Kalman Filter Implementation on an Accelerometer sensor data for three state estimation of a dynamic system

Toshak Singhal, Akshat Harit, and D N Vishwakarma

Abstract**—** Kalman Filter is used in many system estimation applications like state estimation, digital signal processing, sensor integration, Navigational Systems, etc. Kalman Filter is frequently used for the purpose of filtering accelerometer data to give position and velocity coordinates. This paper presents a Kalman filter implementation using a system model based on constant acceleration and analyzes its performance for different use cases. Furthermore relation of q and R parameters of Kalman filter is also presented with respect to errors in measurements. The paper also discusses the different implementations that can be used for optimally predicting acceleration along with velocity and position.

Keywords**—**Kalman Filter, State Estimation, Kalman Filter error Analysis, Accelerometer sensor

I. Introduction

KALMAN Filter is a digital filter used to filter noise on a series of measurements observed over a time interval. Recent advancements have been made and various successive filters such as Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF) have been derived from it. It is an algorithm used to solve the linear quadratic Gaussian (LQG) estimation problem. It operates recursively on the data stream of a dynamic system to give an optimum estimate of the current system state. It has numerous applications in various fields like Power System state estimation [2][3], Aircraft Guidance and navigational control systems. The Kalman filter algorithm is based on two steps; first the prediction step in which the current estimate of state variables, with random noise included is given. The prediction step only involves data measurement before the time at which system state is to be calculated. These estimates are used along with the measurement, with random Gaussian noise, to give the correct state of the system. The algorithm works by using a weighted average model on the predicted value and the current value. The more certain measurement is given more weight. The filter works in the discrete time domain. Implementations are available for continuous time version, called Kalman-Bucy filter. Another variant, the Unscented Kalman Filter (UKF) [4] is used when state transition and observation models are highly non-linear i.e. cases in which EKF gives poor performance. Also Kalman filter has been proven to give excellent results in the sensor data fusion [5] sometimes along with Fuzzy logic. Kalman filter in sensor data fusion treats one sensor data as measurement and other as prediction. It has been very frequently used to integrate GPS (Global Positioning System) and IMU (Inertial Measurement Unit) unit employed in both Airborne and terrestrial automated vehicles.

Toshak. Singhal is a 3rd year B. Tech student in Electrical Engineering at Indian Institute of Technology (BHU), Varanasi, India (phone: 00917388065750; fax: N/A; e-mail: toshak.singhal.eee10@itbhu.ac.in).

Akshat. Harit is a 3rd year B. Tech student in Electrical Engineering at Indian Institute of Technology (BHU), Varanasi, India (phone: 00918853234750; fax: N/A; e-mail: akshat.harit.eee10@itbhu.ac.in).

D N. Vishwakarma is with the Electrical Engineering Department, Indian Institute of Technology (BHU), Varanasi, India (e-mail: dnv.eee@itbhu.ac.in).

II. Nomenclature

x_k	System State Matrix
w_k	Process Noise
z_k	Measurement Result Matrix
v_k	Measurement Noise
Φ_k	State Transition Matrix
P_k	State Error Covariance Matrix
H_k	Measurement transition Matrix
K_k	Kalman Gain
Q	Process Noise Covariance
R	Measurement Noise Covariance
E	Expectation Operator

III. System Model

System modelled in this paper is a three state system with acceleration, velocity and position being the three states. The process noise added is White Gaussian noise with signal to noise ratio equal to -2. Similarily, the measurement noise is also White Gaussian noise with signal to noise ratio equal to -2. Now system equations can be given as

$$x_{k+1} = x_k + v_k \tag{1}$$
$$z_k = x_k + b_k \tag{2}$$

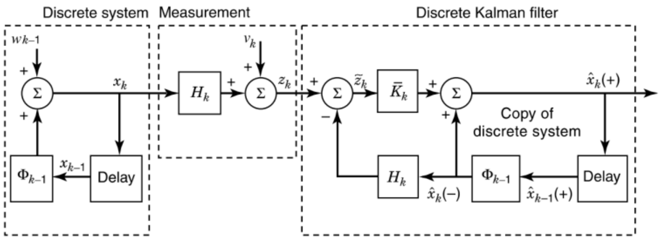

Fig. 1 Block Diagram of Data Acquisition System

I. KALMAN FILTER MATHEMATICAL FORMULATION

A. *Equations [6]*

1) *System dynamic model*

$$x_k = \Phi_{k-1} x_{k-1} + w_{k-1} \quad (3)$$

$$w_k = N(0, Q_k) \quad (4)$$

The above equations represent how our system is modeled. Φ is the state transition matrix. w_k is the process noise. It is assumed to be zero mean Gaussian noise

2) *Measurement Model*

$$z_k = H_k x_k + v_k \quad (5)$$

$$v_k = N(0, R_k) \quad (6)$$

It is assumed that measurement is related to state by the above equation, where H is the measurement sensitivity matrix and v_k the measurement noise. This is also assumed to be white Gaussian noise (zero mean).

3) *Initial Conditions*

$$E(x_0) = \hat{x}_k \quad (7)$$

$$E(\hat{x}_0 \hat{x}_0^T) = P_0 \quad (8)$$

P_0 is the priori covariance matrix. It is initialized as above. Expectation of x is assumed to be optimal estimate of initial value.

4) *Independence Assumptions*

$$E(w_k v_j^T) = 0 \text{ for all k and j} \quad (9)$$

The process noise and measurement noise are assumed to be independent of each other

5) *State Estimate Extrapolation*

$$\hat{x}_k = \Phi_{k-1} x_{k-1}(+) \quad (10)$$

This equation represents the prediction step of Kalman Filter. As we do not know about the particular value of noise signal and any other estimate of the system, we take the prediction value using our state transition matrix. The above can include control input as well, if necessary or required.

6) *Error Covariance Extrapolation*

$$P_k(-) = \Phi_{k-1} P_{k-1}(+) \Phi_{k-1}^T + Q_{k-1} \quad (11)$$

This step models the effect of time on covariance matrix of estimation certainty as a function of previous posteriori value P_{k-1}.

7) *State Estimate Observational Update*

$$\hat{x}_k(+) = \hat{x}_k(-) + \overline{K}_k[z_k - H_k \hat{x}_k(-)] \quad (12)$$

This equation gives the Kalman output of the current signal. K_k is the Kalman gain, which represents the relative weight of the past measurements, based on system modeling and on the current measure input through the sensor measurement

8) *Error Covariance Update*

$$P_k(+) = [I - \overline{K}_k H_k] P_k(-) \quad (13)$$

Error covariance is updated in this equation using Kalman gain and posteriori value. This implements the effect that conditioning on the measurement has on the covariance matrix of estimation uncertainty.

9) *Kalman Gain Matrix*

$$\overline{K}_k = P_k(-) H_k^T [H_k P_k(-) H_k^T + R_k]^{-1} \quad (14)$$

In this equation the Kalman Gain (K_k) is updated using the new values generated during this particular set of measurements.

As each of the parameters is recursively computed based on its previous value, the previous on its previous value till initial condition, the Kalman filter incorporates the information obtained by all the previous values in its prediction. It does so without actually storing that data and uses simple equations in a loop, making it computationally inexpensive. It is also necessary to point out that Kalman gain and error covariance equations are independent of actual observations. These parameters can be used to obtain preliminary information about the estimator performance. Since algorithm is recursive, it can be implemented easily as the dimension of the matrix doesn't increase with time and this is very helpful in the problems with multi-state or multidimensional problems.

II. SIMULATION SYSTEM PARAMETERS

A modeling of a constant acceleration system was done. The model chosen was Weiner process acceleration model. [7] This model requires

$$\Phi_k = e^{A\Delta t} = \begin{pmatrix} 1 & \Delta t & \Delta t^2 \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{pmatrix}$$

$$Q_k = q\begin{pmatrix} \frac{\Delta t^5}{20} & \frac{\Delta t^4}{8} & \frac{\Delta t^3}{6} \\ \frac{\Delta t^4}{8} & \frac{\Delta t^3}{3} & \frac{\Delta t^2}{2} \\ \frac{\Delta t^3}{6} & \frac{\Delta t^2}{2} & \Delta t \end{pmatrix}$$

$$\Delta t = 0.05$$

$$q = \sigma_q^2$$

$$H_k = \left[001\right]$$

$$R_k = \left[\sigma_r^2\right]$$

The assumption of constant acceleration is quite suitable for land vehicle dynamics as most of the time a land vehicle can be regarded as quasi-static in terms of acceleration. We will be examining the effects of this assumption on variable acceleration system also.

III. Simulation Results

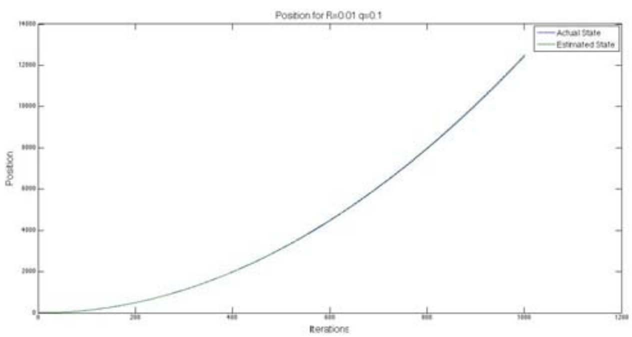

Fig. 2 Position is error-free for R=0.01 q=0.1

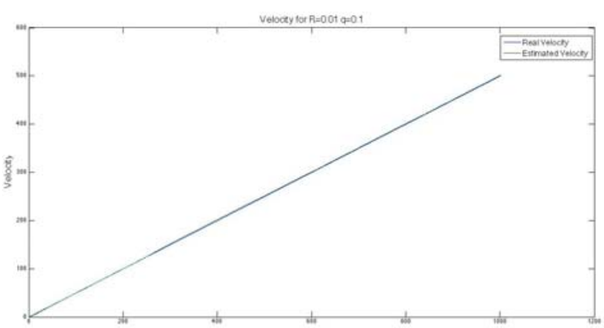

Fig. 3 Velocity is error-free for R=0.01 q=0.1

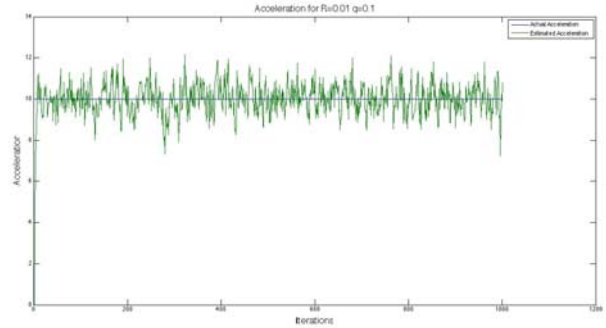

Fig. 4 Acceleration is noisy but with low convergence time for R=0.01 q=0.1

We found that the simulations gave adequate results for the system modeled. The simulations were run for various values of R/q. In the first case i.e. R/q=0.1, we found that velocity and position curves closely follow the actual position. However, the acceleration output was still noisy. However in the second case with R/q=2.8, we found that although the acceleration output is sufficiently noise free, an offset is present in the velocity and position curves. Hence there is a trade-off between predicting the acceleration and position, velocity correctly. Further the offset of velocity was found to be constant while the error in position coordinate was continually increasing with the number of iterations. This may be attributed to the difference in convergence time of acceleration w.r.t differing R/q values. Increasing R/q value is found to increase the convergence time. This leads to larger number of iterations for velocity to approach the correct value, hence contributing to the increasing offset.

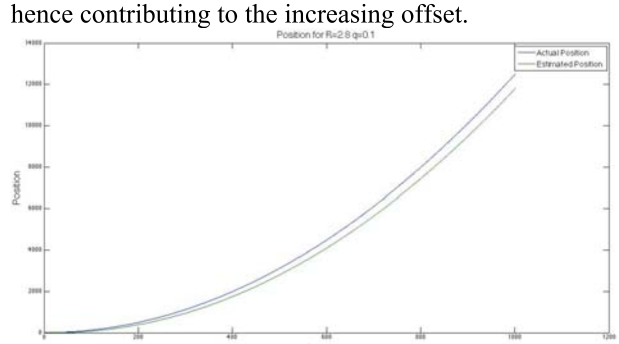

Fig. 5 Position shows increasing error for R=2.8 q=0.1

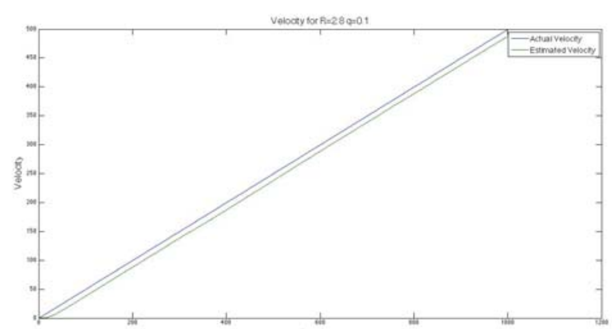

Fig. 6 Velocity shows constant offset for R=2.8 q=0.1

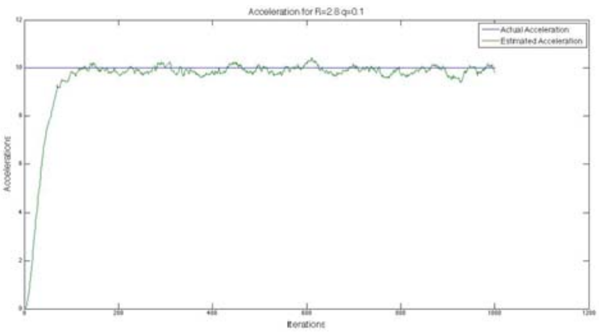

Fig. 7 Acceleration is noise free with high convergence time for R=2.8 q=0.1

We also used the same model for variable acceleration and found that it gave satisfactory results. The acceleration was assumed to be +10 for first 500 iterations and -10 for next 500 iterations. This approximates the acceleration of a body vertically thrown up in the air. Although for the cases where there is high acceleration changes in quick succession it is hard to predict the correct value using this model as the convergence time for the better results is significant and cannot be used for fast changing scenarios. This model thus in general can be used for bodies that do not show highly variable acceleration characteristics.

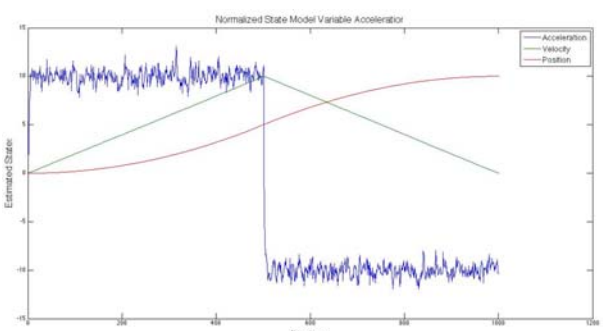

Fig. 8 Accurate prediction of velocity and position for variable acceleration for R=0.01 q=0.1 for variable acceleration

IV. Estimation of Process Noise (Q) and Measurement Noise(R) covariance

In the Kalman filter, the weight of the current data and recursively computed predicted value is calculated on the basis of two matrices supplied by the user. These matrices, represented in equation (4) and (6)

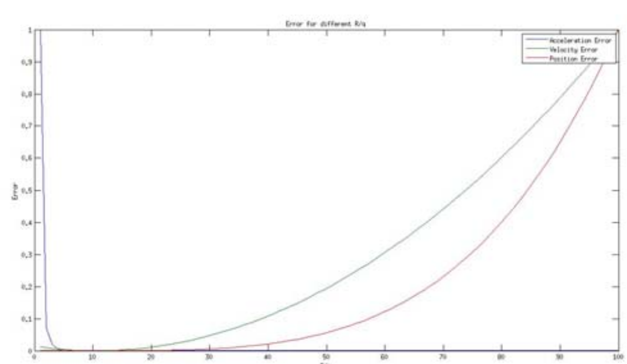

Fig. 9 Error versus R/Q

as Q and R, are process noise and measurement noise covariance respectively. In our model, we control Q through the parameter q, as Δt is sensor dependent. These matrices depend on noise in signal. However as we do not know the noise distribution exactly prior to the experiment, the estimation of Q and R can be done using various algorithms like Auto-covariance Least Square Method and Riccati's Equation. However, frequently hit and trial method is used.

V. Error Analysis And Optimizing Algorithm

Since in the Kalman Filter main problem lies in Optimizing and calculating the error with respect to the actual state, in this paper we have applied an Algorithm that considers both the convergence period i.e. the number of iterations and over the time deviation from the actual or true state. The error function used in this paper is:

$$\text{Error}(R/q) = \sum(\text{Estimated} - \text{Actual})^2$$

Based on the graph we might conclude that R/q value of 28 might be a good compromise on minimizing the errors of acceleration, velocity and position. However, based on the graphs found and practical scenarios it is usually more important to minimize position error, which reduces with reduction in R/q that was observed from the increasing convergence time on rising R/q values. This initial time lends to an offset in velocity and leads to error in velocity estimation. Variable acceleration cumulates this effect leading to a larger offset.

VI. Discussion

It is usually good practice to ensure low convergence time for acceleration at the cost of reducing the accuracy in prediction of acceleration. If we need best estimates for acceleration as well as position and velocity, an obvious approach would be to incorporate variable acceleration in such a way to counter the effects of the offsets. A different approach would be to apply a separate one-dimension Kalman filter on acceleration with different R/Q value that is used solely for the prediction of acceleration values. This solution requires less computational time at the cost of large overhead in modelling. If the implementation is offline or separate processors are available this computation can be carried out in parallel using different threads for the two Kalman filters resulting in faster computation times. However even in a single processor online implementation the calculation time is quite less for this model.

VII. Results and Conclusions

Based on simulation results we found that an accelerometer sensor based navigation system for one-dimensional movement gives good results even from a noisy sensor. In this way we can measure linear acceleration in three independent axes. This model can consequently be extended for gyroscopes. Combining all these in a single cohesive unit gives us an Inertial Navigation System.

REFERENCES

[1] R. E. Kalman, A new approach to linear filtering and prediction problems, Trans. ASME, J. Basic Engrg. ser. D, vol. 82, pp. 35-45, 1960

[2] J.J. Allemong, State Estimation Fundamentals for Successful Deployment, 2005 IEEE Power Engineering Society General Meeting, vol 1, pp. 800-801, June 2005.

[3] R.E. Larson, W.F. Tinney, J. Peschon, State Estimation in Power Systems: Part I: Theory and Feasibility, Vol PAS-89, Iss 3, pp. 345-352, March 1970

[4] Julier, S.J. , Unscented filtering and nonlinear estimation, Proceedings of the IEEE, Volume: 92 , Issue: 3, pp 401- 422, 2004

[5] Francois Caron, Emmanuel Duos, Denis Pomorski, Philippe Vanheeghe, GPS/IMU Data Fusion using Multisensor Kalman Filtering: Introduction of Contextual Aspects, Information fusion, Vol 7, pp. 221-230, 2006

[6] Grewal, M. S. and Andrews, A. P. (2002) References, in Kalman Filtering: Theory and Practice Using MATLAB ®, Second Edition, John Wiley & Sons, Inc., New York, USA. doi: 10.1002/0471266388.ref

[7] Y. Bar-Shalom, X. Rong Li, T. Kirubajan, Estimation with applications to tracking and navigation, Wiley-Interscience, 2001.