AN ABSTRACT OF THE THESIS OF

Ratanak So for the degree of Master of Science in Electrical and Computer Engineering

presented on May 15, 2015.

Title: Extended Kalman Filter Simulink Model for Nonlinear System Modeling

Abstract approved: ____

Ted K.A. Brekken

The objective of the work presented herein is the development of the extended Kalman filter for nonlinear system modeling. A standard Kalman filter is a well-known filter for estimating the state of a system, assuming the system is linear and it has a Gaussian distribution in its noise. In reality, linear systems don't really exist. As a result, the standard Kalman filter is inadequate for modeling most systems. This need could be addressed by changing the standard Kalman filter to work in a nonlinear system. The extended Kalman filter Simulink model proposed in this work allows modeling in nonlinear systems through local linearization. This approach is validated by accurately estimating the nonlinear system behavior for a permanent magnet synchronous motor.

Extended Kalman Filter Simulink Model for Nonlinear System Modeling

by

Ratanak So

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented May 15, 2015
Commencement June 2015

Master of Science thesis of Ratanak So presented on May 15, 2015.

APPROVED:

Major Professor, representing Electrical and Computer Engineering

Director of the School of Electrical Engineering and Computer Science

Dean of the Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Ratanak So, Author

ACKNOWLEDGEMENTS

The author expresses sincere appreciation to Dr. Ted K.A. Brekken for his wonderful mentoring, his passion and enthusiasm for control and modeling of Wave Energy Converters (WECs), and his support and encouragement. In addition, the author would like to thank his colleagues for their kindness, helpfulness, time and support. Finally, a big thank you to his sister and family for always supporting and believing in him. Without their encouragement and support, this would not have been possible.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1: Introduction

The Kalman filter is named after Rudolph E. Kalman, who published his paper in 1960 [1] [2]. The filter is essentially a set of mathematical equations that implement a two stage estimator built from a predictor and corrector [2]. This estimator is optimal in the sense that it minimizes the estimated error covariance. The Kalman filter has proven to be a useful tool for estimating the variables of a wide range of processes, including the process variables used in system control [3] [4]. Although the filter is originally developed for use in spacecraft navigation systems, it is now widely used in many other systems including satellite navigation systems, computer vision applications, and object tracking software [5] [6] [7].

This thesis is divided into three sections: discussion of a standard Kalman filter, effect of nonlinear function on Gaussians, and discussion of an extended Kalman filter.

Chapter 2: The Standard Kalman Filter

A linear system is simply a process that can be described by the following equations [3] [4].

State Equation:

$$x_{k+1} = Ax_k + Bu_k + w_k \tag{2.1}$$

Output Equation:

$$y_k = Cx_k + v_k \tag{2.2}$$

In the above equations, A, B, and C are constant matrices, k is the time index, x is the state of the system, u is the input to the system, and y is the measured output. The variables w and v are defined as the process noise and the measurement noise, respectively. The noise variables are in vector format containing more than one element. Although the x vector contains all of the information about the present states of the system, only some states can be measured and those states have noise. The measured states are the elements of y vector.

The Kalman filter algorithm involves two stages: a prediction stage and a measurement update stage. The standard Kalman filter equations for the prediction stage are shown in the following [8]:

$$K_k = P_k C^T (C P_k C^T + R)^{-1} \tag{2.3}$$

$$\hat{x}_{k+1} = (A\hat{x}_k + Bu_k) + K_k(y_k - C\hat{x}_k) \tag{2.4}$$

$$P_{k+1} = A(I - K_k C) P_k A^T + Q \tag{2.5}$$

Where P is the estimation error covariance, R is the measurement noise covariance, K is the Kalman gain, and Q is the process noise covariance. Equations (2.3), (2.4), and (2.5) are the Kalman gain, the state estimation, and the estimation error covariance. Examination of (2.3) illustrates the adaptive nature of the filter, where the value of K

determines the accuracy of the measurement. If the measurement noise is large, R will be large, so K will be small, indicating a large error in the next estimate. On the other hand, if the measurement noise is small, R will be small, K will be large and the next estimate is more accurate.

The state estimation equation given in (2.4) consists of two terms. The first term, used to derive the state estimate at time $k+1$, is A times the state estimate at time k plus B times the known input (disturbance) at time k. This value would be the state estimate if the measurement is unavailable. The second term of (2.4) is the correction term, representing the amount by which to correct the updated state estimate due to the available measurement.

Chapter 3: The Effect of Nonlinear Function on Gaussians

When a Gaussian distribution is mapped through a linear function, the output is always a Gaussian [9].

For example, a Gaussian distribution with mean $\bar{x}$ and covariance σ_{xx} can be written as below:

$$x \sim N(\bar{x}, \sigma_{xx})$$

A linear transformation $y = Ax + b$ is used to determine whether an output is still a Gaussian, as supported in the following proof:

The mean of y:

$$\bar{y} = E\{y\} = E\{Ax + b\} = AE\{x\} + b = A\bar{x} + b$$

The covariance of y:

$$\begin{gathered} \sigma_{yy} \triangleq E\{(y - \bar{y})(y - \bar{y})^T\} \\ = E\{[(Ax + b) - (A\bar{x} + b)][(Ax + b) - (A\bar{x} + b)]^T\} \\ = E\{[A(x - \bar{x})][A(x - \bar{x})]^T\} \\ = E\{A(x - \bar{x})(x - \bar{x})^T A^T\} \\ = AE\{(x - \bar{x})(x - \bar{x})^T\}A^T \\ = A\sigma_{xx}A^T \end{gathered}$$

y is defined as:

$$y \sim N(A\bar{x} + b, A\sigma_{xx}A^T)$$

Since the standard Kalman filter is a linear system and it has a Gaussian distribution in its noise, it is imperative that the Gaussian distribution be maintained from input to output as stated and proved above [4]. For a linear system, this criteria is met. However for a nonlinear system, the output is no longer Gaussian. Therefore, the standard Kalman filter algorithm is less accurate due to the fact that the output is no longer Gaussian.

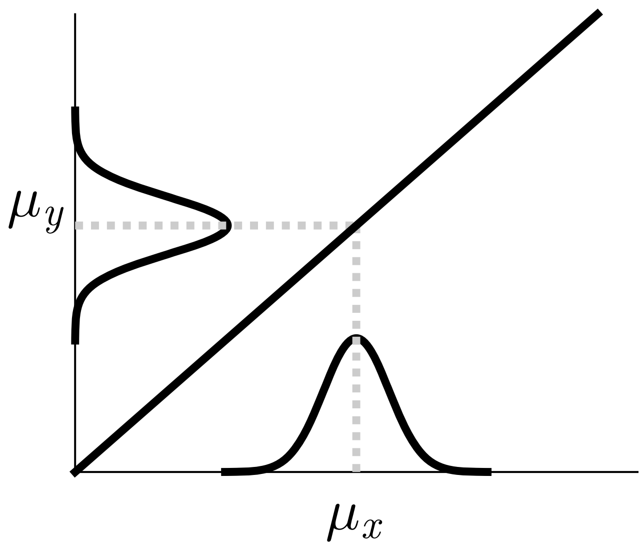

Figure 3.1: This figure shows the effect of a linear transformation. A Gaussian distribution is maintained when it is mapped through a linear function.

This fundamental understand is very important for modelers because it separates those who understand why and when to use the extended Kalman filter and those who don't.

One solution to resolve this nonlinearity issue is to utilize local linearization, where the basic concept is to take the best estimate value of a function and then linearize around that best estimate, discussed further in the following section.

Chapter 4: The Extended Kalman Filter

4.1 Taylor Series Expansion

The key to nonlinear Kalman filtering is to expand the nonlinear terms of the system equation in a Taylor series expansion [10] around a nominal point. A Taylor series expansion of a nonlinear function can be written as [8]:

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(\bar{x}) \triangle x^n}{n!} = f(\bar{x}) + f'(\bar{x}) \triangle x + \frac{f''(\bar{x}) \triangle x^2}{2} + ... \tag{4.1}$$

Linearizing a function means expanding "first-order" Taylor series around some expansion point.

The first-order Taylor series expansion of a function $f(x)$ is equal to:

$$f(x) \approx f(\bar{x}) + f'(\bar{x}) \triangle x \tag{4.2}$$

This approximation is more accurate with smaller $\triangle x$.

4.2 The Linearized Kalman Filter

In the traditional linearized Kalman filter, a first-order Taylor series is used to expand the state equation and output equation around a nominal state. The nominal state is a function of time, so it's sometimes called a trajectory. The nominal trajectory is based on a guess of what the system behavior might look like. For example, if the system equations represent the dynamics of an airplane, the nominal state might be the planned flight trajectory. The actual flight trajectory may differ from the nominal trajectory due to disturbances, and other factors. If the actual trajectory is close to the nominal trajectory, the Taylor series linearization is reasonably accurate [8].

The linearized Kalman fiter can be derived using the concept of Taylor series. The nonlinear system is matched to a linear system whose states represent the deviations from a nominal trajectory of the nonlinear system. Then, the filter can be used to estimate the deviations from the nominal trajectory. This calculation indirectly gives an estimate of the states of the nonlinear system. Below is a general nonlinear system model [8]:

State equation:

$$x_{k+1} = f(x_k, u_k) + w_k \tag{4.3}$$

Output equation:

$$y_k = h(x_k) + v_k \tag{4.4}$$

The state equation $f(\cdot)$ and the measurement $h(\cdot)$ are nonlinear functions. The Taylor series linearization of the nonlinear system equation and output equations are shown below [8]:

$$x_{k+1} = f(x_k, u_k) + w_k \approx f(\bar{x}_k, u_k) + f'(\bar{x}_k, u_k) \triangle x_k + w_k \tag{4.5}$$

$$y_k = h(x_k) + v_k \approx h(\bar{x}_k) + h'(\bar{x}_k) \triangle x_k + v_k \tag{4.6}$$

$$\triangle x_{k+1} = f'(\bar{x}_k, u_k) \triangle x_k + w_k \tag{4.7}$$

$$\triangle y_k = h'(\bar{x}_k) \triangle x_k + v_k \tag{4.8}$$

There are two important points to remember when using the linearized Kalman filter [8]:

1. After the Kalman filter is used to estimate $\triangle x$, the estimate of $\triangle x$ needs to be added to the nominal state $\bar{x}$ in order to get an estimate of the state x. This is because $\triangle x = x - \bar{x}$ [11].

2. If the true state x gets too far away from the nominal state $\bar{x}$, then the linearized Kalman filter will not give good results because neglected higher order terms became more significant.

4.3 The Extended Kalman Filter's Algorithm

The traditional linearized Kalman filter derived above works for systems with a predetermined nominal trajectory. However, for systems where a nominal trajectory is not known, this approach is not valid. An extended Kalman filter addresses this issue by using an estimate of x as the nominal trajectory in the linearized Kalman filter [8].

Below is the extended Kalman filter's algorithm [8]:

State equation:

$$x_{k+1} = f(x_k, u_k) + w_k \tag{4.9}$$

Output equation:

$$y_k = h(x_k) + v_k \tag{4.10}$$

At each time step, compute the following derivative matrices, evaluated at the current state estimate:

$$A_k = f'(\hat{x}_k, u_k)$$
$$C_k = h'(\hat{x}_k)$$

Execute the following Kalman filter equations:

$$K_k = P_k C_k^T (C_k P_k C_k^T + R)^{-1} \tag{4.11}$$

$$\hat{x}_{k+1} = f(\hat{x}_k, u_k) + K_k(y_k - h(\hat{x}_k)) \tag{4.12}$$

$$P_{k+1} = A_k(I - K_k C_k) P_k A_k^T + Q \tag{4.13}$$

4.4 The Extended Kalman Filter for a Motor State Estimation

To illustrate the use of the extended Kalman filter, a two-phase permanent magnet synchronous motor is modeled, where the filter is used to estimate the speed and position of the rotor by only using measurements of the motor voltages and currents [8] [12].

The system equations are [8] [12]:

$$\dot{I}_a = \frac{-R}{L} I_a + \frac{\omega\lambda}{L} sin(\theta) + \frac{u_a + \triangle u_a}{L} \tag{4.14}$$

$$\dot{I}_b = \frac{-R}{L} I_b + \frac{\omega\lambda}{L} cos(\theta) + \frac{u_b + \triangle u_b}{L} \tag{4.15}$$

$$\dot{\omega} = \frac{-3\lambda}{2J} I_a sin(\theta) + \frac{3\lambda}{2J} I_b cos(\theta) - \frac{F\omega}{J} + \triangle\alpha \tag{4.16}$$

$$\dot{\theta} = \omega \tag{4.17}$$

$$y = \begin{bmatrix} I_a \\ I_b \end{bmatrix} + \begin{bmatrix} v_a \\ v_b \end{bmatrix} \tag{4.18}$$

Nomenclature

I_a and I_b are the current in the two windings
θ is the angular position of the rotor
ω is the velocity of the rotor
R is the motor winding's resistance
L is the motor winding's inductance
λ is the flux constant of the motor
F is the coefficient of viscous friction that acts on the motor shaft and its load
J is the moment of inertia of the motor shaft and its load
u_a and u_b are the voltage that are applied across the two motor windings
$\triangle u_a$ and $\triangle u_b$ are noise terms due to errors in u_a and u_b
$\triangle\alpha$ is a noise term due to uncertainty in the load torque
y is the measurement.

The model assumes that the two winding currents can be measured. The measurements are distorted by measurement noises v_a and v_b, which are caused by events such as electrical noise, and quantization errors in the microcontroller [8].

In order to apply the extended Kalman filter to the motor, first step is to define the states of the system. If a variable is differentiated in the system equations, that quantity is a state. From (4.14)-(4.17), the system has four states and the state vector x can be defined as:

$$x = \begin{bmatrix} I_a \\ I_b \\ \omega \\ \theta \end{bmatrix} \tag{4.19}$$

$$x_{k+1} = f(x_k, u_k) + w_k \tag{4.20}$$

The system equation is obtained by "discretizing" the differential equations to obtain (4.21) below where $\triangle t$ is the step size used for estimation in the microcontroller or DSP.

$$x_{k+1} = x_k + f(x_k, u_k) \triangle t + w_k \triangle t \tag{4.21}$$

$$\begin{aligned} x_{k+1} \quad = \quad & x_k + \\ & \begin{bmatrix} \frac{-R}{L} x_k(1) + \frac{\lambda sin x_k(4)}{L} x_k(3) + \frac{1}{L} u_{ak} \\ \frac{-R}{L} x_k(2) + \frac{\lambda cos x_k(4)}{L} x_k(3) + \frac{1}{L} u_{bk} \\ \frac{-3\lambda sin x_k(4)}{2J} x_k(1) + \frac{3\lambda cos x_k(4)}{2J} x_k(2) - \frac{F}{J} x_k(3) \\ x_k(3) \end{bmatrix} \\ & * \triangle t + \\ & \begin{bmatrix} \frac{\triangle u_{ak}}{L} \\ \frac{\triangle u_{bk}}{L} \\ \triangle \alpha \\ 0 \end{bmatrix} \triangle t \end{aligned} \tag{4.22}$$

$$y_k = h(x_k) + v_k \tag{4.23}$$

$$y = \begin{bmatrix} x_k(1) \\ x_k(2) \end{bmatrix} + \begin{bmatrix} v_{ak} \\ v_{bk} \end{bmatrix} \tag{4.24}$$

To use the filter, compute the derivatives of $f(\hat{x}_k, u_k)$ and $h(\hat{x}_k)$ with respect to $\hat{x}_k$. The vectors A_k and C_k as can be written as:

$$A_k = f'(\hat{x}_k, u_k) = \begin{bmatrix} \frac{\partial f_1}{\partial \hat{x}_1} & \frac{\partial f_1}{\partial \hat{x}_2} & \frac{\partial f_1}{\partial \hat{x}_3} & \frac{\partial f_1}{\partial \hat{x}_4} \\ \frac{\partial f_2}{\partial \hat{x}_1} & \frac{\partial f_2}{\partial \hat{x}_2} & \frac{\partial f_2}{\partial \hat{x}_3} & \frac{\partial f_2}{\partial \hat{x}_4} \\ \frac{\partial f_3}{\partial \hat{x}_1} & \frac{\partial f_3}{\partial \hat{x}_2} & \frac{\partial f_3}{\partial \hat{x}_3} & \frac{\partial f_3}{\partial \hat{x}_4} \\ \frac{\partial f_4}{\partial \hat{x}_1} & \frac{\partial f_4}{\partial \hat{x}_2} & \frac{\partial f_4}{\partial \hat{x}_3} & \frac{\partial f_4}{\partial \hat{x}_4} \end{bmatrix} \tag{4.25}$$

$$C_k = h'(\hat{x}_k) \tag{4.26}$$

Simulink Model

The Simulink model that illustrates the use of the extended Kalman filter for the state estimation of a permanent magnet synchronous motor is shown in Fig. 4.1.

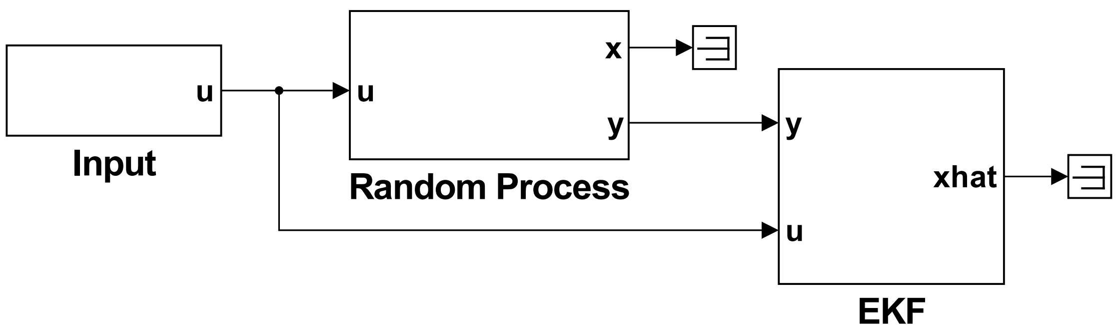

Figure 4.1: Simulink model of the extended Kalman filter. The Input represents the voltages u_a and u_b. The Random Process represents the real states of the dynamic system $(I_a, I_b, \omega, \theta)$. For modeling purposes, noise is added to these real states to become the measured states $(I_a + v_{ak}, I_b + v_{bk})$ available to the extended Kalman filter. Finally, the EKF represents the extended Kalman filter's algorithm that estimates the system states $(\hat{I}_a, \hat{I}_b, \hat{\omega}, \hat{\theta})$.

The subsystems of "Random Process" and "EKF" are shown in Figs. 4.7 and 4.8, respectively.

In this example, phase a and b voltages are represented in a "Input" block. The input vector u is then sent to the "Random Process" block and outputs: 1) a state matrix x which consists of the true currents I_a & I_b, the true rotor speed ω, and the true rotor position θ, 2) an output state matrix y which is the measurement of the currents I_a & I_b that contaminated by noise.

Inputs that go to the extended Kalman filter "EKF" block are voltage vector u and measurement current vector y. The filter outputs the estimated state $\hat{x}$ that contains the estimation of the current $\hat{I}_a$ & $\hat{I}_b$, the estimation of rotor speed $\hat{\omega}$, and the estimation of rotor position $\hat{\theta}$.

Table 4.1 shows the motor's parameters with their values used in the model. A normal (Gaussian) distributed random signal with zero mean and $0.05A$ standard deviation is used for the measurement noise.

Table 4.1: Motor's Variables

Parameter	Description	Value
R	Resistor	2 Ω
L	Inductor	3 mH
J	Moment of Inertia	2e-3 kgm^2
F	Coefficient of viscous friction	1e-3
$ContNoise$	Std dev of uncertainty in control inputs	1 mA
$MeasNoise$	Std dev of measurement noise	50 mA
dt	Simulation time step	0.2 ms
dT	Measurement time step	1 ms

Simulation Results

Figs. 4.2 and 4.3 show the real and measured currents I_a and I_b. The measurements of currents I_a and I_b are noisy but they follow the dynamic of the real currents.

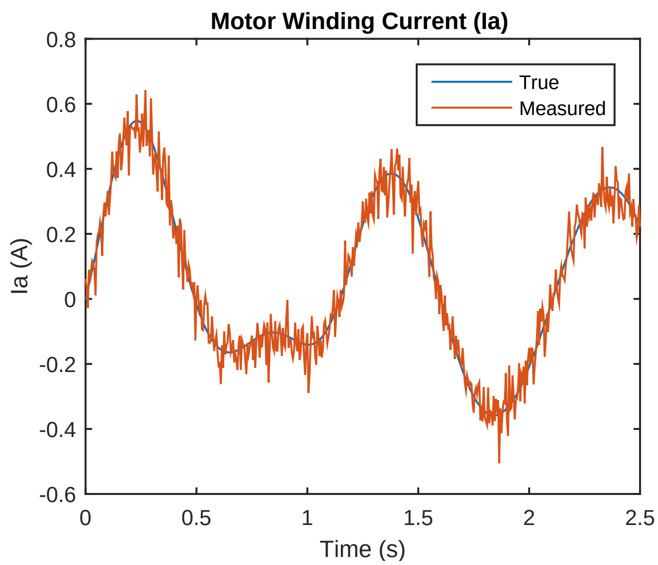

Figure 4.2: Real and measured motor winding current (I_a). The standard deviation of measurement noise is $0.05A$.

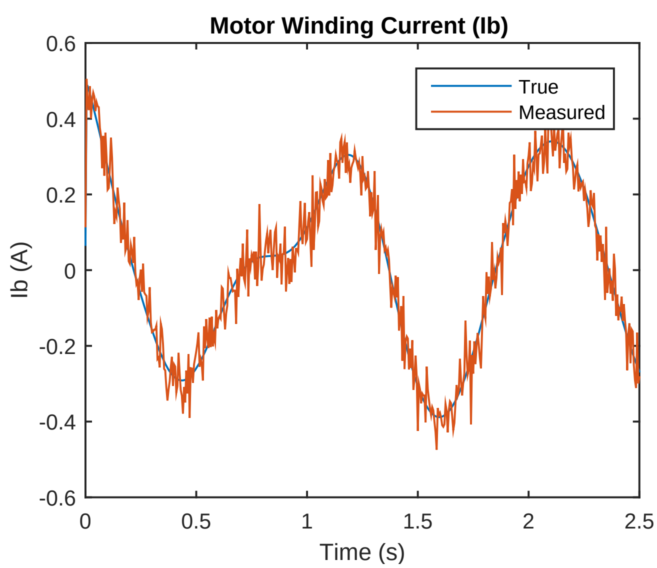

Figure 4.3: Real and measured motor winding current (I_b).

The unique aspect of this example is the rotor position and velocity can be estimated and they accurately matched the true values as shown in Figs. 4.4 and 4.5 without using an encoder. The Simulink model also adds an additional function to keep the rotor position between 0 and 2π. This method is a good practice because it allows the rotor position to display within one graph.

It should be noted that the assumption is the sense resistors are used and the extended Kalman filter is running on a microcontroller. With this filter, money can be saved from avoiding an encoder.

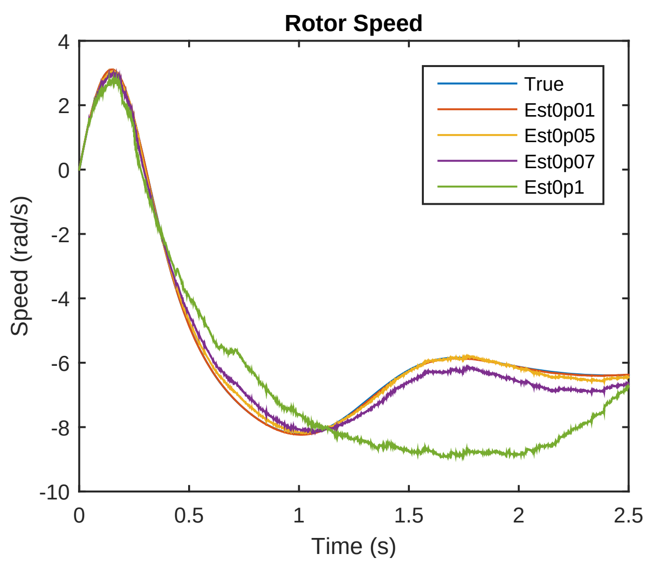

Figure 4.4: This figure shows the real and estimated rotor speed with different measurement noise. As the standard deviation of the measurement noise becomes larger, the estimated rotor speed is less accurate. For example, $0p1$ means the standard deviation of the measurement noise is $0.1A$.

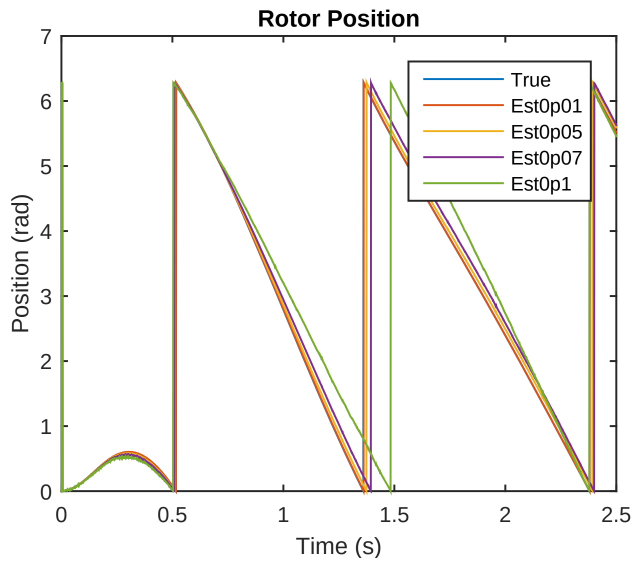

Figure 4.5: This figure shows the real and estimated rotor position. The Simulink model has a subsystem that keeps the rotor position between 0 and 2π. The estimation is less accurate as the measurement noise is stronger.

Fig. 4.6 below shows the trace of estimation error covariance. This plot reflects the accuracy of the measurement currents I_a and I_b. For example, $Est0p01$ represents the trace of estimation error covariance when the standard deviation of the measurement noise is set to $0.01A$. The estimation error covariance is more severe when the measurement becomes noisy. It is interesting to note that between 0.3 and 0.4 seconds, the error is stronger because of a rapid change in the dynamic system. Meanwhile, when the system approaches the steady state, the presence of the error is reduced.

Table 4.2: The Comparison Between the two Filters

The Standard Kalman Filter	The Extended Kalman Filter
Kalman Gain	
$K_k = P_k C^T (C P_k C^T + R)^{-1}$	$K_k = P_k C_k^T (C_k P_k C_k^T + R)^{-1}$
Update the Estimate via Measurement	
$\hat{x}_{k+1} = (A\hat{x}_k + Bu_k) + K_k(y_k - C\hat{x}_k)$	$\hat{x}_{k+1} = f(\hat{x}_k, u_k) + K_k(y_k - h(\hat{x}_k))$
Update the Error Covariance	
$P_{k+1} = A(I - K_k C) P_k A^T + Q$	$P_{k+1} = A_k(I - K_k C_k) P_k A_k^T + Q$

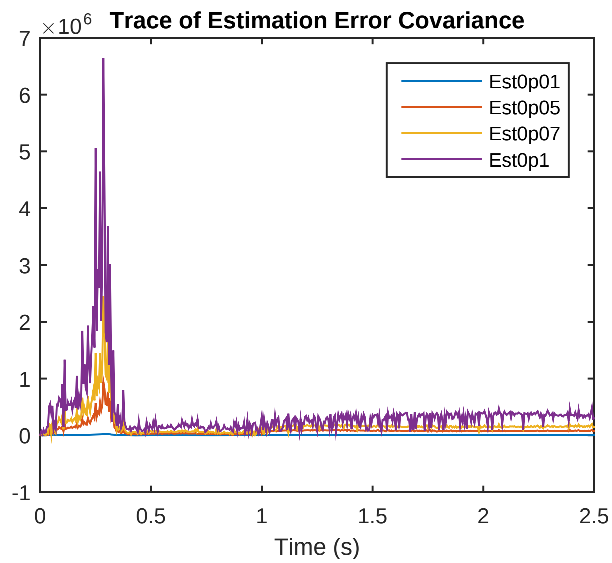

Figure 4.6: This figure shows the trace of estimation error covariance. When the standard deviation of the measurement noise is small (e.g., $0.01A$), labeled $Est0p01$, the estimation error is also small. On the other hand, if the standard deviation of the measurement noise is large (e.g., $0.1A$), the estimation error is large.

It is important to recognize the limitations of the extended Kalman filter. This filter only works if the sample rate and the measurement noise are small enough. In other words, if the measurement data is poor and the sample rate is too big, it is impossible for the filter to estimate the system state accurately.

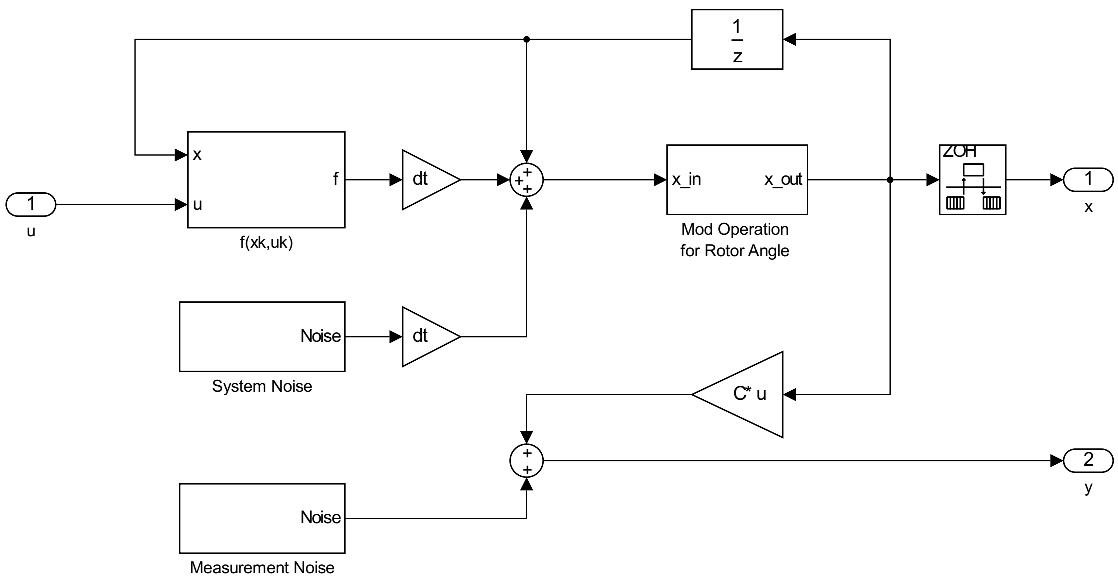

Figure 4.7: This figure shows the Simulink model of the random process. This block takes input u which represents the a and b voltages and models both system and measurement noise. The outputs of this system are the true system states x and the measurement states y.

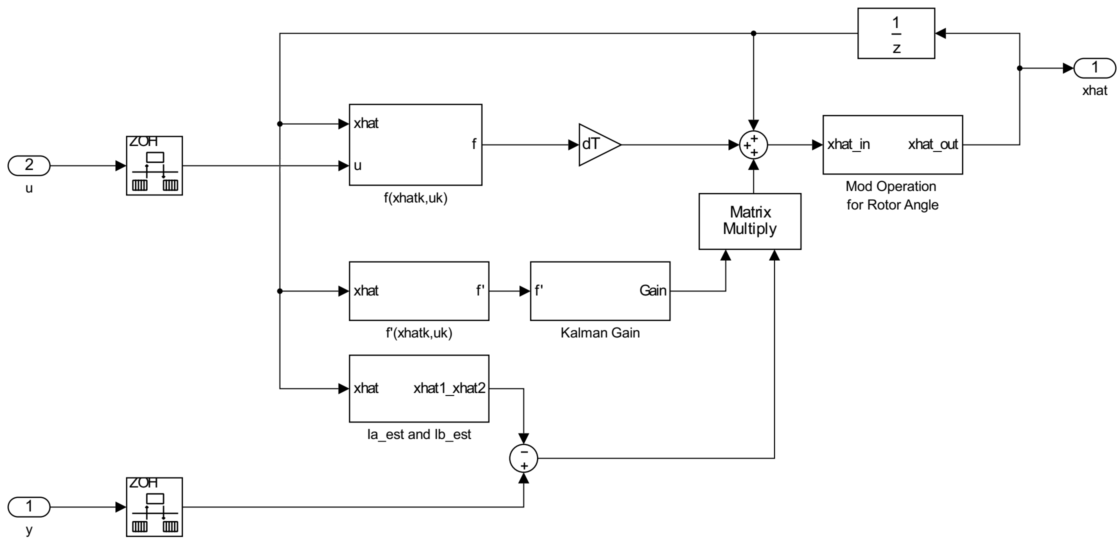

Figure 4.8: This figure shows the Simulink model of the extended Kalman filter's algorithm. This block needs to know the measurement of currents I_a & I_b and true voltages u_a & u_b in order to estimate all the states of the system.

4.5 A Comparision Between the two Filters

Table 4.2 shows a comparison between the two filters. A standard Kalman filter is transformed to an extended Kalman filter by:

- ***Kalman gain:*** Replaced C by C_k
- ***Update the estimated state:*** Replaced $A\hat{x}_k + Bu_k$ by $f(\hat{x}_k, u_k)$ and $C\hat{x}_k$ by $h(\hat{x}_k)$
- ***Update the error covariance:*** Replaced A by A_k and C by C_k

Chapter 5: Conclusion

This thesis provides an in-depth explanation to how each filter works and a practical example of an electrical engineering application. A standard Kalman filter which was originally developed in the 1960s is used for applications that the systems are linear. On the other hand, most applications, including the example of the permanent magnet synchronous motor, have nonlinear systems. Therefore, a standard Kalman filter is no longer valid for these nonlinear systems. Due to a great need to work with nonlinear systems, the standard Kalman filter is modified by applying a linearization method based on a Taylor series expansion and this filter is then known as the extended Kalman filter. Both filter algorithms can be coded in various platforms. The Simulink model is chosen because it provides a great visualization to illustrate the real process and the filter itself.

Bibliography

[1] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Journal of Fluids Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[2] G. Welch and G. Bishop, “Course 8–an introduction to the kalman filter,” *SIGGRAPH 2001 Courses*, 2001.

[3] D. Simon, “Kalman filtering,” *Embedded Systems Programming*, vol. 14, no. 6, pp. 72–79, 2001.

[4] R. Faragher *et al.*, “Understanding the basis of the kalman filter via a simple and intuitive derivation,” 2012.

[5] M. Rhudy and Y. Gu, “Understanding nonlinear kalman filters, part ii: An implementation guide,” *Interactive Robotics Letters*, 2013.

[6] E. Kamen and J. Su, “Kalman filter applications,” in *Introduction to Optimal Estimation*, pp. 225–267, Springer, 1999.

[7] G. W. Morrison and D. H. Pike, “Kalman filtering applied to statistical forecasting,” *Management Science*, vol. 23, no. 7, pp. 768–774, 1977.

[8] D. Simon, “Using nonlinear kalman filtering to estimate signals,” *Embedded Systems Design*, vol. 19, no. 7, p. 38, 2006.

[9] “Using matlab for teaching transformations of random variables,” in *Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on*, pp. 1–4, Aug 2013.

[10] M. Abramowitz and I. A. Stegun, *Handbook of mathematical functions: with formulas, graphs, and mathematical tables.* No. 55, Courier Corporation, 1964.

[11] A. Jordan, “Linearization of non-linear state equation,” 2006.

[12] R. Dhaouadi, N. Mohan, and L. Norum, “Design and implementation of an extended kalman filter for the state estimation of a permanent magnet synchronous motor,” *Power Electronics, IEEE Transactions on*, vol. 6, no. 3, pp. 491–497, 1991.