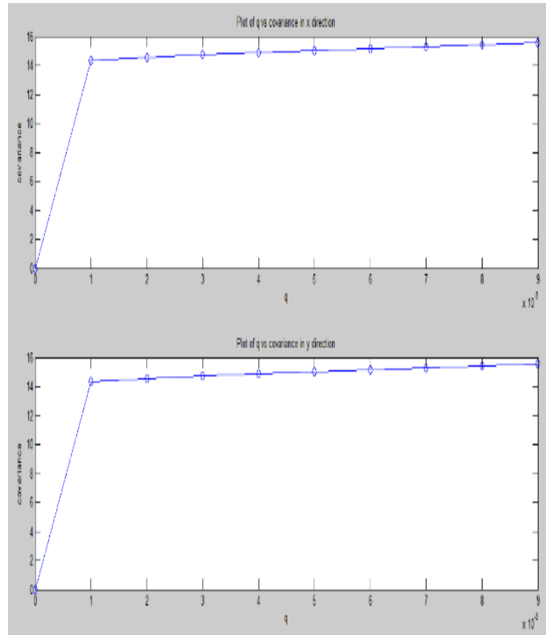

Figure 2: Plant noise ‘q’ versus covariance in both X amd Y axises

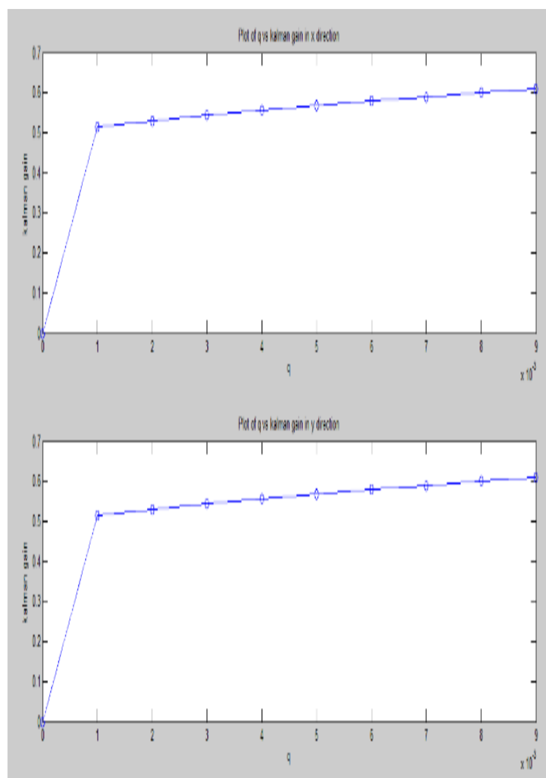

Figure 3: Plant noise ‘q’ versus Kalman gain

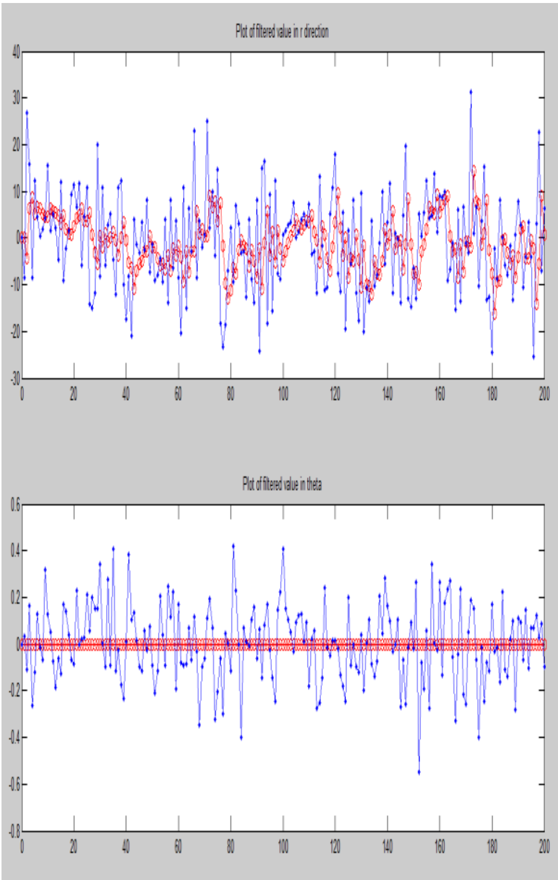

Figure 4: Comparison between error in measurement vector and the error in the updated measurement vector (both are in spherical coordinates) for Extended Kalman Filter Algorithm

6. Conclusion and Future Work

In this paper we have presented a general Implementation of Kalman Filter for tracking objects that follow Gaussian distribution. We then extended it for a non-linear systems and understood that Kalman filter algorithm cannot be used if the system parameters and/or measurements are nonlinear. So for nonlinear systems we have few other algorithms like Particle filters[5][9] (which is beyond the scope of this paper) that will help in better estimation. The algorithms discussed in this paper can also be used in other applications such as Navigation, Guidance control, Robotics, Econometrics.

References

[1] Eli Brookner, "Tracking and Kalman Filtering Made Easy", John Wiley and Sons, Inc, 1998.
[2] Baseem R.Mahafza, "Radar Systems Analysis and Design Using MATLAB, Chapman and Hall/CRC, 2000.
[3] Yaakov Bar-Shalom, X.Rong Li, Thangalingam Kirubarajan, "Estimation with Application to tracking and Navigation: , John Wiley and Sons, Inc, 2001.
[4] Greg Welch and Gary Bishop, " An Introduction to the Kalman Filter", TR 95-041, Department of Computer Science, University of North Carolina at Chapel Hill, 2002.
[5] Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp,‖ A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking‖, IEEE transactions on signal processing, vol. 50, no. 2, February 2002.
[6] N. G. B. Ristic, S. Arulampalam, Beyond the Kalman Filter – Particle Filters for Tracking Applications. Artech House, September 2004.
[7] Maria Isabel Ribeiro, "Kalman and Extended Kalman Filters: Concept, Derivation and Properties", M. Isabel Ribeiro, 2004.
[8] Gabriel A. Terejanu, "Extended Kalman Filter Tutorial", Department of Computer Science and Engineering, University at Buffalo, Buffalo, NY 14260.
[9] Victor B. Frencl and Jo˜ao B. R. do Val, ―Tracking with Range Rate Measurements: Turn Rate Estimation and Particle Filtering‖, IEEE conference 2012, 978-1-4673-0658-4/12.

Nousheen Fahmedha is from Karnataka, India and was born on 2nd August 1993. She is currently in the final year of BE in Electronics and Communication engineering from Sir M Visvesvaraya Institute of Technolgy, Bangalore which is affiliated to Visvesvaraya Technologial University, Belgaum.
Email: nousheen2893@gmail.com

P Chaitanya Prakash is from Karnataka, India and was born on 26th February 1993. He is currently in the final year of BE in Electronics and Communication engineering from Sir M Visvesvaraya Institute of Technology, Bangalore which is affiliated to Visvesvaraya Technological University, Belgaum.

Pooja A is from Karnataka, India and was born on 6th July 1993. She is currently in the final year of BE in Electronics and Communication engineering from Sir M Visvesvaraya Institute of Technology, Bangalore which is affiliated to Visvesvaraya Technological University, Belgaum.

Rachana R is from Karnataka, India and was born on 6th June 1993. She is currently in the final year of BE in Electronics and Communication engineering from Sir M Visvesvaraya Institute of Technology, Bangalore which is affiliated to Visvesvaraya Technological University, Belgaum.