Investigation of Adaptive Robust Kalman Filtering Algorithms for GPS/DR Navigation System Filters

MOSTAFA Elzoghby\textsuperscript{1,}\textsuperscript{*}, USMAN Arif\textsuperscript{1}, FU Li\textsuperscript{1} and XI Zhi Yu\textsuperscript{1}

\textsuperscript{1}School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China

\textsuperscript{*}E-mail: mostafa@buaa.edu.cn

Abstract: The conventional Kalman filter (KF) algorithm is suitable if the characteristic noise covariance for states as well as measurements is readily known but in most cases these are unknown. Similarly robustness is required instead of smoothing if states are changing abruptly. Such an adaptive as well as robust Kalman filter is vital for many real time applications, like target tracking and navigating aerial vehicles. A number of adaptive as well as robust Kalman filtering methods are available in the literature. In order to investigate the performance of some of these methods, we have selected three different Kalman filters, namely Sage Husa KF, Modified Adaptive Robust KF and Adaptively Robust KF, which are easily simulate able as well as implementable for real time applications. These methods are simulated for land based vehicle and the results are compared with conventional Kalman filter. Results show that the Modified Adaptive Robust KF is best amongst the selected methods and can be used for Navigation applications.

1. Introduction

Since Kalman filter was established in 1960; it has been utilized in almost every prediction, estimation and filtering problem. It has proven its grounds in the field of inertial navigation, trajectory estimation, radar tracking, global navigation satellite systems (GNSS), Magnetic detection and Location as well as controllers and observer models [1].

After the initial work on Kalman filter, a number of variants of Kalman filters have been developed. The main categories include linear Kalman filters, adaptive Kalman filters, and nonlinear Kalman filters. But basic idea is same, estimate the state by fusing the information of measurements with linear/nonlinear state prediction equations. The fusion is performed based on the information provided by process noise matrix $Q$ and measurement noise matrix $R$ [2]. Therefore the $Q$ and $R$ matrices play a vital role in Kalman filtering state estimation. If $Q$ and $R$ are known, Kalman filter gives the optimal estimate. $R$ can be obtained using data sheets of the measurement sensors which provide general statistical properties for the class of sensors, while $Q$ is gathered analytically, by the experience of the designer. But if these are not chosen properly, the Kalman filter could lead to a non optimal solution. Therefore many modified Kalman filter algorithms have been proposed which can estimate the $R$ and $Q$ along with the states.

Usually the matrices $Q$ and $R$ are considered as constants, but in order to get optimal state, $Q$ and $R$ must adjust themselves according to the changing states as well as sensor accuracies, as sensors have different accuracies under different static and dynamic conditions. Such kind of Kalman filters are
called adaptive Kalman filters. Similarly if state is changing abruptly, e.g. in target tracking and motion estimation, robustness becomes more essential as than smoothing. The robustness can be obtained by changing Kalman gain $K$, Process covariance $P$, adjusting measurement noise matrix $R$, or combination of these. Such filters are called as robust Kalman filters.

For real time applications, adaptiveness as well as robustness is required to obtain optimal state. A number of adaptive and robust Kalman filters have been proposed. [3] Have devised a robust Kalman filter for outlier detection. It removes the outliers from robotic sensors data by introducing weights for each data sample. Similarly [4] has opted an adaptively robust Kalman filter for GPS/INS integration. This filter performs GPS/INS integration based on outlier detection. Another commonly used adaptive Kalman filter is Sage Husa which estimates optimal $Q$ and $R$ on alternate iterations. [5] Has utilized Sage Husa KF for INS initial alignment. Similarly in [6] have used Sage Husa KF to de-noise gyroscopic signal. Most applications of Sage Husa include static data, where robustness requirement is very less. When robustness is required, Sage Husa can diverge, instead of converging. This is because Sage Husa Kalman Filter is an adaptive filter, not a robust filter. Similarly, in [7] strong tracking algorithm is proposed which is robust but not adaptive. An adaptive robust algorithm is also proposed in [8]. But the problem with this method is that it is not real time implementable, [9] has used it and it took 96 hours to complete the simulation.

In our work, we have selected three Kalman filters namely Adaptive Sage Husa KF, Modified Adaptive Robust Kalman filter, which is based on Sage Husa KF for adaptivity and Strong Tracking KF for robustness, and another adaptively robust method given in [10]. They are implemented, and compared with linear/ conventional version of Kalman Filter. Data for navigation is collected via land based vehicle which has GPS, heading rate gyro and an odometer. The initial values for $Q$ and $R$ matrices are based on analytical assumptions only. These three filters are implemented and simulated. The results are compared to investigate the performance of each method for real time applications.

2. Formulations Material and Methods

2.1. GPS/DR Problem Description

Our problem is a ground vehicle that is travelling on surface of the earth with linear velocity $V$. The position of ground vehicle is described in NED reference frame, with $N_x$ denoting position of vehicle along true North and $N_y$ denoting position of vehicle along true East respectively. The yaw angle is denoted by $\psi$, which gives the angle between the North axis of NED reference frame and the vertical line along the length of the vehicle. The vehicle speed is measured by the odometer. The relationship between odometer and vehicle speed is:

$$ V = SN $$

Where $N$ is odometer readings in counts per second and $S$ is scale of odometer.

2.1.1 The state equations. The east position $E_x$, north position $N_y$, heading $\psi$, speed $v$, gyro rate $\dot{\psi}$, bias $B$ and odometer scale $S$ can be expressed in state space form as:

$$ E_x(k+1) = E_x(k) + V(k)T \sin(\psi(k)) + w_1 $$

$$ N_y(k+1) = N_y(k) + V(k)T \cos(\psi(k)) + w_2 $$

$$ \dot{\psi}(k+1) = \dot{\psi}(k) + T \dot{\psi}(k) + w_3 $$

$$ V(k+1) = V(k) + w_4 $$

$$ B(k+1) = B(k) + w_5 $$

$$ V(k) - N(k)S(k) $$

$$ S(k+1) = S(k) + \frac{V(k) - N(k)S(k)}{N(k)} + w_7 $$

A processes noise $w_k$ is white, zero-mean, uncorrelated, and has known covariance matrix $Q_k$. 

2.1.2 The observation equations. The sensors which attached on the body of the vehicle are GPS, Single axis gyro for yaw rotation measurement and odometer. Then, the measurements equations are expressed as:

\[
\begin{align*}
E_{gps} &= E_x(k) + v_1 \\
N_{gps} &= N_y(k) + v_2 \\
\psi_{gps} &= \psi(k) + v_3 \\
V_{speed}(k) &= n(k) \hat{S}(k) + v_4 \\
\hat{\psi}_{gyro}(k) &= \psi(k) + B + v_5
\end{align*}
\]

The measurement noise \( v_k \) is a zero mean white noise and the measurement noise covariance matrix \( R_k \).

2.2 Adaptive Sage Husa Kalman Filter
Sage Husa is one of the most commonly used methods. [5, 11] used it for initial INS alignment. [6] used it for de noising FOG measurements. In all these examples, the data and states are static and Sage Husa proves to be very efficient in estimating correct \( Q \) and \( R \) for static data. In case of dynamic data, where states are changing e.g in INS where vehicle is moving or in case of target tracking, where the target is changing its states quite fast, Sage Husa fails to deliver optimal values of \( Q \) and \( R \), and can lead to diverged states. We have also conducted a series of simulations to observe this. Here we are including Sage Husa for reference and it would be compared with the other adaptive filters later.

The Sage Husa consists of Sate prediction, State correction and Statistical properties \( Q \) and \( R \) update stages. The state prediction and correction are same as for linear Kalman filter.

2.2.1 Q and R Update on Every Iteration

\[
\begin{align*}
\hat{R}(k+1) &= [1 - d(k)] \hat{R}(k) + d(k)[\hat{Z}(k+1)\hat{Z}(k+1)^T] \\
&- H(k+1)P(k+1|k)H(k+1)^T
\end{align*}
\]

\[
\begin{align*}
\hat{Q}(k+1) &= [1 - d(k)] \hat{Q}(k) \\
&+ d(k)[K(k+1)\hat{Z}(k+1)\hat{Z}(k+1)^T]K(k+1)^T \\
&+ P(k+1) - \Phi(k+1|k)P(k|k)\Phi^T(k+1|k)
\end{align*}
\]

\[
\hat{r}(k+1) = [1 - d(k)] \hat{r}(k) + d(k)[Z(k+1) - H(k+1)\hat{X}(k+1|k)]
\]

\[
\hat{q}(k+1) = [1 - d(k)] \hat{q}(k) + d(k)[\hat{X}(k+1) - \Phi(k+1|k)\hat{X}(k)]
\]

Where \( d \) is attenuation factor and \( b \) is the forgetting factor, the value of \( d \) is given as

\[
d(k) = \frac{1 - b}{1 - b^{k+1}} \quad 0 < b < 1
\]

The selection of forgetting factor \( b \) is of prime importance. Whether suitable for selecting of forgetting factor is the key of the filter divergence and precision [12]. In general, trial and error analysis is used mostly; while some scholars use the optimal estimation theory in filter and iteratively find the forgetting factor in real time. The value of \( b \) is usually chosen between 0.95 to 0.99 [11]. But also another method has been proposed in [11] to estimate \( b \).

In case if data is static and also if we readily know \( Q \), we can quantify \( R \). Similarly if we know \( R \), we can Quantify \( Q \), using Sage Husa adaptive algorithm. If both are unknown, we can calculate \( Q \) and \( R \) using periodic switching between \( Q \) and \( R \) calculations. In our case, we have wrongly initialized \( Q \) and \( R \) initially, and let Sage Husa converge to optimal \( R \) and \( Q \) respectively.
2.3 Strong Tracking Kalman Filter

As previously described, the Sage Husa Adaptive Kaman filter can estimate optimal Q and R if new measurements have less information and system is less dynamic. Kalman filter works by calculating the optimal gain K which depends on R and P. In case if information/measurement is changing very fast, R and Q or may be P must be updated in such a way so that our Kalman filter could utilize the new measurement to give a real time effect. This can be done through following.

- Choose optimal $R$
- Choose optimal $P$
- Choose optimal $R$ and $P$

Strong tracking is an algorithm that can be used to update $P$ by depressing the accuracy so that the states are optimal as well as real time.

The generic algorithm for strong tracking algorithm is given as follows:

2.3.1 Prediction Stage.

$$\tilde{X}(k+1|k) = \Phi(k+1|k) \tilde{X}(k|k) + B(k)U(k) \quad (19)$$

When the measurements arrive, calculate the following:

$$\lambda(k+1) = \text{diag} [\lambda_4 (k+1), \lambda_2 (k+1), \lambda_3 (k+1), ..., \lambda_n (k+1)] \quad (20)$$
Where $\lambda_i(k+1)$ can be obtained as

$$
\lambda_i(k+1) = \begin{cases} 
    a_i C_{k+1} & \text{if } (a_i C_{k+1} > 1) \\
    1 & \text{if } (a_i C_{k+1} \leq 1)
\end{cases}
$$

(21)

Where $C_{k+1}$ can be found by

$$
C_{k+1} = \frac{\text{Tr}[V_{o(k+1)} - R_{(k+1)} - H_{(k+1)} Q_k H_{(k+1)}^T]}{\text{Tr}(\sum_{i=1}^{n} \alpha_i [\Phi_{k+1,k} P_k \Phi_{k+1,k}^T H_{k+1,k}^T H_{k+1,k} H_{k+1,k}^T])}
$$

(22)

And

$$
V_{o(k+1)} = \begin{cases} 
    Z_{k+1}^T Z_{k+1} T & \text{if } (k = 0) \\
    pV_{o(k)} + \bar{Z}_{k+1}^T \bar{Z}_{k+1} T & \text{if } (k \geq 1 \text{ and } 0 \leq p < 1)
\end{cases}
$$

(23)

From [7] the value of $p$ is chosen to be 0.95. And by Using the above variables, calculate $P_{k+1,k}$

$$
P_{k+1,k} = \lambda_{k+1,k} \Phi_{k+1,k} P_k \Phi_{k+1,k}^T + Q_k
$$

(24)

2.4 Hybrid between SAGE Husa and Strong Tracking (Modified Adaptive Robust KF)

Strong tracking is necessary when state is changing quickly and we need more information, while Sage Husa works under less dynamic conditions. Therefore we must check whether we must update $R$ and $Q$ using Strong tracking or Sage Husa. This check is provided by $\bar{Z}(k/k - 1)$ and $\bar{Z}(k/k - 1)\bar{Z}(k/k - 1)$ provides the mean square error (MSE). When $\bar{Z}(k/k - 1)$ is large, therefore the MSE is large, and we need strong tracking to make $\bar{Z}(k/k - 1)$ small, otherwise use Sage Husa.

$$
\bar{Z}(k,k - 1) = Z(k,k - 1) - HX(k,k - 1)
$$

(25)

**Figure 4.** Linear KF and Modified Adaptive Robust KF Results for absolute position error

**Figure 5.** Linear KF and Modified Adaptive Robust KF Results for absolute heading error
2.5 Adaptively Robust Kalman Filter

The adaptively robust Kalman Filter is described in [10] and is based on switching algorithm depending on adaptive factor $\alpha$ which decides either adaptiveness or robustness is required. Robustness is provided by least square method while adaptiveness is provided by modified equations of $P$ and $K$ where as $Q$ and $R$ are updated using Sage Husa Kalman Filtering equations. The complete equations for implementation are describe in [10] and are discussed below.

After prediction, estimate residual given by:

$$\tilde{Z}(k, k-1) = Z(k, k-1) - HX(k, k-1)$$

(26)

2.5.1 Calculate Learning Statistics $S$. Different learning statistic are given in the literature [10], based on different conditions. Here we took $\tilde{Z}(k+1)$ and $\tilde{X}(k+1|k)$ as two different observations, and calculate learning statistic $S$ based on Ratio of Variances method.

$$S = \frac{V_k' P_k V_k}{m V_k' P_k V_k}$$

(27)

2.5.2 Calculate $\alpha$. Adaptive factor $\alpha$ is calculated as an exponential function [10]. We have taken $c=1$.

$$\alpha = \begin{cases} 1 & S \leq c \\ e^{-(S-c)^2} & S > c \end{cases}$$

(28)

- If $\alpha = 0$, Use Least Square Estimation given by:

$$X = \frac{H^T Z}{H (P_k H)^T}$$

(29)

- If $\alpha = 1$, Use standard Kalman Filter for state correction

- If $0 < \alpha < 1$, Use adaptive filter, whose correction equations are given as:

$$P_k = \frac{1}{R} \quad \text{and} \quad P_{Xk} = \frac{1}{R}$$

(30)
\[ \hat{X}(k+1) = \left[ H^T (k+1) P_k H(k+1) + \alpha P_x \right]^{-1} \left[ H^T (k+1) P_k Z(k+1) + \alpha P_x \hat{X}(k+1|k) \right] \]
\[ K(k+1) = \frac{1}{\alpha} \left\{ P(k+1|k) H(k+1) \left[ \frac{1}{\alpha} H^T (k+1) P(k+1|k) H(k+1) + R(k+1) \right]^{-1} \right\} \]
\[ P(k+1|k+1) = P(k+1|k) - K(k+1) H(k+1) P(k+1|k) \]

3. Results and observations
The absolute position, velocity and heading are shown in the figures (1, 2 and 3) for Linear KF and Sage Husa. In figures (4, 5 and 6), a comparison of Linear KF and Modified Adaptive Robust KF is shown. Also in figures (7, 8 and 9) the comparison between Linear KF and Adaptive Robust KF is shown. The summarized results are shown in the Figure 10. The graph gives the complete overview of the results. As can be seen, the best position results are obtained by Sage Husa KF while it diverges in velocity calculation. The overall best results are obtained by the Modified Adaptive Robust KF which converges for all states and gives the best results. Adaptively Robust Kalman filter has same accuracy as linear Kalman filter. This is because the simulations are made for land based vehicle whose states don’t change abruptly, except for velocity. If states are changing fast like in the application of target tracking or aerial vehicle navigation, adaptively robust KF can prove efficient over linear Kalman filter.
Results clearly show that the Modified Adaptive Robust method gives the best convergence. Even if state changes fast like velocity or state is changing smoothly, like east and north positions, Modified Adaptive Robust Kalman filter converged to the optimal values of state. During our simulation work, we made following observations:

- While implementing Sage Husa Q and R must not be updated simultaneously but alternately on each iteration.
- Sage Husa works very well if either one of Q or R is known, and the other is updated.
- The best approach is to take static data with known covariance R. Use Sage Husa to update Q. Then use this Q and Sage Husa to update R for dynamic data.

If data is static, Sage Husa gives best convergence while if robustness is also required, modified adaptive must be used.

![Figure 10. Mean Square error Comparison between Filters](image)

4. Conclusion and Error Analysis
The results; the noise added to the Kalman filter is truly Gaussian, whose mean is zero. The gain K settles to a constant value, thus the system of conventional Kalman Filter in our case becomes a complementary filter. Therefore if we decrease Q and increase R, the system acts like an averaging low pass filter, thus smoothing out all the white Gaussian noise. This is due to additive nature of added noise that the conventional Kalman filter is giving us good results.

Analyzing Sage Husa Kalman Filter, the values for the variance of noise added to the North position, East Position, Heading and velocity are 8.5, 8.5, 1 x 10^{-5} and 0.1. As seen in the following figures, for initial wrong estimates of these parameters, the covariance values of these parameters converge approximately to the real values except heading. Heading converges to the correct value up till 200th iteration. At 200th iteration, there is some abnormality in the heading measurement. Therefore convergence of heading is lost here and covariance values in R for heading takes some abnormal value. As it is clear from the results of the Figure 10, heading error is more in case of Sage Husa KF as compared to other Kalman filters. The problem can be solved using some check for these abnormal values.

In case of Fusion between strong tracking and Sage Husa, the problem defined above is solved by using $\tilde{Z}(k/k - 1)\tilde{Z}(k/k - 1)$ as a check condition. When abnormality arrives, this term gets larger, the abnormality is taken as information and the Kalman Filter moves to strong tracking case. But shown in Figure 10, the absolute position error of this filter is a little larger as compared to the “only Sage Husa” case. This is because the values of R and Q had not been converged to the true values and filter switched to strong tracking algorithm. This is caused by limited data. Therefore it is strongly recommended that while implementing such a filter, only Sage Husa must be used until convergence of R and Q is achieved and then move to Sage Husa + Strong Tracking Filtering.
Adaptively Robust Kalman filter depends on Learning Statistics \( S \) and Adaptive Factor \( \alpha \). We have treated \( \tilde{Z}(k/k-1) \) and \( X(k/k-1) \) as two separate measurements and calculated \( s \) using ratio of variance components. The value of \( \alpha \) is calculated as an Exponential Function of \( s \). \( \alpha = 1 \) implies that \( \tilde{Z}(k/k-1) \) is not high and we should continue with conventional Kalman Filter. \( 0 < \alpha < 1 \) brings the filter into robust state. Kalman gain is increased by the factor of \( 1/\alpha \) and system relies more on measurements. The filter acts as a high pass filter, utilizing latest information. Thus in our case its accuracy is lower than conventional Kalman Filter. The Adaptively Robust Filter would perform better in the scenarios where much robustness is involved and states and measurements are varying abruptly.

**Table 1** Converged Values of Covariance in Three Selected Adaptive Filters

| Parameters          | Original Variances | Initialized Variance | Variance by Sage Husa | Variance by Strong Tracking+ Sage Husa | Variance by Adaptive Robust |
|---------------------|--------------------|----------------------|------------------------|----------------------------------------|----------------------------|
| North Position      | 9.4                | 50                   | 8.87                   | 2.24                                   | 10.56                      |
| East Position       | 8.9                | 50                   | 8.73                   | 1.82                                   | 10.40                      |
| Velocity            | 3.25e-06           | 0.01                 | 0.0082                 | 0.0013                                 | 0.0088                      |
| Heading             | 0.097              | 1                    | 0.27                   | 0.21                                   | 0.275                      |

**Figure 11.** Covariance \( R \) for North Position  
**Figure 12.** Covariance \( R \) for East Position  
**Figure 13.** Covariance \( R \) for Heading  
**Figure 14.** Covariance \( R \) for Velocity

Table 1 clarifies the Converged Values of Covariance in Three Selected Adaptive Filters. North and East position indicate low dynamic data while Velocity and Heading constitutes fast varying data. Therefore Combination of Fused Strong Tracking and Sage Husa can better estimates the covariance in fast varying data case while adaptive robust and Sage Husa have better convergence for slow...
varying data sets. In figures (11, 12, 13 and 14) we can notice the varying in the Covariance $\mathbf{R}$ for position, velocity and heading with respect to the three filters.

5. References

[1] Grewal M S and Andrews A P 2010 Applications of Kalman Filtering in Aerospace 1960 to the Present IEEE Control Systems Magazine 30 69–78.

[2] Welch G and Bishop G 2006 An Introduction to the Kalman Filter UNC-Chapel Hill, TR 95-041.

[3] Ting J A, Theodorou E and Schaal S 2007 A Kalman Filter for Robust Outlier Detection IEEE Conference on Intelligent Robots and Systems.

[4] He W, Lian B and Tang C 2014 GNSS/INS Integrated Navigation System Based on Adaptive Robust Kalman Filter Restraining Outliers International Conference on Communications in China.

[5] Xin S W 2014 Application of Sage-Husa Adaptive Filtering Algorithm for High Precision SINS Initial Alignment 11th International Computer Conference on Wavelet Active Media Technology and Information Processing.

[6] Narasimhappa M, Rangababu P, Sabat S L, Nayak J 2012 A modified Sage-Husa adaptive Kalman filter for denoising Fiber Optic Gyroscope signal Annual IEEE India Conference 1266 - 1271.

[7] Gupta M and Kumar R 2014 International Journal of Advance Research In Science And Engineering 3.

[8] Simon D J 2006 Optimal State Estimation (John Wiley & Sons, Inc).

[9] Kosanam S 2004 Kalman Filtering For Uncertain Noise Covariances, Cleveland State Univerisity.

[10] Yang Y 2010 Adaptively Robust Kalman Filters with Applications in Navigation (Springer-Verlag Berlin Heidelberg).

[11] Zhang H 2011 Application of a New Adaptive Kalman Filitering Algorithm in Initial Alignment of INS International Conference on Mechatronics and Automation, Beijing.

[12] D.Huang 1990 Inertial Navigation System (Publishing Company of National Industry Press).

Acknowledgements

This study is supported by a grant from the National Natural Fund of China, Grant No. 61375082.