Vol 4(Iss.2), pp.222-242, 2024 Science Mundi ISSN:2788-5844 http://sciencemundi.net

# Second order Extended Ensemble Kalman Filter with Stochastically Perturbed Innovation

**Cavin Oyugi Ongere**[*1]
**David Angwenyi**[2]
**Robert Oryiema**[3]

[1] *cavinvoic@gmail.com*
[2] *dangwenyi@mmust.ac.ke*
[3] *robertoryiema@gmail.com*

[1,2,3] *Department of Mathematics, Masinde Muliro University of Science and Technology, P. O. Box 190-50100, Kakamega, Kenya.*

## ABSTRACT

Studies have shown several forms of non-linear dynamic filters. However, Extended Kalman filters have proved to provide more accurate values of the state of dynamic systems over period of time. Though, the results of estimation by use extended Kalman filters are accurate, there is involvement of computation of high dimension covariance matrix that are very expensive. Although Bayesian methods offer a robust and accurate approach, they are often hindered by the computational complexity involved in computing high-dimensional matrices. This study introduces a new filter, the Second Order Extended Ensemble Filter with pertubed innovation (SoEEFPI), designed to numerically address the inversion of high-dimensional covariance matrices and then stochastically perturbing the innovation. The SoEEFPI is derived from the numerical expansion of the expected values of non-linear terms in the stochastically perturbed Kushner-Stratonovich equation, utilizing a second-order Taylor series expansion. Validation of the SoEEFPI is conducted on a three-dimensional stochastic Lorenz 63 model, with simulations performed using MATLAB software. In the validation process , SoEEKFPII is compared with First Order Extended Ensemble Filter (FoEEF), First Order Extended Kalman Bucy Filter (FoEKBF), Second order Extended Ensemble Filter (SoEEF), Bootstrap Particle Filter, and Second Order Extended Kalman Bucy Filter (SoEKBF). Results indicated that SoEEFPI outperformed the other filters (KBF, FoEEF, SoEEF) across all three variables of the Lorenz 63 model: $x_1, x_2$ and $x_3$. While SoEKBF exhibited the lowest root mean square error (RMSE), its computational cost is significantly higher due to the integration of high-dimensional covariance, making SoEEFPI a more desirable option since its covariance computation is performed empirically.

**Mathematics Subject Classification:** Primary 35A01; Secondary 65L10.

**Keywords:** Non-linaer filtering, Non-linear state space dynamic models, Bayesian methods

# 1 Introduction

The introduction of a First Order Extended Ensemble Filter (FoEEF) offers an alternative approach, calculating covariance matrices empirically instead of relying on matrix multiplication, thus expediting the convergence process [12]. However, the first-order term used in the Taylor series expansion of nonlinear terms may yield less accurate results compared to filters developed using a second-order Taylor polynomial. Recently, Kevin Midenyo developed a second Order Extended Ensemble Filter which proved more efficient in estimation than the First Order Extended Ensemble Filter [9]. However there is a need for a new filter which incorporates the noisy components of the dynamic system. Therefore, this study has developed a Second Order Extended Ensemble Filter (SoEEF) with stochastic perturbed innovation, potentially more efficient for initializing neural network weights. The SoEEFPI utilizes an empirical method to estimate the inverse of covariance matrices, addressing the computational challenges and costs associated with previous Kalman filters. As a second-order estimate, the SoEEFPI provides greater accuracy compared to first-order estimations. Its efficient implementation will benefit various sectors, such as manufacturing, image recognition, healthcare, and transportation in estimations.

# 2 Literature Review

[12] Proposed the first-order extended ensemble filter to solve the computationally intensive initialization of weights in artificial neural networks. Unlike the Kalman filter-based approach, this method avoids the direct computation of the matrix products and their inverses, thus improving the learning rate. The first-order extended ensemble filter uses an empirical estimate of the matrix products and inverses through a weighted ensemble of samples. The algorithm first generates a set of particles, each representing a possible value of the weight parameters. The computation of the matrix products and inverses in the first-order extended ensemble filter is replaced by the empirical estimates of the mean and covariance of the particles. The mean and covariance are updated using the weighted samples, and the covariance is modified to account for the resampling process. This modification ensures that the covariance of the new set of particles is equal to the covariance of the previous set of particles, which is an essential property for maintaining the diversity of the particles [10].

## 2.1 Non-Linear Filters

Kalman filters have been applied in many areas and more significantly in the non-linear filtering methods that are just but the extension of the Kalman filter. When working with the non linear models, it is prudent to make an assumption that the probability densities and the conditional densities are Gaussian. In the non-linear filters, the Extended Kalman Filters (EKF) has been utilized and it is just an extension of the Kalman filters. In EKF, the evolution of the time state vector is well described by a dynamical model. In this case, we have different equations and the observation mode amd the noice part as shown below;

$$y_k = f(y_{k-1}) + w_{k-1} \tag{2.1}$$

where; $w_k \sim N(0, Q)$

$$x_k = h(x_k) + A_k \tag{2.2}$$

where; $A_k \sim N(0, R)$,

The initial state $y_0$ is a random vector and the mean is;

$$\mu_0 = E[y_0]$$

and the covariance is given by;

$$\rho_0 = \left[(y_0 - \mu_0)(y_0 - \mu_0)^T\right]$$

where;

$y_k$ is $n \times 1$ state vector,
$w_k$ is $n \times 1$ process noice vector,
$x_k$ is $m \times 1$ measurement vector,
$A_k$ is $m \times 1$ measurement noice vector,
$f(y - k)$ $n \times 1$ process non-linear vector,
$h(y_k)$ $m \times 1$ observation non-linear vector finction,
$k$ is the index of time.

The EKF algorithm is shown in the following steps.

**Step 1:** Initializing estimated initial state vector

$$\hat{y}_0^n = \mu \tag{2.3}$$

The error covariance matrix $\rho_0$

**Step 2:** Predicting the state vector.

$$\hat{y}_{k+1} = f(\hat{y}_k, \mu_k) \tag{2.4}$$

then predict error covariance ahead;

$$P_k = V_k P_k V^T + Q_k \tag{2.5}$$

**Step 3:** Calculation of the observation matrix.

$$H_{y_{k+1}} = \frac{\partial h}{\partial y} \tag{2.6}$$

which is the Jacobian of the non-linear measurement function $h(y_k)$.

**Step 4:** Acquiring new measurement vector.

$$x_{(k+1)} \tag{2.7}$$

**Step 5:** Calculating the Kalman gain matrix.

$$M_k = P_k H (H_k P_k H^T + T)^{-1} \tag{2.8}$$

the correction of predicted results, that is, updated state vector with measurement $Z_k$ is given by,

$$y_k = \hat{y}_k + K_k \Big( Z_k - H(\hat{y}_k) \Big)$$

the update, error covariance

$$P_k = (1 - K_k H_k) P_k \tag{2.9}$$

**Step 5:** Corrected $\hat{y}_k$ (output) now becomes the previous state in the next iteration and the process is then repeated.

The EKF does not perform well when noise is non-Gaussian and the dynamic system must be linear. Though applied widely, the high computational cost makes it more expensive.

## 2.2 Kalman-Bucy Filter

The weak form of the Kushner-Stratonovich equation, when integrated over $y$, gives the estimate $\hat{y}$:

$$d\hat{y} = F(t)\hat{y}_t dt + Q_t H^T(t)(dx_t - H(t)\hat{y}_t dt) \text{ [3]} \tag{2.10}$$

The equation of covariance is described as:

$$dQ_t = F(t)Q_t dt + Q_t F^T(t)dt + G(t)G^T(t)dt - Q_t H^T(t)R^{-1}(t)H(t)Q_t dt \tag{2.11}$$

The optimal estimate of $y$ given $x$ is provided by equations2.10 and 2.11 above equations. The prediction steps involve the mean estimate:

$$d\hat{y} = F(t)\hat{y}_t dt \tag{2.12}$$

and the covariance:

$$dQ_t = F(t)Q_t dt + Q_t F^T(t)dt + G(t)G^T(t)dt$$

The new measurement is captured by the additive term in the predicted estimator:

$$Q_t H^T(t)\big(dx_t - H(t)\hat{y}_t dt\big)$$

## 2.3 Bootstrap Particle Filter

In this approach, Monte Carlo methods are utilized to numerically approximate estimates of mean and covariance. According to [15] [8] ,the posterior distribution is approximated as:

$$q(y_t|x_{1:t})$$

The set of particles is denoted as:

$$M_t = (y_t^i, w_t^i); \quad i = 1, 2, ..., M.$$

Here $y_t^i$ represents the dynamic state, $w_t^i$ signifies the weight of the particle at time $t$, and $x_{1:t}$ denotes the measurements. The filtering probability density is defined as:

$$Q\big(y_t|x_{1:t}\big) = \sum_{i=1}^{M} w_t^i \delta(y_t - y_t^i) \tag{2.13}$$

## 2.4 Ensemble Kalman-Bucy Filter

The Ensemble Kalman-Bucy Filter (EKBF) is based on the particle filter and is designed for modeling continuous dynamic processes. The covariance is estimated using Monte Carlo techniques [3]. The evolution of the state is represented as:

$$dy_t^i = F(t)y_t^i dt + Z(t)dB_t^i + Q_t^M H^T(t)R^{-1}(t)\big(dx_t + R^{\frac{1}{2}}(t)\eta_i^t - H(t)y_t^i dt\big) \tag{2.14}$$

The mean of the ensemble is given by:

$$\hat{y}_t = \frac{1}{N}\sum_{i=1}^{N} y_t^i \tag{2.15}$$

The covariance is estimated as:

$$Q_t = \frac{1}{N-1}\sum_{i=1}^{N}(y_t^i - \hat{y}_t)(y_t^i - \hat{y}_t)^T \tag{2.16}$$

## 2.5 First Order Extended Kalman-Bucy Filter (FoEKBF)

In scalar forms, the first-order model equations are given as follows:

$$\begin{aligned} \text{signal}: &\ dy_t = f(y_t)dt + g(t)(t)dB_t, \quad x_{t_0}, t_0 < t \\ \text{measurement}: &\ dx_t = h(y_t)dt + R^{\frac{1}{2}}(t)d\eta_t, \quad t_0 \le t \end{aligned}$$

Substituting $g(y,t)$ with $g(t)$ in corresponding derivatives and using Taylor series expansion around the mean, we arrive at:

$$d\hat{y}_t = f(\hat{y}_t)dt + Q_t\nabla[h](\hat{y}_t)R^{-1}(t)\big(dx_t - h(\hat{x}_t)dt\big)$$

### 2.6 Second Order Extended Kalman-Bucy Filter (SoEKBF)

The Second Order Extended Kalman-Bucy Filter (SoEKBF) is derived from a second-order Taylor expansion around the mean [5]. The equations are as follows:

$$d\hat{y}_t = f(\hat{y}_t)dt + \frac{1}{2}\Delta[f]\Big(\hat{y}_t Q_t dt + Q\nabla[h]^T(\hat{y}_t)R^{-1}(t)(dx_t - \big(h(\hat{y}_t) + \frac{1}{2}\Delta[h](\hat{y}_t q_t)dt\big)\Big) \tag{2.17}$$

$$\begin{aligned} dQ_t =& Q_t\nabla[f]^T(\hat{y}_t)dt + \nabla[f](\hat{y}_t)Q_t dt + g(t)g^T(t)dt - Q_t\nabla[h]^T(\hat{y})R^{-1}\nabla[h](\hat{y}_t)Q_t dt \\ &+ \frac{1}{2}Q_t\Delta[h]^T(\hat{y}_t)R^{-1}(t)\big(dx_t - (h(\hat{y}_t)) + \frac{1}{2}\Delta[h](\hat{y}_t Q_t)dt\big)Q_t \end{aligned} \tag{2.18}$$

where $\delta[f]$ and $\delta[h]$ are second-order derivatives.

### 2.7 First Order Extended Ensemble Filter (FoEEF)

The First Order Extended Ensemble Filter (FoEEF) is based on an ensemble of weights for a neural network model [12]. The process involves:

1. Initializing ensemble weights $w$, filter object $\hat{w}_0$, and covariance matrix $P_0$.
2. Projecting state forward: $f(w_t^i)df + g(w_t^i)q^{\frac{1}{2}}(t)dw_t^{*i}$.
3. Computing Kalman gain: $ph_w(w_i^i)r^-1(t)$.
4. Updating state with measurement information and iterating over the ensemble to update weights $\hat{w}_t$ and covariance matrix $P_t$ [11].

### 2.8 Second Order Extended Ensemble Filter (SoEEF)

The Second Order Extended Ensemble Filter (SoEEF), developed by Kevin Midenyo in 2023 [9], improves on the First Order version by incorporating second-order Taylor expansions and Monte Carlo methods [5]. The filter equation is:

$$dx_t^i = f(x_t^i,\theta)dt + g(x_t^i)q^{\frac{1}{2}}(t)B_t + Mh_x(x_t^i)r^{-1}(t)(d(z_t) - (\frac{1}{2}M_t(h_{xx}(xx_t^i) - h(x_t^i))dt; \quad t_0 \leq t \tag{2.19}$$

Empirical approximations of mean $\hat{x}_t$ and covariance $M_t$ are given by:

$$\hat{x}_t = \frac{1}{N}\sum_{i=1}^{N} x_t^i; \quad M_t = \frac{1}{N-1}\sum_{i=1}^{N}(x_t^i - \hat{x}_t)^2 \tag{2.20}$$

This method was validated using the Lorenz 63 system, showing improved performance with increased ensemble size. Numerous Filters have been discussed and it was established that Kalman Bucy filters were very accurate in their applications in non-linear dynamic systems, however the cost of computing of the high dimensional covariance is very high [11]. The first-order extended ensemble filter [12, 11] proved to be a better filter as it solved the computational costs by estimating the expected covariances empirically . However, it applied Taylor series to the first order thus slightly less accurate as compared to filters where the estimation of expected covariance id done to second order. KevIn Midenyo developed a new filter, second order extended ensemble filter and in application it proved better than FoEEF as the expected values of the covariances was approximated by Taylor series to the second order.This thesis is thus building on the past work by expanding Kushner Stratonovich equation to the second order around the mean and stochastically perturbing the innovation to take care of any other uncertainties by developing second order extended ensemble filter with stochastically perturbed innovation. The new filter performs better than SoEEF and FoEEF and it is computationally cheaper than SoEKBF and other filter.

# 3 Development of Filter

## 3.1 Stochastic calculus

There is a lot of underlying mathematics when developing filters. The method of derivation involves dynamic state-space model hence a need for a deeper discussion on stochastic calculus. This section gives an introduction , step by step derivation , explanation of stochastic calculus needed to derive Second Order Extended Ensemble Kalman Filter with stochastically perturbed Innovation.

Assuming that a state of a system is according to

$$dy_t = f(y_t, t)dt + g(y_t, t)dB_t;\ t_0 \leq t$$

and a noisy measurements of the system given by

$$dz = h(y_t, t)dt + \theta dv;\ t_0 \leq t$$

Where $f$ is the drift function, $g$ refers to the volatility , $B$ and $v$ are the independent Brownian motion at time $t$. Then the conditional probability density $p(y,t)$ of the state at time $t$ is given by the Kushner-Stratonovich equation;

$$dp(y,t) = M[p(y,t)]dt + p(y,t)\big(h(y,t) - E_t(y,t)\big)\theta^{-T}\theta^{-T}\big(dz - E_t h(y,t)dt\big) \tag{3.1}$$

where

$$M[p] = -\sum \frac{\partial(f_i p)}{\partial y_i} + \frac{1}{2}\sum(gg^T)_{ij}\frac{\partial^2 p}{\partial y_i \partial y_j}$$

is the foward Kolmogorv operation and $dp(x,t) = p(x,t+dt) - p(x,t)$ gives the variation of conditional probability and the innovation is given by $dz - E_t h(x,t)$, that is the difference between the measurement and the expected value. Throughout this paper, because the derivation of Second Order Extended Ensemble Filter with stochastically perturbed innovation involves dynamic processes, stochastic calculus will be utilized in solving differential and integral equations to help come up with the filter [2].

### 3.2 Itô formula

The Taylor expansion of the KS model is given by

$$\begin{aligned} h(T_{t+\delta t}, t+\delta t) &= h(y_t, t) + \partial(y_t, t)]\delta_t + \partial_{xy}[h(y_t, t)]\delta y_t \\ &+ \frac{1}{2}(\partial_{tt}[h(y_t, t)]\delta t^2 + 2\partial_{tt}[h(y_t, t)]\delta_t \delta y_t \\ &+ \partial_{yy}[h(y_t, t)]\delta y_t^2) + \cdots \end{aligned}$$

In the above equation, values with powers greater than 2 are insignificant hence ignored;

$$\delta y_t = f(y_t, t)\delta t + g(y_t, t)\delta B_t \tag{3.2}$$

Then,

$$\delta h(y_t, t) = \partial(y_t, t)]\delta t + \partial y[h(y_t, t)]\delta y_t + \frac{1}{2}\partial_{yy}[h(y_t, t)]g^2(y_t, t)\delta_t + 0\delta t \tag{3.3}$$

The derivation of equation 3.3 is borrowed from the work of [1, 6].

From the above equation, terms containing orders greater than 2 are assigned zero because they are negligible. Substituting equation 3.2 and 3.3 and assuming the higher order terms are negligible, we will get:

$$\begin{aligned} \delta h(y_t, t) &= \partial_t[h(y_t, t)]\delta_t + \partial_y[h(y_t, t)](f(y_t, t)\delta_t) + g(y_t, t)\delta B_t) + \frac{1}{2}\partial_{yy}[h(y_t, t)]g^2(y_t, t)\delta t. \\ &= \partial_t[h(y_t, t)]\delta_t + \partial_y[h(y_t, t)](f(y_t, t)\delta_t) + \frac{1}{2}\partial_{yy}[h(y_t, t)]g^2(y_t, t)\delta t \\ &+ \partial_y[h(y_t, t)]g(y_t, t)\delta_t B_t \end{aligned} \tag{3.4}$$

when we extend the equation 3.4 to vector form through substituting n dimensional column vector, $y_t$, in the place of scalar $y_t$ and by use of differential operators we get;

$$\nabla = (\frac{\partial}{\partial y_1}\frac{\partial}{\partial y_2}\vdots\frac{\partial}{\partial y_n})$$

and

$$\begin{bmatrix} \frac{\partial^2}{\partial {y_1}^2} & \frac{\partial^2}{\partial y_1 \partial y_2} & \cdots & \frac{\partial^2}{\partial y_1 \partial y_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2}{\partial y_n \partial y_1} & \frac{\partial^2}{\partial y_n \partial y_2} & \cdots & \frac{\partial^2}{\partial {y_n}^2} \end{bmatrix}$$

Then equation 3.2 will yield

$$\delta h = \partial_t[h]\delta t + \nabla[h]^T f(y_t, t)\delta t + \frac{1}{2}tr[g(y_t, t)g^T(y_t, t)]\Delta[h]\delta t + (\nabla[h]^T g(y_t, t))\delta_t B_t \tag{3.5}$$

Taking the limit in the mean of $\delta h$ as $\delta t$ and $\delta t$ tending to zero, we get:

$$\delta h = \partial_t[h]dt + \nabla[h]^T dy_t + \frac{1}{2}tr[g(y_t, t)g^T(y_t, t)]\Delta[h]dt \tag{3.6}$$

equation 3.4 is referred to as Itô formulae. This formula is significant in the calculation of derivatives of stochastic calculus [7]. In fact,it is equivalent to chain rule in deterministic calculus [7].

### 3.3 Fokker-Plank equation

By considering the scalar of the Itô stochastic differential equation 3.4. It can be shown that the process for $\{y_t, t \in [t_o, I]\}$ generates equation 1 is a molecular process.
The density function of the Marker process is given by

$$\Phi_t(y) = \Phi(y,t), \forall\, t \in [t_o, T] \tag{3.7}$$

together with the transition probability density function we have;

$$\Phi_{t|T}(y/z) = \Phi(y,t)(Z,T), \forall\, T < t \in [t_o, T] \tag{3.8}$$

Fokker plank is an equation of the evolution of the density function, $\Phi_t(y)$ and the conditional density, $\Phi_{t|T}(y/Z)\, \forall\, t > T \in [t_o, T]$. The Process $\{y_t, t > 0\}$ that is generated by equation 3.4 is a Markov process, then we can say that, given $t_1 < t_2 < t_3$;

$$\Phi_{t_3|t_1,t_2}(y|z,x) = \Phi_{t_3|t_2}(y/x) \tag{3.9}$$

and the following equation satisfies;

$$\Phi_{t_3|t_1}(y/z) = \int \Phi_{t_3|t_1}(y/x)\Phi_{t_2|t_1}(x/z)dx \tag{3.10}$$

equation 3.10 above is the Chapman-Kolmogor equation and is applying to all Markov processes. We will then use the equation 3.10 and Taylor expansion to obtain the equation of evolution of transition probability density function, $\Phi_{t|T}(y/x) = \Phi(y,t|x,T)$, and is given by

$$\frac{\partial \Phi_{t|T}(y/x)}{\partial t} = \frac{-\partial(\Phi_{t|T}(y/x)f(y,t))}{\partial y} + \frac{\frac{1}{2}\partial^2(\Phi_{t|T}(y/x)g^2(y_t,t))}{\partial y^2} \tag{3.11}$$

Equation 3.11 is the Fokker plank equation and also referred to as Kolmogorov's forward equation.
Then we take expectations to equation 3.11 with respect to $\Phi_t(x)$;

$$E_T[\Phi_{t|T}(y/x)] = \int \Phi_{t|T}(y/x)\Phi_T(x)dx = \Phi_t(y) \tag{3.12}$$

The equation of each the evolution of probability density function, $\Phi_t(y)$ is thus;

$$\frac{\partial \Phi_t(y)}{\partial t} = \frac{-\partial(\Phi_t(y/x)f(y,t))}{\partial y} + \frac{\frac{1}{2}\partial^2(\Phi_t(y_t)g^2(y,t))}{\partial y^2} \tag{3.13}$$

By use of the vector equation 2, the corresponding Fokker plank is got by the evolution of every element in equation 3.13.

$$\frac{\partial \Phi_t(y,t)}{\partial t} = -\sum_{k-1}^{n} \frac{\partial \Phi_t(y,t) f_k(y,t)}{\partial y_k} + \frac{1}{2}\sum_{k,1=1}^{n} \frac{\partial^2(\Phi_t(y,t)(g(y,t)g^T(y,t))_{k1})}{\partial y_k \partial y_i} = \ell \Phi_t(y,t) \tag{3.14}$$

we therefore have;

$$\ell\partial\Phi_t(y,t) = -\sum_{k-1}^{n}\frac{\partial\Phi_t(y,t)f_k(y,t)}{\partial y_k} + \frac{1}{2}\sum_{k,1=1}^{n}\frac{\partial^2(\Phi_t(y,t)(g(y,t)g^T(y,t))_{k1})}{\partial y_k\partial x_i} \tag{3.15}$$

The solution of the equation 3.14 gives the probability density function $\Phi_t(y)$. It should be noted that most SDE for nonlinear drafts are cumbersome to solve hence numerical solutions are preferred. We then derive a form of Fokker- plank equation forming the basis of numerical methods.

## 3.4 Weak form of Fokker plank equation

Considering the scalar Fokker Plank equation 3.13. Let $\Theta(y) \in_c^\infty (\mathbb{R},\mathbb{R})$ meaning $\Theta(y)$ is differentiable function to infinity from $\mathbb{R}$ to $\mathbb{R}$, and the compact support is set $\mathbb{R}$. The value of

$$\Phi_{to}(y) = \Phi_o(y) \tag{3.16}$$

Multiplying equation 3.13 by $P(x)$ and intergrity over domain;

$$\int\frac{\partial\Phi_t(y)}{\partial t} = -\int\frac{\partial\Phi_t(y)f(y,t)}{\partial y}P(y)dy + \frac{1}{2}\int\frac{\partial^2(\Phi_t(y)g^2(y,t))}{\partial y^2}P(y)dy \tag{3.17}$$

when we intergrate 3.17 by parts we obtain

$$\int\frac{\partial\Phi_t(y)}{\partial t}P(y)dy = -\int\partial\Phi_t(y)f(y,t)\frac{\partial P(y)}{\partial y}dy + \frac{1}{2}\int\frac{\partial(\Phi_t(y)g^2(y,t))}{\partial y^2}\frac{\partial P(y)}{\partial y}dy \tag{3.18}$$

$$= \int\frac{\partial P(y)}{\partial y}(\Phi_t(y)f(y,t) - \frac{1}{2}\frac{\partial(\Phi_t(y)g^2(y,t))}{\partial y})dy \tag{3.19}$$

By intergrating the R.H.S of equation 3.19, we have;

$$\int\frac{\partial\Phi_t(y)}{\partial t}P(y)dy = \int\frac{\partial P(y)}{\partial y}\Phi_t(y)f(y,t)dy + \frac{1}{2}\int\frac{\partial^2 P(y)}{\partial y^2}\Phi_t(y)g^2(y,t)dy \tag{3.20}$$

Getting expected values of 3.20,

$$\frac{\partial/E_t[P(y)]}{\partial t} = /E_t[\frac{\partial P(y)}{\partial y}f(y,t)] + \frac{1}{2}/E_t[\frac{\partial^2 P(y)}{\partial y^2}g^2(y,t)] \tag{3.21}$$

Then defining $/E_t[P]$ as

$$/E_t[P] = \Phi[P] \tag{3.22}$$

$$= \int\Phi_t(y)P(y)dy \tag{3.23}$$

we have,
$d\Phi_t[P] = \Phi_t[\ell^* P]dt$ but the operator

$$\ell^* = f\frac{\partial}{\partial y} + \frac{1}{2}g^2\frac{\partial^2}{\partial y^2} \tag{3.24}$$

The above eqaution 3.24 is the weck form Fokker plank equation. We can now obtain different numerical methods based on choice of the test function $P(y)$

## 3.5 Equation of Evolution of the Mean

Given the equation of the mean is

$$d\Psi_t(\hat{x}_t) = \hat{f}dt$$

Let;
$\hat{x}_t = \Psi_t(x) = \int x\Psi_t(x)dx$ and
$\hat{f}_t = \Psi_t[f(x,t)] = \int f(x,t)\Psi_t(x)dx$ then

$$a_t = \Psi_t[x_t^2] - (\hat{x}_t)^2 \, and \, da_t = d\Psi_t[x_t^2] - d(\hat{x})^2$$

using its formulae:

$$d\Psi_t(x_t^2) = 2\Psi_t[fx_t]d\hat{x} + \Psi[q^2]$$

hence

$$da_t = 2\Psi_t[fx_t]d\hat{x} + \Psi[q^2] + 2\hat{x}_t\hat{f}_t dt \tag{3.25}$$

## 3.6 Kushner stratonovich

We can describe Kushner-stratonovich equation as the purtabation of Fokker Plank equation through the addition of the knowledge from measurement by use of Bayesian approach.
The equation therefore estimate $\hat{y}_t$ at time $t$ through combination of the noisy dynamic with the noisy measurement.

$$dP(y,t) = L[P(y,t)]dt + P(y,t)[h()g, t - E_t h(y,t)]^T \tag{3.26}$$

$$\eta\eta[d2 - E_t h(y,t)dt] \tag{3.27}$$

where

$$\ell P = -\sum \frac{\sigma\partial(\delta_i, P)}{\partial y_i} + \frac{1}{2}\sum(\delta\delta^T)_{ij}\frac{\partial^2 P}{\partial y_i \partial y}$$

which is Kolmogorov formed operation and the

$$dP(y,t) = P(y, t+dt) - P(y,t)$$

By apply as

$$q(\delta_t, \delta y_r) = 1 + (h-h)^T R^{-1}(\delta y_t - \hat{h}_i\delta t) + \cdots$$

## 3.7 Evolution of the Mean and Covariance

Mean refers to the first moment whereas covariance refers to the second moment.

$$d\hat{y}_t = \hat{f}dt + (\widehat{y_t h^T} - \hat{y}_t\hat{h}^T)R_{(t)}^{-1}(dz_t - \hat{h}d(t)); \;\; y_{t_o} = y(0) \tag{3.28}$$

$$\begin{aligned}(dM_t)_{ij} &= (\widehat{y_i f_i} - \hat{y}_i \hat{f}_i)dt + (\widehat{f_i y_j} - \hat{f}_i \hat{y}_j)dt + (\widehat{GQG^T})_{ij}dt - \\ &(\widehat{y_i h} - \hat{y}_i \hat{h})^T R^{-1}(\widehat{h y_j} - \hat{h} y_j - \hat{h}\hat{j})dt + (\widehat{y_i y_j h} - \\ &\widehat{y_i y_j}\hat{h} - \hat{y}_i \widehat{y_j h} - \hat{y}_j \widehat{y_i h} + z\hat{y}_i \hat{y}_j \hat{h})^T R^{-1}(t)(dz_t - \hat{h}dt); \quad M_{t_0} = M(0)\end{aligned} \tag{3.29}$$

The estimate $\hat{y}$ is solution to the equation 3.28 and $M_t$ is the solution of equation 3.29. The two equations give exact equation of the evolution of mean and covariance. To simplify the expressions, we can use scalar and write as;

$$signal: \quad dy_t = f(y_t, \theta)dt + g(y_t)q^{\frac{1}{2}}dB_t; \quad y_{t_0} = y(0), t_0 \leq t \tag{3.30}$$

$$measurement: \quad dz_t = h(y_t)dt + r^{\frac{1}{2}}d\eta_t; \quad t_0 \leq t \tag{3.31}$$

The equations for evolution of the conditional mean and variance for scalar are given by;

$$d\hat{y}_t = \hat{f}dt + (\widehat{y_t h} - \hat{y}_t \hat{h})r^{-1}(t)(dz_t - \hat{h}dt); \quad y_{t_0} = y(0) \tag{3.32}$$

$$\begin{aligned}dM_t &= 2(\widehat{yf} - \hat{y}\hat{f})dt - \Big(\widehat{qg^2}dt - (\widehat{yh} - \hat{y}\hat{h})^2 r^{-1} \\ &\quad + (\widehat{y^2 h} - \hat{y}^2\hat{h} - 2\hat{y}y\hat{h} + 2\hat{y}^2\widehat{yh} + 2\hat{y}^2\hat{h})r^{-1}(dz_t - \hat{h}dt)\Big); \\ dM_0 &= M(0)\end{aligned} \tag{3.33}$$

where,

$$\hat{f} = \int f(y)M(y|Z_t)dy_t \tag{3.34}$$

solving 3.30 and 3.31 will give exact filter. Here the solution involves the calculation of conditional expected values which is difficult as they involve integration over non-linear functions. We then resort to approximating the expected values. The second order approximation of exact filter is made by neglecting the third and higher order

### 3.7.1 Second-Order Approximate Filter

### 3.7.2 Theorem 1: Second-Order Approximate Filter

Given continuous functions $f(y)$ and $h(y)$, with existing first and second derivatives $f_y, f_{yy}, h_y, h_{yy}$, the second-order approximation of the exact filter neglecting higher-order terms is:

$$d\hat{y}_t = f(\hat{y})dt + \frac{1}{2}M_t f_{yy}(\hat{y}dt + Mh_y(\hat{y})r^{-1}(t)\big(dz_t - (h\hat{y}) + \frac{1}{2}M_t h_{yy}(\hat{y})\big) \tag{3.35}$$

$$\begin{aligned}dM_t &= 2Mf_y(\hat{y})dt + \big(q(t)g^2(\hat{y}) + Md(t)g_y^2(\hat{y})\big)dt - \big(M_t h_y(\hat{y})\big)^2 r^{-1} \\ &\quad + \frac{1}{2}M_t^2 h_{yy}(\hat{y})r^{-1}(t)\big(dz_t - (h(\hat{y}) + \frac{1}{2}M_t h_{yy}(\hat{y}))dt\big)\end{aligned} \tag{3.36}$$

**Proof:** By applying a Taylor expansion for $f(y)$ and $h(y)$ around $\hat{y}$, and taking expectations up to the second order, we have:

$$f(y) \approx f(\hat{y}) + (y-\hat{y})f_y(\hat{y}) + \frac{1}{2}(y-\hat{y})^2 f_{yy}(\hat{y}) \tag{3.37}$$

$$h(y) \approx h(\hat{y}) + (y-\hat{y})h_y + \frac{1}{2}(y-\hat{y})^2 h_{yy}(\hat{y}) \tag{3.38}$$

The approximations for the remaining terms can be similarly derived using Taylor expansions. Taking expectations and using covariance terms leads to the final second-order equations.

## 3.8 First-Order Approximate Filter

### 3.8.1 Theorem 2: First-Order Approximate Filter

For continuous functions $f(y)$ and $h(y)$, with existing first derivatives $f_y$ and $h_y$, the first-order approximation of the exact filter is:

$$dy_t = f(\hat{y})dt + Mh_y(\hat{y})r^{-1}(t)(dz_t) - h(\hat{y})dt \tag{3.39}$$

$$dM_t = 2Mf_y(\hat{y})dt + \Big(q(t)g^2(\hat{y}) + Mq(t)g_y^2(\hat{y})\Big)dt - \Big(M_t h_y(\hat{y})\Big)^2 r^{-1} \tag{3.40}$$

**Proof**

This is derived from the second-order filter by setting the second-order derivatives $f_{yy}$ and $h_{yy}$ to zero, leading to first-order terms only.

## 3.9 Second-Order Extended Ensemble Filter (soEEF)

The second-order extended ensemble filter (soEEF) is developed based on the nonlinear model with second-order approximation:

$$dy_t^i = f(y_t^i,\theta)dt + g(y_t^i)q^{\frac{1}{2}}(t)B_t + M_t h_y(y_t^i)r^{-1}(t)(d(z_t) - (\frac{1}{2}M_t(h_{yy}(y_t^i) - h(y_t^i))dt; \quad t_0 \le t \tag{3.41}$$

the empirical mean $\hat{y}_t$ and covariance $M_t$ are estimated as:

$$\hat{y}_t = \frac{1}{N}\sum_{i=1}^{N} y_t^i; \quad t_o \le t \tag{3.42}$$

$$M_t = \frac{1}{N-1}\sum_{i=1}^{N}(y_t^i - \hat{y}_t)^2; \quad t_0 \le t \tag{3.43}$$

### 3.10 SoEEF with Stochastically Perturbed Innovation, S0EEFPI

In this section, you introduce the SoEEFPI (Second Order Extended Esemble Filter with Stochastically Perturbed Innovation) model, derived by adding a stochastic component to the deterministic perturbation term of SOEEF. This results in the following McKean-Vlasov Stochastic Differential Equation (SDE) for the system's state evolution, using mean-field theory (MFT) [13] as an approximation tool. In MFT, all interactions are replaced by an effective interaction with the mean field.

The key stochastic differential equation is:

$$d(\bar{y}_t) = f(\bar{y}_t)dt + g(\bar{y}_t)q^{\frac{1}{2}}(t)d\beta_t + Mh_y(\bar{y}_t)r^{-1}(t)\Big[dz_t - 2h(\bar{y}_t)dt + R^{\frac{1}{2}}(t)d\bar{u}_{(t)}\Big] \tag{3.44}$$

This describes the evolution of the system's state with stochastic innovations. In the finite system, where $M$ independent copies of the state variables interact, the equation becomes:

$$d(y^i_t) = f(y^i_t)dt + g(y^i_t)q^{\frac{1}{2}}(t)d\beta^i_t + Mh_y(y^i_t)r^{-1}(t)\Big[dz_t + -2h({y^i}_t)dt + R^{\frac{1}{2}}(t)d{u^i}_{(t)}\Big] \tag{3.45}$$

This captures the stochastic interaction hypothesis for each state in the system.

The SoEEF evolution equation in its controlled Stratonovich form is then:

$$dy_t = f(y_t)dt + g(y_t)q^{\frac{1}{2}}(t)d\beta_t + Mh_y(y_t)r^{-1}(t)\big[dz_t + R^{\frac{1}{2}}(t)du_{(t)} - 2h(y_t)dt\big] \tag{3.46}$$

This equation is eventually transformed into the Itô form, and the corresponding Fokker-Planck equation describes the evolution of the filtering density $\pi(y_t|Z_t)$.

### 3.11 Exactness of SoEEF with Stochastically Perturbed Innovation

We begin with filtering posterior

$$\pi_{t_0}(y \mid Z_0) = \Pi^*_{t_0}(y \mid Z_0) \tag{3.47}$$

$\pi^*_{t_0}(y \mid Z_0)$ is true posterior at initial time $t_0$ and filter posterior matches time posterior at all times, t and hence we prove that filtering in equation (3.8.5) is exact. It makes sense to show that the equations of evolution of true posterior and the filtering are the same. By multiplying equaton (3.9.0) by $-\pi_t$ we get;

$$-\pi_t N = \pi_t Mh - 2\pi_t q \tag{3.48}$$

$$-\pi_t M(h - \hat{h}_t) - \pi_t M\hat{h}_t + 2\pi_t q \tag{3.49}$$

by introducing $\pi_t M\hat{h}_t - \pi_t\alpha\hat{h}_t$ from equation (3.8.9) we get,

$$-\pi(h - \hat{h}_t) = R(t)\nabla\cdot(\pi_t\alpha_k) \tag{3.50}$$

substituting equation (3.9.3) to equation (3.9.2) we get;

$$-\pi_t N = MR(t)\nabla\cdot(\pi_t M) - \pi_t M\hat{h}_t + 2\pi_t q \tag{3.51}$$

But

$$\nabla \cdot \big(\pi[MRM^T]big) = \pi MR\nabla \cdot (M) + MR\nabla \cdot (\pi M) \tag{3.52}$$

and applying in 3.9.4 and noting that

$$q(y,t) = \frac{R}{2}\sum_{k=1}^{n} M_k(y_t)\frac{\partial k_j}{\partial y_t} \tag{3.53}$$

we get

$$-\Pi N = -\nabla\big(\pi[MR\alpha^T]\big) + \Pi M\hat{h}_t \tag{3.54}$$

Taking divergence on both sides of equation 3.9.5 we get;

$$-\nabla \cdot (\pi N) = -\sum_{i,k=1}^{n} \frac{\partial^L}{\partial_{y_i}\partial y_k}\big(\pi[MRM^T]_{IK}\big) + \nabla \cdot (\pi M\hat{h}_t) \tag{3.55}$$

we substitute equation (3.8.6) and equation (3.9.6) for;

$$-\nabla \cdot (\pi_t N) + \sum_{i,k=1}^{n} \frac{\partial^L}{\partial_{y_i}\partial_{y_t}}\big(\pi[MRM^T]_i k\big) \tag{3.56}$$

and $\nabla cdot(\pi, M)$ in equation 3.8.7 to get

$$d\pi_t = \mathcal{L}(\pi_t)dt + (h - \hat{h}_t)^T R^{-1}(t)(dy_t - \hat{h}_t dt) \tag{3.57}$$

Thus proving exactness.

# 4 Filter Validation

The Lorenz 63 system, first introduced by Edward Lorenz [4], is a set of three ordinary differential equations that exhibit chaotic behaviour for specific parameter values and initial conditions [14]. The system is given by:

$$\frac{dx_1}{dt} = a(x_2 - x_1) \quad \frac{dx_2}{dt} = x_1(b - x_3) - x_2 \quad \frac{dx_3}{dt} = x_1x_2 - cx_3 \tag{4.1}$$

where $a = 10$, $b = 28$ and $c = \frac{8}{3}$. These parameter values determine the system's chaotic behavior, which is highly sensitive to initial conditions. Small changes in these conditions can result in significantly different trajectories over time, a hallmark of chaos.

Lorenz 63 is widely used as a testbed for data assimilation because it exemplifies how small initial perturbations lead to large changes in outcomes, making it an ideal model for testing filters [4]. In this case, a new filter, the Stochastically Perturbed SoEEKF, was applied to the Lorenz 63 model and compared with several other filters, including:

1. First Order Extended Ensemble Kalman Filter with perturbed Innovation (FoEEKPi)
2. First Order Extended Kalman-Bucy Filter (FoKBF)
3. Second Order Extended Kalman-Bucy Filter (SoEKBF)
4. The Bootstrap Particle Filter (KBF)
5. First Order Extended Ensemble Filter (FoEEF)
6. Second Order Extended Ensemble Filter (SoEEF)

## 4.1 Stochastic Lorenz 63 Model

The stochastic version of the Lorenz 63 model incorporates randomness through Brownian motion and is given by:

$$dy_t = f(y_t, \theta)dt + G(y_t)Q^{\frac{1}{2}}(t)dB_t; \quad x_{t_0} = x(0), t_0 \leq t$$

where $f(y)$ represents the Lorenz 63 dynamics, $B_t$ is a 3-dimensional standard Brownian motion, and $G$ and $Q$ represent the noise covariance. The measurement equation associated with this model is:

$$d\bar{x}_t = h(y_t)dt + R^{\frac{1}{2}}(t)d\eta_t; t_0 \leq t \tag{4.2}$$

where $h(y)$ describes the observation process, and $\eta_t$ is also a 3-dimensional Brownian motion.

## 4.2 Simulation Setup

The simulation of the Lorenz 63 system used the following setup:

1. Time increment: $dt = 0.001$
2. Simulation time: $T = 0$ to $T = 30$
3. Initial condition: $y_{t_0} = [-5.91652, -5.52332, 24.5723]^T$
4. The system ran for $30,000$ iterations with nine ensemble sizes: $10, 15, 22, 26, 29, 34, 41, 46, 49$

## 4.3 Perturbed SoEEKF

The stochastic SoEEKF (Second Order Extended Ensemble Kalman Filter) perturbed version used for the experiment is described by:

$$dy_t = f(y_t)dt + g(y_t)q^{\frac{1}{2}}(t)d\beta_t + Mh_y(y_t)r^{-1}(t)\left[dz_t + R^{\frac{1}{2}}(t)du_{(t)} - 2h(y_t)dt\right] \tag{4.3}$$

## 4.4 Performance Evaluation

The performance of the newly developed SoEEKF filter was tested using the Lorenz 63 system and compared with several other filters. These include: Bootstrap Particle Filter (BPF) First Order Extended Ensemble Kalman Filter (FoEEF), First Order Extended, Kalman-Bucy Filter (FoEKBF), Second Order Extended Kalman-Bucy Filter (SoEKBF), Second Order Extended, Ensemble Kalman Filter (SoEEF)

By evaluating the accuracy and robustness of these filters on the Lorenz 63 system, the new filter's performance was validated. The chaotic nature of the Lorenz 63 system made it an ideal candidate to test the filter's ability to handle sensitivity to initial conditions, noise, and nonlinearity.

# 5 Results

In Figure 1, the Root Mean Square Error (RMSE) is plotted on the y-axis against the reciprocal of the ensemble size $M$ on the x-axis. The graph compares the performance of six different filters: Bootstrap Particle Filter (BPF): represented by a thin blue line, First Order Extended Ensemble Filter (FoEEF): represented by a green line, First Order Extended Kalman-Bucy Filter (FoEKBF): represented by an orange line, Second Order Extended Kalman-Bucy Filter (SoEKBF): represented by a purple line, Second Order Extended Ensemble Kalman Filter (SoEEF): represented by a sky blue line and Stochastically Perturbed SoEEF (SoEEFPI): represented by a maroon line

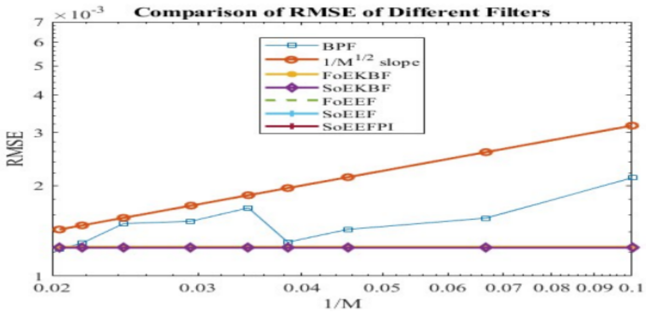


Figure 1: Root mean square error for the Reciprocal of Ensemble.
The sizes of the Ensamble used are $10, 15, 22, 26, 29, 34, 41, 46, 49$ Other settings; $dt = 0.001,\ R = 0.17,\ and\ G = 0.2000$.

From the graph, several observations can be made; As the ensemble size increases, the RMSE of the Bootstrap Particle Filter (BPF) rises, indicating that this filter may become less effective with larger ensembles.

In contrast, the RMSE for the other filters—FoEKBF, SoEKBF, FoEEF, SoEEF, and SoEEFPI—decreases as the ensemble size increases, suggesting improved performance with larger ensembles.

Among these filters, SoEKBF shows the lowest RMSE, reflecting its strong performance in reducing estimation error. However, it is important to note that the computation of the inverse of covariance matrices for SoEKBF is quite expensive, which raises concerns about its practicality for larger applications.

The Stochastically Perturbed SoEEF (SoEEFPI) consistently demonstrates lower RMSE compared to the other filters, making it a compelling option. Although SoEKBF has the best performance in terms of RMSE, its high computational cost underscores the need for an efficient alternative. SoEEFPI effectively balances performance and computational efficiency, proving to be a superior choice in this analysis.

Figure 2 effectively illustrates the performance of the SoEEFPI filter in comparison to traditional filters in the context of chaotic systems. The close alignment of SoEEFPI's trajectory with the true state, combined with its error performance, underscores its effectiveness as a reliable data assimilation method within the Lorenz 63 model framework. The findings emphasize the importance of selecting filters that not only provide accurate estimates but also maintain computational efficiency in chaotic environments.

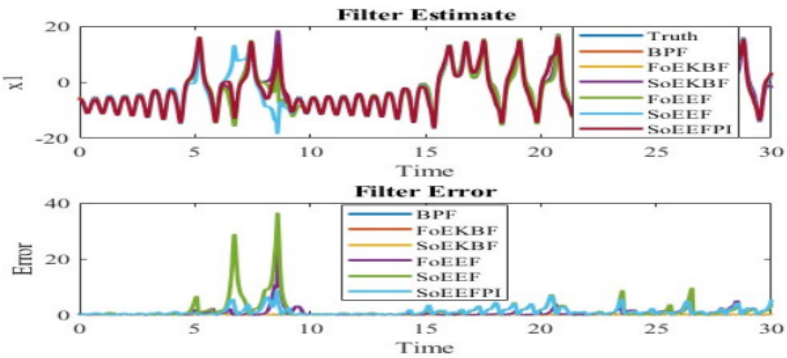


Figure 2: Filter Estimate and Filter Error for first variable, $x1$ in Lorenz 63 model 94

Figure 3 illustrates the comparison between the true trajectory of the second variable $x_2$ of the Lorenz 63 model and the estimates produced by six different filters. The plot reveals that there is no noticeable deviation between the output of SoEEFPI and the true trajectories of the Lorenz 63 system, indicating that SoEEFPI outperforms the other filters in accurately tracking the true state. This highlights the effectiveness of SoEEFPI in providing reliable estimates in chaotic systems like the Lorenz 63 model.

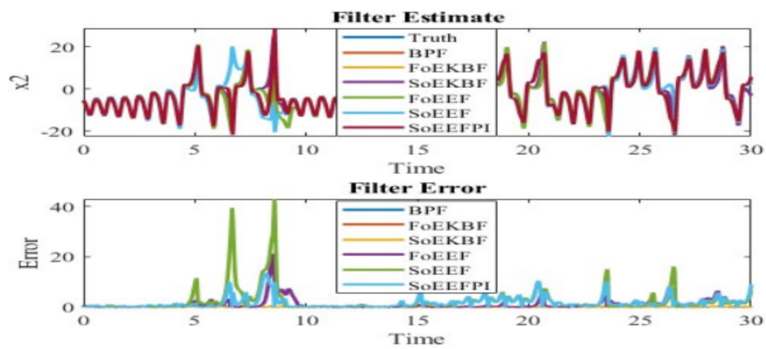


Figure 3: Filter Estimate and Filter Error for $x2$ in Lorenz 63 Model 94

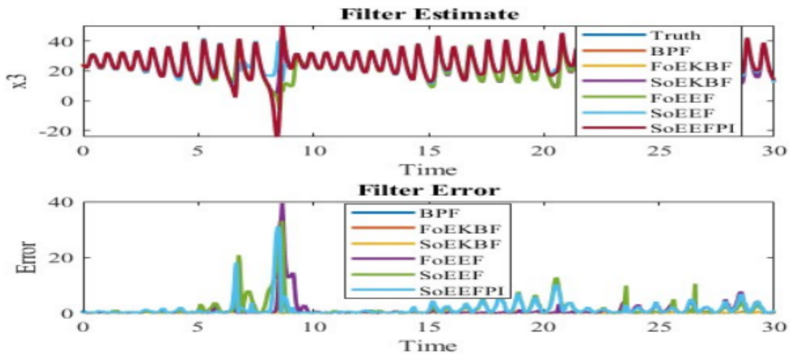


Figure 4: Filter Estimate and Filter Error for $x3$ in Lorenz 63 Model 94

Figure 4 consists of two graphs. The first graph displays the comparison of the trajectories of evolution for the new filter SoEEFPI alongside SoEEF, SoEKBF, FoEKBF, and BPF, against the true state generated by the third variable in the Lorenz 63 model. These trajectories are obtained using nine ensembles measured between times $T = 0$ and $T = 30$. The second graph presents the filter errors measured across the same time interval for all six filters. This comprehensive analysis provides insight into the performance and accuracy of each filtering approach in relation to the true state of the Lorenz 63 model.

## 6 Conclusion

The complexity of integrating the covariance of high-dimensional data and the associated computational costs have led to the emergence of ensemble Kalman filters. The plots presented compare six filters: First Order Extended Ensemble Filter (FoEEF), First Order Extended Kalman Bucy Filter (FoEKBF), Second Order Extended Kalman Bucy Filter (SoEKBF), and Second Order Extended Ensemble Kalman Filter (SoEEF). The experiment was conducted on a three-dimensional stochastic Lorenz 63 model, with simulations performed using MATLAB software. Results indicate that SoEEFPI outperformed the other filters (KBF, FoEEF, SoEEF, and FoEKBF) across all three variables of the Lorenz 63 model: $x_1$, $x_2$, and $x_3$. While SoEKBF exhibited the lowest root mean square error (RMSE), its computational cost is significantly higher due to the integration of high-dimensional covariance, making SoEEFPI a more desirable option since its covariance computation is performed empirically. The graphs further demonstrate that SoEEFPI is advantageous for parallel computing, as it does not require resampling, unlike particle filters. Additionally, the performance of SoEEFPI improves with increasing ensemble sizes.

## References


[1] Angwenyi, D. (2019). Time-continuous state and parameter estimation with application to hyperbolic SPDEs, (*Doctoral dissertation, Universität Potsdam*).

[2] Cohen, S. N., & Elliott, R. J. (2015). Stochastic calculus and applications (Vol. 2), *New York: Birkhäuser*.

[3] Evensen, Geir (2003) The Ensemble Kalman Filter: theoretical formulation and practical implementation, *Journal of Ocean Dynamics*, volume:53, Serial:1616-7228, https://doi.org/10.1007/s10236-003-0036-9

[4] Goodliff, M., Fletcher, S., Kliewer, A., Forsythe, J., & Jones, A. (2020). Detection of non-Gaussian behavior using machine learning techniques: a case study on the Lorenz 63 model, *Journal of Geophysical Research: Atmospheres*, 125(2), e2019JD031551.

[5] Hazazi, Muhammad Asaduddin and Sihabuddin, Agus(2019)Extended Kalman filter in recurrent neural network: USDIDR forecasting case study, *Indonesian Journal of Computing and Cybernetics Systems*, 13.

[6] Itô, K. (1944). 109. Stochastic Integral. *Proceedings of the Imperial Academy*, 20(8), 519-524.

[7] Lamberton, D., & Lapeyre, B. (2011). Introduction to stochastic calculus applied to finance. *Chapman and Hall/CRC*.

[8] Marron, M and Garcia, JC and Sotelo, MA and Cabello, M and Pizarro, D and Huerta, F and Cerro, J(2007)Comparing a Kalman filter and a particle filter in a multiple objects tracking application, *IEEE International Symposium on Intelligent Signal Processing.*

[9] Midenyo, K., Angwenyi, D., & Oganga, D. (2024). Second Order Extended Ensemble Filter for Non-linear Filtering.

[10] Neelima, M., & Prabha, I. S. (2024). Optimized deep network based spoof detection in automatic speaker verification system, *Multimedia Tools and Applications*, 83(5), 13073-13091.

[11] Oryiema, Robert and David, Angwenyi and Kevin Midenyo (2021)Extended Ensemble Filter for High-dimensional Nonlinear State Space Models, *Journal of Advances in Mathematics and Computer Science*, 36.

[12] Robert, Oryiema and Angwenyi, David and Achiles, Nyogesa and On'gala, Jacob[2022]Initialization and Estimation of Weights and Bias using Bayesian Technique, *Asian Journal of Probability and Statistics*, 17.

[13] Sen, N., & Caines, P. E. (2016). Nonlinear filtering theory for McKean–Vlasov type stochastic differential equations, *SIAM Journal on Control and Optimization*, 54(1), 153-174.

[14] Tandeo, P., Ailliot, P., Ruiz, J., Hannart, A., Chapron, B., Cuzol, A., ... & Fablet, R. (2015). Combining analog method and ensemble data assimilation: application to the Lorenz-63 chaotic system. In Machine Learning and Data Mining Approaches to Climate Science: proceedings of the 4th International Workshop on Climate Informatics , *Springer International Publishing*,3-12.

[15] Xia, Nan and Qiu, Tian-Shuang and Li, Jing-Chun and Li, Shu-Fang(2013)A nonlinear filtering algorithm combining the Kalman filter and the particle filter, *ACTA ELECTONICA SINICA*, 41.