

Nonlinear Filtering Methodologies for Parameter Estimation

Brett Matzuka

Mikio Aoi

Adam Attarian

Hien Tran

Department of Mathematics
North Carolina State University, Raleigh, NC 27607
Phone: 808-896-9965 Email: bjmatzuk@ncsu.edu

July 19, 2012

Abstract

Filtering is a methodology used to combine a set of observations with a model to get the optimal state. This technique can be extended to not just estimate the state of the system, but also the unknown model parameters. Estimating the model parameters given a set of data is often referred to as the inverse problem. Filtering provides many benefits to the inverse problem by providing estimates in real time and allowing model errors to be taken into account. Assuming a linear state space and Gaussian noises, the optimal filter is the Kalman filter. However, these assumptions rarely hold for many problems of interest, so a number of extensions have been proposed in the literature to deal with nonlinear dynamics. To determine the best method for a given problem, we do a comprehensive comparison study of five filtering methods in the estimation of the state of the system and its unknown model parameters. The performance of the methods are tested across several test problems which cover a wide range of potential issues that can be encountered including numerical stiffness, the size of system and parameters, chaotic dynamics, the nature of nonlinearities, the type of error model, the data sampling frequency, and other issues. These examples are used to give recommendations as to which filter is best under the various conditions.

1 Introduction

The estimation of the state of a system given a set of observations on the system is ubiquitous for many problems in science. Given that a mathematical model is an approximation of the true dynamics of the underlying system and that any measurement of the system dynamics is noisy in general, we wish to find an optimal method to combine these so as to get the most accurate estimation of the state of the system, and any model parameters. The optimal method to do this would be the Bayes filter; however, in general, the solution of the Bayes filter can rarely be solved analytically, and numerical approximation is often intractable, except in a small set of restrictive cases [2].

One such special case of the Bayes filter is the celebrated Kalman filter [16]. Assuming that the model is linear and that the errors in both the model and the observations are Gaussian, the Bayes filter can be shown to simplify to the Kalman filter. The Kalman filter is a recursive algorithm which calculates the optimal state of the system by taking a weighted average of the probability distribution from the model and the probability distribution from the measurement. It is deterministic in nature and characterizes the entire optimal estimate through the propagation of the mean and covariance of the estimate at each step. However, if these restrictive assumptions do not hold and the model dynamics are nonlinear or the noise distributions are non-Gaussian,

the Kalman filter fails and adjustments have to be made to account for this. Several variants of the Kalman filter have been developed to overcome these shortcomings.

One such method is to approximate the nonlinear model dynamics by a linearization about the current state. This linear model is then propagated forward under the Kalman filter equations along with the observations and is used to approximate the optimal mean and covariance for the state of the system. This still requires the assumption of Gaussian noise to hold for both the model and measurements, which may not be true for the model as a normal distribution is not maintained through a nonlinear transformation. This approach is known as the Extended Kalman Filter (EKF) [7, 8, 13, 17].

Another approach to deal with the nonlinear model dynamics is to do a statistical linearization [6]. Instead of linearizing the model dynamics, we perform a linearization upon the distribution itself by carefully choosing a set of sigma points that characterize the distribution and capture the important features. By propagating these points through the unscented transform, we get an accurate representation of the posterior distribution. This is done for both the model and observations, and the resulting distributions are used in the classical Kalman filter equations. This approach still assumes Gaussian distributions and is deemed the Unscented Kalman Filter (UKF) [6, 13, 15, 21, 23].

The last of the deterministic approaches is derived by applying the Gaussian distribution to the Bayes filter for any nonlinear function. Exploiting the properties of the highly efficient cubature numerical integration technique for the multi-dimensional integral given in the Bayes filter, we get a variation of the Kalman filter. Much like the Unscented Kalman Filter, this method requires the calculation of cubature points to characterize the integrals, which are used to calculate the distributions more accurately, and finally used in the classical Kalman filter equations. This approach is derived from the Gaussian assumption and is known as the Cubature Kalman Filter (CKF) [2].

Another set of approaches has been derived by using sampling techniques, as opposed to deterministic, to characterize the distributions. These methods sample a large number of points from the assumed distribution and propagate them forward. The characterization of the distribution is now done using straightforward calculations of the mean and variance of these samples. The accuracy of these approximation methods now depends on the sampling as opposed to the previous approaches which relied upon the accuracy of the linearizations or the numerical integration. There are two types of these methods we will discuss, the Ensemble Kalman filter (EnKF) and the Ensemble Transform Kalman Filter (ETKF), though many other methods exist. For example, we refer the interested reader to [1, 5, 8, 9, 10] for more information on these methods and others.

From here, since we are dealing with systems that will change sequentially in time, we shall focus our efforts on state-space models using discrete time. Therefore, difference equations shall mainly be presented to describe the dynamics of a system over time. The discrete formulation shall be used as measurements are only taken at discrete intervals in application and the modeling framework can easily be extended to continuous time.

In this paper, a comprehensive comparison study of the methods presented above will be performed on several problems. We shall draw conclusions about which methods give the best results for certain problems taking into account the assumptions, cpu time required for solutions, and the ability to solve the types of problems presented. The paper is organized as follows. Background and derivation of the Bayes filter and Kalman filter will be presented in Section 2. The problem statement we are interested in will be presented in Section 3. Section 4 will present the deterministic methods we mentioned above, and Section 5 will present the sampling based methods. The application of these methods to test problems will be presented in Section 6. Finally, Section 7 contains our concluding remarks.

2 Background and Derivation

2.1 Bayes Filter

As we have stated, the optimal method to combine our model information along with our measurement information to get the best estimate of the state of the system and the model parameters, is the Bayes filter.

To begin our discussion of the Bayes filter, we start by defining our state-space representation as follows:

$$x_{i+1} = f(i, x_i, u_i, \theta) + w_i \quad (1)$$

$$y_{i+1} = h(i + 1, x_{i+1}, u_{i+1}, \theta) + v_{i+1}, \quad (2)$$

where $x_{i+1} \in \mathcal{X}^{n_x}$ and $y_{i+1} \in \mathcal{X}^{n_y}$ are the model state and measurement, respectively, w_i and v_{i+1} are assumed to be independent and identically distributed, which we shall refer to as i.i.d., noise processes with zero means and covariances Q_i and R_{i+1} , $u_i \in \mathcal{X}^{n_u}$ is an exogenous control, and θ are the model parameters. Then, f and h are assumed to be nonlinear functions for the state and observations, respectively.

Our goal, from a Bayesian perspective, is to use the measurements, $y_{1:i}$, up to time i , to give us an understanding of the state x_i . This requires us to calculate the distribution, $p(x_i|y_{1:i})$, assuming that the initial distribution $p(x_0|y_0) = p(x_0)$, which is just the distribution of the state without any observations at the current time, known as the prior. Given this, we can calculate $p(x_i|y_{1:i})$ in two steps: a prediction and update.

Assuming we are starting at time $i - 1$, with $p(x_{i-1}|y_{1:i-1})$, the prediction step involves using the model to calculate the prior probability density function, pdf, of the state at time i by using the Chapman-Kolmogorov equation [4],

$$p(x_i|y_{1:i-1}) = \int p(x_i|x_{i-1})p(x_{i-1}|y_{1:i-1})dx_{i-1}. \quad (3)$$

This utilizes the fact that $p(x_i, x_{i-1}|y_{1:i-1}) = p(x_i|x_{i-1})$ since our model is a first order Markov process. The pdf $p(x_i|x_{i-1})$ is given by our state-space model (1) with statistics governed by w_i .

Now given a measurement y_i becomes available at time i , we can apply Bayes rule to update the prior estimate and get our posterior, $p(x_i|y_{1:i})$,

$$p(x_i|y_{1:i}) = \frac{p(y_i|x_i)p(x_i, y_{1:i-1})}{p(y_i, y_{1:i-1})}, \quad (4)$$

where $p(y_i, y_{1:i-1}) = \int p(y_i, x_i)p(x_i|y_{1:i-1})dx_i$.

The posterior is dependent upon the likelihood function which is defined by the measurement equation (2), with statistics given by v_{i+1} . Hence, the posterior is just an update of the prior given a measurement to get the density at the current state [4] and the Bayes filter is simply a recursion relationship between (3) and (4).

2.2 Kalman Filter

Assuming linear dynamics and Gaussian noise processes, the Bayes filter simplifies to the Kalman filter. Operating with Gaussian distributions, we can characterize the entire optimal estimate through the propagation of the mean and covariance. Letting $x_{i+1|i}$ be the state at time $i + 1$ given the state at time i , $\hat{x}_{i+1|i}$ the mean of the prior distribution, and $P_{x_{i+1|i}}$ the covariance of the prior distribution, we consider the state space formulation as follows:

$$x_{i+1|i} = Fx_{i|i} + w_i \quad (5)$$

$$y_{i+1} = Hx_{i+1} + v_{i+1}, \quad (6)$$

where w_i and v_{i+1} are normally distributed with mean 0 and covariances, Q and R , respectively, and F and H are matrix operators. For simplification, we shall use the following notation to describe distributions, $w_i \sim N(0, Q)$ and $v_{i+1} \sim N(0, R)$ where \sim is denoted to mean distributed and $N(0, R)$ is normal with mean 0 and covariance R . This gives $p(x_{i+1}) \sim N(\hat{x}_{i+1|i}, P_{x_{i+1|i}})$ with

$$P_{x_{i+1|i}} = \langle (x_{i+1} - \hat{x}_{i+1|i})(x_{i+1} - \hat{x}_{i+1|i})^T \rangle, \quad (7)$$

where $\langle A \rangle$ denotes the mean of A .

Substituting into this expression for the covariance, we obtain the following equation for the covariance prediction,

$$P_{x_{i+1|i}} = \langle (Fx_{i|i} + w_{i+1} - F\hat{x}_{i|i})(Fx_{i|i} + w_{i+1} - F\hat{x}_{i|i})^T \rangle \quad (8)$$

$$= F \langle (x_{i|i} - \hat{x}_{i|i})(x_{i|i} - \hat{x}_{i|i})^T \rangle F^T + Q = FP_{x_{i|i}}F^T + Q. \quad (9)$$

Assuming Gaussian distributions, we have:

$$p(x_{i+1}) \propto e^{-\frac{1}{2}(x - \hat{x}_{i+1|i})^T (P_{x_{i+1|i}})^{-1} (x - \hat{x}_{i+1|i})} \quad (10)$$

$$p(y_{i+1}|x_{i+1}) \propto e^{-\frac{1}{2}(y_{i+1} - Hx)^T R^{-1} (y_{i+1} - Hx)}. \quad (11)$$

From these densities, and applying Bayes formula, $p(x_{i+1}|y_{i+1}) \propto p(x_{i+1})p(y_{i+1}|x_{i+1})$, where \propto denotes proportionality, the posterior is given by

$$p(x_{i+1}|y_{i+1}) \propto e^{-\frac{1}{2}J[x]}, \quad (12)$$

where $J[x] = (x - \hat{x}_{i+1|i})^T (P_{x_{i+1|i}})^{-1} (x - \hat{x}_{i+1|i}) + (y_{i+1} - Hx)^T R^{-1} (y_{i+1} - Hx)$.

To obtain the best estimate, we want to maximize the conditional density, $p(x_{i+1}|y_{i+1})$. This is the same as minimizing $J[x]$. The necessary condition for $J[x]$ to be minimized is

$$\frac{dJ}{dx} = P_{x_{i+1|i}}^{-1} (x - \hat{x}_{i+1|i}) + (x - \hat{x}_{i+1|i})^T P_{x_{i+1|i}}^{-1} - H^T R^{-1} (y_{i+1} - Hx) + (y_{i+1} - Hx)^T R^{-1} (-H) \quad (13)$$

$$= 2(P_{x_{i+1|i}})^{-1} (x - \hat{x}_{i+1|i}) - 2H^T R^{-1} (y_{i+1} - Hx) = 0, \quad (14)$$

which implies

$$(P_{x_{i+1|i}})^{-1} (x - \hat{x}_{i+1|i}) = H^T R^{-1} (y_{i+1} - Hx).$$

Isolating the dependent variable, x , yields

$$(P_{x_{i+1|i}}^{-1} + H^T R^{-1} H)x = P_{x_{i+1|i}}^{-1} \hat{x}_{i+1|i} + H^T R^{-1} y_{i+1}.$$

Adding and subtracting $H^T R^{-1} H \hat{x}_{i+1|i}$, we then multiply through by the inverse of the leading term, $(P_{x_{i+1|i}}^{-1} + H^T R^{-1} H)$, to obtain

$$\begin{aligned} \hat{x}_{i+1|i+1} = & (P_{x_{i+1|i}}^{-1} + H^T R^{-1} H)^{-1} P_{x_{i+1|i}}^{-1} \hat{x}_{i+1|i} + (P_{x_{i+1|i}}^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1} y_{i+1} + \\ & [(P_{x_{i+1|i}}^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1} H \hat{x}_{i+1|i} - (P_{x_{i+1|i}}^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1} H \hat{x}_{i+1|i}], \end{aligned}$$

where $\hat{x}_{i+1|i+1}$ is the mean of the posterior distribution.

Rearranging the terms, we arrive at

$$\begin{aligned} \hat{x}_{i+1|i+1} = & (P_{x_{i+1|i}}^{-1} + H^T R^{-1} H)^{-1} P_{x_{i+1|i}}^{-1} \hat{x}_{i+1|i} + (P_{x_{i+1|i}}^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1} H \hat{x}_{i+1|i} + \\ & (P_{x_{i+1|i}}^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1} (y_{i+1} - H \hat{x}_{i+1|i}). \end{aligned}$$

Adding the $\hat{x}_{i+1|i}$ terms together and simplifying, we have

$$\hat{x}_{i+1|i+1} = \hat{x}_{i+1|i} + K_{i+1}(y_{i+1} - H\hat{x}_{i+1|i}), \quad (15)$$

where K_{i+1} is defined below

$$K_{i+1} = (P_{x_{i+1|i}}^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1}, \quad (16)$$

and is known as the Kalman gain. Equation (15) is the mean update for the state. A more common expression for the Kalman gain can be obtained by multiplying K_{i+1} by $(HP_{x_{i+1|i}}H^T + R)(HP_{x_{i+1|i}}H^T + R)^{-1}$,

$$K_{i+1} = (P_{x_{i+1|i}}^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1} (HP_{x_{i+1|i}}H^T + R)(HP_{x_{i+1|i}}H^T + R)^{-1}. \quad (17)$$

Multiplying R by $(P_{x_{i+1|i}}H^T)^{-1}P_{x_{i+1|i}}H^T$ in $(HP_{x_{i+1|i}}H^T + R)(HP_{x_{i+1|i}}H^T + R)^{-1}$, we obtain

$$(HP_{x_{i+1|i}}H^T + R)(HP_{x_{i+1|i}}H^T + R)^{-1} = (HP_{x_{i+1|i}}H^T + R(P_{x_{i+1|i}}H^T)^{-1}P_{x_{i+1|i}}H^T)(HP_{x_{i+1|i}}H^T + R)^{-1}. \quad (18)$$

Substituting (18) into (17) and expanding yields

$$K_{i+1} = (P_{x_{i+1|i}}^{-1} + H^T R^{-1} H)^{-1} [H^T R^{-1} HP_{x_{i+1|i}}H^T + H^T (P_{x_{i+1|i}}H^T)^{-1} P_{x_{i+1|i}}H^T] (HP_{x_{i+1|i}}H^T + R)^{-1}. \quad (19)$$

Substituting $(P_{x_{i+1|i}}H^T)^{-1} = (H^T)^{-1}P_{x_{i+1|i}}^{-1}$ into (19) yields

$$K_{i+1} = (P_{x_{i+1|i}}^{-1} + H^T R^{-1} H)^{-1} [H^T R^{-1} H(P_{x_{i+1|i}}H^T) + H^T (H^T)^{-1} P_{x_{i+1|i}}^{-1} (P_{x_{i+1|i}}H^T)] (HP_{x_{i+1|i}}H^T + R)^{-1}.$$

Rearranging terms, we obtain

$$K_{i+1} = (P_{x_{i+1|i}}^{-1} + H^T R^{-1} H)^{-1} [(H^T R^{-1} H + P_{x_{i+1|i}}^{-1})(P_{x_{i+1|i}}H^T)] (HP_{x_{i+1|i}}H^T + R)^{-1}.$$

Simplifying, we finally arrive at

$$K_{i+1} = P_{x_{i+1|i}}H^T(HP_{x_{i+1|i}}H^T + R)^{-1}, \quad (20)$$

which is the more common expression for the Kalman Gain.

From (15), $\hat{x}_{i+1|i+1} = \hat{x}_{i+1|i} + K_{i+1}(y_{i+1} - H\hat{x}_{i+1|i})$, by substituting this expression into the covariance formula,

$$\langle (x_{i+1} - \hat{x}_{i+1|i+1})(x_{i+1} - \hat{x}_{i+1|i+1})^T \rangle, \quad (21)$$

we obtain

$$(x_{i+1} - \hat{x}_{i+1|i+1}) = x_{i+1} - \hat{x}_{i+1|i} - K_{i+1}(y_{i+1} - Hx_{i+1} - H(\hat{x}_{i+1|i} - x_{i+1})).$$

Let $e_{i+1|i+1} = x_{i+1} - \hat{x}_{i+1|i+1}$ be the error between the true state and the estimated state. Using the output equation (6), we obtain

$$e_{i+1|i+1} = (I - K_{i+1}H)e_{i+1|i} - K_{i+1}(Hx_{i+1} + v_{i+1} - Hx_{i+1}) \quad (22)$$

$$e_{i+1|i+1} = (I - K_{i+1}H)e_{i+1|i} - K_{i+1}v_{i+1}. \quad (23)$$

Using (23), the covariance update is given by

$$\langle e_{i+1|i+1}e_{i+1|i+1}^T \rangle = (I - K_{i+1}H)\langle e_{i+1|i}e_{i+1|i}^T \rangle(I - K_{i+1}H)^T + K_{i+1}RK_{i+1}^T \quad (24)$$

$$\langle e_{i+1|i+1}e_{i+1|i+1}^T \rangle = (I - K_{i+1}H)\langle e_{i+1|i}e_{i+1|i}^T \rangle - (I - K_{i+1}H)\langle e_{i+1|i}e_{i+1|i}^T \rangle(K_{i+1}H)^T + K_{i+1}RK_{i+1}^T. \quad (25)$$

Substituting (16) into $(I - K_{i+1}H)$ and simplifying, we obtain

$$(I - K_{i+1}H) = (P_{x_{i+1}|i}^{-1} + H^T R^{-1} H)^{-1} (P_{x_{i+1}|i}^{-1} + H^T R^{-1} H) - (P_{x_{i+1}|i}^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1} H \quad (26)$$

$$= (P_{x_{i+1}|i}^{-1} + H^T R^{-1} H)^{-1} P_{x_{i+1}|i}^{-1}. \quad (27)$$

Applying the definition of the covariance, $\langle e_{i+1|i} e_{i+1|i}^T \rangle = P_{x_{i+1}|i}$, along with substituting (27) and expanding (25) yields

$$\begin{aligned} \langle e_{i+1|i+1} e_{i+1|i+1}^T \rangle &= (I - K_{i+1}H) P_{x_{i+1}|i} - ((P_{x_{i+1}|i}^{-1} + H^T R^{-1} H)^{-1} P_{x_{i+1}|i}^{-1}) P_{x_{i+1}|i} H^T K_{i+1}^T + K_{i+1} R K_{i+1}^T \\ &= (I - K_{i+1}H) P_{x_{i+1}|i} - (P_{x_{i+1}|i}^{-1} + H^T R^{-1} H)^{-1} H^T K_{i+1}^T \\ &\quad + ((P_{x_{i+1}|i}^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1}) R K_{i+1}^T \\ &= (I - K_{i+1}H) P_{x_{i+1}|i} - (P_{x_{i+1}|i}^{-1} + H^T R^{-1} H)^{-1} H^T K_{i+1}^T \\ &\quad + (P_{x_{i+1}|i}^{-1} + H^T R^{-1} H)^{-1} H^T K_{i+1}^T. \end{aligned}$$

Simplifying, we obtain the covariance update

$$P_{x_{i+1}|i+1} = (I - K_{i+1}H) P_{x_{i+1}|i}. \quad (28)$$

In summary, the above derivation for the Kalman filter essentially contains two steps.

Prediction

$$\hat{x}_{i+1|i} = F x_{i|i} \quad (29)$$

$$P_{x_{i+1}|i} = F P_{x_{i|i}} F^T + Q \quad (30)$$

Update

$$\hat{x}_{i+1|i+1} = \hat{x}_{i+1|i} + K_{i+1} (y_{i+1} - H \hat{x}_{i+1|i}) \quad (31)$$

$$K_{i+1} = P_{x_{i+1}|i} H^T (H P_{x_{i+1}|i} H^T + R)^{-1} \quad (32)$$

$$P_{x_{i+1}|i+1} = (I - K_{i+1}H) P_{x_{i+1}|i}. \quad (33)$$

As we have stated, however, this only holds if the model dynamics and observation equation are both linear and the noise processes are Gaussian. For cases when these do not hold, modifications to the Kalman filter have been developed.

3 Problem Statement

Consider a state space model of the form

$$x_{i+1} = f(i, x_i, u_i, \theta) + w_i \quad (34)$$

$$y_{i+1} = h(i+1, x_{i+1}, u_i, \theta) + v_{i+1}, \quad (35)$$

where f is a nonlinear function of the state $x_{i+1} \in \mathfrak{R}^{n_x}$, and h is a nonlinear function relating the observation, $y_{i+1} \in \mathfrak{R}^{n_y}$, to the state. $u_i \in \mathfrak{R}^{n_u}$ is a exogenous control, w_i and v_{i+1} are i.i.d. noise processes, and θ are the model parameters.

Our objective is to find the optimal parameter estimates to give the state the best fit to the data. Classically, parameter estimation is done by using an optimization technique to minimize a cost function. Generally, the cost function is the sum of the squares of the difference between the model, y_k , and observations, y_k^{obs} , given by

$$J(\theta) = \sum_{k=1}^N \gamma_k [y_k^{obs} - y_k]^2, \quad (36)$$

where γ_k is a weighting parameter and $y_k^{obs} - y_k$ is the residual. This approach is commonly referred to as nonlinear least squares.

Utilizing the filtering technique to estimate parameters, there are multiple methods in which we can achieve our goal. First, and most direct, is to modify the state space representation to accommodate our objective. Since we are assuming the observations are coming from our model and the parameters are our main focus, we can proceed by assuming that the parameters have zero dynamics

$$\begin{aligned} \theta_{i+1} &= \theta_i + r_i \\ y_{i+1} &= h(i+1, x_{i+1}, u_{i+1}, \theta_{i+1}) + v_{i+1}, \end{aligned}$$

where r_i and v_{i+1} are i.i.d. noise processes. However, our goal is to not only estimate the parameters, but also to fit the states as accurately as possible. This methodology is known as dual estimation. There are two methodologies that directly achieve this. First is joint estimation. This is done by concatenating the parameters into the state-space. This gives the following

$$\begin{aligned} z &= [x; \theta] \\ z_{i+1} &= f(i, z_i, u_i) + w_i \\ \theta_{i+1} &= \theta_i + r_i. \end{aligned}$$

A second approach is called the dual filter. This is done by concurrently running a state filter and parameter filter in parallel. The state filter estimates the state, x_i , using the parameter value from the parameter filter at time, $i-1$. While, the parameter filter estimates the parameter, θ_i , using the state estimate at time, $i-1$. These both propagate forward in time and obtain estimates at each time step. So, one estimates, $p(x|y, \theta)$, while the other filter estimates, $p(\theta|x, y)$.

Since most models of practical interest are nonlinear in nature, we cannot directly apply the Kalman filter to get these estimates. As stated, adjustments to the filtering framework must be implemented to account for the nonlinearities. In the following sections, we shall present methods that have been proposed for overcoming the difficulties of estimating nonlinear systems. The sections will be divided into deterministic filters and sampling filters.

4 Deterministic Kalman Filters

4.1 The Extended Kalman Filter

Because of the Kalman filter's optimality and simplicity, the extended Kalman filter (EKF) maintains the basic framework and idea of the original Kalman filter, but uses a first order linearization of the model dynamics to approximate the error statistics. Assuming we have a general nonlinear system as below:

$$x_{i+1}^f = f(i, x_i^f, u_i, \theta) + w_i \quad (37)$$

$$y_{i+1}^f = h(i+1, x_{i+1}^f, u_i, \theta) + v_{i+1}, \quad (38)$$

and assuming we model this system by the approximate equation

$$\hat{x}_{i+1}^- = f(i, \hat{x}_i, u_i, \theta), \quad (39)$$

we can subtract (39) from (37) to obtain

$$x_{i+1}^t - \hat{x}_{i+1}^- = f(i, x_i^t, u_i, \theta) - f(i, \hat{x}_i, u_i, \theta) + w_i. \quad (40)$$

We can perform a Taylor expansion to approximate the system

$$f(x) = f(x_0) + \frac{df}{dx}(x_0)(x - x_0) + \frac{1}{2}(x - x_0)^T H_f(x_0)(x - x_0)^T + \mathcal{O}(\|x - x_0\|^3),$$

where H_f is the Hessian matrix. Applying this expansion to the original system, we obtain

$$x_{i+1}^t - \hat{x}_{i+1}^- = f(i, \hat{x}_i, u_i, \theta) + \frac{df}{dx}(\hat{x}_i)(x_i^t - \hat{x}_i) + \frac{1}{2}(x_i^t - \hat{x}_i)H_f(\hat{x}_i)(x_i^t - \hat{x}_i)^T + \dots - f(i, \hat{x}_i, u_i, \theta) + w_i.$$

Squaring both sides and taking the expected value, we get the following equation

$$\begin{aligned} P_{x_{i+1}|i} &= F_x(\hat{x}_i)E[(x_i^t - \hat{x}_i)(x_i^t - \hat{x}_i)^T]F_x(\hat{x}_i)^T + \frac{1}{2}F_x(\hat{x}_i)E[(x_i^t - \hat{x}_i)(x_i^t - \hat{x}_i)^T]F_{xx}^T(\hat{x}_i)E[(x_i^t - \hat{x}_i)] \\ &\quad + \frac{1}{4}E[(x_i^t - \hat{x}_i)(x_i^t - \hat{x}_i)^T]H_f H_f^T E[(x_i^t - \hat{x}_i)(x_i^t - \hat{x}_i)^T] + \dots + Q, \end{aligned}$$

which simplifies by discarding the moments of third or higher order and assuming higher order terms are negligible, to become

$$P_{x_{i+1}|i} \approx F_x(\hat{x}_i)P_{x_i|i}F_x(\hat{x}_i)^T + Q, \quad (41)$$

where $F_x(\hat{x}_i) = \frac{\partial f(i, x)}{\partial x}|_{x=\hat{x}_i}$, Q is the model error covariance, and $P_{x_{i+1}|i}$ is the predicted error covariance.

Applying the same procedure to the nonlinear observation equation h , we utilize a linearization of the observation equation to be used in the update as follows

$$H_{i+1} = \frac{\partial h(i, x)}{\partial x}|_{x=\hat{x}_{i+1}^-}.$$

Using the above derivations in the classical Kalman filter equations, we get the extended Kalman filter. While the initial state estimation portion of the algorithm is unchanged, these Jacobians are utilized in the initial covariance estimation and in the estimation update for both the covariance and the state, through the gain. Though we utilize the same procedure as the Kalman filter, the extended Kalman filter is sub-optimal for nonlinear problems, and reduces to the Kalman filter for linear problems.

The algorithm for the extended Kalman filter is summarized as follows:

Initialize: for $i = 0$

$$\hat{x}_0 = E[x_0]$$

$$P_0 = E[(x_0 - E[x_0])(x_0 - E[x_0])^T]$$

algorithm: for $i = 1, 2, 3, \dots$

State Estimation

$$\hat{x}_i^- = f(i, \hat{x}_{i-1})$$

Covariance Estimation

$$P_i^- = F_{i,i-1} P_{i-1} F_{i,i-1}^T + Q_{i-1}$$

where F is the linearization of the model w.r.t. the state

$$F_{i,i-1} = \left. \frac{\partial f(i, x)}{\partial x} \right|_{x=\hat{x}_{i-1}}$$

Kalman Gain Matrix

$$K = P_i^- H_i^T [H_i P_i^- H_i^T + R_i]^{-1}$$

where H is the linearization of the observation w.r.t. the state

$$H_i = \left. \frac{\partial h(i, x)}{\partial x} \right|_{x=\hat{x}_i^-}$$

State Estimation Update

$$\hat{x}_i = \hat{x}_i^- + K[y_i - h(i, \hat{x}_i^-)]$$

Covariance Estimation Update

$$P_i = [I - KH_i] P_i^-$$

Reviewing the algorithm, we see that this extension of the Kalman filter is applicable for nonlinear systems. Though we still require Gaussian distributions for both the model and observations, we have made steps to more accurately estimate the state and parameters of the system we are modeling. However, the disadvantage of the extended Kalman filter is that it is approximating the nonlinear model by a linear counterpart, through a first order Taylor expansion. The success of this relies heavily on the assumption that the linearization of the nonlinear state-space model deviates from the linear model no more than first order [13]. In cases where the nonlinearities are large, or the initial guess is too far away, the filter will achieve less accurate results, and can even diverge and fail to achieve meaningful results [23]. This is likely due to the approximation of the error statistics by the linearized model, since a Gaussian distribution cannot be guaranteed once propagated through a nonlinear operator.

To increase the accuracy of nonlinear models, another Kalman filter has been constructed that better preserves the distribution statistics as it passes through the nonlinearities. Unlike the EKF, this new filter maintains at least 2nd order accuracy for all distributions, and at least 3rd order accuracy for Gaussian distributions [13].

4.2 The Unscented Kalman Filter

As mentioned above, the unscented Kalman filter addresses the shortcomings of the EKF through the 'unscented transformation.' Again, we assume that the errors are Gaussian random variables (GRV). Instead of

using a linear approximation to the nonlinear model, we construct a set of sample points which will completely capture the true mean and covariance of the GRV, and once propagated through the nonlinear model, will capture the posterior mean and covariance to the 3rd order for any nonlinearity [23]. The advantages of this are not just the increase of accuracy, but also the decrease in complexity by not requiring the calculation of a derivative, or Jacobian. To better understand, we shall explain the unscented transformation.

The unscented transformation (UT) is a method used to calculate the statistics of a random variable which undergoes a nonlinear transformation [15]. Assume we have a random variable, x , with dimension L , which has mean \hat{x} and covariance, P_x . This random variable gets propagated through a nonlinear function, $y = g(x)$. To calculate the statistics of y , we form a matrix, χ , of $2L + 1$ sigma vectors, χ_i . This matrix, χ is constructed as follows,

$$\chi = [\hat{x}, \hat{x} + (\sqrt{(L + \lambda)P_x})_L, \hat{x} - (\sqrt{(L + \lambda)P_x})_L], \quad (42)$$

where \hat{x} is a column vector and $(\sqrt{(L + \lambda)P_x})_L$ is an $L \times L$ matrix. The corresponding weights (W_i) for the sigma vectors are as follows

$$W_0^{(m)} = \frac{\lambda}{(L + \lambda)} \quad (43)$$

$$W_0^{(c)} = \frac{\lambda}{(L + \lambda)} + (1 - \alpha^2 + \beta) \quad (44)$$

$$W_i^{(m)} = W_i^{(c)} = \frac{1}{2(L + \lambda)} \quad i = 1, \dots, 2L, \quad (45)$$

where $\lambda = \alpha^2(L + \kappa) - L$ is a scaling parameter. α is used to determine the spread of the sigma points around \hat{x} , and takes values between $1 \geq \alpha \geq 10^{-4}$ depending on the problem. κ is a secondary scaling parameter which is optimally set to $3 - L$ [13]. Lastly, β is used to incorporate prior knowledge of the distribution of x (for Gaussian distributions, $\beta = 2$ is optimal) [23]. $(\sqrt{(L + \lambda)P_x})_i$ is the i th row of the matrix square root [23]. The sigma vectors are then propagated through the nonlinear function accordingly,

$$y_i = g(\chi_i), \quad i = 0, \dots, 2L, \quad (46)$$

with the mean and covariance of y being approximated using a weighted sample mean and covariance of the posterior sigma points,

$$\hat{y} \approx \sum_{i=0}^{2L} W_i^{(m)} y_i \quad (47)$$

$$P_y \approx \sum_{i=0}^{2L} W_i^{(c)} (y_i - \hat{y})(y_i - \hat{y})^T. \quad (48)$$

As stated earlier, the UT results in approximations that are accurate to the third order for Gaussian inputs for all nonlinearities, and accurate to at least second order for all other inputs [13]. Extending the UT for use in the Unscented Kalman filter (UKF) is straightforward, and is shown below (assuming additive Gaussian

noise),

Initialize: for $t = 0$

$$\hat{x}_0 = E[x_0]$$

$$P_0 = E[(x_0 - E[x_0])(x_0 - E[x_0])^T]$$

algorithm: for $t = 1, 2, 3, \dots$

Calculate Sigma Points

$$\chi_{t-1} = [\hat{x}, \hat{x} + (\sqrt{(L + \lambda)P_{t-1}})_L, \hat{x} - (\sqrt{(L + \lambda)P_{t-1}})_L]$$

time-update equations

$$\chi_{t,t-1}^* = F(\chi_{t-1}, u_t, w_t)$$

where u_t is an exogenous input and w_t are parameters

$$\hat{x}_t^- = \sum_{i=0}^{2L} W_i^{(m)} \chi_{i,t,t-1}^*$$

$$P_t^- = \sum_{i=0}^{2L} W_i^{(c)} (\chi_{i,t,t-1}^* - \hat{x}_t^-)(\chi_{i,t,t-1}^* - \hat{x}_t^-)^T + R^v$$

time-update equations for observation: augment sigma points

$$\chi_{t,t-1} = [\chi_{0,t,t-1}^* \quad \chi_{0,t,t-1}^* + (\sqrt{(L + \lambda)P_t^-})_L, \hat{x} - (\sqrt{(L + \lambda)P_t^-})_L]$$

$$Y_{t,t-1} = (\chi_{t,t-1})$$

$$\hat{y}_t^- = \sum_{i=0}^{2L} W_i^{(m)} Y_{i,t,t-1}$$

measurement-update equations

$$P_{\hat{y}_t \hat{y}_t} = \sum_{i=0}^{2L} W_i^{(c)} (Y_{i,t,t-1} - \hat{y}_t^-)(Y_{i,t,t-1} - \hat{y}_t^-)^T + R^n$$

$$P_{x_t y_t} = \sum_{i=0}^{2L} W_i^{(c)} (\chi_{i,t,t-1} - \hat{x}_t^-)(Y_{i,t,t-1} - \hat{y}_t^-)^T$$

$$K_t = P_{x_t y_t} P_{\hat{y}_t \hat{y}_t}^{-1}$$

$$\hat{x}_t = \hat{x}_t^- + K_t (y_t - \hat{y}_t^-)$$

$$P_t = P_t^- - K_t P_{\hat{y}_t \hat{y}_t} K_t^T$$

where R^v is the process noise covariance and R^n is the measurement noise covariance.

The unscented Kalman filter as presented assumes additive Gaussian noise. There are multiple variations of this algorithm for numerical purposes, one of which is the square root unscented Kalman filter. Also, a more general form of the UKF is also available, which does not assume additive noise, but propagates the measurement and process covariance through the nonlinear function. These extensions can be found in most of our referenced materials [13, 15, 21, 23, 24].

4.3 The Cubature Kalman Filter

Similar to the previous deterministic filters, the cubature Kalman filter (CKF) relies upon an approximation of the nonlinear function. However, unlike the previous methods, the CKF closely follows the Bayes filter theory and takes advantage of the assumption of a normal distribution by using numerical integration techniques to calculate the time and measurement updates. This underlying assumption of a Gaussian distribution for the predictive density and the measurement density is central as it leads to a Gaussian posterior distribution [2].

The prediction step, or time update, is focused on calculating the mean, $\hat{x}_{i+1|i}$, and covariance, $P_{i+1|i}$, of the predictive density as follows

$$\hat{x}_{i+1|i} = E[x_{i+1}|u_{1:i}, y_{1:i}].$$

Substituting in for x_{i+1} from our model (34), we obtain

$$\hat{x}_{i+1|i} = E[f(i, x_i, u_i, \theta) + w_i|u_{1:i}, y_{1:i}] = E[f(i, x_i, u_i, \theta)|u_{1:i}, y_{1:i}].$$

This follows because w_i is assumed to be a zero-mean noise process that is uncorrelated with previous measurements. This yields

$$\begin{aligned} \hat{x}_{i+1|i} &= \int_{\mathfrak{R}^{n_x}} f(i, x_i, u_i, \theta) p(x_i|u_{1:i}, y_{1:i}) dx_i \\ &= \int_{\mathfrak{R}^{n_x}} f(i, x_i, u_i, \theta) \mathbb{N}(x_i; \hat{x}_{i|i}, P_{i|i}) dx_i, \end{aligned}$$

where $\mathbb{N}(\cdot, \cdot, \cdot)$ is standard notation for a Gaussian density. Following the same procedure, we obtain the predictive covariance

$$P_{i+1|i} = E[(x_{i+1} - \hat{x}_{i+1|i})(x_{i+1} - \hat{x}_{i+1|i})^T | u_{1:i}, y_{1:i}].$$

Again, substituting (34) in for x_{i+1} , using $f(x_i)$ for shorthand to represent $f(i, x_i, u_i, \theta)$, we obtain

$$\begin{aligned} P_{i+1|i} &= E[(f(x_i) + w_i - \hat{x}_{i+1|i})(f(x_i) + w_i - \hat{x}_{i+1|i})^T | u_{1:i}, y_{1:i}] \\ &= E[f(x_i)f(x_i)^T + f(x_i)w_i^T - f(x_i)\hat{x}_{i+1|i}^T + w_i f(x_i)^T \\ &\quad + w_i w_i^T - w_i \hat{x}_{i+1|i}^T - \hat{x}_{i+1|i} f(x_i)^T - \hat{x}_{i+1|i} w_i^T + \hat{x}_{i+1|i} \hat{x}_{i+1|i}^T | u_{1:i}, y_{1:i}]. \end{aligned}$$

Assuming that w_i is independent of both $f(x_i)$ and $\hat{x}_{i+1|i}$, we can use $E[xy] = E[x]E[y]$ and $E[E[x]] = E[x]$. Knowing that $w_i \sim N(0, Q)$ and $\hat{x}_{i+1|i} = E[f(x_i)|u_{1:i}, y_{1:i}]$, we can simplify the previous expression to be

$$\begin{aligned} P_{i+1|i} &= E[f(x_i)f(x_i)^T - f(x_i)\hat{x}_{i+1|i}^T + w_i w_i^T - \hat{x}_{i+1|i} f(x_i)^T + \hat{x}_{i+1|i} \hat{x}_{i+1|i}^T | u_{1:i}, y_{1:i}] \\ &= E[f(x_i)f(x_i)^T] - E[f(x_i)\hat{x}_{i+1|i}^T] + E[w_i w_i^T] - E[\hat{x}_{i+1|i} f(x_i)^T] + E[\hat{x}_{i+1|i} \hat{x}_{i+1|i}^T]. \end{aligned}$$

Simplifying $-2E[f(x_i)\hat{x}_{i+1|i}^T] + E[\hat{x}_{i+1|i}\hat{x}_{i+1|i}^T]$ by using $E[f(x_i)]E[\hat{x}_{i+1|i}^T] = \hat{x}_{i+1|i}\hat{x}_{i+1|i}^T$, we finally arrive at

$$P_{i+1|i} = \int_{\mathfrak{R}^{n_x}} f(i, x_i, u_i, \theta) f(i, x_i, u_i, \theta)^T \mathbb{N}(x_i; \hat{x}_{i|i}, P_{i|i}) dx_i - \hat{x}_{i+1|i} \hat{x}_{i+1|i}^T + Q.$$

For the measurement update, again on the basis that measurements are from Gaussian distributions, we can write the likelihood function to be

$$p(y_{i+1}|u_{1:i}, y_{1:i}) = \mathbb{N}(y_{i+1}; \hat{y}_{i+1|i}, P_{zz,i+1|i}),$$

where the measurement prediction, calculated using the same procedure as the prediction step, is

$$\hat{y}_{i+1} = \int_{\mathfrak{X}^{n_x}} h(i+1, x_{i+1}, u_{i+1}, \theta) \mathbb{N}(x_{i+1}; \hat{x}_{i+1|i}, P_{i+1|i}) dx_{i+1}.$$

The associated covariance, calculated using the same procedure as the prediction step, is given by

$$P_{zz,i+1|i} = \int_{\mathfrak{X}^{n_x}} h(x_{i+1})h(x_{i+1})^T \times \mathbb{N}(x_{i+1}; \hat{x}_{i+1|i}, P_{i+1|i}) dx_{i+1} - \hat{z}_{i+1|i} \hat{z}_{i+1|i}^T + R_i.$$

Therefore, the joint conditional Gaussian density of the state and measurement is

$$p\left(\begin{bmatrix} x_{i+1}^T \\ y_{i+1}^T \end{bmatrix} \middle| y_{1:i}, u_{1:i}\right) = \left(\mathbb{N}\left(\begin{bmatrix} \hat{x}_{i+1|i} \\ \hat{y}_{i+1|i} \end{bmatrix}, \begin{pmatrix} P_{i+1|i} & P_{xz,i+1|i} \\ P_{xz,i+1|i}^T & P_{zz,i+1|i} \end{pmatrix}\right) \right),$$

where the cross covariance is calculated as follows, substituting (35),

$$\begin{aligned} P_{xz,i+1|i} &= E[(x_{i+1} - \hat{x}_{i+1|i})(y_{i+1} - \hat{y}_{i+1|i})] \\ &= E[(x_{i+1} - \hat{x}_{i+1|i})(h(x_{i+1}) + v_{i+1} - \hat{y}_{i+1|i})], \end{aligned}$$

where carrying out the calculation, assuming v_{i+1} is independent of x_{i+1} and $\hat{x}_{i+1|i}$, and also noting that $E(x_{i+1}|y_{1:i}, u_{1:i}) = \hat{x}_{i+1|i}$ and $E(h(x_{i+1})|y_{1:i}, u_{1:i}) = \hat{y}_{i+1|i}$, we obtain

$$\begin{aligned} P_{xz,i+1|i} &= E(x_{i+1}h(x_{i+1})^T + x_{i+1}v_{i+1}^T - x_{i+1}\hat{y}_{i+1|i}^T - \hat{x}_{i+1|i}h(x_{i+1})^T - \hat{x}_{i+1|i}v_{i+1}^T + \hat{x}_{i+1|i}\hat{y}_{i+1|i}) \\ &= E(x_{i+1}h(x_{i+1})^T) - E(x_{i+1}\hat{y}_{i+1|i}^T) - E(\hat{x}_{i+1|i}h(x_{i+1})^T) + E(\hat{x}_{i+1|i}\hat{y}_{i+1|i}) \\ &= \int_{\mathfrak{X}^{n_x}} x_{i+1}h(i+1, x_{i+1}, u_{i+1}, \theta)^T \mathbb{N}(x_{i+1}; \hat{x}_{i+1|i}, P_{i+1|i}) dx_{i+1} - \hat{x}_{i+1|i}\hat{y}_{i+1|i}. \end{aligned}$$

Given a new measurement y_{i+1} , we calculate the posterior density $p(x_{i+1}|y_{1:i+1}, u_{1:i+1})$ from the Bayesian filter, resulting in

$$p(x_{i+1}|y_{1:i+1}, u_{1:i+1}) = \mathbb{N}(x_{i+1}; \hat{x}_{i+1|i+1}, P_{i+1|i+1}), \quad (49)$$

where

$$\hat{x}_{i+1|i+1} = \hat{x}_{i+1|i} + K_{i+1}(y_{i+1} - \hat{y}_{i+1|i}) \quad (50)$$

$$P_{i+1|i+1} = P_{i+1|i} - K_{i+1}P_{zz,i+1|i}K_{i+1}^T \quad (51)$$

$$K_{i+1} = P_{xz,i+1|i}P_{zz,i+1|i}^{-1}, \quad (52)$$

which, if $f(\dots)$ and $h(\dots)$ are linear functions of the state, simplifies to the classical Kalman filter.

The remaining question now becomes how to calculate an integral of the form,

$$\int_{\mathfrak{X}^n} g(x)\mathbb{N}(x)dx = \int_{\mathfrak{X}^n} g(x)\exp(-x^T x)dx.$$

Arasaratnam [2, 3] proposes that the best way to do this is (i) transform it into a more familiar spherical-radial integration form, and (ii) subsequently apply a third-degree spherical-radial rule .

First, a transformation must be made to change the variables from the Cartesian space $x \in \mathbb{R}^n$ to a radius r and direction y as follows: let $x = ry$ with $y^T y = 1$, so that $x^T x = r$ for $r \in [0, \infty)$ [2]. Then, the integral becomes

$$I(f) = \int_0^\infty \int_{U_n} f(ry) r^{n-1} \exp(-r^2) d\sigma(y) dr,$$

where U_n is the surface of the sphere defined by $U_n = \{y \in \mathbb{R}^n | y^T y = 1\}$ and $\sigma(\dots)$ is the spherical surface measure [2]. This integral can be broken up into a radial component and spherical component which can subsequently be numerically computed using a cubature rule and quadrature rule. Radial integral

$$I = \int_0^\infty S(r) r^{n-1} \exp(-r^2) dr,$$

where $S(r)$ is defined by the spherical integral with unit weighting function $w(y) = 1$,

$$S(r) = \int_{U_n} f(ry) d\sigma(y).$$

Carrying out both computations and combining them into a third-degree spherical-radial rule, we can calculate the standard Gaussian weighted integral

$$I_{\mathbb{N}}(f) = \int_{\mathbb{R}^n} f(x) \times \mathbb{N}(x; 0, I) dx \approx \sum_{i=1}^m \omega_i f(\xi_i),$$

where

$$\xi_i = \sqrt{\frac{m}{2}} [1]_i$$

$$\omega_i = \frac{1}{m}, \quad i = 1, 2, \dots, m = 2n.$$

Here $[1]_i$ denotes a generator, where a generator \mathbf{u} in a fully symmetric region is defined by $\mathbf{u} = (u_1, u_2, \dots, u_r, 0, \dots, 0) \in \mathbb{R}^n$ with $u_i \geq u_{i+1} > 0, i = 1, 2, \dots, (r-1)$. For example, $[1] \in \mathbb{R}^2$ is represented by the points

$$\left(\begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix} \right),$$

using the notation that $[u_1, u_2, \dots, u_r]_i$ is the i -th point from the set $[u_1, u_2, \dots, u_r]$. For more information on the numerical integration details, see [2, 3].

In summary, the algorithm for the cubature Kalman filter (assuming additive noise) proceeds as follows,

Initialize: for $t = 0$

$$\hat{x}_0 = E[x_0]$$

$$P_0 = E[(x_0 - E[x_0])(x_0 - E[x_0])^T]$$

algorithm: for $t = 1, 2, 3, \dots$

Calculate Matrix Square Root

$$P_{t-1|t-1} = S_{t-1|t-1} S_{t-1|t-1}^T$$

Evaluate Cubature Points: for $i = 1, 2, \dots, 2n_x$

$$X_{i,t-1|t-1} = S_{t-1|t-1} \xi_i + \hat{x}_{t-1|t-1}$$

time-update equations

$$X_{i,t|t-1}^* = f(X_{i,t-1|t-1}, u_{t-1}, \theta)$$

where u_{t-1} is an exogenous input and θ are parameters

$$\hat{x}_{t|t-1} = \frac{1}{2n_x} \sum_{i=1}^{2n_x} X_{i,t|t-1}^*$$

$$P_{t|t-1} = \frac{1}{2n_x} \sum_{i=1}^{2n_x} X_{i,t|t-1}^* X_{i,t|t-1}^{*T} - \hat{x}_{t|t-1} \hat{x}_{t|t-1}^T + Q_{t-1}$$

time-update equations for observation: calculate matrix square root

$$P_{t|t-1} = S_{t|t-1} S_{t|t-1}^T$$

Evaluate Cubature Points: for $i = 1, 2, \dots, 2n_x$

$$X_{i,t|t-1} = S_{t|t-1} \xi_i + \hat{x}_{t|t-1}$$

$$Y_{i,t|t-1} = h(X_{i,t|t-1}, u_{t-1}, \theta)$$

measurement-update equations

$$\hat{y}_{t|t-1} = \frac{1}{2n_x} \sum_{i=1}^{2n_x} Y_{i,t|t-1}$$

$$P_{zz,t|t-1} = \frac{1}{2n_x} \sum_{i=1}^{2n_x} Y_{i,t|t-1} Y_{i,t|t-1}^T - \hat{y}_{t|t-1} \hat{y}_{t|t-1}^T + R_t$$

$$P_{xz,t|t-1} = \frac{1}{2n_x} \sum_{i=1}^{2n_x} \omega_i X_{i,t|t-1} Y_{i,t|t-1}^T - \hat{x}_{t|t-1} \hat{y}_{t|t-1}^T$$

$$K_t = P_{xz,t|t-1} P_{zz,t|t-1}^{-1}$$

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t (y_t - \hat{y}_{t|t-1})$$

$$P_{t|t} = P_{t|t-1} - K_t P_{zz,t|t-1} K_t^T$$

where Q_{t-1} is the process noise covariance and R_t is the measurement noise covariance.

It should be noted that a distinct advantage of the cubature Kalman filter over the EKF is the lack of necessity to compute derivatives. Other advantages include the lack of tuning parameters, which has been known to have a distinct impact on the effectiveness of convergence on the UKF, and $2n$ points to propagate as opposed to $2n + 1$. Again, similar to the unscented Kalman filter, there are multiple variations of this

algorithm for numerical purposes, including a square root implementation. For details on this, we refer the interested reader to [2].

5 Sampling Based Kalman Filters

5.1 Ensemble Kalman Filter

The main idea around the ensemble Kalman filter is to approximate the error statistics of our estimate by a set of particles sampled from the distribution. Unlike the deterministic methods, we will calculate the prior and posterior error covariances by the ensemble covariance matrices around the corresponding ensemble mean, instead of the classical covariance equations given in the linear Kalman filter. That is, we compute

$$P_{x_{i+1|i}} = \frac{1}{K-1} U_{i+1|i} U_{i+1|i}^T \quad (53)$$

$$P_{x_{i+1|i+1}} = \frac{1}{K-1} U_{i+1|i+1} U_{i+1|i+1}^T \quad (54)$$

where

$$U_{i+1|i} = [x_{i+1|i}^1 - \hat{x}_{i+1|i}; x_{i+1|i}^2 - \hat{x}_{i+1|i}; \dots; x_{i+1|i}^K - \hat{x}_{i+1|i}] \quad (55)$$

$$U_{i+1|i+1} = [x_{i+1|i+1}^1 - \hat{x}_{i+1|i+1}; x_{i+1|i+1}^2 - \hat{x}_{i+1|i+1}; \dots; x_{i+1|i+1}^K - \hat{x}_{i+1|i+1}]. \quad (56)$$

with $x_{i+1|i}^j$ being the j^{th} particle being propagated through the model, and $x_{i+1|i+1}^j$ is the update of each particle. With this, we define K to be the number of particles in which to approximate the distribution and $\hat{x} = K^{-1} \sum_{j=1}^K x^j$. From now on, we shall define U to be the ensemble perturbation matrix as it gives us the deviation of each particle from the mean. The advantage of this methodology is that unlike the deterministic methods, there is no need to do an expensive calculation, such as the linearization or statistical transform, to get our error statistics. In place of these, we just integrate each ensemble member in time through our dynamical model.

To get the update equations, we first must calculate the Kalman gain, as before. Substituting our ensemble error covariance, we obtain

$$\begin{aligned} K_{i+1} &= P_{x_{i+1|i}} H^T (H P_{x_{i+1|i}} H^T + R)^{-1} \\ &= (K-1)^{-1} U U^T H^T (H (K-1)^{-1} U U^T H^T + R)^{-1} \\ &= (K-1)^{-1} U (H U)^T ((K-1)^{-1} (H U) (H U)^T + R)^{-1}. \end{aligned}$$

In the above expression, and also for the remainder of this paper, we omit the subscript on U for simplicity.

Looking at the Kalman gain equation, we see that anywhere the linear operator H appears, it is coupled to the ensemble perturbation matrix, U . Due to this consequence, given a nonlinear observation function h , we can take advantage by replacing HU with the following approximation

$$V = [h(x_{i+1}^1) - \hat{y}_{i+1}; h(x_{i+1}^2) - \hat{y}_{i+1}; \dots; h(x_{i+1}^K) - \hat{y}_{i+1}],$$

where \hat{y}_{i+1} is the mean of the observation function given each sample, $\hat{y}_{i+1} = \sum_{j=1}^K h(x_{i+1}^j)$.

Again, we notice that we avoid the need for costly calculations by using the ensemble approximations in place of deterministic methods.

To get our final estimates, we apply the classical Kalman filter update equations to each ensemble member

$$x_{i+1|i+1}^j = x_{i+1|i}^j + K_{i+1}(y_{i+1}^k - h(x_{i+1|i}^j)), \quad (57)$$

where K_{i+1} is defined as above, and the observation, y_{i+1} , is perturbed accordingly

$$y_{i+1}^k = y_{i+1} + \psi_{i+1}^k, \quad (58)$$

where ψ_{i+1}^k is a Gaussian random variable with mean zero and covariance R . This perturbation is a Monte Carlo method applied to the Kalman filter formula which yields an asymptotically correct analysis error covariance estimate for large ensemble sizes [19]. In practice, to keep these perturbations unbiased, we generate these random perturbations by first randomly drawing a $M \times K$ size matrix, A , where K is the number of particles and M is dimension of the data, and take the singular value decomposition of $(K - 1)^{-1}AA^T = F \Sigma F^T$. Therefore, we have unbiased random vectors of y_{i+1}^k , which is just the column vectors of the matrix $T = ((K - 1)R)^{\frac{1}{2}}F \Sigma^{-\frac{1}{2}}F^T A$ [19]. Since we are perturbing the measurements, we can define the measurement error covariance matrix to be

$$R = (K - 1)^{-1}EE^T,$$

where $E = [e_1, e_2, \dots, e_K]$ are the ensemble measurement perturbations, with mean zero. In summary, the algorithm for the ensemble Kalman filter is given as follows. An alternate implementation is given by Evensen [9], and many other implementations also exist.

Initialize: for $i = 0$

$$\hat{x}_0 = E[x_0]$$

$$P_0 = E[(x_0 - E[x_0])(x_0 - E[x_0])^T]$$

algorithm: for $i = 1, 2, 3, \dots$

State Estimation

Sample N particles from initial distribution: for $j = 1, \dots, N$

$$x_i^{-j} = f(i-1, x_{i-1}^{-j})$$

$$\hat{x}_i^- = (N)^{-1} \sum_{j=1}^N x_i^{-j}$$

Covariance Estimation

$$U = [x_i^{-1} - \hat{x}_i^-; x_i^{-2} - \hat{x}_i^-; \dots; x_i^{-N} - \hat{x}_i^-]$$

$$P_i^- = (K-1)^{-1} U U^T$$

Measurement Ensemble

$$y_i^j = y_i + \psi_i^j$$

$$Y = [y_i^1, y_i^2, \dots, y_i^N]$$

$$E = [\psi_i^1, \psi_i^2, \dots, \psi_i^N]$$

$$R = (N-1)^{-1} E E^T$$

$$V = [h(i, x_i^{-1}) - \hat{y}_i; h(i, x_i^{-2}) - \hat{y}_i; \dots; h(i, x_i^{-N}) - \hat{y}_i]$$

unbiased random vectors

$$[F, \Sigma] = \text{svd}((N-1)^{-1} E E^T)$$

$$T = ((N-1)R)^{\frac{1}{2}} F \Sigma^{-\frac{1}{2}} F^T E$$

Kalman Gain Matrix

$$K_i = (K-1)^{-1} U V^T ((K-1)^{-1} V V^T + R)^{-1}$$

State Estimation Update

$$Innov = T - Y$$

$$x_i^j = x_i^{-j} + K_i * Innov$$

Posterior Covariance Update

$$W = [x_i^1 - \hat{x}_i; x_i^2 - \hat{x}_i; \dots; x_i^N - \hat{x}_i]$$

$$P_t = (K-1)^{-1} W W^T$$

5.2 Ensemble Transform Kalman Filter

Just like the EnKF, the ensemble transform Kalman filter (ETKF) relies on sampling from the proposed normal distribution to obtain the error statistics. However, the main idea of the ETKF is to take the square root of the covariance matrix, and transform it to a space where it is more robust and well-conditioned [8, 10, 13, 19, 24].

There are multiple methods for taking the square root, the one we shall focus on from here is the ETKF as derived by Bishop [5]. The basic idea is to find a transformation matrix, T , so that

$$U_{i+1|i+1}U_{i+1|i}T = UT \quad \text{and} \quad UT(UT)^T = (K-1)R_{i+1|i+1},$$

where $R_{i+1|i+1}$ is the sample posterior covariance matrix as previously defined. The posterior ensemble is then generated by taking the posterior mean and adding each column vector from U . Using the identity

$$A^T(AA^T + R)^{-1} = (I + A^T(R^{-1})A)^{-1}A^TR^{-1}, \quad (59)$$

and letting $V = A$, we apply this to the Kalman gain and arrive at the following expression

$$K_{i+1} = (K-1)^{-1}U(I + (K-1)^{-1}V^TR^{-1}V)^{-1}V^TR^{-1}. \quad (60)$$

Substituting (60) and (53) into the posterior covariance formula, we obtain

$$P_{x_{i+1|i+1}} = (I - \frac{U}{K-1}(I + (K-1)^{-1}V^TR^{-1}V)^{-1}V^TR^{-1}H)\frac{UU^T}{K-1}. \quad (61)$$

Factoring out $\frac{U}{K-1}$ from (61) yields

$$P_{x_{i+1|i+1}} = \frac{U}{K-1}(I - (I + \frac{V^TR^{-1}V}{K-1})^{-1}\frac{V^TR^{-1}V}{K-1})U^T. \quad (62)$$

Therefore, we finally obtain

$$P_{x_{i+1|i+1}} = \frac{1}{K-1}U(I - (I + B)^{-1}B)U^T, \quad (63)$$

where $B = \frac{V^TR^{-1}V}{K-1}$. Knowing that $I - (I + B)^{-1}B = (I + B)^{-1}$, we obtain

$$P_{x_{i+1|i+1}} = U((K-1)I + V^TR^{-1}V)^{-1}U^T = U\Sigma U^T = U\frac{TT^T}{K-1}U^T. \quad (64)$$

Applying this to the previous analysis given for the EnKF, we obtain the algorithm for the ETKF.

Initialize: for $i = 0$

$$\hat{x}_0 = E[x_0]$$

$$P_0 = E[(x_0 - E[x_0])(x_0 - E[x_0])^T]$$

algorithm: for $i = 1, 2, 3, \dots$

State Estimation

Sample N particles from initial distribution: for $j = 1, \dots, N$

$$x_i^{-j} = f(i-1, x_{i-1}^j)$$

$$\hat{x}_i^- = (N)^{-1} \sum_{j=1}^N x_i^{-j}$$

Covariance Estimation

$$U = [x_i^{-1} - \hat{x}_i^-; x_i^{-2} - \hat{x}_i^-; \dots; x_i^{-N} - \hat{x}_i^-]$$

$$P_i^- = (K-1)^{-1} U U^T$$

Measurement Ensemble

$$y_i^j = y_i + \psi_i^j$$

$$Y = [y_i^1, y_i^2, \dots, y_i^N]$$

$$E = [\psi_i^1, \psi_i^2, \dots, \psi_i^N]$$

$$R = (N-1)^{-1} E E^T$$

$$V = [h(i, x_i^{-1}) - \hat{y}_i; h(i, x_i^{-2}) - \hat{y}_i; \dots; h(i, x_i^{-N}) - \hat{y}_i]$$

SVD of (64)

$$[X, \Gamma] = \text{svd}((K-1)I + V^T R^{-1} V)$$

Kalman Gain Matrix

$$K_i = U(X\Gamma^{-1}X^T)V^T R^{-1}$$

State Estimation Update

$$\hat{x}_i = \hat{x}_i^- + K_i(y_i - h(\hat{x}_i^-))$$

Transformation Matrix

$$T = \sqrt{K-1} X \Gamma^{-\frac{1}{2}} X^T$$

$$W = U T$$

Posterior Ensemble and Covariance

$$A = \hat{x}_i + W$$

$$P_i = (K-1)^{-1} A A^T$$

6 Applications

In this section, we will compare the performance of the filtering methodologies presented in the previous sections on several test problems. Their performance will be judged based upon their ability to fit the states,

and more importantly, the model parameters accurately given varying levels of state measurements. As stated previously, we will only compare the dual filters and joint filter estimation methods, and not the parameter filters. All computations were done using code written in MATLAB.

For the examples, all data was simulated from the appropriate model with known model parameters using Gaussian noise added to each state. The truth solution will be the model solution with known parameters with no noise, except in the last example.

6.1 Example 6.1: Lorenz Equations

The first example we shall use is the famous Lorenz 63 equations first studied by Lorenz in 1963 [18]. The system is made up of 3 state equations and 3 model parameters, with nonlinearities present in 2 of the 3 state equations. This test case was chosen because of the well known properties of the chaotic behavior of the Lorenz equations and strong nonlinearities. Due to the nature of the system, a small perturbation away from the correct state can lead to large qualitative changes which will easily be seen quantitatively. This system of ordinary differential equations is given by

$$\begin{aligned}\frac{dx}{dt} &= \sigma(y - x) \\ \frac{dy}{dt} &= x(\rho - z) - y \\ \frac{dz}{dt} &= xy - \beta z.\end{aligned}$$

For our test case, we generate data using the following initial conditions and parameter values with a parameter regime change occurring at time, $t = 10$.

$$\begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} = \begin{pmatrix} 0.9 \\ 1 \\ 1.1 \end{pmatrix}, \quad \begin{pmatrix} \sigma \\ \rho \\ \beta \end{pmatrix} = \begin{pmatrix} 10; 7 \\ 28; 21 \\ \frac{8}{3}; 1 \end{pmatrix}, \quad w_k = 0.1, \quad \Delta t = 0.01.$$

This was done to test each algorithm's ability to accurately track the parameters and states when perturbations to the systems dynamics occur. For each case, we start with the initial condition given from above, but with initial parameter values perturbed largely away from the true solution. The initial parameter values and covariances are

$$\begin{pmatrix} \sigma_0 \\ \rho_0 \\ \beta_0 \end{pmatrix} = \begin{pmatrix} 5 \\ 21 \\ \frac{1}{3} \end{pmatrix}, \quad Q = \begin{pmatrix} 0.001 & 0 & 0 \\ 0 & 0.001 & 0 \\ 0 & 0 & 0.001 \end{pmatrix}, \quad R = \begin{pmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.1 \end{pmatrix}, \quad Q_{\sigma, \rho, \beta} = \begin{pmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{pmatrix}.$$

Since the data is simulated, we know the observation and process noise covariance. Those were appropriately set to reflect that knowledge. However, as we are more interested in the application of filtering for parameter estimation, we perturb the initial values of the parameters away from the truth and set the covariance large to reflect that we do not know what the true parameter is and our initial guess likely has large error. Going forward with the analysis, we ran each filter with full state observations and present the results, deterministic filters first. From Figure 1, we can see that all of the deterministic methods tracked the state quite accurately over the whole simulation; from the subplot, the figure shows a more detailed look at the accurate state estimation for all methods. It is noted that the results for both the joint and dual UKF and CKF are nearly identical. More importantly, the accuracy of the methods' abilities to track the parameters through

time, and the parameter regime change, is shown in Figure 2 and Figure 3. The dual filters mean parameter estimates are presented in Figure 2, with 3 standard deviations plotted around each estimate, and the true parameter value plotted as the solid lines. As before, the difference between the CKF and UKF is negligible, and both track the parameter with the same qualitative features. As can be seen, both dual estimation methods track the parameter values well in time, with a small lag when the regime shift occurs, but quickly adapts and shifts to the correct value. From Figure 3, the joint estimation methods seem to exhibit more variation for the mean value, with erratic changes occurring frequently to the estimate. This is due to the nature of the joint estimation technique. As we are appending the parameters on as unobserved states, the method is partitioning the error equally among the states and the parameters, and can pass it back and forth. This translates into the filter moving the state to best fit the data at the cost of the parameter shifting greatly. As the parameter shifts drastically, the state is once again inaccurate and requires another sharp change in the parameter to accommodate this. This overcorrection leads to the observed qualitative behavior as the error gets propagated back and forth between the state and parameter. Because the dual filter compartmentalizes the error for the state and parameter into two distinct filters running concurrently, this behavior is not as noticeable. However, since the state filter is using estimates from the parameter filter, and vice versa, to update, the dual filter can also exhibit variation as the error of each filter is still passed, but better handled due to each filter's updated covariance. Table 1 presents the sum of square truth (SST), the difference between the filter's state estimate and the known true solution, the sum of square parameter (SSP), the difference between the true parameter and the estimated at final time, and each parameter estimate with one standard deviation.

Table 1: Results for Deterministic Filters with full state observations; Joint and Dual

	truth	j. ukf	j. ckf	j. ekf	d. ukf	d. ckf
SST	0	3.03	3.03	1.14	0.79	0.81
SSP	0	0.12	0.12	0.86	0.031	0.031
σ (1 std)	10; 7	7.22 (3.04)	7.22 (3.27)	6.91 (0.91)	6.93 (1.02)	6.93 (1.02)
ρ (1 std)	28; 21	21.23 (2.86)	21.23 (3.14)	20.63 (0.81)	20.83 (0.47)	20.83 (0.47)
β (1 std)	$\frac{8}{3}; 1$	0.87 (0.94)	0.87 (1.14)	0.84 (0.37)	1.00 (0.10)	1.00 (0.10)

Next, the ensemble filters are presented. Figure 4 shows the ensemble filters' ability to track the states, x_1 , over the parameter shift. The ensemble filters do well to accurately track the state initially, top of Figure 4, but fail to catch the parameter regime shift and therefore, fails to track the state after this event, bottom of Figure 4. This is more noticeable in Figure 5 where the ensemble parameter estimates are shown, with the solid line being the true parameter value. This inability to catch the shift is due to the nature of the sampling technique underlying the method. At each update, the samples are shifted based upon the covariance, which is shrinking due to the increased confidence and accuracy in the estimates. Therefore, when the change occurs, the ensemble filters struggle to sample from enough of the parameter space to quickly adjust, making the shift occur slowly and ultimately inaccurately. Table 2 presents the sum of squares truth, sum of squares for the parameters, and each parameter estimate with standard deviation for the ensemble filters.

An update that can be implemented to account for the ensemble shrinkage noted above is to append a process noise unto the prior. Instead of calculating the prior solely by propagating the particles through the model

$$x_i^{-j} = f(x_{i-1}^j, i-1, u_{i-1}, \theta),$$

a process noise is added to account for a known underlying model error, which prevents the shrinkage phenomena described previously. The prior calculation then becomes

$$x_i^{-j} = f(x_{i-1}^j, i-1, u_{i-1}, \theta) + w_i,$$

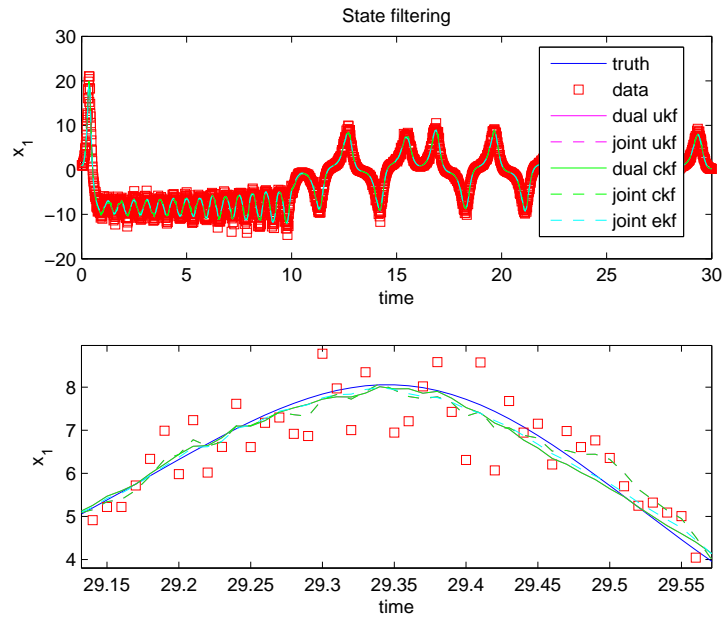


Figure 1: Plot of States due to joint and dual filters for x_1

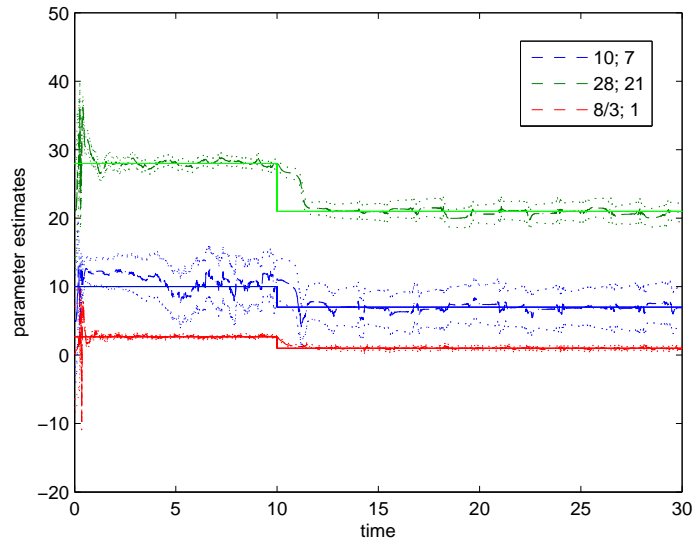


Figure 2: Plot of Parameter estimates for the Dual UKF and Dual CKF along with 3 standard deviations (dotted)

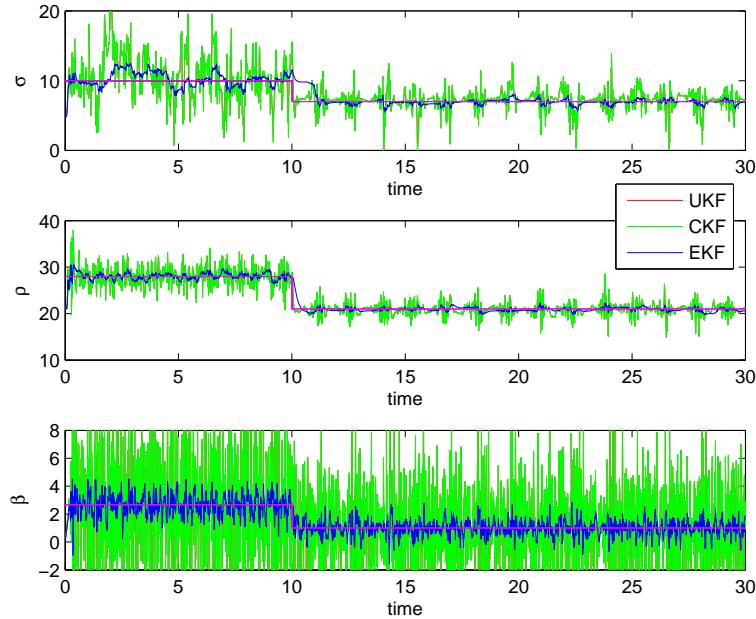


Figure 3: Plot of Parameter estimates for the Joint EKF, Joint UKF and Joint CKF

where $w_i \sim N(0, Q)$. This perturbs the prior ensemble with covariance, Q , to maintain a minimum known error. For the case described above, this process noise will allow the parameters to span the space more readily, accounting for the shift and properly estimating the parameters.

Table 2: Results for Ensemble Filters with full state observations

	truth	EnKF	ETKF
SST	0	203.69	153.96
SSP	0	42.73	82.94
σ (1 std)	10; 7	11.06 (0.005)	15.49(7.54e-4)
ρ (1 std)	28; 21	25.77 (0.0045)	23.69 (5.48e-4)
β (1 std)	$\frac{8}{3}$; 1	2.87 (5.76e-4)	2.88 (5.92e-5)

The new implementation of the ensemble filters are now presented. Figure 6 shows the updated ensemble filters' ability to track the states, x_1 and x_3 , over the parameter shift. From Figure 6, we can see the updated ensemble filters do well to accurately track the states over the entire interval, including the parameter shift. Next, the parameter estimates for the updated ensemble filters are presented in Figure 7, with the true values plotted as the solid lines. Similar to the deterministic methods, the updated ensemble methods track the parameter estimates accurately over time, including quickly adapting to the parameter shift. However, more similar to the joint methods than the dual filters, the updated ensemble methods exhibit a large degree of variation around the mean. This is due to the nature of the updated ensemble filters. Since ensemble methods are based upon random sampling, samples can be drawn that maximize the likelihood for the observed state, but with a large error in the parameter. These deviations in the parameters cause errors in the next update of the state. To account for these errors, the filter corrects the parameters. However, this causes the previous

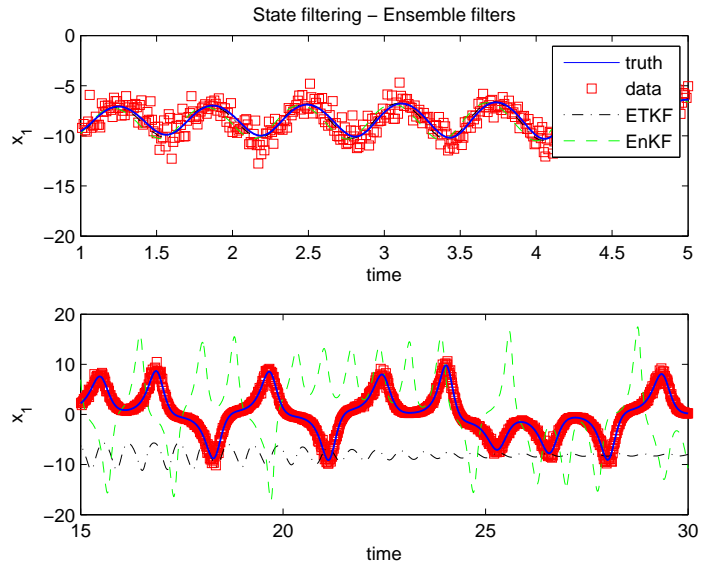


Figure 4: Plot of x_1 of the Lorenz Equations using Ensemble Filters

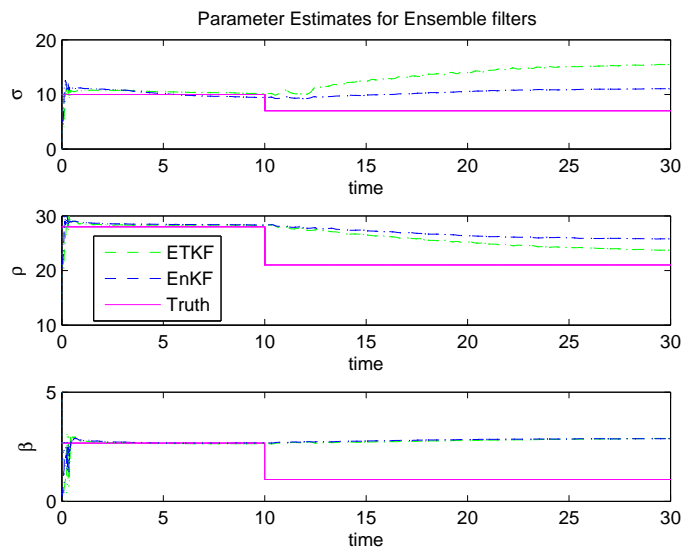


Figure 5: Plot of Parameter estimates for both Ensemble Filters

state estimates to be less accurate, which leads to adjustment in the states, and the qualitative features in Figure 5. These deviations in the parameters are made worse by the updated sampling due to the random perturbing of the particles with the process noise. This leads to extra deviation in the estimates shown in the parameters compared to the joint and dual methods. One way to account for these deviations would be to decrease the process noise. Table 3 presents the sum of squares truth, sum of squares for the parameters, and each parameter estimate with standard deviation for the updated ensemble filters.

Table 3: Results for Updated Ensemble Filters with full state observations

	truth	EnKF	ETKF
SST	0	7.88	3.02
SSP	0	0.43	1.34
σ (1 std)	10; 7	6.96 (2.69)	6.54(3.29)
ρ (1 std)	28; 21	21.60 (2.99)	19.94 (2.64)
β (1 std)	$\frac{8}{3}$; 1	0.74 (1.05)	1.09 (1.06)

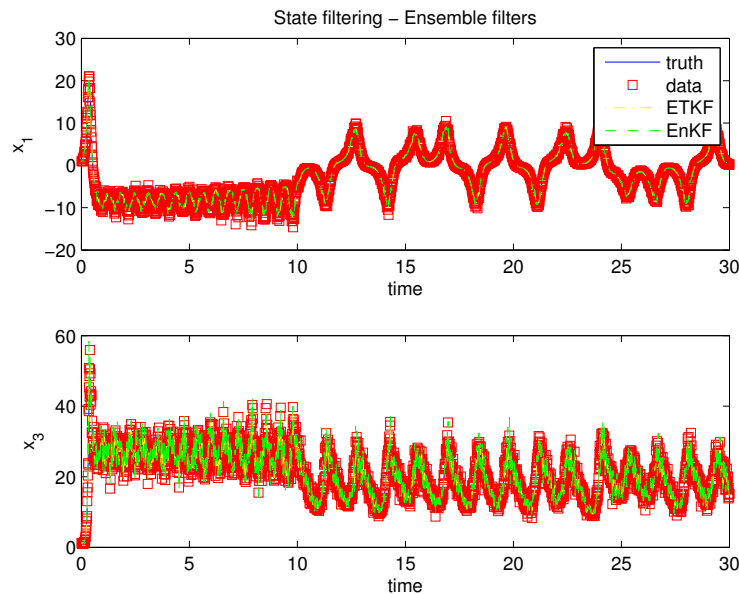


Figure 6: Plot of x_1 and x_3 of the Lorenz Equations using Updated Ensemble Filters

Next, we consider the scenario where we keep the parameters in the chaotic regime for the whole simulation, and only use observations for x_1 . This was done to study the filters' ability to estimate more states than we have observations. This is important as rarely in practice does one obtain full state observations for the system. Doing this, it is seen that the methods seem to deal quite differently to the new situation. From Figure 8, all filters do well to track x_1 for which observations are present, however, most of the methods struggle to track x_2 , except for the joint EKF and dual UKF. The bottom plot in Figure 8 does not display the dual CKF as it fails to track the state and largely deviates, in orders of magnitude, away from the truth. Figure 9 shows the parameter estimates for the methods with observations for x_1 only. Other than the joint EKF, no method seems to show much success in terms of tracking the states or parameters with observations for x_1 .

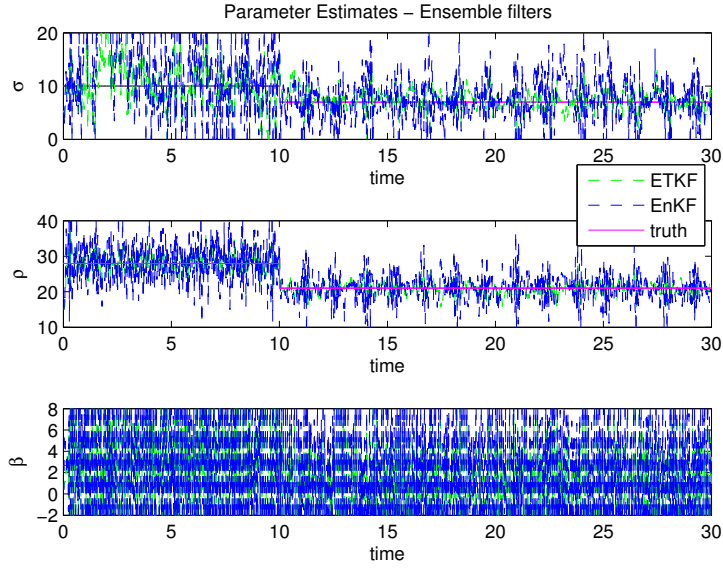


Figure 7: Plot of Parameter estimates for both Updated Ensemble Filters

Table 4: Results for Deterministic Filters given observations for x_1

	truth	j. ukf	j. ckf	j. ekf	d. ukf	d. ckf
SST	0	1.61e4	509.4	6.21	128.48	2.07e9
SSP	0	146.85	26.95	1.45	213.99	343.97
σ (1 std)	10	12.61 (4.86)	10.12 (32.83)	9.43 (0.7)	7.39 (.17)	0.002 (0.001)
ρ (1 std)	28	17.46 (3.47)	23.59 (47.4)	26.94 (1.2)	42.34 (1.90)	27.25 (111.29)
β (1 std)	$\frac{8}{3}$	8.04 (7.03)	5.40 (28.18)	2.65 (0.46)	1.42 (3.14)	18.27 (127.75)

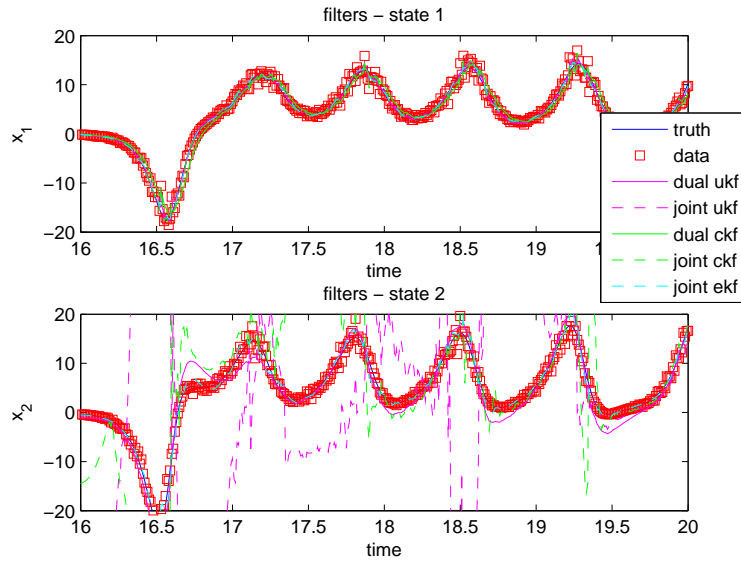


Figure 8: Plot of x_1 and x_2 for dual and joint filters given observations only for x_1

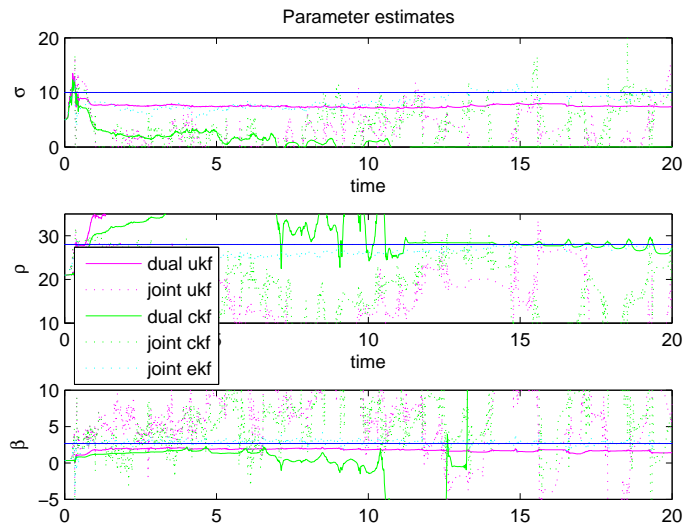


Figure 9: Plot of Parameter estimates given observations only for x_1

Moving onto the ensemble methods, Figure 10 shows the ensemble methods' ability to track both the state with observations, x_1 , as well as the unobserved states, x_3 . As seen, the ensemble methods have no trouble with the unobserved states, tracking all states accurately over the whole simulation. Reviewing their parameter estimates, Figure 11 shows the mean parameter estimate for both ensemble filters and the true parameter value. The ensemble methods quickly adapt their parameter estimates to a neighborhood of the true solution by time 3. This success by the ensemble methods seems to be due to their underlying theory of sampling. Instead of a deterministic covariance, which can lead to large errors which causes large discrepancies in the estimate, the ensemble methods use a sampling strategy which prevents these errors from propagating into the estimate as easily. This is because though the ensemble might have a few samples with a large variation from the true posterior, these outliers are averaged out by the rest of the ensemble which are properly approximating the posterior. The deterministic methods require the covariance to be accurate each step, and one inaccurate step gets propagated forward leading to less and less accurate estimates, as we observed earlier. From Table 5, note the accuracy of the ensemble methods for limited state observations, only x_1 .

Table 5: Results for Ensemble Filters given observations for x_1 only

	truth	EnKF	ETKF
SST	0	0.81	0.95
SSP	0	0.27	0.086
σ (1 std)	10	10.35 (0.007)	9.87 (0.026)
ρ (1 std)	28	27.62 (0.0038)	27.74 (0.019)
β (1 std)	$\frac{8}{3}$	2.70 (0.0006)	2.73 (0.00195)

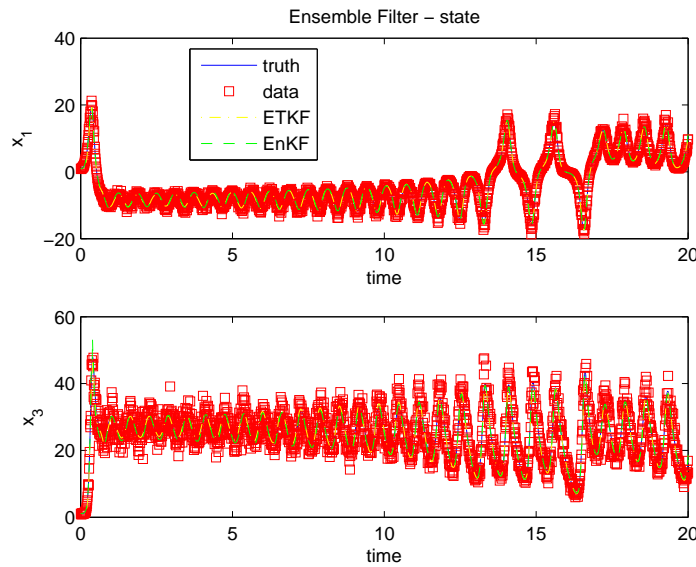


Figure 10: Plot of x_1 and x_3 of the Lorenz Equations using Ensemble Filters given observations only for x_1

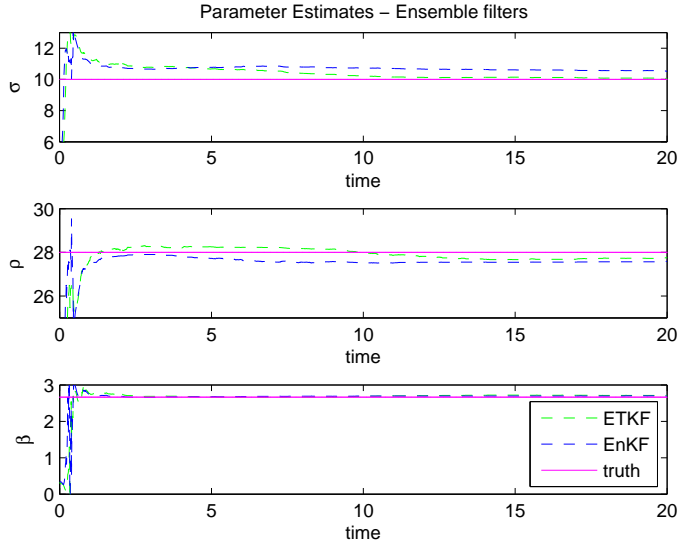


Figure 11: Plot of Parameter Estimates using Ensemble Filters given observations only for x_1

6.2 Example 6.2: HIV model

The second example is a HIV model with quadratic nonlinearities in 2 of the state equations and 6 model parameters. HIV is a viral disease that infects healthy T-cells in the body. A simple mathematical model describing the dynamics of HIV in humans is given by [11],

$$\dot{T} = \lambda - dT - kVT \quad (65)$$

$$\dot{T}^* = kTV - \delta T^* \quad (66)$$

$$\dot{V} = N\delta T^* - cV, \quad (67)$$

where λ is the recruitment of uninfected T-cells, d is the per capita death rate of uninfected cells, k is the infection rate, δ is the death rate of uninfected cells T^* , N is the number of new HIV virions V per infected cells and c is the clearance rate. This test case was not only chosen because of the difficulties it provides due to the nonlinearities and its biological background, but more importantly, because of the varying orders of magnitude the parameters and states. Due to the stiff nature of the system, numerical issues can arise which can cause inaccuracies in the estimates if not appropriately handled by the filter.

To test this model, we create data using the following initial conditions and parameter values,

$$\begin{pmatrix} T_0 \\ T_0^* \\ V_0 \end{pmatrix} = \begin{pmatrix} 1000 \\ 0.0001 \\ 0.001 \end{pmatrix}, \quad \begin{pmatrix} \lambda \\ d \\ k \\ \delta \\ N \\ c \end{pmatrix} = \begin{pmatrix} 10 \\ 0.01 \\ 8e^{-4} \\ 0.7 \\ 100 \\ 13 \end{pmatrix}, \quad w_k = 0.04, \quad \Delta t = 0.5.$$

Unlike the previous example, for this case, we create data using a multiplicative error model, with errors given by $x(1 + \epsilon)$ as opposed to $x + \epsilon$, like the previous example. This was done as it is often more realistic

for biological problems to exhibit multiplicative error in the data as opposed to additive. Also, unlike the previous example, the sampling rate is set to a much larger interval. This is done to test the filters' ability to accurately estimate the parameters over a sparse temporal data set. This is once again important as it is often difficult to obtain dense data sets for biological and biomedical experiments. Using the analysis provided by Fink et al. [11], we establish that we can only accurately estimate λ , k , and δ due to the linear dependence and correlations between the parameters. So, similar to the previous example, we start with the initial conditions given from above, but with the parameter values we wish to estimate perturbed away from the true values, with the other parameters held constant at their known values. To deal with the issues caused by the multiplicative error model and the stiffness, we use a log transformation to both the model and data. That is, we define the transformation $y_{new} = \log_{10}(y)$, where y is the original data, and $x_{new} = \log_{10}(x)$, where x is the original state dynamics. To obtain our new model, we use $\frac{dx_{new}}{dt} = \frac{1}{\log(10)x} \dot{x}$ and substitute $10^{x_{new}}$ for x . The above transformation will scale the errors in the data from multiplicative to additive, fitting the assumptions of the filter, and will also scale the states and parameters so as to prevent numerical errors from the varying orders of magnitude. The initial covariances and parameters are then given by the following, using the log transformed model and data,

$$\begin{pmatrix} \lambda_0 \\ k_0 \\ \delta_0 \end{pmatrix} = \begin{pmatrix} 6 \\ 0.0005 \\ 0.3 \end{pmatrix}, Q = \begin{pmatrix} 0.0001 & 0 & 0 \\ 0 & 0.0001 & 0 \\ 0 & 0 & 0.0001 \end{pmatrix}, R = \begin{pmatrix} 0.02 & 0 & 0 \\ 0 & 0.02 & 0 \\ 0 & 0 & 0.02 \end{pmatrix},$$

$$Q_{\lambda,k,\delta} = \begin{pmatrix} 0.0001 & 0 & 0 \\ 0 & 0.0001 & 0 \\ 0 & 0 & 0.0001 \end{pmatrix}.$$

The observation and process covariances are set appropriately knowing the nature of the simulated data. However, the initial covariance for the states and parameters, P_{x_0} and P_{q_0} , are set to be large to establish that our initial values are perturbed from the truth and likely contain a large degree of error. As the parameterization of the model rarely contributes to the process noise, and can even be negligible, the parameter process covariance is often set to be very small, to account for this in application. For our example, we utilize this as can be seen from the above covariances. Figure 12, transforming the states and parameters back to the original scaling, we can see that, except for the joint CKF, the filters are all accurately tracking the states qualitatively and quantitatively. Some overshoot on the peaks can be seen by both dual filters and both ensemble filters, but damps out as the state approaches steady state. Figure 13 depicts the same overshoot, which can be attributed to the filters convergence to the parameter estimates. Reviewing Figure 14, all 3 parameters for both dual filters are taking longer to approach the correct parameter estimate, while for the ETKF, k is slow to converge to the proper value. This can often be attributed to the model and the sensitivity of the parameters to the data. If the parameters are not very sensitive, or sensitive only in selected regions, then the estimates will take longer to converge or won't change much until those regions are evaluated by the filter. For this example, we have far fewer measurements than the previous, which can explain the slower convergence in parameter estimates to the truth.

Comparing the ensemble filters to the deterministic, the EnKF took longer to converge to the proper state estimates and had the largest error because of this. The ETKF took much less time and this is likely due to the sampling method implemented. Because we are filtering around the log-transformed model and data, small errors in log space equate to large deviations in the original space. The ETKF handled this better by using the SVD to ensure more accurate distributions. As the small errors accumulated in the EnKF sampling, the EnKF took longer to accurately converge to the correct estimates. However, the EnKF more accurately estimates

Table 6: Results for the Deterministic Filters for the HIV model in \log_{10} space

	truth	j. ukf	j. ckf	j. ekf	d. ukf	d. ckf
SST	0	2.9e-4	0.60	0.0026	0.019	0.01
SSP	0	1.12e-5	7484.4	1.02e-5	1.61e-4	9.66e-5
λ (1 std)	1	1.002 (0.182)	-6.37 (1.25)	1.01 (0.025)	0.99 (0.47)	1.01 (0.47)
k (1 std)	-3.097	-3.1 (0.162)	-10.09 (0.438)	-3.1 (0.033)	-3.09 (0.17)	-3.09 (0.17)
δ (1 std)	-0.1549	-0.156 (0.129)	-86.068 (N/A)	-0.156 (0.038)	-0.162 (0.101)	-0.158 (0.1)

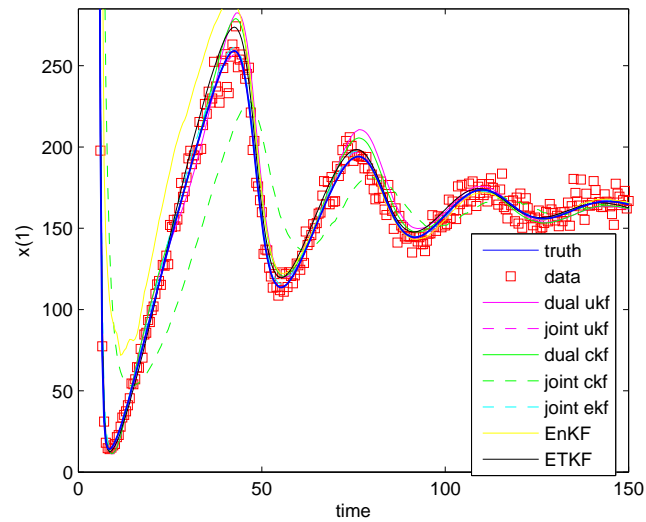


Figure 12: Plot of filter estimates for the number of uninfected T cells

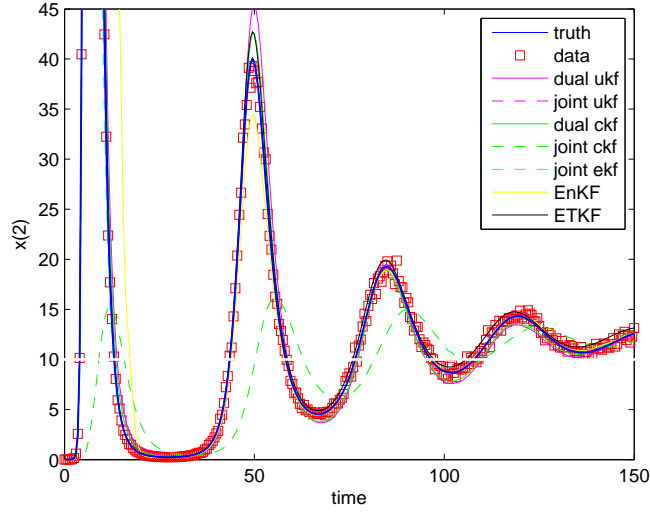


Figure 13: Plot of filter estimates for the number of infected T cells

λ better than the ETKF. For our tests, we are using 100 particles to approximate the distribution. Enlarging the number of particles could improve the robustness of the estimates, but at the cost of computational time. Since each particle results in a function evaluation, the computation time increases with a larger number of particles. Since the deterministic filters all propagate the covariance through a deterministic algorithm, they do not encounter this sampling error. However, because they are solely based upon their deterministic methodology for estimating the covariance, if an error is encountered, the repercussions are directly felt in the estimate and usually results in inaccuracies and divergence. The ensemble methods, though they incorporate a sampling error, do use averaging to weed out these poor estimates and are therefore far more robust to errors in the covariance over the course of a run. Reviewing Tables 6 and 7, we can again see the accuracy for both EnKF and ETKF methods for this case.

Table 7: Results for the Ensemble Filters for the HIV model in \log_{10} space

	truth	EnKF	ETKF
SST	0	0.24	0.002
SSP	0	1.9e-5	1.1e-4
λ (1 std)	1	1.0033 (1.6e-4)	1.0097 (7.4e-4)
k (1 std)	-3.097	-3.099 (1e-4)	-3.094 (1e-3)
δ (1 std)	-0.1549	-0.156 (1e-4)	-1.574 (6.3e-4)

6.3 Example 6.3: Autoregulation model

The final example is a model used to describe cerebral autoregulation coupled with intercranial hemodynamics. Many models have been developed for this purpose, but Ursino and Lodi's [22] (U-L) model is selected as it is simple, but still retains the important features to describe the cerebral blood flow. The goal is to estimate the blood flow velocity in response to changes in arterial blood pressure. Viewing arterial pressure as a function of time, $p_a = p_a(t)$, and using the diagram of cerebral dynamics to construct a circuit analogy, the

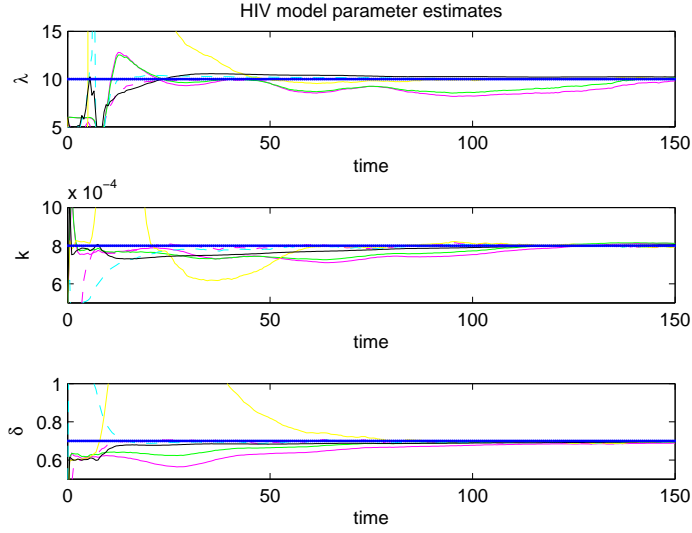


Figure 14: Plot of parameter estimates for λ , k and δ for all the filters; blue dashed line is the true value, solid magenta line is the dual ukf, dashed magenta is the joint ukf, solid green line is the dual ckf, dashed green is the joint ckf, solid cyan line is the joint ekf, solid yellow is the EnKF, and solid black is the ETKF

U-L model is given by

$$\begin{aligned}\frac{dp_{ic}}{dt} &= \frac{k_E p_{ic}}{1 + C_a k_E p_{ic}} \left[C_a \frac{dp_a}{dt} + \frac{dC_a}{dt} (p_a - p_{ic}) + \frac{p_c - p_{ic}}{R_f} - \frac{p_{ic} - p_{vs}}{R_0} \right] \\ \frac{dC_a}{dt} &= \frac{1}{\tau} [-C_a + \sigma(x)],\end{aligned}$$

with observation function

$$v(t_i) = \frac{1}{A_c R_{pv}} \left[\frac{(p_a - p_{ic})^3}{p_a^2 - 2p_a p_{ic} + p_{ic} + \frac{k_R}{C_a^2 R_{pv}}} \right],$$

where

$$\begin{aligned}\sigma(Gx) &= \frac{(C_{an} + \Delta C_a/2) + (C_{an} - \Delta C_a/2)e^{Gx/k_\sigma}}{1 + e^{Gx/k_\sigma}} \\ R_a &= \frac{k_R}{V_a^2}, \quad V_a = C_a(p_a - p_{ic}) \\ p_c &= \frac{p_a R_{pv} + p_{ic} R_a}{R_{pv} + R_a},\end{aligned}$$

and $x = \frac{q - q_n}{q_n}$, where q is the arterial blood flow defined to be $q = \frac{p_a - p_c}{R_a}$. Simplifying to our general state space framework, $\vec{x} = [p_{ic}, C_a]^T$ is the state vector of intracranial pressure and arterial compliance, respectively, $v(x, t, \theta)$ is the observation function, and the parameters, θ , are given below

$$\theta = \{q_n, p_{vs}, k_E, \tau, G, R_0, R_{pv}, R_f, \Delta C_{a1}, \Delta C_{a2}, C_{an}, k_R, A_c\}.$$

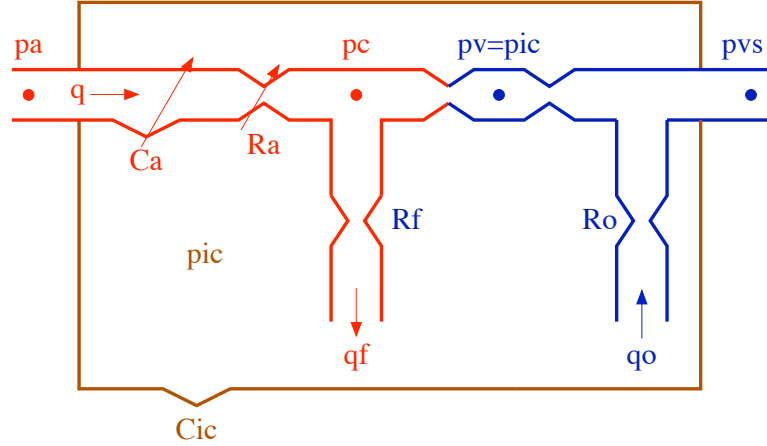


Figure 15: Circuit diagram of cerebral dynamics

This model has many challenges including nonlinearities, oscillatory input function, p_a , many parameters, single state observation, and stiffness due to varying orders of magnitude in parameter values. Because of the large number of parameters, we run a subset selection following the same procedure as described in the previous example [11]. This led to 5 of the 13 parameters being found to be identifiable and uncorrelated; they are $k_E, \tau, G, C_{an}, k_R$. Before estimating the real data, we first simulated data from the model using nominal values for the parameters and initial conditions, using arterial pressure input from a standard patient. Using stepsize, $\Delta t = 0.08$, and noise level, $w_k = 0.02$, we created the data with initial conditions, $[p_{ic}, C_a, \hat{v}]^T = [4.35, 0.15, 77.28]^T$, and parameter given by Table 8.

It is noted that from Table 8 the values that were arterial pressure dependent were calculated using the standard patient pressure data. After creating the data, we then went on to estimate the parameters by using a log-transform solely on the parameters.

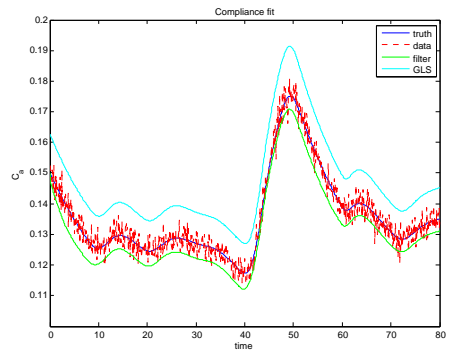
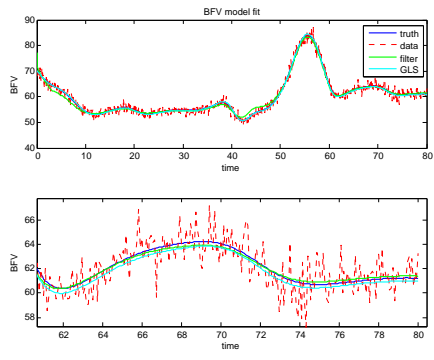
Setting the covariances appropriately, and perturbing the parameter values to be estimated, we get

$$\begin{pmatrix} k_{e0} \\ \tau_0 \\ G_0 \\ C_{an0} \\ k_{R0} \end{pmatrix} = \begin{pmatrix} 0.041 \\ 11 \\ 0.6 \\ 0.27 \\ 3e4 \end{pmatrix}, Q = \begin{pmatrix} 0.0001 & 0 & 0 \\ 0 & 0.0001 & 0 \\ 0 & 0 & 0.0001 \end{pmatrix}$$

$$Q_{k_e, \tau, G, C_{an}, k_R} = \begin{pmatrix} 0.0001 & 0 & 0 & 0 & 0 \\ 0 & 0.0001 & 0 & 0 & 0 \\ 0 & 0 & 0.0001 & 0 & 0 \\ 0 & 0 & 0 & 0.0001 & 0 \\ 0 & 0 & 0 & 0 & 0.0001 \end{pmatrix}$$

with $R = 0.02$, $P_{x_0} = 10I$, and $P_{q_0} = 10I$, where I is the identity matrix.

Running all the filters over the simulated data, we found that only the ensemble methods successfully converged. While the dual UKF could track the blood flow velocity, \hat{v} , it failed to get reasonable compliance, C_a , results. Along with that, the dual UKF not only failed to estimate the parameters, the covariance and confidence in the parameter estimates diverged over time. The dual CKF failed to solve as it encountered numerical errors and completely diverged. The joint filters also failed to estimate due to the limited state observations coupled with the large number of parameters. The lack of success of the deterministic filters



(a) simulated data, \hat{v} , and filter and NLS fit using Levenberg-Marquardt (b) hidden state, C_a , and filter and NLS result due to estimation

Figure 16: Simulated data, filter and LM fits

Table 8: Parameters of the hemodynamic model of CA. Superscript * indicates that the value of the parameter was determined by the suggested value in [22]. Subscript n indicates that the parameter represents a basal value.

Parameter	Description	Initial Value
p_{cn} (mmHg)	Capillary pressure	25*
p_{icn} (mmHg)	ICP	9.5*
q_n (ml/s)	Arterial flow	15.114
p_{vsn} (mmHg)	Venous pressure	6*
V_{an} (ml)	Arterial volume	13.5*
q_{fn} (ml/s)	CSF formation rate	2/300*
k_E (ml ⁻¹)	Intracranial elastance	.11*
τ (s)	CA relaxation time	20*
G (unitless)	CA gain	1.5*
p_{an} (mmHg)	Arterial pressure	mean arterial pressure from data
R_0 (mmHg · s/ml)	CSF outflow resist.	$(p_{icn} - p_{vsn})/q_{fn}$
R_f (mmHg · s/ml)	CSF formation resist.	$(p_{cn} - p_{icn})/q_{fn}$
C_{an} (ml/mmHg)	Arterial compliance	$V_{an}/(p_{an} - p_{icn})$
$C_{a,mx}$ (ml/mmHg)	Max C_a	$6 C_{an}$
$C_{a,mn}$ (ml/mmHg)	Min C_a	$.5 C_{an}$
ΔC_{a1} (ml/mmHg)	σ amplitude 1	$2(C_{a,mx} - C_{an})$
ΔC_{a2} (ml/mmHg)	σ amplitude 2	$2(C_{an} - C_{a,mn})$
R_{an} (mmHg · s/ml)	Arterial resistance	$(p_{an} - p_{cn})/q_n$
R_{pv} (mmHg · s/ml)	Pial vein resistance	$R_{an} \frac{p_{icn} - p_{cn}}{p_{cn} - p_{an}}$
k_R (mmHg ³ · s/ml)	Resistance coefficient	$V_{an}^2 \frac{p_{an} - p_{cn}}{C_{an}^2 q_n}$
A_c (cm ²)	Area of MCA	0.25
C_{a0} (ml/mmHg)	Initial C_a	C_{an}
P_{ic0} (mmHg)	Initial ICP	p_{icn}

stems from their covariance estimation methodology. As they each are using a linearization or numerical approximation to characterize the covariance of the posterior distribution, they are not capturing the true posterior of this system. As these errors accumulate, the confidence in the estimates diverges and the filter resolves to solely fitting the data. This creates poor confidence in the state estimates and a complete failure in the parameter estimates. The ensemble methods succeed in this instance because their covariance calculation is based upon sampling which is more robust to errors. As before, the few poor approximations are cancelled out by the large number of accurate particles, keeping the estimate and covariance closer to the true posterior. To compare the ensemble filter's success to a standard approach in real application, we also ran a nonlinear least squares estimation using Levenberg-Marquardt, LM. From Figure 16, we can see that the ensemble method, ETKF, is comparable to the nonlinear least squares fit using Levenberg-Marquardt for blood flow; however, the filter seems to more accurately approximate the hidden state, C_a . The EnKF also converged successfully for this problem, but the results were left out as they were less accurate than the ETKF. From Figure 17, we can see how the filter does in converging to the parameter estimates. As mentioned in the previous example, if the parameters are only sensitive in specific areas, they will only shift once those regions are encountered. This can explain the convergence plots for each parameter. The full results are displayed in Table 9.

Moving onto the real patient data, and using the knowledge gained from the simulated data, we go on to

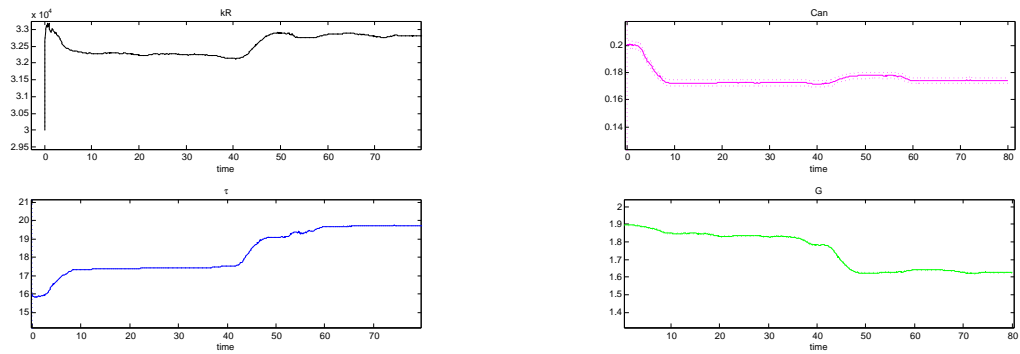


Figure 17: parameter estimates for 4 of the 5 parameters

Table 9: Results of ETKF vs. LM for simulated blood flow data with parameters in \log_{10} space

	truth	NLS	ETKF
SST	0	2.09e3	1.82e3
SSP	0	25.98	24.92
Ke (1 std)	-0.9586	-0.6861	-1.4686 (0.0048)
τ (1 std)	1.301	2.0237	1.2334 (0.002)
G (1 std)	0.1761	1.0035	0.0227 (0.0019)
C_{an} (1 std)	-0.8342	0.0917	-0.9075 (0.003)
kR (1 std)	4.6335	4.5453	4.7344 (0.0004)

use the ETKF to estimate both the blood flow velocity and parameters for the autoregulation model. Again, we compare the results to the standard nonlinear least squares approach using LM. Using the initial conditions and parameter estimates from the simulated data, we can see the fit to the blood flow velocity in Figure 18 for both methods. The sum of squares residual (SSR) and parameter estimates are given in Table 10. One issue in comparing the filter to the nonlinear least squares approach is that we are looking at the SSR which will be biased towards the NLS approach. Since NLS is based upon a minimization of the residuals, it should exhibit a theoretical minimum for the problem. However, if we increase the covariance for the model and put a large confidence in the data, the filter will track the data without any regard for the model, and it will minimize the residual, but with little regard to the model. So, though the SSR is useful in terms of knowing how well the model may be fitting the data in a relative viewpoint, it is not the best metric for comparison of these two methods. More thought needs to be put into an optimal method to objectively compare the efficiency of each method for a given model fit. Reviewing the results, we can see the qualitative differences in the two estimation methods, with the lower SSR for the NLS likely due to the under shoot of the filter when the blood flow dips before reaching its maximum. Reviewing the estimates, we can also see substantial differences which account for the different dynamics. All in all, the ETKF produces comparable results to the NLS and fits the data reasonably well.

Table 10: Results of ETKF vs. LM for real blood flow data with parameters in \log_{10} space

	NLS	ETKF
SSR	3.58e3	3.80e3
Ke (1 std)	-0.9318	-0.8861 (0.0022)
τ (1 std)	1.0022	1.0892 (0.0009)
G (1 std)	0.3304	0.2765 (0.0009)
C_{an} (1 std)	-0.7545	-0.7212 (0.0009)
kR (1 std)	4.6149	4.6848 (0.0009)

7 Conclusion

In comparing a multitude of nonlinear filtering techniques for parameter estimation, we have seen a number of advantages for each method. The joint extended Kalman filter seems to exhibit the best start point for any state and parameter estimation problem due to its simplicity in implementation coupled with accurate results. It handled the nonlinear dynamics in each example well, even with limited state knowledge. It resolved the parameter accurately, and with high confidence, in each example and was robust to each examples challenges. Though it did not present the best fit to the states in each case, it did resolve the parameter estimates accurately, and with high confidence, while being robust to each challenge presented. The UKF, both dual and joint, exhibited success in estimation of both the states and parameters as long as there was adequate data present. These methods struggled in the Lorenz example when only one state was being observed. Their level of robustness and accuracy were good, but dependent upon high levels of state knowledge. However, the ability to run without model linearizations gives this method a distinct advantage in runtime and simplicity that the EKF does not exhibit. The joint CKF and dual CKF exhibited equivalent results to the UKF methods, but with failure in many cases where the UKF methods still succeeded. The negligible difference in accuracy coupled with the higher failure rate makes the CKF method's less usefulness in applications. The ensemble method's only shortcoming is its inability to span the space well in long runs. Since it does well to converge to

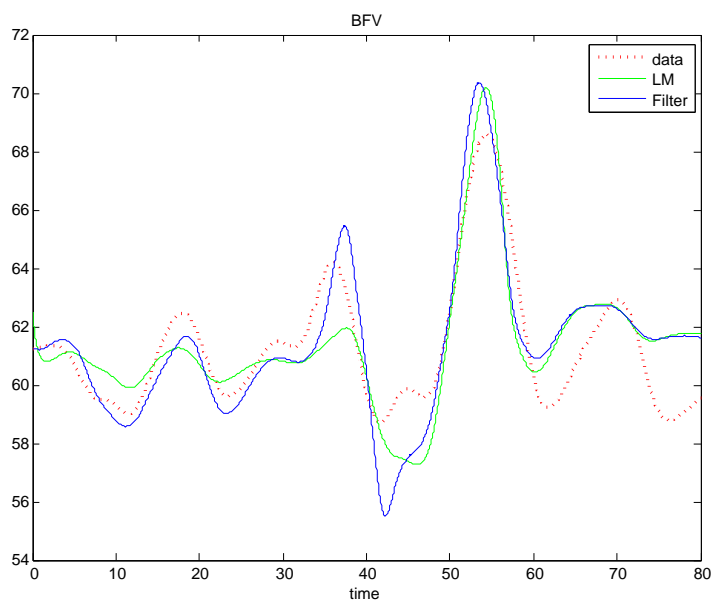


Figure 18: Plot of real data, filter estimate and NLS approaches

the appropriate answer, the confidence in the estimate increases and large changes take longer to be corrected. Updating the ensemble with the model and parameter noise processes, the ensemble Kalman filters overcame this adversity and produced the best results in state and parameter estimates. The only other potential drawback is the increase in computational time as it scales proportionally to the number of particles. However, if time is not an issue, the ensemble methods were among the best estimates, with highest confidence, in all three examples. The ensemble methods were also the only methods to converge on the real data. While the other methods could not sufficiently handle the covariance updates for the final example, the ensemble methods demonstrated a high level of robustness and success in the most challenging problems. In cases of extreme complexity and challenge, the ensemble methods seem to be the best method to choose.

Overall, although the extended Kalman filter is the best start point due to simplicity, the ensemble filters were the best filter for our applications. The UKF obtained the same level of accuracy as the CKF in all examples where both converged, however, the UKF converged more frequently than the CKF. The dual UKF and CKF exhibited less variation in estimates and failed less frequently than their joint counterparts. The EKF succeeded in estimating both the states and parameters in the first two applications, but failed in the last application. The ensemble filters managed to succeed in every example including the real data problem.

We have used nonlinear filtering techniques for the estimation of both the states and parameters and shown comparable results when compared to the standard cost function minimization method. In cases where online parameter estimation is necessary or parameter regimes can shift temporally, nonlinear filtering has shown favorable results.

8 Acknowledgements

This research was supported in part by the National Institute of Allergy and Infectious Disease under grant NIAID 9R01AI071915 and by the National Science Foundation under grant NSF-DMS 1022688. The authors would like to thank John Harlim for stimulating and fruitful conversations during the course of their efforts.

References

- [1] Anderson, B. "An Ensemble Adjustment Filter for Data Assimilation." Monthly Weather Review, Vol. 129, 2001, Pgs. 2884-2903
- [2] Arasaratnam, I. and Haykin, S. "Cubature Kalman Filters." IEEE Transactions in Automatic Control. Vol. 54, 2009. Pgs. 1254-1269.
- [3] Arasaratnam, I., Haykin, S., and Hurd, T. "Cubature Kalman Filtering for Continuous-Discrete Systems: Theory and Simulations." IEEE Transactions on Signal Processing. Vol. 58, No. 10, 2010, Pgs. 4977-4993
- [4] Arulampalam, S. M., Maskell, S., Gordon, N., and Clapp, T. "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking." IEEE Transactions on Signal Processing. Vol. 50, no. 2, 2002, Pgs. 174-188.
- [5] Bishop, C.H., Etherton, B.J., and Majumdar, S.J. "Adaptive Sampling with the Ensemble Transform Kalman Filter. Part I: Theoretical Aspects." Monthly Weather Review. Vol. 129, Pgs. 420-436.
- [6] Candy, J. V. Bayesian Signal Processing: Classical, Modern and Particle Filtering Methods. New Jersey: John Wiley and Sons, Inc., 2009.

- [7] Daum, F. "Nonlinear Filters: Beyond the Kalman Filter." IEEE AE Systems Magazine. Vol. 20, No. 8, 2005. Pgs. 57-69.
- [8] Evensen, G. Data Assimilation: The Ensemble Kalman Filter. New York: Springer, 2009.
- [9] Evensen, G. "The Ensemble Kalman Filter for Combined State and Parameter Estimation." IEEE Control Systems Magazine. Vol. 29, 2009. Pgs. 83-104.
- [10] Evensen, G. "The Ensemble Kalman Filter: Theoretical Formulation and Practical Implementation." Ocean Dynamics. Vol. 54, 2003. Pgs. 539-560.
- [11] Fink, M., Attarian, A., and Tran, H. "Subset Selection for Parameter Estimation in an HIV Model." PAMM. Vol. 7, No. 1, Pgs. 1121501-1121502.
- [12] Gershgorin, Majda, and Harlim, J. "Mathematical Strategies for Filtering Turbulent Dynamical Systems." Discrete Contin. Dynam. Syst. A, Vol. 27, No. 2, 2010. Pgs. 441-486
- [13] Haykin, S. Kalman Filtering and Neural Networks. New York: John Wiley and Sons, Inc., 2001.
- [14] Hunt, B., Kostelich, E., and Szunyogh, I. "Efficient Data Assimilation for Spatiotemporal Chaos: A Local Ensemble Transform Kalman Filter." Physica D, Vol. 230, 2007, Pgs. 123-137.
- [15] Julier, S., J. and Uhlmann, J. K. "A New Extension of the Kalman filter to Nonlinear Systems." In Proc. of Aerosense: The 11th Int. Symp. on Aerospace/Defense Sensing, Simulation and Controls. 1997.
- [16] Kalman, R. E. "A New Approach to Linear Filtering and Prediction Problems." ASME J. Basic Engineering. Vol. 82, 1960, Pgs. 34-45.
- [17] Lewis, F. L. Optimal Estimation. New York: John Wiley and Sons, Inc., 1986.
- [18] Lorenz, E. N. "Deterministic Nonperiodic Flow." Journal of the Atmospheric Sciences. Vol. 20, 1963. Pgs. 130-141.
- [19] Majda and Harlim, J. Filtering Complex Turbulent Systems. Cambridge University Press, United Kingdom, 2012.
- [20] Ristic, B., Arulampalam, S., and Gordon, N. Beyond the Kalman Filter: Particle filters for tracking applications. Boston: Artech House, 2004.
- [21] Sarkka, S. "On Unscented Kalman Filtering for State Estimation of Continuous-Time Nonlinear Systems." IEEE Transactions on Automatic Control. Vol. 52, No. 9, 2007. Pgs. 1631-1641.
- [22] Ursino, M. and Lodi, C. A. "A Simple Mathematical Model of the Interaction between Intracranial Pressure and Cerebral Hemodynamics." Journal of Applied Physiology. Vol. 82, 1997. Pgs. 1256-1269.
- [23] Wann, E. and Van Der Merwe, R. "The Unscented Kalman Filter for Nonlinear Estimation." Adaptive Systems for Signal Processing, Communications, and Control Symposium, IEEE. 2001. Pgs. 153-158.
- [24] Wan, E. and Van Der Merwe, R. "The Square-Root Unscented Kalman Filter for State and Parameter Estimation." IEEE International Conference on Acoustics, Speech and Signal Processing. Vol. 6, 2001. Pgs. 3461-3464.