A Comparative Study of the Particle Filter and the Ensemble Kalman Filter

by

Syamantak Datta Gupta

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2009

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Non-linear Bayesian estimation, or estimation of the state of a non-linear stochastic system from a set of indirect noisy measurements is a problem encountered in several fields of science. The particle filter and the ensemble Kalman filter are both used to get sub-optimal solutions of Bayesian inference problems, particularly for high-dimensional non-Gaussian and non-linear models. Both are essentially Monte Carlo techniques that compute their results using a set of estimated trajectories of the variable to be monitored. It has been shown that in a linear and Gaussian environment, solutions obtained from both these filters converge to the optimal solution obtained by the Kalman Filter. However, it is of interest to explore how the two filters compare to each other in basic methodology and construction, especially due to the similarity between them. In this work, we take up a specific problem of Bayesian inference in a restricted framework and compare analytically the results obtained from the particle filter and the ensemble Kalman filter. We show that for the chosen model, under certain assumptions, the two filters become methodologically analogous as the sample size goes to infinity.

Acknowledgements

First and foremost, I would like to thank sincerely my supervisor Professor Ravi R. Mazumdar for his support, encouragement and guidance. Working under his supervision has been a great learning opportunity for me. His encouragement has helped me learn a lot and has also induced in me a desire to learn more. Interactions with him has not only helped me expand my academic horizons, but has helped on a personal level as well. It has truly been s privilege to be his student.

I would also like to express my sincere gratitude to Professor Patrick Mitran, whose course, Statistical Signal Processing provided a great deal of background knowledge for my research.

Finally, I would like to thank my parents, who have encouraged and motivated me greatly and without whose love, care and support, it would have been impossible for me to be where I am today.

Contents

List of Tables **vii**

List of Figures **viii**

1 Introduction **1**

1.1 Background 1

1.2 The Optimum Solution 1

1.3 Sub-optimal Approaches 2

1.4 Monte Carlo Methods 3

1.5 Problem Description 4

2 Related Work **6**

2.1 Convergence Results for the Particle Filter 6

2.2 Convergence Results for the Ensemble Kalman Filter 8

2.3 Experimental Comparisons of the Two Methods 8

3 Analytical Approach to the Problem **10**

3.1 A Generalized Problem Formulation 10

3.2 The Analytical Approach 11

3.3 A More Specific Model 13

4 The Particle Filter **15**

4.1 Introduction 15

4.2 Description of the Algorithm 16

4.3 Choices of Proposal Distribution and Importance Weights 17

4.4 Parameters for the Specified Model 18

4.5 Resampling 19

5 The Ensemble Kalman Filter **22**

5.1 The Discrete Kalman Filter 23

5.2 The Ensemble Kalman Filter Algorithm 24

6 An Analytical Comparison of the Two Schemes **28**

6.1 Introduction 28

6.2 Convergence Results for the Ensemble Kalman Filter Estimates . . 29

6.3 A Relation between the Expectations of the Solutions of the Two Methods 34

6.4 A Relation between the Covariances of the Solutions of the Two Methods 36

6.5 General Remarks 38

7 Simulation Results **40**

8 Conclusion and Future Work **51**

APPENDIX **55**

References **60**

List of Tables

7.1 Average estimation errors for the particle filter and the ensemble Kalman filter for different sample sizes 50

List of Figures

4.1 Flow chart illustrating particle filter algorithm 21

5.1 Flow chart illustrating ensemble Kalman filter algorithm 27

7.1 $N_s = 5$, error versus time plot for particle filter 41
7.2 $N_s = 5$, error versus time plot for ensemble Kalman filter 42
7.3 $N_s = 20$, error versus time plot for particle filter 42
7.4 $N_s = 20$, error versus time plot for ensemble Kalman filter 43
7.5 $N_s = 50$, error versus time plot for particle filter 43
7.6 $N_s = 50$, error versus time plot for ensemble Kalman filter 44
7.7 $N_s = 100$, error versus time plot for particle filter 44
7.8 $N_s = 100$, error versus time plot for ensemble Kalman filter 45
7.9 $N_s = 5$, average error versus time plot for particle filter 46
7.10 $N_s = 5$, average error versus time plot for ensemble Kalman filter . 46
7.11 $N_s = 20$, average error versus time plot for particle filter 47
7.12 $N_s = 20$, average error versus time plot for ensemble Kalman filter . 47
7.13 $N_s = 50$, average error versus time plot for particle filter 48
7.14 $N_s = 50$, average error versus time plot for ensemble Kalman filter . 48
7.15 $N_s = 100$, average error versus time plot for particle filter 49
7.16 $N_s = 100$, average error versus time plot for ensemble Kalman filter 49

Chapter 1

Introduction

1.1 Background

Estimation of the state of a stochastic system from indirect noisy measurements is a problem encountered in several fields of science. These include a diverse class of problems in econometrics, biostatistics, geology and meteorology as well as many typical statistical signal processing problems such as target tracking, time series analysis, communications and satellite navigation. In all these problems, one essentially has the task of accurately estimating a certain set of variables that evolve over time, from a set of noisy measurements. Such a problem comes under the category of **Bayesian inference** problems, a sub-class of statistical inference where the likelihood of a hypothesis is updated sequentially in the light of observed data. Many such problems are formulated as discrete time hidden Markov models, where it is assumed that the present state of the system depends only on the state at the preceding instant. The evolution of the state variables and their mathematical relation with the observed data may be known or may be hypothesized based on experience. Either ways, the model connecting the observations and the variables of interest is in general a probabilistic one, due to the presence of noise and/or other uncertainties; and as such there is a need to determine a scheme that would lead to an optimum or sub-optimal solution.

1.2 The Optimum Solution

When the dynamics of this model are entirely linear and the noises involved are additive, following Gaussian distributions with known parameters, the optimum solution is given by the Kalman filter (Kalman [1960]). The **discrete Kalman filter** is a very robust and useful tool that has found its application in a wide variety

of problems encountered in various fields of science and technology, and is based on minimizing the estimation errors. However, a linear and Gaussian environment accounts for a very small subset of Bayesian inference problems; in most cases the system dynamics are non-linear and the noises non-Gaussian. In such situations analytical solutions are often intractable. Higher dimensionality of the system also adds to the complexity of the problem. Even for linear and Gaussian models, the Kalman filter may not be a feasible scheme to apply when the state dimensions are too high. For instance, if the state dimension is $N = 10^6$, then execution of the Kalman filter involves storage of an $N \times N$ matrix that would occupy a huge amount of memory.

1.3 Sub-optimal Approaches

Among these, the **extended Kalman filter** (Maskell and Gordon [2001]) can be utilised when the problem involves one or more non-linear function. Essentially, it linearizes the non-linear function locally at several regions using the first term of its Taylor series expansion. Some versions of this filter also use a few higher order terms of the expansion, but these are not used extensively for the obvious rise in computational complexity. The method assumes the noises to be Gaussian and uses the equations of the discrete Kalman filter to obtain the final estimate at each step of estimation. Because it assumes a Gaussian environment, this method would not work well when the distributions are significantly non-Gaussian. Moreover, it does not give good results under severe non-linearity, because then the local linearisations do not emulate well enough the original function.

While the extended Kalman filter attempts to emulate an optimal solution by linearizing the non-linear functions, the **approximate grid-based methods** attempt the same by discretising a continuous state space. In the latter, the continuous state domain is divided into a finite number of states around certain points within the domain and probability density functions involved in the estimation are reduced to probability mass functions. Prediction and update equations are formed using the conditional probabilities of each state with respect to the observations. For this model to approximate closely the actual dynamics of the state variable, in general, the discrete grid must be dense enough. It is intuitive that if the original space is known beforehand to be sparse, and the regions of high occurrence are known too, this method can be useful. However, in most cases one has no a priori knowledge of the distribution of the state space, and hence it is not possible to partition it unevenly by assigning greater resolution to the regions of greater likelihood. Another disadvantage of this method is the inevitable truncation of certain portions of the state space.

1.4 Monte Carlo Methods

The ensemble Kalman filter and the particle filter both are essentially Monte Carlo estimation methods and work on the following basic principle. First, a domain of possible input points is defined. For Bayesian estimation problems, this is equivalent to defining the a priori probability distribution. Next, a fixed number of sample points are generated from this domain or distribution. Using these sample points, finally, the required variables or parameters are estimated by performing a numerical integration. It is intuitively evident, therefore, that these methods rely on the law of large numbers, as they tend to replace integrations involving probability terms with deterministically computed sums and averages, effectively approximating probability with relative frequency of occurrence when the sample size is sufficiently large.

The fact that Monte Carlo methods such as the ones mentioned above can be used to solve complicated integrals numerically was known for a considerable time. However, the implementation of these methods for practical computational purposes was not feasible until recently. Over the recent years, thanks to advances in the fields of computing there has been an increased interest and popularity in these techniques. Monte Carlo methods are widely applied to a variety of problems in several fields of science. Many of these problems involve simulation of a physical system, many other involve prediction and estimation of unknown variables. The particle filter and the ensemble Kalman filter are both extensively used to get sub-optimal solutions of Bayesian inference problems, particularly in case of high-dimensional non-Gaussian and non-linear models.

The **particle filter** (Moral [1996]) is a recursive filtering method that generates multiple copies of the variable of interest from a sample population, associates a specific weight to each of these copies and then computes their weighted average to get the final estimate. Samples for the unknown states are drawn from an approximate distribution, and the optimal estimate is obtained by taking a weighted average of the samples, where the weights are assigned using the principle of importance sampling. This method has been called bootstrap filtering, sequential Monte Carlo method, the condensation algorithm, interacting particle approximation and survival of the fittest by different authors and researchers (Maskell and Gordon [2001]). The Monte Carlo characterisations tend to approach the original distribution as the sample size becomes sufficiently large, and the filter approaches the optimal Bayesian estimate.

Simply put, this technique draws samples for the state variable from a dummy distribution $q(\cdot)$ in lieu of the original distribution $p(\cdot)$ as sampling from the latter directly may not be feasible. The dummy distribution, also known as the proposal distribution $q(\cdot)$ is related to the original distribution in terms of importance

weights. Finally the required estimate is computed by combining the sample points drawn from $q(\cdot)$ with the corresponding importance weights. At each point, the state estimates are being developed simultaneously from each realisation of the sample along separate trajectories. The use of importance weights basically ensure that trajectories that more closely emulate the observations are assigned greater importances.

The **ensemble Kalman filter** (Evensen [1994]), on the other hand is an approximate extension of the discrete Kalman Filter used for non-linear Bayesian filtering. This method involves generating ensembles of model states and arriving at the final result using those samples and the observed measurements. An ensemble of forecast estimates is predicted here based on the estimates at the previous instant, and then those forecasts are tuned using an ensemble Kalman gain once the most recent observations arrive. It may be noted that the ensemble Kalman filter works best under Gaussian environments and do not give desired results when the probability distribution of the relevant variable deviates significantly from the Gaussian distribution.

It has been shown that under a linear and Gaussian environment, solutions obtained from both these filters converge to the optimal solution obtained by the Kalman Filter (Mandel et al. [2009], Butala et al. [2008], Saygin [2004]). Several convergence results for the particle filter under different conditions have also been derived and analyzed. Issues such as the asymptotic convergence of the filter solution to the optimal solution, error accumulation with time, convergence of the mean square error and convergence of the empirical distributions to the true ones have been addressed (Crisan and Doucet [2002]). Given the similar natures of the two filtering mechanisms, it would be an interesting problem to explore how they compare and relate to each other in basic methodology and construction.

1.5 Problem Description

In this work, we would take up a specific problem of Bayesian inference in a restricted framework. Namely, the noises involved are assumed to be zero-mean Gaussian with known covariance matrices, and the observations are linearly related to the states. For a model of this kind, an analytical comparison of the results obtained from the particle filter and the ensemble Kalman filter would be done. It would be shown that for the given model, the two methods closely resemble each other in their approaches, though they differ in their ultimate results.

More specifically, it would be shown that when the sample size is sufficiently large, the two filters essentially follow the same procedure to create estimate trajectories. However, even when the sample size is significantly large, the particle

filter, because it ascribes greater importance weights to trajectories that are more likely to produce the observed measurements, is, in general, more likely to give more accurate results.

The basic problem being considered is the determination or prediction of a parameter $\boldsymbol{\theta}$ that changes over some variable, usually time, using a series of noisy observations $\boldsymbol{x}$. The parameter of interest $\boldsymbol{\theta}$ is generally assumed to evolve over time following some known probabilistic model, depending on its previous state(s) and external disturbances. The observed variable $\boldsymbol{x}$ is some function of $\boldsymbol{\theta}$, contaminated with some noise.

Let us now enumerate the contents of this work. A discrete time formulation of the problem would be considered, where state evolution and availability of measurement both are assumed to occur at the same instants. Starting from a general description of the problem we would narrow our attention to a special case. We would then look at the development of a general Bayesian filtering approach, applicable to both linear and non-linear models; and show how such an analytical method would fail to provide a direct solution in many cases because of the integrals that it involve. Then, we would take up separately the two methods of interest in this study, discuss their working principles and formulate the structure of the solutions given by them. Here, we would attempt to solve a standard problem of Bayesian inference in case of a Gaussian system where the observations are linearly related to the parameter of interest but the state evolution dynamics of the parameter itself are non-linear.

We would then study how the two solutions relate to each other. We would establish, under certain assumptions, an analytical relation between the set of solutions provided by the two methods and show that as the ensemble size goes to infinity, the ensemble Kalman filter trajectories approximate the particle filter trajectories. The general similarity would also be demonstrated through simulation. Finally, we would briefly discuss the implications of this results and the future directions of work in this regard.

Chapter 2

Related Work

As seen in the previous chapter, except for a very limited scenario, it is not easy to estimate the optimum solutions for the states of a stochastic process directly from a set of noisy observations. Subsequently, several sub-optimal approximate methods have been derived over the years. The fact that solutions for high-dimensional non-linear state estimation problems can be derived by manipulating a large number of sample points was well known from a theoretical perspective, and with the recent advances in computing abilities, such methods have begun to gain popularity in a wide range of practical fields.

The particle filter and the ensemble Kalman filter are both sequential Monte Carlo methods that estimate the state variables using a large number of sampled data points. A detailed and rigorous mathematical formulation of the particle filter can be found in Moral [1996] where the technique is elaborately explained starting from the first principles, while Doucet et al. [2000], too, provides a comprehensive study. Ensemble Kalman filters, on the other hand were introduced as an approximation of the Kalman filter in Evensen [1994]. Several convergence results have been derived for the particle filter, and some for the ensemble Kalman filter. However, analytical comparisons between the two filters with an aim to establish a correlation between the two do not seem to have been abundant.

2.1 Convergence Results for the Particle Filter

A thorough discussion on the convergence results for the particle filter is available in Crisan and Doucet [2002]. After providing a detailed mathematical framework for the filter, this paper explores some results on almost sure convergence, convergence of the mean square error and a large deviations result. Specifically, it has been shown that under the assumption that the transition Markov kernel is Feller, and

that the likelihood function (this is the conditional probability density function of the observed variables given the state variables) is bounded, continuous and strictly positive, the empirical distribution obtained from the particle filter converges almost surely to the true distribution of the state. Further, it shows that convergence of the mean square error towards zero is guaranteed and it occurs with a rate in $1/N$ when a standard resampling scheme is used and the importance weights are upper bounded.

Another result presented in this work implies that a uniform convergence of the particle filter is ensured if the 'true' optimal filter is quickly mixing, while a considerable amount of error accumulation will prevent such a convergence when the optimal filter has a 'long memory'. This indicates that for Markov processes, one would expect a uniform convergence. Again, in a fairly recent work, Hu et al. [2008], it has been shown that the approximate solution given by the particle filter converges to the true optimal estimate, even when the function to be estimated is unbounded, as the sample size goes to infinity.

In a recent study (Bengtsson et al. [2008]) that analytically explores the performance of sequential Monte Carlo based methods, and the particle filter in particular, it has been shown that for models with high dimensions, the sequential importance sampling method tends to collapse to a single point mass within a few cycles of observation. The paper provides general conditions under which the maximum of the importance weights associated with the individual trajectories of the particle filter approaches unity, if the particle size is sub-exponential in the cube root of the system dimension. The convergence result is derived for a Gaussian setting, but it has been argued that the result would also hold for observation models with any other independent and identically distributed (iid) kernel. The study also asserts that even though methods such as resampling may be employed as a remedy to this degeneracy phenomenon for small scale models, they would not be able to eliminate the problem of slow convergence rates for systems with large dimensions.

In Saygin [2004] it has been shown that for the linear Gaussian state model, the optimal solution given by the interactive particle systems converges asymptotically to the real predictor conditional density given by the optimal Kalman Filter. The proof is first given for a uni-dimensional model and is then extended for the multi-dimensional case. Further, this work has explored how large the sample size is required to be for the filter to follow this asymptotic behaviour.

2.2 Convergence Results for the Ensemble Kalman Filter

Though the ensemble Kalman filter has not yet been as rigorously analyzed as its counterpart, some recent works have studied the nature of its convergence under certain restricted conditions. In Furrer and Bengtsson [2007], it has been mentioned that an asymptotic convergence of the ensemble Kalman filter can be shown using Slutsky's theorem, without providing a rigorous proof. Butala et al. [2008] provides a more convincing result by showing that as the size of the ensemble grows exponentially, the ensemble Kalman filter estimates converge in probability to LMMSE optimal estimates obtained by the Kalman Filter for a linear and Gaussian model. This work also provides a formal argument for the proposition that the ensemble Kalman filter is a Monte Carlo method that converges to a well defined limit.

In another recent work (Mandel et al. [2009]), it has been shown that the filter converges to the discrete Kalman filter as the ensemble size goes to infinity for a linear and Gaussian model and constant state space dimension. This work has first proved that the ensemble members are exchangeable random variables bounded in L^p and has then used this result, Slutsky's theorem and the Weak Law of Large Numbers to establish their final conclusion.

Since the ensemble Kalman filter approximates covariance terms by averaging over a large number of data points, it is expected to give better results as the ensemble size increases. This has been demonstrated in Gillijns et al. [2006], where simulation results show a steady fall in estimation errors as the ensemble size grows.

2.3 Experimental Comparisons of the Two Methods

Some studies have examined and compared the performances of the two filtering methods in a real experiment. In these experiments, in general, the particle filter has been seen to outperform its counterpart in terms of accuracy, because of its more sound mathematical foundations and also because of the fact that it makes lesser assumptions about the nature of the distribution to be estimated. The ensemble Kalman filter implicitly assumes the distribution to be Gaussian by relying only on the first two moments for estimation, and this makes it unreliable when the system dynamics are significantly non-Gaussian.

In Nakamura et al. [2009], the authors explore the merits of the two filters with respect to each other for data assimilation and tsunami simulation models. The models used are Gaussian and non-linear; and experimental results show that the

particle filter outperforms its rival quite convincingly in terms of accurate estimation.

In some of the practical applications though, a choice between the two filters involve a trade-off between accuracy and computational burden; and the ensemble Kalman filter may in fact be preferred because it is relatively simpler in formulation and also because it might involve less computational burden.

The work Weerts and El Serafy [2006], for instance, has taken up the task of comparing the performances of several non-linear filters (and in particular the two filters in our study), in the context of flood prediction, a key issue in hydrology. This paper suggests that since the particle filter utilises the full prior density, without making assumptions on the prior distribution of the model states, as opposed to the ensemble Kalman filter, it has a greater sensitivity towards the tails of the distribution and this makes it extremely advantageous in flood forecasting. However, this advantage is achieved at the cost of higher computational complexities. Also, it is said that the particle filter is more affected by error in measurement or system modeling, while the ensemble Kalman filter is more robust in that respect. It is thus stated that the ensemble Kalman filter appears to be more efficient than the particle filter for low flows, because of its relatively lesser susceptibility to uncertainties and misspecification of model parameters.

The theoretical studies imply that both the models would give optimal solutions under certain restricted conditions, as the sample size or ensemble size goes to infinity. The experimental studies indicate that even though the particle filter is theoretically superior, in some practical applications the ensemble Kalman filter may in fact be preferred to reduce computational burden. Since both the filters are used to solve similar estimation problems and also use similar techniques, it is of interest to explore how the schemes relate to each other from an analytical perspective. With this motivation, let us now proceed to describe the details of the actual problem that would be approached.

Chapter 3

Analytical Approach to the Problem

3.1 A Generalized Problem Formulation

Having briefly discussed the objectives of this study, let us now first describe the problem in details. Subsequently we would formulate the steps of a general solution. As mentioned before, the problem that we are concerned with is to estimate states of a hidden Markov chain from a set of noisy observations. For the ease of construction, a discrete time formulation is being considered, where the state evolution occurs at fixed instants, and these are the same instants when measured observations are made available. Clearly, since we have two sets of variables in this problem, of which one, viz., the state vector, evolves over time and the other, viz., the set of measurements, changes accordingly, we would require two mathematical models, or two sets of equations to describe the system. One of them would describe the evolution of the state with time while the other would relate the measured data with the present state. For the most general case, these two sets of equations would have the following forms.

The state evolution of the parameter of interest $\boldsymbol{\theta}_t$ would be described by the following hidden Markov model:

$$\boldsymbol{\theta}_{t+1} = f_t(\boldsymbol{\theta}_t, \boldsymbol{w}_t) \; \forall \; t \in \mathbb{N} \tag{3.1}$$

The observations would be related to the state variables as per the following equations:

$$\boldsymbol{x}_t = h_t(\boldsymbol{\theta}_t, \boldsymbol{v}_t) \; \forall \; t \in \mathbb{N} \tag{3.2}$$

where $f_t : \mathbb{R}^{N_\theta} \times \mathbb{R}^{N_w} \to \mathbb{R}^{N_\theta}$ and $h_t : \mathbb{R}^{N_\theta} \times \mathbb{R}^{N_v} \to \mathbb{R}^{N_x}$ are any functions, possibly non-linear.

3.2 The Analytical Approach

Let us now proceed to construct an analytical solution of this inference problem. Since both the state evolution and the observations are mixed with unknown disturbances in the form of random noise, and hence are probabilistic, it is useful to consider probabilistic models for estimation. The general framework of methods that solve such problems is typically based on the Bayesian approach. The aim is to determine the posterior probability density function (pdf) of the state using all the information available from past history of the state and the measured data (Maskell and Gordon [2001]). Given the transition equation (which, in our case is equation (3.1)) that describes the evolution of a hidden Markov process $\{\boldsymbol{\theta}_t; t \in \mathbb{N}\}$, an observation equation (which, in this case is equation (3.2)) that describes the conditional likelihood of the observations given the process, and the sequence of observations $\{\boldsymbol{x}_t; t \in \mathbb{N}\}$ over t; this method attempts to find a best estimate of the conditional probability $p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_t | \boldsymbol{x}_t, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_t)$ (denoted by $p(\boldsymbol{\theta}_{1:t} | \boldsymbol{x}_{1:t})$).

Often the requirement is to only obtain the optimal estimate for the present instant and not the entire path of trajectory; in which case we attempt to obtain the conditional distribution of the variable at the present instant only. Instead of determining $p(\boldsymbol{\theta}_{1:t} | \boldsymbol{x}_{1:t})$, then, we only need to find out $p(\boldsymbol{\theta}_t | \boldsymbol{x}_{1:t})$. Since the measurement needs to be updated every time a new data entry is received, a recursive filter would be convenient for the purpose. Once this probability is obtained, estimation of any function $g(\cdot)$ of $\boldsymbol{\theta}_t$ can be done (provided $g(\boldsymbol{\theta}_{1:t}) p(\boldsymbol{\theta}_{1:t} | \boldsymbol{x}_{1:t})$ is integrable); a very special (yet common) example is where $g(\boldsymbol{\alpha}) = \boldsymbol{\alpha}$, in which case the requirement would be the estimation of $\boldsymbol{\theta}_t$ itself. In these kind of estimation problems, the initial conditional pdf $p(\boldsymbol{\theta}_0 | \boldsymbol{x}_0)$ is either known or is assigned an a priori value. Depending on the nature of the functions involved, this initialisation may or may not influence the convergence of the filter significantly.

Such a filter has two stages: prediction and update. In the prediction stage, the state pdf at some instant t is forecasted based on all the prior information up to the instant $t - 1$. In the update stage, this forecasted estimate is tuned and modified using the latest measurements for the instant t.

Let us suppose that up to the instant $t - 1$, $t \in \mathbb{N}$, the pdf $p(\boldsymbol{\theta}_{t-1} | \boldsymbol{x}_{1:t-1})$ is available, i.e., it is either known or has been estimated. Because of the Markov property of the process, we also have $p(\boldsymbol{\theta}_t | \boldsymbol{\theta}_{t-1}, \boldsymbol{x}_{1:t-1}) = p(\boldsymbol{\theta}_t | \boldsymbol{\theta}_{t-1})$, i.e., the distribution at the present instant depends only on the distribution at the previous instant and is independent of distributions at any instant prior to the last instant.

At the prediction stage, then, the posterior for the next instant can be predicted as follows.

$$\begin{aligned} p(\boldsymbol{\theta}_t|\boldsymbol{x}_{1:t-1}) &= \int p(\boldsymbol{\theta}_t \cap \boldsymbol{\theta}_{t-1}|\boldsymbol{x}_{1:t-1}) d\boldsymbol{\theta}_{t-1} \\ &= \int \frac{p(\boldsymbol{\theta}_t \cap \boldsymbol{\theta}_{t-1} \cap \boldsymbol{x}_{1:t-1})}{p(\boldsymbol{x}_{1:t-1})} d\boldsymbol{\theta}_{t-1} \\ &= \int \frac{p(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1}, \boldsymbol{x}_{1:t-1}) p(\boldsymbol{\theta}_{t-1} \cap \boldsymbol{x}_{1:t-1})}{p(\boldsymbol{x}_{1:t-1})} d\boldsymbol{\theta}_{t-1} \\ &= \int p(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1}, \boldsymbol{x}_{1:t-1}) \frac{p(\boldsymbol{\theta}_{t-1} \cap \boldsymbol{x}_{1:t-1})}{p(\boldsymbol{x}_{1:t-1})} d\boldsymbol{\theta}_{t-1} \\ &= \int p(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1}, \boldsymbol{x}_{1:t-1}) p(\boldsymbol{\theta}_{t-1}|\boldsymbol{x}_{1:t-1}) d\boldsymbol{\theta}_{t-1} \end{aligned} \tag{3.3}$$

Using the fact the $\boldsymbol{\theta}_t$ is Markovian, and hence depend only on $\boldsymbol{\theta}_{t-1}$, we get

$$p(\boldsymbol{\theta}_t|\boldsymbol{x}_{1:t-1}) = \int p(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1}) p(\boldsymbol{\theta}_{t-1}|\boldsymbol{x}_{1:t-1}) d\boldsymbol{\theta}_{t-1} \tag{3.4}$$

At each step, the integration is over the entire domain of $\boldsymbol{\theta}_{t-1}$ which is in fact the domain of $\boldsymbol{\theta}_t$, in general, for all t.

In the next stage, once measured data for the current instant arrive, the above prediction (or the prior) is modified to obtain the optimal posterior density, using Bayes' rule.

$$p(\boldsymbol{\theta}_t|\boldsymbol{x}_{1:t}) = \frac{p(\boldsymbol{x}_t|\boldsymbol{\theta}_t) p(\boldsymbol{\theta}_t|\boldsymbol{x}_{1:t-1})}{p(\boldsymbol{x}_t|\boldsymbol{x}_{1:t-1})} \tag{3.5}$$

where the denominator is a normalizing constant, given by the Chapman-Kolmogorov equation as follows.

$$p(\boldsymbol{x}_t|\boldsymbol{x}_{1:t-1}) = \int p(\boldsymbol{x}_t|\boldsymbol{\theta}_t) p(\boldsymbol{\theta}_t|\boldsymbol{x}_{1:t}) d\boldsymbol{\theta}_t \tag{3.6}$$

In the above sets of equations, $p(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1})$ is the state transition probability distribution for $\boldsymbol{\theta}_t$ while $p(\boldsymbol{x}_t|\boldsymbol{\theta}_t)$ is the conditional probability of $\boldsymbol{x}_t$ given $\boldsymbol{\theta}_t$. Ideally, both these probabilities should be available when the functions involved and noise distributions are known and hence the solution of the problem may appear to be straightforward at a glance. However, even when the functions are known, in general, the integrals involved in (3.4) and (3.6) cannot be determined analytically, except for a few special cases. For instance, an optimum solution can be derived when the functions are linear and the noises are additive and Gaussian. As one

would expect, the solution derived under this specific model is the same as that obtained by the Kalman filter. For most cases though, a direct solution is either not feasible or not tractable and one has to depend on approximate methods for sub-optimal solutions. In fact, the above set of recursive equations only gives a conceptual framework of the solution.

3.3 A More Specific Model

So far we have described the problem from a general point of view where both the functions f_t, h_t may be non-linear, the noises may not be additive and they may not be Gaussian either. Let us now describe the special case of this problem which we are about to explore.

It is assumed that the state evolution follows a known non-linear function while the observations are linearly related to the present states. We thus replace the non-linear function $h_t(\cdot)$ with a matrix $\boldsymbol{H}_t$. Noises involved at state evolution and observations are assumed to additive and zero-mean Gaussian with known covariance matrices.

Under these conditions the parameter of interest $\boldsymbol{\theta}_t$ is modeled as a discrete-time non-linear system with the following dynamics:

$$\boldsymbol{\theta}_{t+1} = f(\boldsymbol{\theta}_t) + \boldsymbol{w}_t \ \forall \ t \in \mathbb{N} \tag{3.7}$$

The observations are described by the following equations:

$$\boldsymbol{x}_t = \boldsymbol{H}_t\boldsymbol{\theta}_t + \boldsymbol{v}_t \ \forall \ t \in \mathbb{N} \tag{3.8}$$

where, $\boldsymbol{\theta}_t$ is the realisation of the unknown parameter $\boldsymbol{\theta}$ at the instant $t \in \mathbb{N}$, and $\boldsymbol{x}_t$ is the observation vector at each instant t. We assume that the noises $\boldsymbol{w}_t, \boldsymbol{v}_t$ are i.i.d. and follow Gaussian distributions with mean zero and known covariance matrices $\boldsymbol{Q}_t, \boldsymbol{R}_t$ respectively. The function $f(\boldsymbol{\theta}_t)$ that determines evolution of the state is known and defined for $t > 0$. It is also assumed that $\boldsymbol{\theta}_t, \boldsymbol{w}_t$ and $\boldsymbol{v}_t$ are uncorrelated. The state variable $\boldsymbol{\theta}_t$ and the observation $\boldsymbol{x}_t$ are both vectors with finite dimensions N and M respectively and in general, $N > M$. Then, $\boldsymbol{w}_t$ and $\boldsymbol{v}_t$ are vectors of dimensions N and M respectively; $\boldsymbol{H}_t$ is a real matrix of dimension $M \times N$; and $\boldsymbol{Q}_t, \boldsymbol{R}_t$ are $N \times N$ and $M \times M$ square matrices.

Evidently, for the specific problem we are interested in, $p(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1})$ is the pdf of a Normal Distribution with mean $f(\boldsymbol{\theta}_{t-1})$ and covariance matrix $\boldsymbol{Q}_t$; while $p(\boldsymbol{x}_t|\boldsymbol{\theta}_t)$ is the density of a Normal distribution with mean $\boldsymbol{H}_t\boldsymbol{\theta}_t$ and covariance matrix $\boldsymbol{R}_t$.

This means that for this problem, the integral in 3.4 would also involve the non-linear function $f(\cdot)$, and depending on its nature it may or may not be tractable.

Chapter 4

The Particle Filter

4.1 Introduction

A Particle filter (Moral [1996], Doucet et al. [2000], Maskell and Gordon [2001]) is essentially a sequential Monte Carlo estimation technique that is used for solving a wide variety of problems involving non-linearity and high dimensionality. In different papers and works, different terms have been used to describe this filtering mechanism. These names include bootstrap filtering, sequential Monte Carlo method, the condensation algorithm, interacting particle approximation and survival of the fittest. There are some minor differences in the different versions of the particle filter that are in use, and some algorithms use additional steps that are not employed by others, but fundamentally they are all based on the same basic principle. The method uses a set of point mass random samples (called 'particles') of probability densities and constructs a representation of the posterior density function by combining them, using a set of so-called 'importance weights'. There are several variants of the particle filter which can be broadly categorized into two groups (Haug [2005]). In one, the same particles are re-used as trajectories while in the other particles are not re-used and fresh particles are generated at each step. Since our aim is to do a comparison of the particle filter with the ensemble Kalman filter, we have chosen to consider the filter that comes to the first category because of its intuitive similarity to the ensemble Kalman filter. Moreover, this model, known as the sequential importance sampling (SIS) particle filter is also more popular compared to its counterpart.

In this scheme, starting from the same initial conditions, several dynamic realisations or trajectories of the state are developed over time, using the available information from past and present. The optimal solution at each step is a weighted average of all these different trajectories. The weights attached to the different paths of realisation indicate the relative likelihood of the corresponding trajectory.

As the number of samples increase, the solution of this filter approaches the optimal Bayesian estimate. To further facilitate the procedure, an additional step is included, which is known as resampling. This is done to eliminate the effects of filter degeneracy, or the situation when only a few of the trajectories being computed have a weight large enough to contribute to the final solution, while the weights associated with the rest are too insignificant to have any prominence. Under such conditions, particles are resampled and the optimum estimate is computed using modified weights.

4.2 Description of the Algorithm

The basic steps involved in the particle filter algorithm are now described (Maskell and Gordon [2001]). Let $\{\hat{\boldsymbol{\theta}}_t^i; t \in \mathbb{N}, i = 1(1)N_s\}$ be a set of samples of size N_s drawn from a distribution that is an approximate representation of the posterior. Let $\boldsymbol{\omega}_t^i$ be the respective weights associated with the sample points. By definition, initial points $\hat{\boldsymbol{\theta}}_0^i = \boldsymbol{\theta}_0$ for all i. The posterior density at an instant t can then be estimated as:

$$p(\boldsymbol{\theta}_{0:t}|\boldsymbol{x}_{1:t}) \approx \sum_{i=1}^{N_s} \boldsymbol{\omega}_t^i \delta(\boldsymbol{\theta}_{0:t} - \hat{\boldsymbol{\theta}}_{0:t}^i) \tag{4.1}$$

where the weights are normalized, i.e.,

$$\sum_{i=1}^{N_s} \boldsymbol{\omega}_t^i = 1 \tag{4.2}$$

It follows that the estimates for $\boldsymbol{\theta}_t$ over time are given by a weighted sum of the individual particle trajectories, where the weights are defined in equations (4.1) and (4.2).

$$\hat{\boldsymbol{\theta}}_{0:t,PF}|\boldsymbol{x}_{1:t} = \sum_{i=1}^{N_s} \boldsymbol{\omega}_t^i \hat{\boldsymbol{\theta}}_{0:t}^i \tag{4.3}$$

When only an estimate at the present instant t is of interest, the above reduces to

$$\hat{\boldsymbol{\theta}}_{t,PF}|\boldsymbol{x}_{1:t} = \sum_{i=1}^{N_s} \boldsymbol{\omega}_t^i \hat{\boldsymbol{\theta}}_t^i \tag{4.4}$$

It is evident, then, that this method consists of two major steps, viz., selection of a suitable distribution to draw samples for $\hat{\boldsymbol{\theta}}_t^i$, and assignment of proper weights $\boldsymbol{\omega}_t^i$ to those samples. An ideal distribution to draw samples from would be the posterior itself, but that is impossible in most cases. As such, a method called **Importance Sampling** is employed. Here one requires to identify a proposal density $q(\cdot)$ from which samples can be drawn easily. The true weights, then are evaluated from the following equations :

$$\boldsymbol{\omega}_t^{i*} = \frac{p(\hat{\boldsymbol{\theta}}_{0:t}^i|\boldsymbol{x}_{1:t})}{q(\hat{\boldsymbol{\theta}}_{0:t}^i|\boldsymbol{x}_{1:t})} \tag{4.5}$$

and, after normalizing,

$$\boldsymbol{\omega}_t^i = \frac{\boldsymbol{\omega}_t^{i*}}{\sum_{i=1}^{N_s} \boldsymbol{\omega}_t^{i*}} \tag{4.6}$$

When the proposal distribution $q(\cdot)$ satisfies the following properties,

$$q(\boldsymbol{\theta}_{0:t}|\boldsymbol{x}_{1:t}) = q(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{0:t-1}, \boldsymbol{x}_{1:t}) q(\boldsymbol{\theta}_{0:t-1}|\boldsymbol{x}_{1:t-1}) \tag{4.7}$$

$$q(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{0:t-1}, \boldsymbol{x}_{1:t}) = q(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1}, \boldsymbol{x}_t) \tag{4.8}$$

the importance weights can be shown to follow a simple recursive relation as follows:

$$\boldsymbol{\omega}_t^i \propto \boldsymbol{\omega}_{t-1}^i \frac{p(\boldsymbol{x}_t|\boldsymbol{\theta}_t^i) p(\hat{\boldsymbol{\theta}}_t^i|\hat{\boldsymbol{\theta}}_{t-1}^i)}{q(\hat{\boldsymbol{\theta}}_t^i|\hat{\boldsymbol{\theta}}_{t-1}^i, \boldsymbol{x}_t)} \tag{4.9}$$

The estimate equation for the posterior then reduces to

$$p(\boldsymbol{\theta}_t|\boldsymbol{x}_{1:t}) \approx \sum_{i=1}^{N_s} \boldsymbol{\omega}_t^i \delta(\boldsymbol{\theta}_t - \hat{\boldsymbol{\theta}}_t^i) \tag{4.10}$$

4.3 Choices of Proposal Distribution and Importance Weights

Critical in this method are the choices of the proposal distribution $q(\cdot)$ and those of the importance weights $\boldsymbol{\omega}_t^i$. As is apparent from the discussion so far, choice of the proposal $q(\cdot)$ plays a crucial role in this methodology during the sampling stage,

while a good choice of importance weights becomes important while combining the samples to get the best estimate. A good choice is to select that $q(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1})$ which minimizes the variance of real weights $\boldsymbol{\omega}_t^{i*}$, given $\hat{\boldsymbol{\theta}}_{t-1}^i$ and $\boldsymbol{x}_t$. Such a choice maximizes the effective sample size $N_{s,eff}$; thereby reducing the effects of degeneracy of the filter. The optimal importance density function based on this consideration can be shown to be

$$q(\boldsymbol{\theta}_t|\hat{\boldsymbol{\theta}}_{t-1}^i, \boldsymbol{x}_t)_{optimal} = p(\boldsymbol{\theta}_t|\hat{\boldsymbol{\theta}}_{t-1}^i, \boldsymbol{x}_k) = \frac{p(\boldsymbol{x}_t|\boldsymbol{\theta}_t, \hat{\boldsymbol{\theta}}_{t-1}^i)p(\boldsymbol{\theta}_t|\hat{\boldsymbol{\theta}}_{t-1}^i)}{p(\boldsymbol{x}_t|\hat{\boldsymbol{\theta}}_{t-1}^i)} \tag{4.11}$$

Under these conditions, the weights given by (4.9) become:

$$\boldsymbol{\omega}_t^i \propto \boldsymbol{\omega}_{t-1}^i p(\boldsymbol{x}_t|\hat{\boldsymbol{\theta}}_{t-1}^i) \tag{4.12}$$

The above gives a general framework for the operation of the particle filter. It may be noted that the model described thus far is one among several variants of the so-called particle filter and these variants have minor differences in their operating principles.

To obtain the optimal importance density it is required to draw samples from $p(\boldsymbol{\theta}_t|\hat{\boldsymbol{\theta}}_{t-1}^i, \boldsymbol{x}_k)$ and subsequently evaluate an integral; which may not be straightforward in many cases, depending on the dynamics of the system. There are special cases, however, where the integration may be analytically feasible. The model used in this discussion, which assumes the noises to be i.i.d. Gaussian and the relation between the observed data and the system parameters to be linear, is one such example where the integration is not intractable and the parameters of the required distributions can be easily determined.

4.4 Parameters for the Specified Model

Referring to the model described by (3.7) and (3.8), we can then construct the following conditional probabilities (Doucet et al. [2000]):

$$p(\hat{\boldsymbol{\theta}}_t^i|\hat{\boldsymbol{\theta}}_{t-1}^i, \boldsymbol{x}_t) = N(\hat{\boldsymbol{\theta}}_t^i; \boldsymbol{\mu}_t^i, \boldsymbol{\psi}_t) \tag{4.13}$$

where

$$\boldsymbol{\mu}_t^i = (\boldsymbol{Q}_{t-1}^{-1} + \boldsymbol{H}_t^T \boldsymbol{R}_t^{-1} \boldsymbol{H}_t)^{-1} (\boldsymbol{Q}_{t-1}^{-1} f(\hat{\boldsymbol{\theta}}_{t-1}^i) + \boldsymbol{H}_t^T \boldsymbol{R}_t^{-1} \boldsymbol{x}_t) \tag{4.14}$$

$$(\boldsymbol{\psi}_t)^{-1} = \boldsymbol{Q}_{t-1}^{-1} + \boldsymbol{H}_t^T \boldsymbol{R}_t^{-1} \boldsymbol{H}_t \; \forall \; i = 1(1)N_s \tag{4.15}$$

where N_s is the sample size.

The importance weights of the different trajectories $\boldsymbol{\omega}_t^i$ would be recursively obtained using the following conditional probability in equation (4.12).

$$p(\boldsymbol{x}_t|\hat{\boldsymbol{\theta}}_{t-1}^i) = N(\boldsymbol{x}_t; \boldsymbol{H}_t f(\hat{\boldsymbol{\theta}}_{t-1}^i), \boldsymbol{R}_t + \boldsymbol{H}_t \boldsymbol{Q}_{t-1} \boldsymbol{H}_t^T) \tag{4.16}$$

where $N(\boldsymbol{a}; \boldsymbol{\mu}, \boldsymbol{\psi})$ denotes the probability density function of a vector $\boldsymbol{a}$ that follows a Normal distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\psi}$.

For the recursive equations described by (4.12) we would start with an initialisation of equal weights for each trajectory, meaning that all the realisations are considered equally important at the initial stage. At the subsequent steps, the importance weights would be modified according to (4.16), which would assign greater importance to trajectories that are more likely to generate the recorded observations. The optimal solution for $\hat{\boldsymbol{\theta}}_t$ at any instant t would be its conditional expectation computed using the probability given by (4.4).

4.5 Resampling

An additional step introduced in this particular algorithm is **resampling**. This is performed to reduce the effects of filter degeneracy. Degeneracy is the situation that arises when only a few of the sampled trajectories contribute to the final computed values by means of their higher weights, while others play a very insignificant role in the estimation as their corresponding weights are exceedingly small. Such a situation would ruin the main purpose of the strategy because final estimation would effectively involve only a small proportion of the samples actually generated. This would also mean that a lot of computation done in generating samples with lesser weights is remaining underutilised.

To avoid degeneracy, therefore, proper steps are taken. After computation of the different weights $\boldsymbol{\omega}_t^i$, the effective sample size $N_{s,eff}$ is defined as follows.

$$N_{s,eff} = \frac{N_s}{1 + Var(\boldsymbol{\omega}_t^{i*})} \tag{4.17}$$

A simple estimate $\hat{N}_{s,eff}$ for $N_{s,eff}$ is given by Doucet et al. [2000] as

$$\hat{N}_{s,eff} = \frac{1}{\sum_{i=1}^{N_s} (\boldsymbol{\omega}_t^i)^2} \tag{4.18}$$

The quantity $\hat{N}_{s,eff}$ is an indicator of the filter degeneracy. It can be taken as an approximate measure of the number of particles out of the ones sampled that actually play a significant role in the estimation. When this number goes below a certain predefined threshold, therefore, resampling is performed, thereby diminishing the scope of degeneracy. A new sample set of $\boldsymbol{x}_t^{i*}$ of size N_s is redrawn from an approximate discrete representation of $p(\boldsymbol{\theta}_t|\boldsymbol{x}_{1:t})_{Resampling}$, where the probability of a particle being chosen is the same as its relative weight computed at the corresponding step (i.e., $p(\boldsymbol{x}_t^{i*} = \boldsymbol{x}_t^j) = \boldsymbol{\omega}_t^i$). This is illustrated in equation (4.19).

$$p(\boldsymbol{\theta}_t|\boldsymbol{x}_{1:t})_{Resampling} \approx \sum_{i=1}^{N_s} \boldsymbol{\omega}_t^i \delta(\boldsymbol{\theta}_t - \boldsymbol{\theta}_t^i) \tag{4.19}$$

In this case, the final estimate is given by the arithmetic mean of the resampled particles, i.e., by replacing the terms $\boldsymbol{\omega}_t^i$ in equation (4.4) with $\frac{1}{N_s}$. It may be noted here that although the method just described is frequently employed in particle filter algorithms, there can be other resampling schemes as well.

Degeneracy of sample paths is caused by the iterative nature of the importance weight update equations (4.9,4.12) and would therefore happen only when sequential importance sampling is done. If the importance weights of the paths at each step are calculated independent of their previous weights, then the effect of weights of certain paths becoming small would not accumulate sequentially, and hence there would be no significant degeneracy. Consequently, there would be no requirement of resampling in such a scenario.

The steps involved in estimation of the unknown state variables by a particle filter are illustrated in the flow-chart of 4.1.

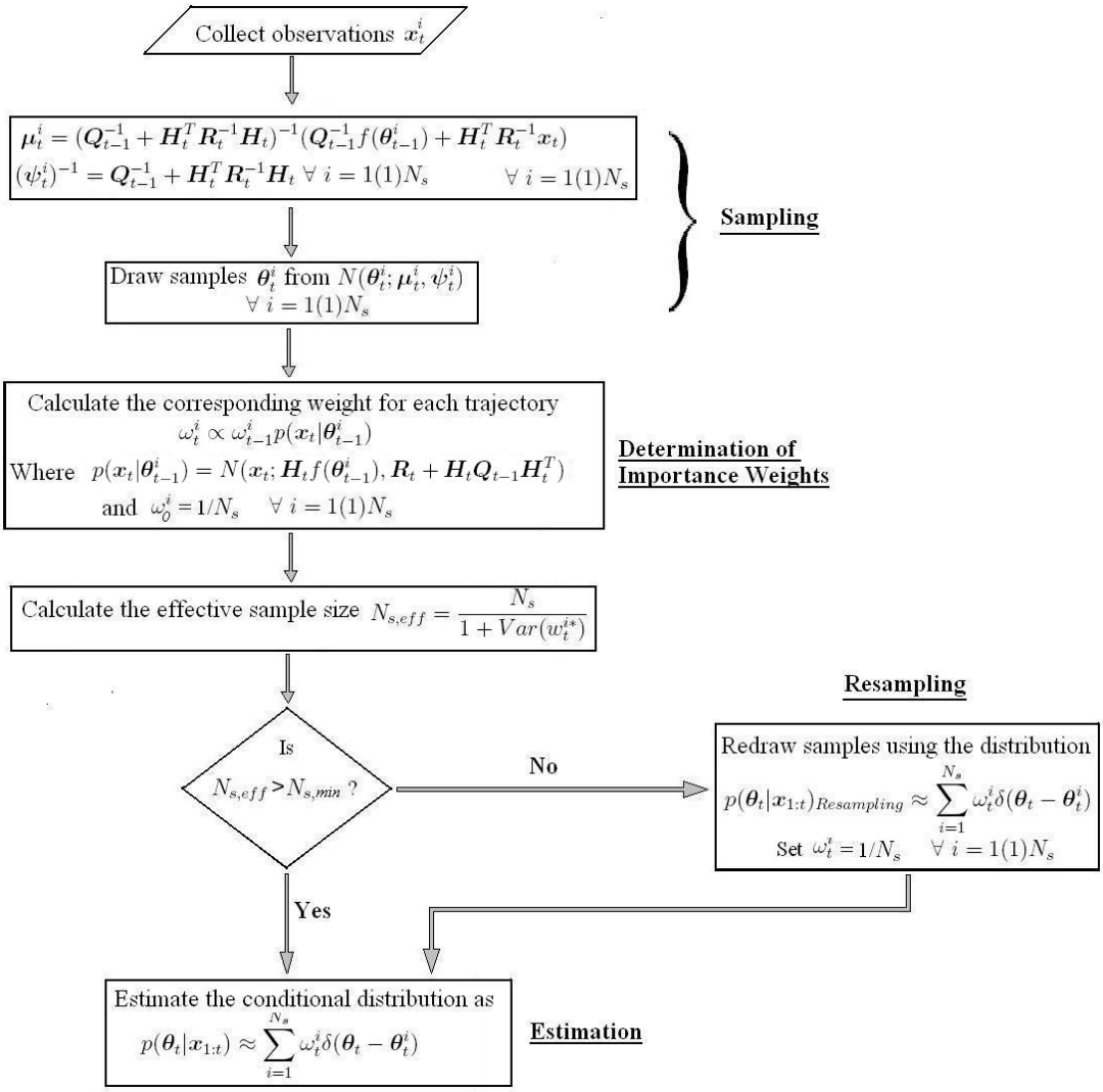

Figure 4.1: Flow chart illustrating particle filter algorithm

Chapter 5

The Ensemble Kalman Filter

We shall now construct the solution of this problem using the ensemble Kalman filter technique (Evensen [2003], Gillijns et al. [2006]). This filter is derived from the classical Kalman filter, a tool that provides optimal solutions for linear Gaussian models. It provides sub-optimal solutions for problems involving extremely high orders and non-linearity, and has in particular gained popularity in the field of weather forecasting, among other areas. It may be noted that this filter does not give very good results under non-Gaussian environments and is hence used mostly when the model is Gaussian. The reason for this is that this filter obtains its estimates using only the first and second moments of the error terms, thereby making an implicit Gaussian assumption. In essence, it is a Monte Carlo approximation of the Kalman filter, where it replaces the actual covariance with the sample covariance calculated over an ensemble of realisations. Different realisations or trajectories of the state evolution are generated using the Kolmogorov Forward Equation.

In order to understand the principle behind this filter, therefore, it is required to first understand the fundamental ideas of the classical discrete Kalman filter (Kalman [1960]), which gives the optimum estimate for a discrete time Bayesian estimation problem in a linear and Gaussian environment. The Kalman filter is a recursive filtering method that uses only the current observed data and the estimate of the state at the last instant to estimate the state at the present instant. Thus, it is ideally suited for the estimation of linear and Gaussian hidden Markov models. As is the case for any Bayesian estimation method, the Kalman filter, too entails the two standard steps of prediction and update. In the prediction stage, estimates are produced based on the last estimates of the state variables, and then subsequently, in the update phase, the predicted estimate is refined and improved using measurement information at the current instant of time.

5.1 The Discrete Kalman Filter

Let us consider the system described by the following equations:

$$\boldsymbol{Y}_{t+1} = \boldsymbol{A}_t \boldsymbol{Y}_t + \boldsymbol{W}_t \tag{5.1}$$

$$\boldsymbol{Z}_t = \boldsymbol{B}_t \boldsymbol{Y}_t + \boldsymbol{V}_t \tag{5.2}$$

where $\boldsymbol{Y}_t, \boldsymbol{Z}_t$ are the state vector and observation vector respectively, $\boldsymbol{W}_t, \boldsymbol{V}_t$ are uncorrelated white Gaussian noises with covariance matrices $\boldsymbol{Q}_t$, $\boldsymbol{R}_t$ respectively. $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$ are matrices defining the system dynamics.

The state estimation equation for the Kalman filter for a linear and Gaussian dynamic system is derived by minimizing the estimated error covariances. The optimal estimation of $\boldsymbol{Y}_t$ for such a system is given by the following equations:

The prediction phase consists of equations (5.3) and (5.4) while the update phase is given by equations (5.5) to (5.7).

$$\boldsymbol{Y}_{t|t-1} = \boldsymbol{A}_t \boldsymbol{Y}_{t-1|t-1} \tag{5.3}$$

$$\boldsymbol{P}_{t|t-1} = \boldsymbol{A}_t \boldsymbol{P}_{t-1|t-1} \boldsymbol{A}_t^T + \boldsymbol{Q}_{t-1} \tag{5.4}$$

$$\boldsymbol{K}_t = \boldsymbol{P}_{t|t-1} \boldsymbol{B}_t^T (\boldsymbol{B}_t \boldsymbol{P}_{t|t-1} \boldsymbol{B}_t^T + \boldsymbol{R}_t)^{-1} \tag{5.5}$$

$$\boldsymbol{Y}_{t|t} = \boldsymbol{Y}_{t|t-1} + \boldsymbol{K}_t (\boldsymbol{Z}_t - \boldsymbol{B}_t \boldsymbol{Y}_{t|t-1}) \tag{5.6}$$

$$\boldsymbol{P}_{t|t} = (1 - \boldsymbol{K}_t \boldsymbol{B}_t) \boldsymbol{P}_{t|t-1} \tag{5.7}$$

In the above set of equations, $\boldsymbol{Y}_{t|t-1}$ is the a priori estimate of $\boldsymbol{Y}_{t|t}$, i.e., the estimate at the prediction stage; $\hat{\boldsymbol{Y}}_t = E[\boldsymbol{Y}_{t|t}]$ is the updated estimate of $\boldsymbol{Y}_t$, $\boldsymbol{P}_{t|t-1}$ is the a priori estimate error covariance and $\boldsymbol{P}_{t|t}$ is the a posteriori estimate error covariance, obtained by updating the a priori using the Kalman gain. It is essentially an indicative measure of the accuracy of the state estimation. The Kalman gain $\boldsymbol{K}_t$ given by (5.5) can be arrived at by minimizing the a posteriori error covariance. These optimal solutions however are only achievable under a linear and Gaussian environment. When the condition of linearity is not met, approximate derivatives of the Kalman filter, such as the extended Kalman filter and the ensemble Kalman filter are used.

5.2 The Ensemble Kalman Filter Algorithm

The ensemble Kalman filter works on the same principle as above, i.e., it too, attempts to minimize the error covariance but in this case the error statistics are modeled using an ensemble of predicted states. Instead of calculating the error covariance matrices in their exact terms, this method approximates them by creating a set of estimate points; thereby reducing the computational burden associated with the inversion of high-dimension matrices. Let us now describe the different steps employed in this scheme (Gillijns et al. [2006]).

Let us consider an instant $t-1$, when the latest observation recorded is $\boldsymbol{x}_{t-1}$. The latest sub-optimal estimate for $\boldsymbol{\theta}$ obtained at this time is that corresponding to $t-1$. The model would first come up with a set of predictions for $\boldsymbol{\theta}$ at the instant t, and subsequently modify this set once new observation $\boldsymbol{x}_t$ is available.

The method starts by generating a finite number of estimate points for the state parameter $\boldsymbol{\theta}_t$ from an a priori distribution. Let us denote this predicted or forecasted ensemble of state estimates by $\boldsymbol{\Theta}_t^f$. and let the fixed sample size be N_s.

$$\boldsymbol{\Theta}_t^f = \{\boldsymbol{\theta}_t^{f_i}\};\ i = 1(1)N_s \tag{5.8}$$

An ensemble of the same size N_s consisting of observations is also generated by adding small perturbations to the current observation. A reasonable method would be to create perturbations that have the same distribution as the observation error. Let the observation ensemble be denoted by $\boldsymbol{X}_t^f$

$$\boldsymbol{X}_t^f = \{\boldsymbol{x}_t^{f_i}\};\ i = 1(1)N_s \tag{5.9}$$

Given a system described by equations (3.7) and (3.8), the samples for the state model ensemble may be drawn using the following rule:

$$\boldsymbol{\theta}_t^{f_i} = f(\hat{\boldsymbol{\theta}}_{t-1}^i) + \boldsymbol{w}_{t-1}^{f_i} \tag{5.10}$$

where $\hat{\boldsymbol{\theta}}_{t-1}^i$ is the updated estimate for the i^{th} trajectory, and $\boldsymbol{w}_{t-1}^{f_i}$ are random noise with covariance $\boldsymbol{Q}_{t-1}$. The observation ensemble at the current instant may be generated by adding zero-mean random noise $\boldsymbol{v}_t^{f_i}$ with covariance matrix $\boldsymbol{R}_t$ to the actual observation. To generate the forecasted observation ensemble, the following equation can be employed:

$$\boldsymbol{x}_t^{f_i} = \boldsymbol{H}_t\boldsymbol{\theta}_t^{f_i} + \boldsymbol{v}_t^{f_i} \tag{5.11}$$

The state ensemble error matrix $\boldsymbol{E}^f_{\theta,t}$ and the observation ensemble error matrix $\boldsymbol{E}^f_{x,t}$ are then defined as follow:

$$\boldsymbol{E}^f_{\theta,t} = [\boldsymbol{\theta}^{f,1}_t - \bar{\boldsymbol{\theta}}^f_t,\ \cdots\ \boldsymbol{\theta}^{f,N_s}_t - \bar{\boldsymbol{\theta}}^f_t];\ i = 1(1)N_s \tag{5.12}$$

$$\boldsymbol{E}^f_{x,t} = [\boldsymbol{x}^{f,1}_t - \bar{\boldsymbol{x}}^f_t,\ \cdots\ \boldsymbol{x}^{f,N_s}_t - \bar{\boldsymbol{x}}^f_t];\ i = 1(1)N_s \tag{5.13}$$

where $\bar{\boldsymbol{\theta}}^f_t$, $\bar{\boldsymbol{x}}^f_t$ are the ensemble averages for the state and the observations; i.e.,

$$\bar{\boldsymbol{\theta}}^f_t = \frac{1}{N_s}\sum_{i=1}^{N_s} \boldsymbol{\theta}^{f_i}_t \tag{5.14}$$

$$\bar{\boldsymbol{x}}^f_t = \frac{1}{N_s}\sum_{i=1}^{N_s} \boldsymbol{x}^{f_i}_t \tag{5.15}$$

Clearly, $\bar{\boldsymbol{\theta}}^f_t$ is the estimate at the prediction stage. Next, the estimated error covariance $\bar{\boldsymbol{P}}^f_{\theta x,t}$ and estimated observation covariance $\bar{\boldsymbol{P}}^f_{xx,t}$ are computed using the following equations.

$$\bar{\boldsymbol{P}}^f_{\theta x,t} = \frac{1}{N_s - 1}\boldsymbol{E}^f_{\theta,t}[\boldsymbol{E}^f_{x,t}]^T \tag{5.16}$$

$$\bar{\boldsymbol{P}}^f_{xx,t} = \frac{1}{N_s - 1}\boldsymbol{E}^f_{x,t}[\boldsymbol{E}^f_{x,t}]^T \tag{5.17}$$

Finally, the updated estimates for each trajectory are computed using the following equations:

$$\hat{\boldsymbol{\theta}}^i_{t,EKF,N_s} = \boldsymbol{\theta}^{f_i}_t + (\bar{\boldsymbol{P}}^f_{\theta x,t})(\bar{\boldsymbol{P}}^f_{xx,t})^{-1}(\boldsymbol{x}^i_t - \boldsymbol{H}_t\boldsymbol{\theta}^{f_i}_t) \tag{5.18}$$

These are the update equations. The $\boldsymbol{x}^i_t$ are generated by adding zero-mean perturbations of covariance matrix $\boldsymbol{R}_t$ to the actual measured observation at the current instant t.

For the sake of simplicity, we would drop the suffix N_s from the above expression in later discussions, and use $\hat{\boldsymbol{\theta}}^i_{t,EKF}$.

At any stage, the best-guess solution is the ensemble mean of the updated realisations, i.e., the mean of $\hat{\boldsymbol{\theta}}^i_{t,EKF}$. Also, at any instant, the relative frequency of a data point, $\frac{1}{N_s}N(\hat{\boldsymbol{\theta}}^i_{t,EKF} \in \phi)$ acts as an estimator of the probability of $P(\boldsymbol{\theta}_t \in \phi)$, where ϕ is some subset in the domain of $\boldsymbol{\theta}_t$.

When required to obtain the estimate for some function $F(\cdot)$ of $\boldsymbol{\theta}$ in this method, the procedure is to approximate the expectation of that function by a weighted sum of the values of the function calculated for each trajectory, in the following way

$$\begin{aligned} E[F(\boldsymbol{\theta}_t)] &= \int F(\boldsymbol{\theta}_t)p(\boldsymbol{\theta}_t)dt \\ &\simeq \int F(\boldsymbol{\theta}_t)\sum_{i=1}^{N_s}\delta(\boldsymbol{\theta}_t - \hat{\boldsymbol{\theta}}^i_{t,EKF})dt \\ &\simeq \frac{1}{N_s}\sum_{i=1}^{N_s}F(\hat{\boldsymbol{\theta}}^i_{t,EKF}) \end{aligned} \tag{5.19}$$

Evidently, as N_s goes to infinity, these relative frequencies would approach the actual probabilities, and the integral approximation would approach the true value of the integral.

It is interesting to compare the above solution with the optimum solution given by the Kalman filter in equations 5.3 to 5.7. There are two differences in the formation of the solutions. Firstly, as one would expect, the predicted value of the next state is a non-linear function of the current state instead of a linear function as was the case for the Kalman filter. Secondly, instead of using the exact cross-covariance and covariance terms the ensemble Kalman filter has replaced them with their estimates. Computation of such estimates would be easier than computation of the corresponding quantities exactly when system dimension is high.

At this point, it is seen that the ensemble Kalman filter implicitly assumes the distributions to be Gaussian. This becomes apparent from the fact that it only uses the first and second moments to estimate the distribution, even though the first two moments would completely define a distribution only when the distribution is Gaussian. When system dynamics are highly non-Gaussian, this assumption affects the performance of this filter and for this reason its use is not recommended in such situations.

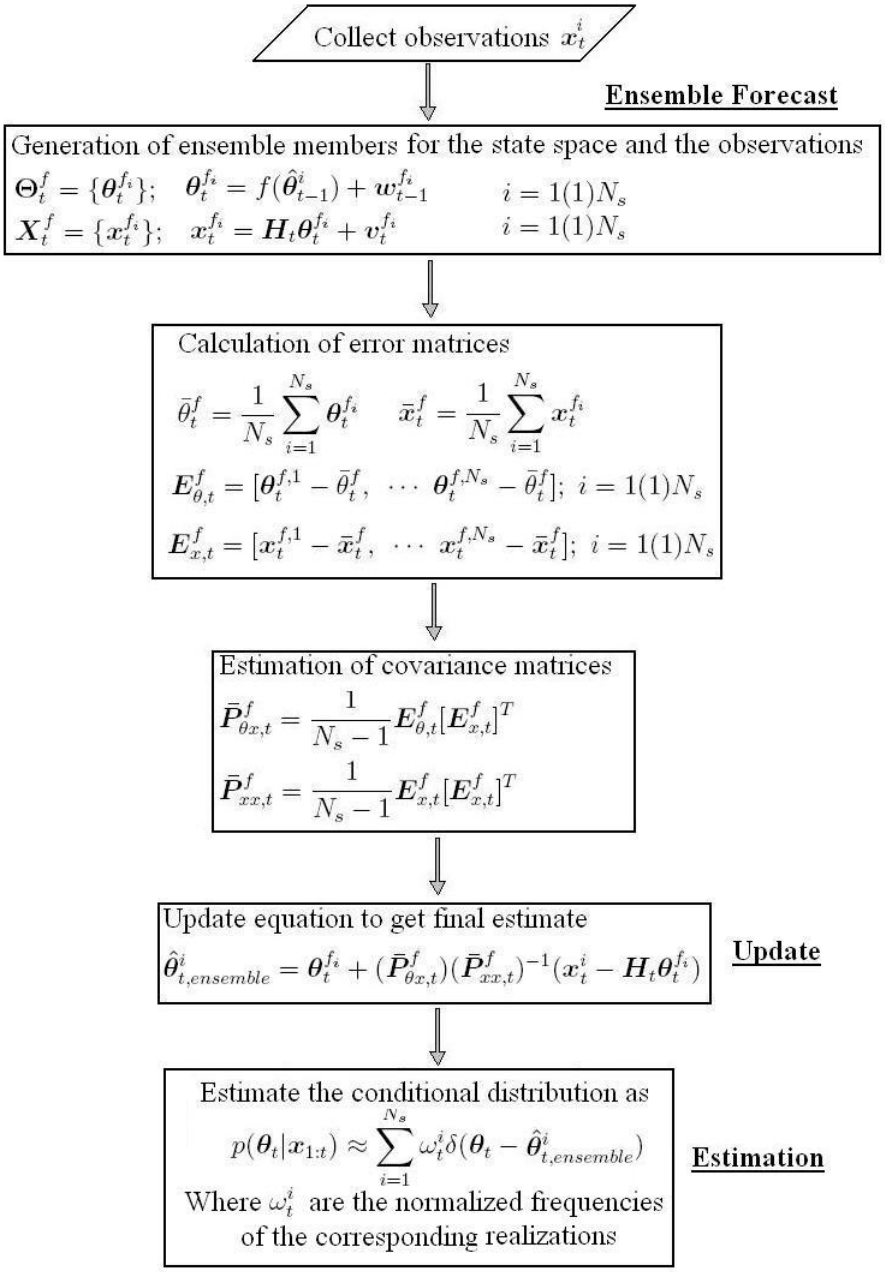

Figure 5.1: Flow chart illustrating ensemble Kalman filter algorithm

Chapter 6

An Analytical Comparison of the Two Schemes

6.1 Introduction

Having illustrated the methodologies followed in the two techniques, let us now proceed to make a comparison of the two filters. From the descriptions of the two methods a close similarity is apparent. It is seen that both methods develop a set of realisations for the variable of interest using certain sequential iterative methods, and obtain the best estimate based on these realisations.

We would now show that the similarity seen with intuition can be established mathematically. More specifically, we would show that, as the sample size goes to infinity, if at any stage, the particle filter and the ensemble Kalman filter start with the same set of ensemble points, then at the next step, the expected values of the estimates of the ensemble Kalman filter trajectories would be equal to the expectations of the sampling distributions of the particle filter, and the covariances of the individual estimates provided by the ensemble Kalman filter would be equal to the covariances of the mentioned sampling distributions. This effectively means that the ensemble Kalman filter methodologically is an approximated version of the particle filter, without the step involving importance weights.

Let us now illustrate a brief outline of the proof. We would first show that the terms involved in the ensemble Kalman filter equations developed at different steps of its derivation would converge in distribution to fixed expressions containing some known matrices. This would follow from the realisation that these terms are approximations of certain covariance quantities related to the state variable $\boldsymbol{\theta}_t$, the observation variable $\boldsymbol{x}_t$ and the noises $\boldsymbol{w}_t$ and $\boldsymbol{v}_t$. We would then, under certain restrictions on the ensemble Kalman filter estimates, obtain the expectation and

covariance of the estimates given by each estimate. Finally, we would relate these quantities with the expectation and covariance of the particle filter.

6.2 Convergence Results for the Ensemble Kalman Filter Estimates

First let us state our assumptions on the bounds of the ensemble Kalman filter estimates.

Let, at any instant t, and for any trajectory i, $t \in \mathbb{N}$, $i \in \{1 \cdots N_s\}$, $\hat{\theta}^i_{t,EKF,N_s,\alpha}$ denote the α^{th} element of the column vector $\hat{\boldsymbol{\theta}}^i_{t,EKF,N_s}$, $\alpha \in \{1 \cdots N\}$. We assume that for all α, β, and for all i, $\hat{\theta}^i_{t,EKF,N_s,\alpha}$, $(\hat{\theta}^i_{t,EKF,N_s,\alpha})^2$ and $\hat{\theta}^i_{t,EKF,N_s,\alpha}\hat{\theta}^i_{t,EKF,N_s,\beta}$ are all **uniformly integrable**.

This means that, at any instant t, and for all trajectories i, for every $\epsilon > 0$, there exist $K_\alpha = K_\alpha(\epsilon)$, $L_\alpha = L_\alpha(\epsilon)$ and $C_{\alpha,\beta} = C_{\alpha,\beta}(\epsilon)$ such that all the following inequalities from 6.1 to 6.3 hold, for all $\alpha, \beta \in \{1 \cdots N\}$.

$$\sup_{N_s \in \mathbb{N}} E[|\hat{\theta}^i_{t,EKF,N_s,\alpha}|\mathbb{I}_{\{|\hat{\theta}^i_{t,EKF,N_s,\alpha}|>K_\alpha\}}] < \epsilon \tag{6.1}$$

$$\sup_{N_s \in \mathbb{N}} E[|(\hat{\theta}^i_{t,EKF,N_s,\alpha})^2|\mathbb{I}_{\{|(\hat{\theta}^i_{t,EKF,N_s,\alpha})^2|>L_\alpha\}}] < \epsilon \tag{6.2}$$

and

$$\sup_{N_s \in \mathbb{N}} E[|\hat{\theta}^i_{t,EKF,N_s,\alpha}\hat{\theta}^i_{t,EKF,N_s,\beta}|\mathbb{I}_{\{|\hat{\theta}^i_{t,EKF,N_s,\alpha}\hat{\theta}^i_{t,EKF,N_s,\beta}|>C_{\alpha,\beta}\}}] < \epsilon \tag{6.3}$$

Let us recall that the variables $\boldsymbol{\theta}$ and $\boldsymbol{x}$ are both vectors with finite dimensions.The sample size for the particle filter and the ensemble size for the ensemble Kalman filter are both assumed to be equal to some integer N_s. For such a formulation, the optimum solution for the particle filter and the ensemble Kalman filter are given by equations (4.13) through (4.16) and (5.8) through (5.18) respectively. Let us first analyze the solutions given by the latter and examine where they would converge to, when the ensemble size is sufficiently large.

We consider equations (5.10) and (5.11) which describe the drawing of samples at the forecast stage. Since the random noises $\boldsymbol{w}_t^{f_i}$ and $\boldsymbol{v}_t^{f_i}$ that are added as perturbations to generate the ensemble sets for the state variables and observations are both zero-mean, it follows that

$$E[\boldsymbol{\theta}^{f_i}_{t+1,EKF}] = f(\hat{\boldsymbol{\theta}}^{i}_{t,EKF}) \tag{6.4}$$

$$E[\boldsymbol{x}^{f_i}_{t+1}] = \boldsymbol{H}_t f(\hat{\boldsymbol{\theta}}^{i}_{t,EKF}) \tag{6.5}$$

$E[\boldsymbol{\theta}^{f_i}_{t+1,EKF}]$, $f(\hat{\boldsymbol{\theta}}^{i}_{t,EKF})$, and $E[\boldsymbol{x}^{f_i}_{t+1}]$ are column vectors of dimensions N, N and M respectively. Let us denote $\tilde{\theta}_j$ as the j^{th} element of $f(\hat{\boldsymbol{\theta}}^{i}_{t,EKF})$, $\bar{\theta}_j$ as the j^{th} element of $E[\boldsymbol{\theta}^{f_i}_{t+1,EKF}]$, $\tilde{x}_k$ as the k^{th} element of $\boldsymbol{x}^{f_i}_{t+1}$ and $\bar{x}_k$ as the k^{th} element of $E[\boldsymbol{x}^{f}_{t}]$ respectively, for $j = 1(1)N$ and $k = 1(1)M$. Also let, w_j and v_k denote the j^{th} and k^{th} element of the noise vectors $\boldsymbol{w}^{f_i}_{t-1}$ and $\boldsymbol{v}^{f_i}_{t}$. Let q_{ab} and r_{cd} be the $(a,b)^{th}$ and $(c,d)^{th}$ elements of the noise covariance matrices $\boldsymbol{Q}_{t-1}$ and $\boldsymbol{R}_t$, where $a = 1(1)N$, $b = 1(1)N$, $c = 1(1)M$, $d = 1(1)M$. Finally, let h_{pq} denote the $(p,q)^{th}$ element of the coefficient matrix $\boldsymbol{H}_t$; $p = 1(1)M$, $q = 1(1)N$.

From equations (6.4) and (6.5) then, we have

$$\tilde{\theta}_j = \bar{\theta}_j + w_j \tag{6.6}$$

$$\tilde{x}_k = \sum_{l=1}^{N} h_{kl}\tilde{\theta}_l + v_k \tag{6.7}$$

and

$$\bar{x}_k = \sum_{l=1}^{N} h_{kl}\bar{\theta}_l \tag{6.8}$$

It is easy to see from the definition of $\bar{\boldsymbol{P}}^{f}_{\theta x,t}$ and $\bar{\boldsymbol{P}}^{f}_{xx,t}$ in equations (5.16) and (5.17) that they are estimates of the cross-covariance matrix of $\boldsymbol{\theta}_t$, $\boldsymbol{x}_t$ and the covariance matrix of $\boldsymbol{x}_t$ respectively, since

$$\boldsymbol{C}(\boldsymbol{\theta}^{f}_{t}, \boldsymbol{x}^{f}_{t}) = E[[\boldsymbol{\theta}^{f}_{t} - E[\boldsymbol{\theta}^{f}_{t}]][\boldsymbol{x}^{f}_{t} - E[\boldsymbol{x}^{f}_{t}]]^T] \tag{6.9}$$

$$\boldsymbol{A}(\boldsymbol{x}^{f}_{t}) = E[[\boldsymbol{x}^{f}_{t} - E[\boldsymbol{x}^{f}_{t}]][\boldsymbol{x}^{f}_{t} - E[\boldsymbol{x}^{f}_{t}]]^T] \tag{6.10}$$

where $\boldsymbol{C}(\boldsymbol{\alpha}, \boldsymbol{\beta})$, $\boldsymbol{A}(\boldsymbol{\alpha})$ denote the cross-covariance matrix of $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and covariance matrix of $\boldsymbol{\beta}$ respectively.

Consequently, as the ensemble size becomes large they would respectively converge in distribution to the cross-covariance of $\boldsymbol{\theta}^{f}_{t}$, $\boldsymbol{x}^{f}_{t}$ and the covariance of $\boldsymbol{x}^{f}_{t}$ in distribution, i.e.,

$$\lim_{N_s \to \infty} \bar{\boldsymbol{P}}^f_{\theta x,t} = \boldsymbol{C}(\boldsymbol{\theta}^f_t, \boldsymbol{x}^f_t) \tag{6.11}$$

$$\lim_{N_s \to \infty} \bar{\boldsymbol{P}}^f_{xx,t} = \boldsymbol{A}(\boldsymbol{x}^f_t) \tag{6.12}$$

Let us now compute the above quantities. As before, for simplicity of notation, we drop some of the suffixes and define $c(i,j)$ as the $(i,j)^{th}$ element of the matrix $\boldsymbol{C}(\boldsymbol{\theta}^f_t, \boldsymbol{x}^f_t)$, and $a(i,j)$ as the $(i,j)^{th}$ element of the matrix $\boldsymbol{A}(\boldsymbol{x}^f_t)$. Then,

$$c(j,k) = E[(\tilde{\theta}_j - \bar{\theta}_j)(\tilde{x}_k - \bar{x}_k)] \tag{6.13}$$

Making use of the results expressed in equations (6.6), (6.7) and (6.8) we get

$$c(j,k) = E[w_j(\tilde{x}_k - \sum_{l=1}^{N} h_{kl}\bar{\theta}_l)] \tag{6.14}$$

Using the expression for $\tilde{x}_k$ and noting that $\boldsymbol{w}_{t-1}$ and $\boldsymbol{v}_t$ are independent and hence uncorrelated,

$$\begin{aligned} c(j,k) &= E[w_j(\sum_{q=1}^{N} h_{kq}(\bar{\theta}_q + w_q) - \sum_{l=1}^{N} h_{kl}\bar{\theta}_l)] \\ &= E[\sum_{q=1}^{N} h_{kq}(w_j w_q)] \\ &= \sum_{q=1}^{N} h_{kq} E[w_j w_q] \end{aligned} \tag{6.15}$$

Thus,

$$c(j,k) = \sum_{q=1}^{N} h_{kq}(q_{jq}) \tag{6.16}$$

In terms of matrices this becomes

$$\boldsymbol{C}(\boldsymbol{\theta}^f_t, \boldsymbol{x}^f_t) = \boldsymbol{Q}_{t-1}\boldsymbol{H}^T_t \tag{6.17}$$

Let us now examine $\bar{\boldsymbol{P}}^f_{xx,t}$ and $\boldsymbol{A}(\boldsymbol{x}^f_t)$. We have,

$$a(j,k) = E[(\tilde{x}_j - \bar{x}_j)(\tilde{x}_k - \bar{x}_k)] \tag{6.18}$$

Or,

$$\begin{aligned} a(j,k) &= E[(\sum_{q=1}^{N} h_{jq}(\bar{\theta}_q + w_q) - \sum_{l=1}^{N} h_{jl}\bar{\theta}_l)(\sum_{p=1}^{N} h_{kp}(\bar{\theta}_p + w_p) - \sum_{t=1}^{N} h_{kt}\bar{\theta}_t)] \\ &\quad + E[v_j v_k] \end{aligned} \tag{6.19}$$

where we have expanded the terms involving x and made use of the fact that $\boldsymbol{w}_{t-1}$ and $\boldsymbol{v}_t$ are independent and hence uncorrelated. From the above, we get

$$\begin{aligned} a(j,k) &= E[(\sum_{q=1}^{N} h_{jq} w_q)(\sum_{p=1}^{N} h_{kp} w_p)] + E[v_j v_k] \\ &= \sum_{q=1}^{N} h_{jq}(\sum_{p=1}^{N} h_{kp} E[w_q w_p]) + E[v_j v_k] \\ &= \sum_{q=1}^{N} h_{jq}(\sum_{p=1}^{N} q_{qp} h_{kp}) + r_{jk} \end{aligned} \tag{6.20}$$

It is easy to see that in terms of matrices, this would be

$$\boldsymbol{A}(\boldsymbol{x}_t^f) = \boldsymbol{H}_t \boldsymbol{Q}_{t-1} \boldsymbol{H}_t^T + \boldsymbol{R}_t \tag{6.21}$$

Since $\bar{\boldsymbol{P}}_{\theta x,t}^f$ and $\bar{\boldsymbol{P}}_{xx,t}^f$ are estimates of $\boldsymbol{C}(\boldsymbol{\theta}_t^f, \boldsymbol{x}_t^f)$ and $\boldsymbol{A}(\boldsymbol{x}_t^f)$, from the results obtained in equations (6.17) and (6.21) we get

$$\lim_{N_s \to \infty} \bar{\boldsymbol{P}}_{\theta x,t}^f = \boldsymbol{Q}_{t-1} \boldsymbol{H}_t^T \tag{6.22}$$

$$\lim_{N_s \to \infty} \bar{\boldsymbol{P}}_{xx,t}^f = \boldsymbol{H}_t \boldsymbol{Q}_{t-1} \boldsymbol{H}_t^T + \boldsymbol{R}_t \tag{6.23}$$

We can then use the above results in equation (5.18) to get the final solution for each trajectory i of the ensemble Kalman filter technique as follows.

$$\lim_{N_s \to \infty} \hat{\boldsymbol{\theta}}_{t,EKF,N_s}^i = \boldsymbol{\theta}_t^{f_i} + (\boldsymbol{Q}_{t-1} \boldsymbol{H}_t^T)(\boldsymbol{H}_t \boldsymbol{Q}_{t-1} \boldsymbol{H}_t^T + \boldsymbol{R}_t)^{-1}(\boldsymbol{x}_t^i - \boldsymbol{H}_t \boldsymbol{\theta}_t^{f_i}) \tag{6.24}$$

The above expression resembles closely the expression for the solution in case of the discrete Kalman filter described in equations (5.5) through (5.7). It is to be noted here that the above holds only when the observation vector and the state vector are linearly related, as has been assumed for this problem.

Having obtained the above results, let us now proceed to compare the solutions arrived at by the two schemes. Let at some instant t, the ensemble estimate $\hat{\boldsymbol{\theta}}^i_{t,EKF,N_s}$ be equal to the estimate $\hat{\boldsymbol{\theta}}^i_{t,PF}$ of the particle filter for some trajectory $i \in \{1, \cdots, N_s\}$, without loss of generality. At this stage, the next step forecasts $\boldsymbol{\theta}^{f_i}_{t+1}$ are generated by adding perturbations $\boldsymbol{w}^{f_i}_t$ to $f(\hat{\boldsymbol{\theta}}^i_t)$. The conditional expectation of the solution generated by the ensemble Kalman filter at the next instant of estimation is then given by

$$\begin{aligned}
\lim_{N_s\to\infty} E[\hat{\boldsymbol{\theta}}^i_{t+1}|\hat{\boldsymbol{\theta}}^i_t] &= E[\lim_{N_s\to\infty} \hat{\boldsymbol{\theta}}^i_{t+1}|\hat{\boldsymbol{\theta}}^i_t] \\
&= E[\boldsymbol{\theta}^{f_i}_{t+1} + (\boldsymbol{Q}_t\boldsymbol{H}^T_{t+1})(\boldsymbol{H}_{t+1}\boldsymbol{Q}_t\boldsymbol{H}^T_{t+1} + \boldsymbol{R}_{t+1})^{-1}(\boldsymbol{x}^i_{t+1} - \boldsymbol{H}_{t+1}\boldsymbol{\theta}^{f_i}_{t+1})] \\
&= f(\hat{\boldsymbol{\theta}}^i_t) + (\boldsymbol{Q}_t\boldsymbol{H}^T_{t+1})(\boldsymbol{H}_{t+1}\boldsymbol{Q}_t\boldsymbol{H}^T_{t+1} + \boldsymbol{R}_{t+1})^{-1}(\boldsymbol{x}_{t+1} - \boldsymbol{H}_{t+1}f(\hat{\boldsymbol{\theta}}^i_t)) \\
&= (I - \boldsymbol{Q}_t\boldsymbol{H}^T_{t+1})(\boldsymbol{H}_{t+1}\boldsymbol{Q}_t\boldsymbol{H}^T_{t+1} + \boldsymbol{R}_{t+1})^{-1}\boldsymbol{H}_{t+1}f(\hat{\boldsymbol{\theta}}^i_t) \\
&\quad + \boldsymbol{Q}_t\boldsymbol{H}^T_1(\boldsymbol{H}_{t+1}\boldsymbol{Q}_t\boldsymbol{H}^T_{t+1} + \boldsymbol{R}_{t+1})^{-1}\boldsymbol{x}_{t+1} \qquad (6.25)
\end{aligned}$$

where we have removed the suffixes EKF, N_s for ease of notation.

In the above derivation, we have used our assumptions of uniform integrability on the elements of $\hat{\boldsymbol{\theta}}^i_{t,EKF,N_s}$, as stated in (6.1) to (6.3), and have hence been able to interchange the limit and expectation. Then, equation (6.4) has been used to obtain $E[\boldsymbol{\theta}^{f_i}_{t+1}]$.

This is the conditional expectation of the estimate of trajectory i of the ensemble Kalman filter at some time $t+1$, given its estimate at the previous instant t.

At this stage, samples for the particle filter trajectories would be drawn from a distribution which would have the following conditional expectation, obtained directly from equation (4.14).

$$\begin{aligned}
\boldsymbol{\mu}^i_{PF,t+1}|\hat{\boldsymbol{\theta}}^i_t &= (\boldsymbol{Q}^{-1}_t + \boldsymbol{H}^T_{t+1}\boldsymbol{R}^{-1}_{t+1}\boldsymbol{H}_{t+1})^{-1}(\boldsymbol{Q}^{-1}_t f(\hat{\boldsymbol{\theta}}^i_t) + \boldsymbol{H}^T_{t+1}\boldsymbol{R}^{-1}_{t+1}\boldsymbol{x}_{t+1}) \\
&= (\boldsymbol{Q}^{-1}_t + \boldsymbol{H}^T_{t+1}\boldsymbol{R}^{-1}_{t+1}\boldsymbol{H}_{t+1})^{-1}\boldsymbol{Q}^{-1}_t f(\hat{\boldsymbol{\theta}}^i_t) \\
&\quad + (\boldsymbol{Q}^{-1}_t + \boldsymbol{H}^T_{t+1}\boldsymbol{R}^{-1}_{t+1}\boldsymbol{H}_{t+1})^{-1}\boldsymbol{H}^T_{t+1}\boldsymbol{R}^{-1}_{t+1}\boldsymbol{x}_{t+1} \qquad (6.26)
\end{aligned}$$

Thus we have,

$$\lim_{N_s \to \infty} E[\hat{\boldsymbol{\theta}}_{t+1}^i | \hat{\boldsymbol{\theta}}_t^i] = (I - \boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T)(\boldsymbol{H}_{t+1} \boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T + \boldsymbol{R}_{t+1})^{-1} \boldsymbol{H}_{t+1} f(\hat{\boldsymbol{\theta}}_t^i) + \boldsymbol{Q}_t \boldsymbol{H}_1^T (\boldsymbol{H}_{t+1} \boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T + \boldsymbol{R}_{t+1})^{-1} \boldsymbol{x}_{t+1} \quad (6.27)$$

$$\boldsymbol{\mu}_{PF,t+1}^i | \hat{\boldsymbol{\theta}}_t^i = (\boldsymbol{Q}_t^{-1} + \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{H}_{t+1})^{-1} \boldsymbol{Q}_t^{-1} f(\hat{\boldsymbol{\theta}}_t^i) + (\boldsymbol{Q}_t^{-1} + \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{H}_{t+1})^{-1} \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{x}_{t+1} \quad (6.28)$$

6.3 A Relation between the Expectations of the Solutions of the Two Methods

We would now prove that, given that we have equal estimates from an ensemble Kalman filter trajectory and a particle filter trajectory at some time instant t, at the next step $t+1$, the expectation of the estimate of the ensemble Kalman filter trajectory will be equal to the mean of the distribution from which samples are drawn for the particle filter trajectory. To prove this result, we make use of the Matrix Inversion Theorem, and obtain.

$$(\boldsymbol{Q}_t^{-1} + \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{H}_{t+1})^{-1} = \boldsymbol{Q}_t - \boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T (\boldsymbol{R}_{t+1} + \boldsymbol{H}_{t+1} \boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T)^{-1} \boldsymbol{H}_{t+1} \boldsymbol{Q}_t \quad (6.29)$$

Post-multiplying the above by $\boldsymbol{Q}_t^{-1}$, we get

$$(\boldsymbol{Q}_t^{-1} + \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{H}_{t+1})^{-1} \boldsymbol{Q}_t^{-1} = \boldsymbol{I} - \boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T (\boldsymbol{R}_{t+1} + \boldsymbol{H}_{t+1} \boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T)^{-1} \boldsymbol{H}_{t+1} \quad (6.30)$$

Again,

$$(\boldsymbol{H}_{t+1} \boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T + \boldsymbol{R}_{t+1})^{-1} = \boldsymbol{R}_{t+1}^{-1} (\boldsymbol{H}_{t+1} \boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} + \boldsymbol{I})^{-1} \quad (6.31)$$

Whence, pre-multiplying both sides by $\boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T$,

$$\boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T (\boldsymbol{H}_{t+1} \boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T + \boldsymbol{R}_{t+1})^{-1} = \boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} (\boldsymbol{H}_{t+1} \boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} + \boldsymbol{I})^{-1} \quad (6.32)$$

Post-multiplying both sides by the factor $(\boldsymbol{H}_{t+1} \boldsymbol{Q}_t \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} + \boldsymbol{I})$ and re-arranging,

$$\boldsymbol{Q}_t\boldsymbol{H}_{t+1}^T\boldsymbol{R}_{t+1}^{-1} - \boldsymbol{Q}_t\boldsymbol{H}_{t+1}^T(\boldsymbol{R}_{t+1} + \boldsymbol{H}_{t+1}\boldsymbol{Q}_t\boldsymbol{H}_{t+1}^T)^{-1}(\boldsymbol{H}_{t+1}\boldsymbol{Q}_t H^T\boldsymbol{R}_{t+1}^{-1} + \boldsymbol{I}) = 0 \quad (6.33)$$

Rearranging once more, we get

$$\begin{aligned}(\boldsymbol{I} - \boldsymbol{Q}_t\boldsymbol{H}_{t+1}^T(\boldsymbol{R}_{t+1} + \boldsymbol{H}_{t+1}\boldsymbol{Q}_t\boldsymbol{H}_{t+1}^T)^{-1}\boldsymbol{H}_{t+1})\boldsymbol{Q}_t\boldsymbol{H}_{t+1}^T\boldsymbol{R}_{t+1}^{-1} \quad &= \\ \boldsymbol{Q}_t\boldsymbol{H}_{t+1}^T(\boldsymbol{H}_{t+1}\boldsymbol{Q}_t\boldsymbol{H}_{t+1}^T + \boldsymbol{R}_{t+1})^{-1} \quad &(6.34)\end{aligned}$$

It has already been shown in equation (6.30) that the first factor on the left-hand side of the last equality is the same as $(\boldsymbol{Q}_t^{-1} + \boldsymbol{H}_{t+1}^T\boldsymbol{R}_{t+1}^{-1}\boldsymbol{H}_{t+1})^{-1}\boldsymbol{Q}_t^{-1}$. Therefore, we now have

$$(\boldsymbol{Q}_t^{-1} + \boldsymbol{H}_{t+1}^T\boldsymbol{R}_{t+1}^{-1}\boldsymbol{H}_{t+1})^{-1}\boldsymbol{H}_{t+1}^T\boldsymbol{R}_{t+1}^{-1} = \boldsymbol{Q}_t\boldsymbol{H}_{t+1}^T(\boldsymbol{H}_{t+1}\boldsymbol{Q}_t\boldsymbol{H}_{t+1}^T + \boldsymbol{R}_{t+1})^{-1} \quad (6.35)$$

Combining the result obtained in (6.30) and (6.35) in the expressions (6.27) and (6.28), we finally get

$$\lim_{N_s\to\infty} E[\hat{\boldsymbol{\theta}}_{t+1,EKF,N_s}^i] = \boldsymbol{\mu}_{t+1,PF,N_s}^i \quad (6.36)$$

Thus we see that if at any step t, the i^{th} sample estimate for the ensemble Kalman filter is equal to that of the particle filter for some i, then at the next step $t+1$, the expected value of the estimate from the corresponding ensemble trajectory would be equal to the expectation of the distribution from which the next estimate of the particle filter trajectory would be drawn, as the sample size N_s goes to infinity. This result is formally stated in Theorem 6.1.

Theorem 6.1 *Let us consider the discrete time system described by equation (3.7) and equation (3.8). Let the states of the system $\boldsymbol{\theta}_t$ at any time t be estimated simultaneously by the particle filter and the ensemble Kalman filter, both of which use samples of equal size N_s.*

Let $\hat{\boldsymbol{\theta}}_{t,PF}^i$, $\hat{\boldsymbol{\theta}}_{t,EKF}^i$ denote the estimates of the i^{th} trajectories of the particle filter and the ensemble Kalman filter respectively, at instant t; and $\hat{\boldsymbol{\theta}}_{t,PF}^i \sim N(\boldsymbol{\mu}_t^i, \boldsymbol{\psi}_t)$.

Then, under the assumption of uniform integrability (as stated in assumptions 6.1 to 6.3), and as $N_s \to \infty$,

$$\hat{\boldsymbol{\theta}}_{t,PF}^i = \hat{\boldsymbol{\theta}}_{t,EKF}^i \Rightarrow E[\hat{\boldsymbol{\theta}}_{t+1,PF}^i] = \boldsymbol{\mu}_{t+1,PF,N_s}^i \quad (6.37)$$

6.4 A Relation between the Covariances of the Solutions of the Two Methods

In order to get a better understanding of how the two methods relate to each other, let us now compare the covariances of the two solutions.

Using our assumptions of uniform integrability on the elements of $\hat{\boldsymbol{\theta}}^i_{t,EKF,N_s}$, their squares and their pairwise products for any time instant t, as stated in 6.1 to 6.3, we have

$$\lim_{N_s\to\infty} Var(\hat{\boldsymbol{\theta}}^i_{t,EKF,N_s}) = Var(\lim_{N_s\to\infty} \hat{\boldsymbol{\theta}}^i_{t,EKF,N_s}) \tag{6.38}$$

The above holds because under our assumption, the limits and expectations become interchangeable for each of the terms involved in the covariance matrix, and thereby become interchangeable for the covariance itself.

Then, from the result obtained in equation (6.24), we get

$$\begin{aligned}\lim_{N_s\to\infty} Var(\hat{\boldsymbol{\theta}}^i_{t+1,EKF,N_s}) = Var(\boldsymbol{\theta}^{f_i}_{t+1} + (\boldsymbol{Q}_t\boldsymbol{H}^T_{t+1})(\boldsymbol{H}_{t+1}\boldsymbol{Q}_t\boldsymbol{H}^T_{t+1} + \\ \boldsymbol{R}_{t+1})^{-1}(\boldsymbol{x}^i_{t+1} - \boldsymbol{H}_{t+1}\boldsymbol{\theta}^{f_i}_{t+1}))\end{aligned} \tag{6.39}$$

It is known that for a given ensemble, the $\boldsymbol{x}^i_{t+1}$'s represent the perturbed measurements. As the process of measurement is independent of the actual process that generates these $\boldsymbol{x}^i_{t+1}$'s, the terms $\boldsymbol{x}_{t+1}$ and $\boldsymbol{\theta}^f_{t+1}$ in equation (6.39) are independent, and hence uncorrelated. The above equation can then be simplified to

$$\begin{aligned}\lim_{N_s\to\infty} Var(\hat{\boldsymbol{\theta}}^i_{t+1,EKF,N_s}) &= Var(I - (\boldsymbol{Q}_t\boldsymbol{H}^T_{t+1})(\boldsymbol{H}_{t+1}\boldsymbol{Q}_t\boldsymbol{H}^T_{t+1} + \boldsymbol{R}_{t+1})^{-1}\boldsymbol{H}_{t+1}\boldsymbol{\theta}^{f_i}_{t+1}) \\ &+ Var((\boldsymbol{Q}_t\boldsymbol{H}^T_{t+1})(\boldsymbol{H}_{t+1}\boldsymbol{Q}_t\boldsymbol{H}^T_{t+1} + \boldsymbol{R}_{t+1})^{-1}\boldsymbol{x}^i_{t+1})\end{aligned} \tag{6.40}$$

We can replace the co-efficient matrices in the above using the results derived in (6.30) and (6.34), whence we get the following.

$$\begin{aligned}\lim_{N_s\to\infty} Var(\hat{\boldsymbol{\theta}}^i_{t+1,EKF,N_s}) &= Var((\boldsymbol{Q}_t^{-1} + \boldsymbol{H}^T_{t+1}\boldsymbol{R}^{-1}_{t+1}\boldsymbol{H}_{t+1})^{-1}\boldsymbol{Q}_t^{-1}\boldsymbol{\theta}^{f_i}_{t+1}) \\ &+ Var((\boldsymbol{Q}_t^{-1} + \boldsymbol{H}^T_{t+1}\boldsymbol{R}^{-1}_{t+1}\boldsymbol{H}_{t+1})^{-1} \\ &\quad \boldsymbol{H}^T_{t+1}\boldsymbol{R}^{-1}_{t+1}\boldsymbol{x}^i_{t+1})\end{aligned} \tag{6.41}$$

Let us denote $(\boldsymbol{Q}_t^{-1} + \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{H}_{t+1})^{-1}$ by $\boldsymbol{B}_t$ for the ease of notations. Since both $\boldsymbol{R}_{t+1}$ and $\boldsymbol{Q}_{t+1}$ are covariance matrices of some variable, they must be symmetric, and therefore, the matrix $\boldsymbol{B}_t$, too is symmetric, given its definition. Then, the covariance relation above reduces to

$$\lim_{N_s \to \infty} Var(\hat{\boldsymbol{\theta}}^i_{t+1,EKF,N_s}) = Var(\boldsymbol{B}_t \boldsymbol{Q}_t^{-1} \boldsymbol{\theta}^{f_i}_{t+1}) + Var(\boldsymbol{B}_t \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{x}^i_{t+1}) \tag{6.42}$$

Or,

$$\begin{aligned} \lim_{N_s \to \infty} Var(\hat{\boldsymbol{\theta}}^i_{t+1,EKF,N_s}) = \boldsymbol{B}_t Var(\boldsymbol{Q}_t^{-1} \boldsymbol{\theta}^{f_i}_{t+1}) \boldsymbol{B}_t^T \\ + \boldsymbol{B}_t Var(\boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{x}^i_{t+1}) \boldsymbol{B}_t^T \end{aligned} \tag{6.43}$$

Because $\boldsymbol{B}_t$ is symmetric, $\boldsymbol{B}_t = \boldsymbol{B}_t^T$, and hence,

$$\begin{aligned} \lim_{N_s \to \infty} Var(\hat{\boldsymbol{\theta}}^i_{t+1,EKF,N_s}) &= \boldsymbol{B}_t Var(\boldsymbol{Q}_t^{-1} \boldsymbol{\theta}^{f_i}_{t+1}) \boldsymbol{B}_t + \boldsymbol{B}_t Var(\boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{x}^i_{t+1}) \boldsymbol{B}_t \\ &= \boldsymbol{B}_t (Var(\boldsymbol{Q}_t^{-1} \boldsymbol{\theta}^{f_i}_{t+1}) \boldsymbol{B}_t + Var(\boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{x}^i_{t+1}) \boldsymbol{B}_t) \\ &= \boldsymbol{B}_t (Var(\boldsymbol{Q}_t^{-1} \boldsymbol{\theta}^{f_i}_{t+1}) + Var(\boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{x}^i_{t+1})) \boldsymbol{B}_t \end{aligned} \tag{6.44}$$

At this stage, we note that $\boldsymbol{\theta}^{f_i}_{t+1}$ and $\boldsymbol{x}^i_{t+1}$ are generated by adding zero-mean noises of covariance matrices $\boldsymbol{Q}_t$ and $\boldsymbol{R}_{t+1}$ to $f(\hat{\boldsymbol{\theta}}_t)$ and $\boldsymbol{x}_{t+1}$ respectively. Since $f(\hat{\boldsymbol{\theta}}_t)$ and $\boldsymbol{x}_{t+1}$ are already known,

$$Var(\boldsymbol{\theta}^{f_i}_{t+1}) = \boldsymbol{Q}_t \tag{6.45}$$

$$Var(\boldsymbol{x}^i_{t+1}) = \boldsymbol{R}_{t+1} \tag{6.46}$$

Using these results, we get

$$\begin{aligned} \lim_{N_s \to \infty} Var(\hat{\boldsymbol{\theta}}^i_{t+1,EKF,N_s}) &= \boldsymbol{B}_t (\boldsymbol{Q}_t^{-1} \boldsymbol{Q}_t \boldsymbol{Q}_t^{-1} + \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{R}_{t+1} \boldsymbol{R}_{t+1}^{-1} \boldsymbol{H}_{t+1}) \boldsymbol{B}_t \\ &= \boldsymbol{B}_t (\boldsymbol{Q}_t^{-1} + \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{H}_{t+1}) \boldsymbol{B}_t \\ &= \boldsymbol{B}_t \boldsymbol{B}_t^{-1} \boldsymbol{B}_t \\ &= \boldsymbol{B}_t \\ &= (\boldsymbol{Q}_t^{-1} + \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{H}_{t+1})^{-1} \end{aligned} \tag{6.47}$$

But by definition $(\boldsymbol{Q}_t^{-1} + \boldsymbol{H}_{t+1}^T \boldsymbol{R}_{t+1}^{-1} \boldsymbol{H}_{t+1})^{-1} = \boldsymbol{\psi}_{t+1}$ is the covariance of the sampling distributions of the particle filter at the step $t+1$. This means that if at any step t, the i^{th} sample estimate for the ensemble Kalman filter is equal to that of the particle filter for some i, then at the next step $t+1$, the covariance of the estimate from the corresponding ensemble trajectory would be equal to the covariance of the distributions from which the next estimate of the particle filter trajectories are drawn, as the sample size N_s goes to infinity.

A formal statement of this result is given in Theorem 6.2.

Theorem 6.2 *Let us consider the discrete time system described by equation (3.7) and equation (3.8). Let the states of the system $\boldsymbol{\theta}_t$ at any time t be estimated simultaneously by the particle filter and the ensemble Kalman filter, both of which use samples of equal size N_s.*

Let $\hat{\boldsymbol{\theta}}_{t,PF}^i$, $\hat{\boldsymbol{\theta}}_{t,EKF}^i$ denote the estimates of the i^{th} trajectories of the particle filter and the ensemble Kalman filter respectively, at instant t; and $\hat{\boldsymbol{\theta}}_{t,PF}^i \sim N(\boldsymbol{\mu}_t^i, \boldsymbol{\psi}_t)$.

Then, under the assumption of uniform integrability (as stated in assumptions 6.1 to 6.3), and as $N_s \to \infty$,

$$\hat{\boldsymbol{\theta}}_{t,PF}^i = \hat{\boldsymbol{\theta}}_{t,EKF}^i \Rightarrow Var[\hat{\boldsymbol{\theta}}_{t+1,EKF,N_s}^i] = \boldsymbol{\psi}_{t+1,PF,N_s} \tag{6.48}$$

6.5 General Remarks

Our results indicate that when the relation between the states and observations is linear and the noises are zero-mean Gaussian with known covariance matrices, then as the sample size goes to infinity, the ensemble Kalman filter method is effectively equivalent to an approximation of the particle filter technique without the final step involving the importance weights. In more precise terms, if at some time t, some trajectory of the ensemble Kalman filter produces an estimate equal to that produced by a particle filter trajectory, then the expectation and covariance of the next step estimates of these two trajectories would be equal. If at some time t the ensemble Kalman filter trajectories have estimates each individually equal to those of the particle filter trajectories; the estimates of the corresponding trajectories of the two filters at the next step $t+1$ would have the same expectation and covariance.

Since the distributions in question are Gaussian the expectation and the covariance are sufficient to completely specify the distributions. This means that for the given model, methodologically, the ensemble Kalman filter closely mimics the particle filter and approaches the latter as the sample size goes to infinity. The procedure to generate updated ensemble estimates is an approximation of drawing

samples for the particle filter trajectories. The two, then, only differ in the final results because one of the two employs importance weights.

When the ensemble size is sufficiently large, the ensemble Kalman filter trajectories would closely approach the particle filter trajectories. Under such large sample sizes, if the particle filter algorithm does not perform the final step, and instead just computes an arithmetic mean of the samples drawn, it would effectively be almost the same as the ensemble Kalman filter. However, taking a weighted average of the sampled points at the end step instead of a simple average ensures that trajectories that are more likely to produce the observed data are given a greater importance. This final step indicates that for the given model, the particle filter would provide better estimates than the ensemble Kalman filter; and can be thus seen as a more precise version of the latter.

Chapter 7

Simulation Results

In this chapter, we would demonstrate and discuss a few simulation results where the states of the same system were simultaneously estimated by the particle filter and the ensemble Kalman filter. These results are consistent with our theoretically established result, where it was shown that the ensemble Kalman filter trajectories tend to follow the particle filter trajectories when the ensemble size is sufficiently large.

We consider a system where the state variable $\boldsymbol{\theta}_t$ is 3-dimensional and the observation vector $\boldsymbol{x}_t$ is 2-dimensional, i.e., in the notations used, $N_\theta = N = 3$ and $N_x = M = 2$. The evolution dynamics of the state variables of the system under consideration is given by the following set of equations.

$$\begin{bmatrix} \theta_{t+1,1} \\ \theta_{t+1,2} \\ \theta_{t+1,3} \end{bmatrix} = \begin{bmatrix} \theta_{t,1} + cos(\theta_{t,2}) + 1.2sin(\theta_{t,3}) \\ 0.5(\theta_{t,1} + \theta_{t,2}) \\ 0.8\theta_{t,3} + 0.5 \end{bmatrix} + \begin{bmatrix} w_{t,1} \\ w_{t,2} \\ w_{t,3} \end{bmatrix} \tag{7.1}$$

The noise covariance matrices $\boldsymbol{Q}_t$ and $\boldsymbol{R}_t$ and the matrix $\boldsymbol{H}_t$ that relates the observed data with the state realisations are assumed to be constant over time and are hence denoted by $\boldsymbol{Q}$, $\boldsymbol{R}$ and $\boldsymbol{H}$ respectively. They are defined as follows.

$$\boldsymbol{Q} = \begin{bmatrix} 1 & 0.5 & 0.2 \\ 0.5 & 0.8 & 0.4 \\ 0.2 & 0.4 & 0.9 \end{bmatrix}$$

$$\boldsymbol{R} = \begin{bmatrix} 1 & 0.6 \\ 0.6 & 0.9 \end{bmatrix}$$

$$\boldsymbol{H} = \begin{bmatrix} 4 & 5 & 1 \\ 3 & 7 & 2 \end{bmatrix}$$

Finally, the states are initialized at the following point:

$$\begin{bmatrix} \theta_{0,1} \\ \theta_{0,2} \\ \theta_{0,3} \end{bmatrix} = \begin{bmatrix} 12 \\ 10 \\ 16 \end{bmatrix}$$

With the system described as above we proceed to simulate its state evolution on MATLAB, for 400 observations, and we use both the particle filter and the ensemble Kalman filter simultaneously to estimate the state variables at every instant, using the above information and the measured observations. For each simulation run, the same sample size N_s is chosen for both the filters to allow a comparison of their performances. We take samples of size 5, 20, 50, 100 and 200. First, estimation error patterns on individual simulation runs are considered, followed by the average errors computed over 25 simulations.

The errors are obtained and plotted against time for both filters and for each of the above sample sizes. These plots are given below (Figures 7.1 to 7.8). In each of these figures, the three lines represent the error accumulated along the three dimensions of θ_t. These errors are the exact errors accumulated on a particular realisation of the state variables and not average errors.

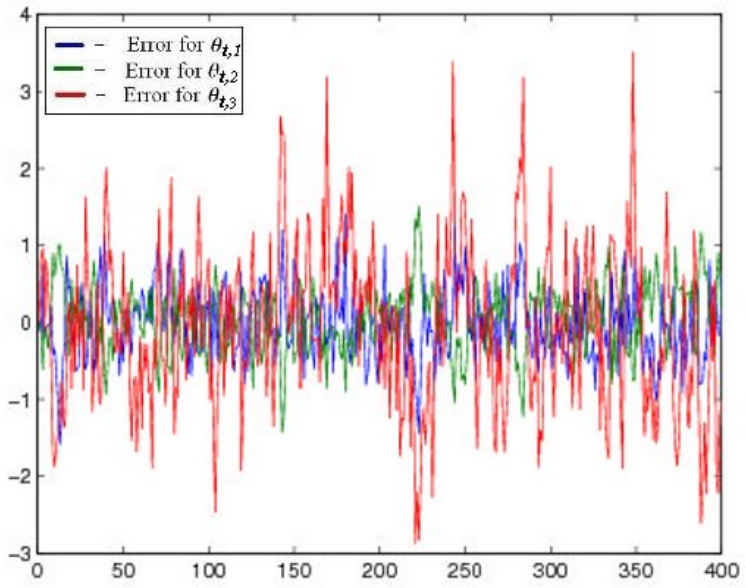

Figure 7.1: $N_s = 5$, error versus time plot for particle filter

While the analysis in the previous chapter lead us to conclusions regarding the relation among the particle filter and Ensemble Kalman filter trajectories, we get to demonstrate the general pattern of their performances via these simulations. It

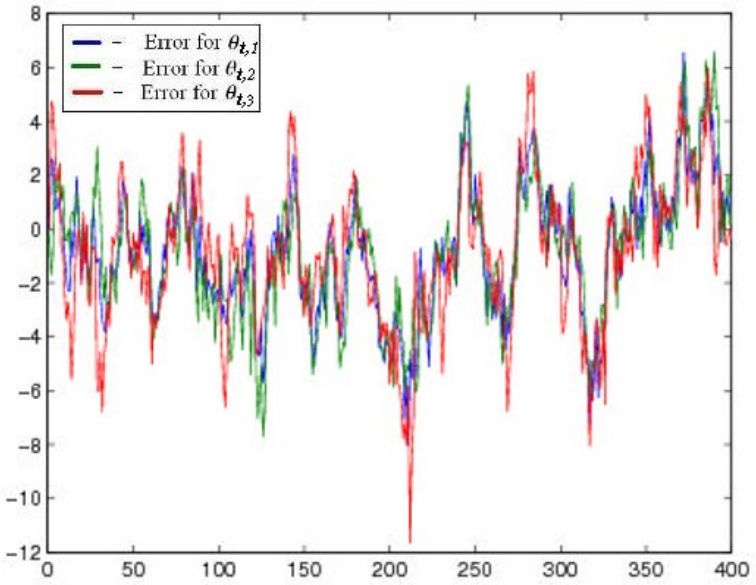

Figure 7.2: $N_s = 5$, error versus time plot for ensemble Kalman filter

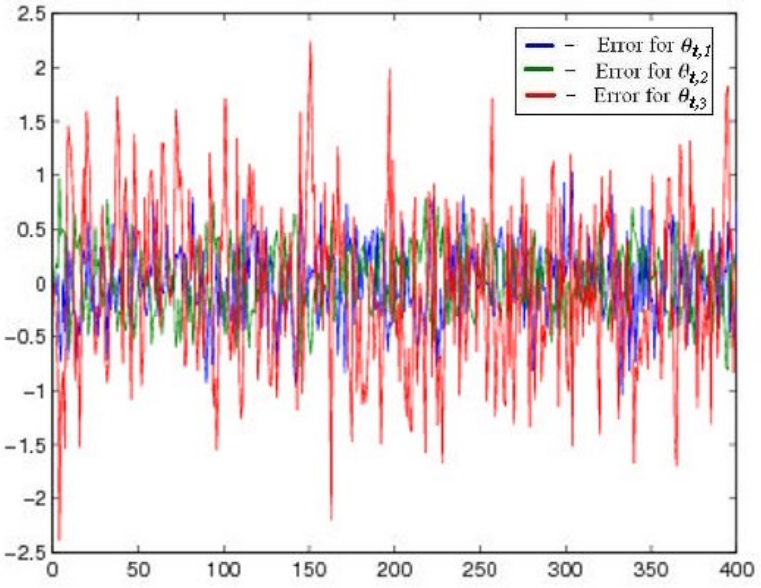

Figure 7.3: $N_s = 20$, error versus time plot for particle filter

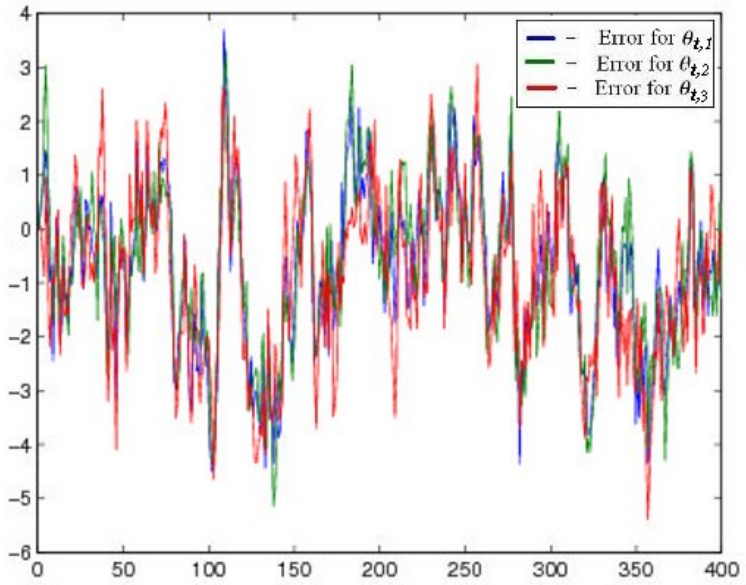

Figure 7.4: $N_s = 20$, error versus time plot for ensemble Kalman filter

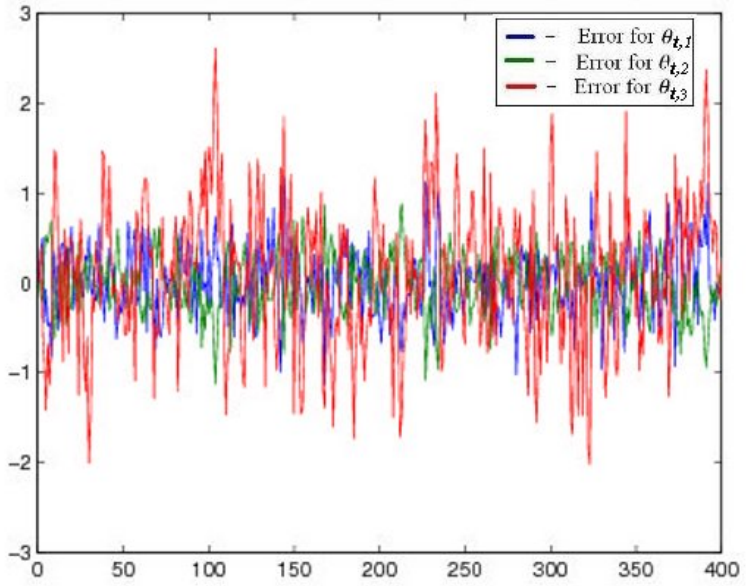

Figure 7.5: $N_s = 50$, error versus time plot for particle filter

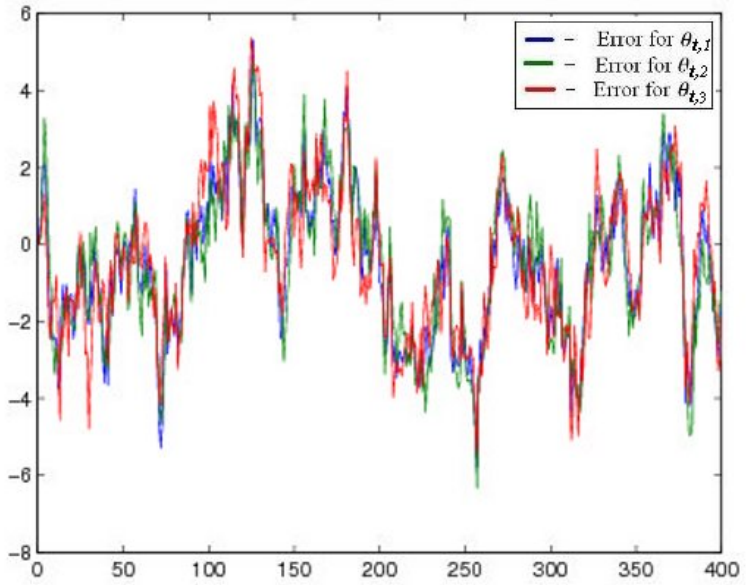

Figure 7.6: $N_s = 50$, error versus time plot for ensemble Kalman filter

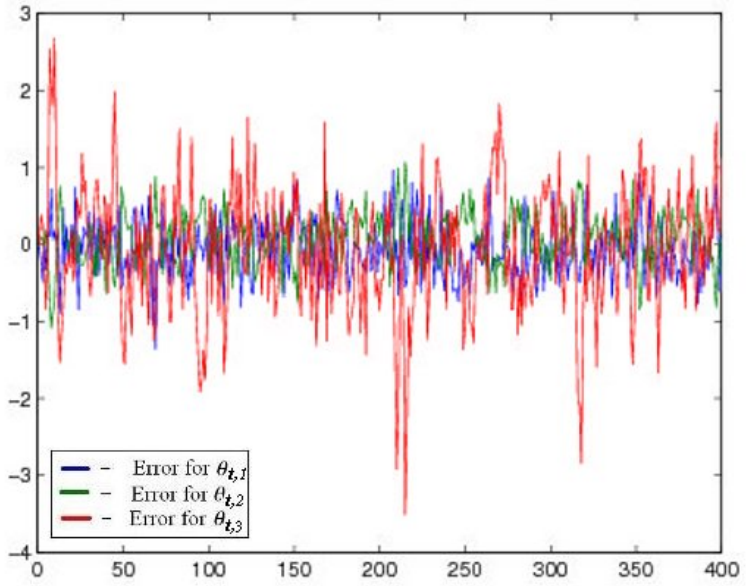

Figure 7.7: $N_s = 100$, error versus time plot for particle filter

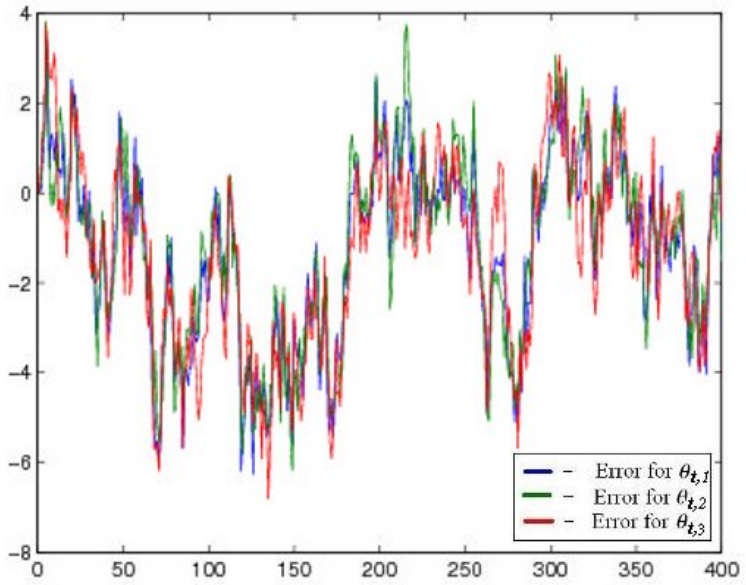

Figure 7.8: $N_s = 100$, error versus time plot for ensemble Kalman filter

is seen that the ensemble Kalman filter has a relatively slower response compared to the particle filter. The former accumulates a considerable error before correcting itself and getting back to the right track, unlike the particle filter. One reason for this is the fact that the particle filter associates importance weights with each trajectories, thereby associating a higher importance to those realisations that are more likely to produce the recorded measurements. Another reason is that while the ensemble Kalman filter uses estimates of the covariance quantities, the particle filter uses directly the values of $\boldsymbol{Q}$ and $\boldsymbol{R}$ to obtain the required parameters of its sampling distributions. Unless the sample size is significantly large, the quantities $\bar{\boldsymbol{P}}^f_{\theta x,t}$ and $\bar{\boldsymbol{P}}^f_{xx,t}$ would not match the true values of their corresponding covariance terms closely enough; and hence the ensemble Kalman filter estimates would not be sufficiently accurate.

Let us now present the average error curves for the same ensemble sizes, each calculated over 25 simulation runs. These plots are given in figures 7.9 to 7.16.

It is seen that, on an average, the particle filter and the ensemble Kalman filter give very similar results. Especially, for $N_s = 100$, the average error plots for the two filters seem to be almost identical, for the same realisation of $\boldsymbol{\theta}_t$. This observation is in unison with our theoretical conclusion, which implied that the two would produce very close results when the sample size is large enough.

The mean error values are tabulated in Table 7.1 and they are seen to be remarkably close.

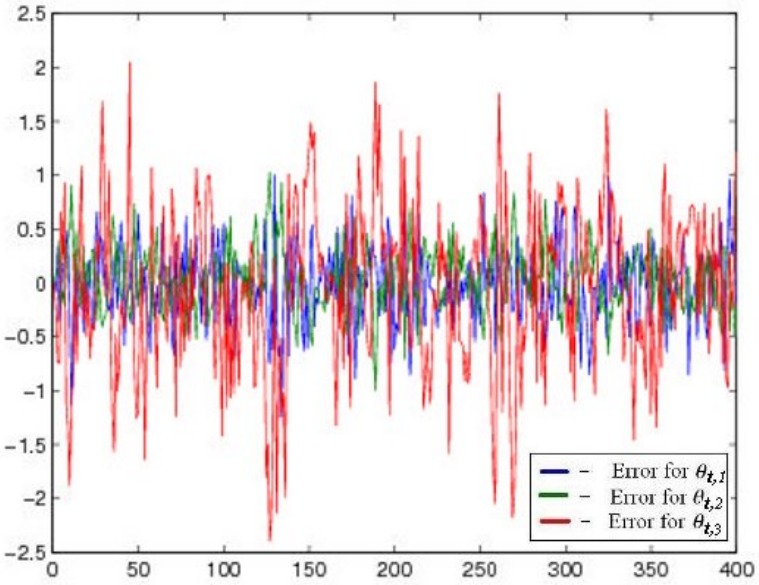

Figure 7.9: $N_s = 5$, average error versus time plot for particle filter

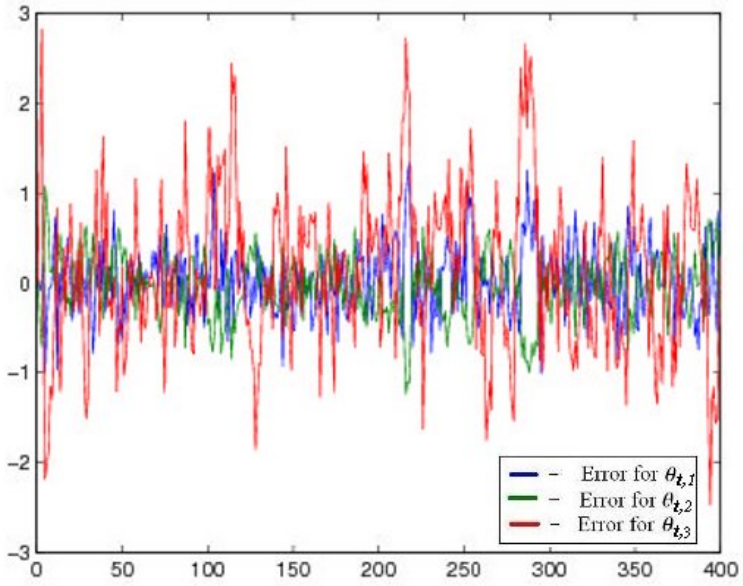

Figure 7.10: $N_s = 5$, average error versus time plot for ensemble Kalman filter

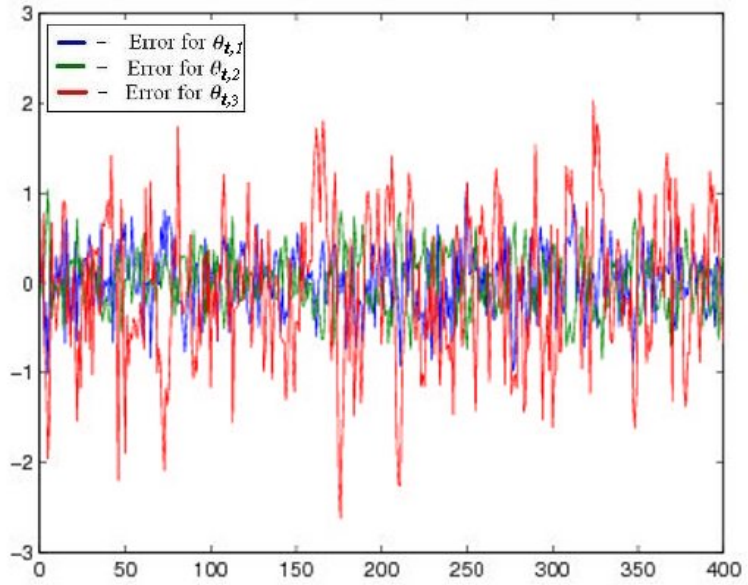

Figure 7.11: $N_s = 20$, average error versus time plot for particle filter

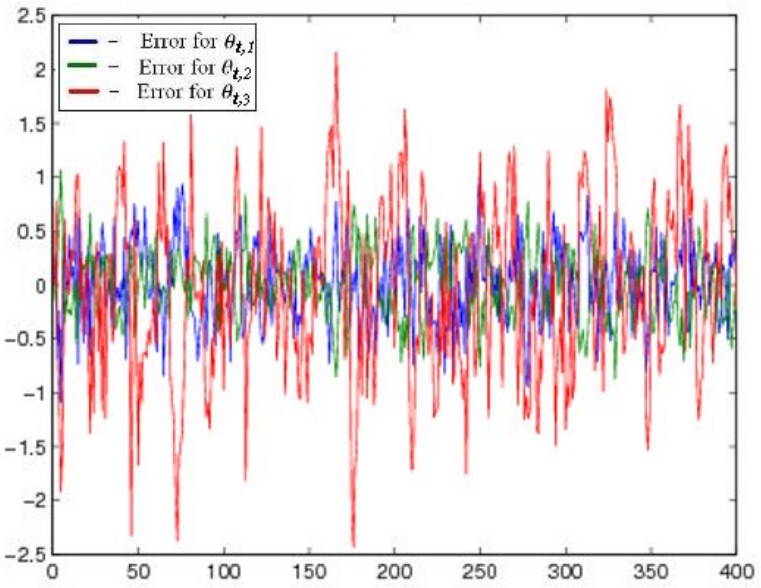

Figure 7.12: $N_s = 20$, average error versus time plot for ensemble Kalman filter

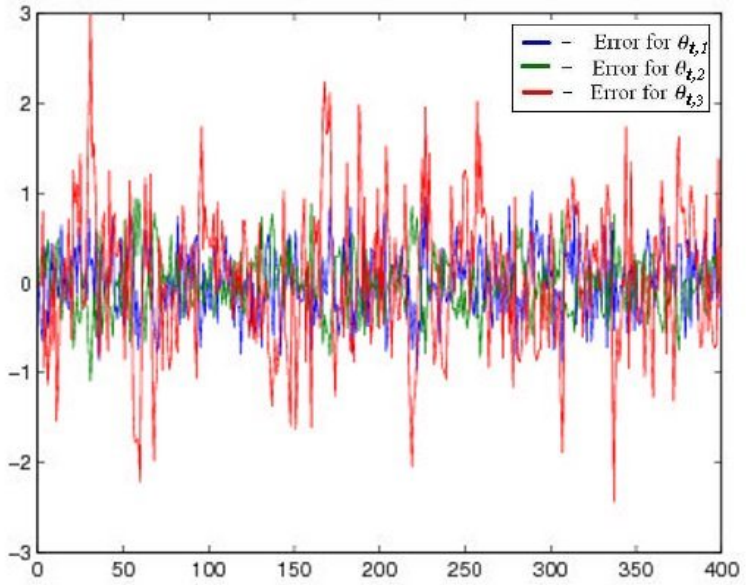

Figure 7.13: $N_s = 50$, average error versus time plot for particle filter

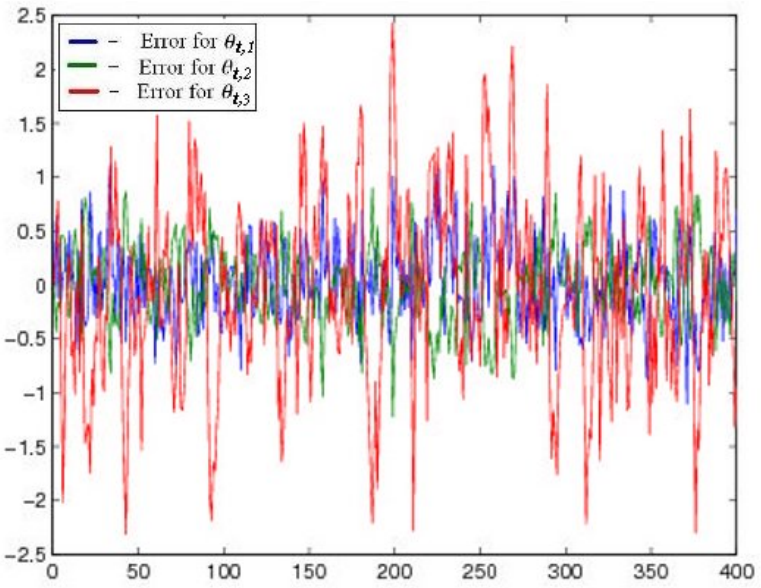

Figure 7.14: $N_s = 50$, average error versus time plot for ensemble Kalman filter

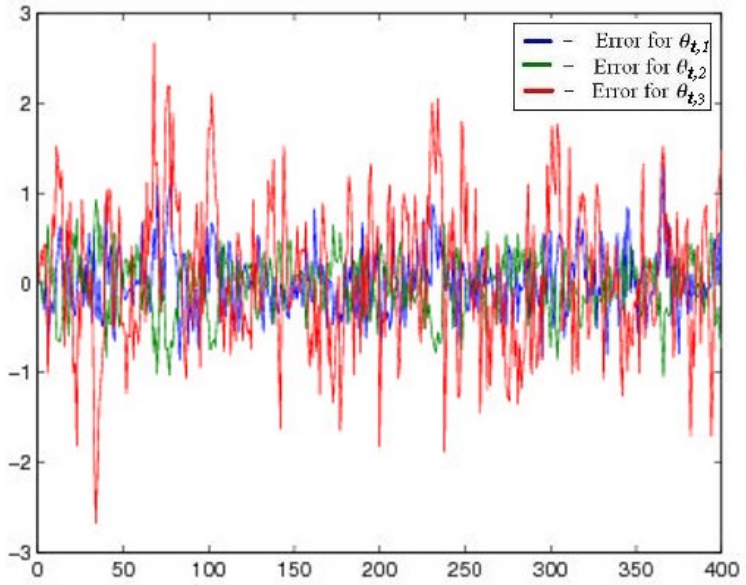

Figure 7.15: $N_s = 100$, average error versus time plot for particle filter

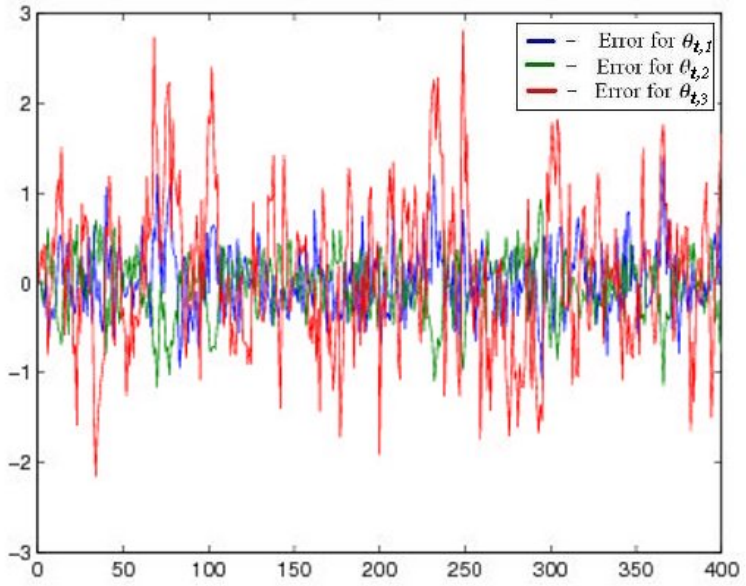

Figure 7.16: $N_s = 100$, average error versus time plot for ensemble Kalman filter

Table 7.1: Average estimation errors for the particle filter and the ensemble Kalman filter for different sample sizes

Average Errors for the particle filter and the ensemble Kalman filter		
Sample Size N_s	Average error for particle filter	Average error for ensemble Kalman filter
5	-0.0281	-0.0209
	-0.0336	-0.0307
	-0.0543	-0.0591
20	0.0117	0.0228
	0.0076	-0.0058
	-0.0795	-0.0505
50	0.0058	0.0217
	-0.0158	-0.0341
	-0.0462	-0.0862
100	0.0209	0.0263
	-0.0401	-0.0476
	0.0856	0.1056

Chapter 8

Conclusion and Future Work

Having described the functioning principles of the two filters and having obtained a relation between them, we would now attempt to interpret these results. In this work we have presented an analytical comparison of the particle filter and the ensemble Kalman filter and explored their inter relations and similarities in the context of estimating a variable from a set of noisy measurements. It has been shown that for a non-linear evolution model of the hidden Markov chain that generates the variable of interest, a linear relation between the states and the observations and a Gaussian setting; the two methods are closely related to each other, when the sample size is sufficiently large. More specifically, if the estimate of an ensemble trajectory is equal to the estimate of a particle trajectory at any step, then the expectations and covariance of the estimate of that ensemble trajectory at the next step would be equal to those of the sampling population from which the next estimate of that particle trajectory would be drawn. In simpler words, the trajectories in the two filters evolve the same way for a large sample size.

Essentially, thus, when the ensemble size is sufficiently large, the two methods would yield a similar set of trajectories; or more precisely, would generate the trajectories in analogous ways, when one starts with the same initial point(s). However, because of the inclusion of the weighted averaging scheme at the final stage of the particle filter, its final result is likely to be closer to the true values of the states as compared to that obtained from its counterpart. Since the ensemble Kalman filter attributes an equal weight to each of the estimates of the ensemble while computing the final estimate; the problem of degeneracy would never occur there, and consequently there would be no need to resample.

It is important to note that the result is based mainly on the convergence of the ensemble estimates of the covariance quantities to their respective true values. Firstly, these results hold under the assumption of uniform integrability as stated in (6.1) to (6.3). Secondly, these convergences occur as the ensemble size goes to

infinity. It would be an interesting problem to determine how large the ensemble size needs to be to ensure that the estimates closely follow the actual values of the corresponding terms. In general, the performance of the ensemble Kalman filter should improve as the ensemble size is increased. Again, when the relation between $\boldsymbol{\theta}_t$ and $\boldsymbol{x}_t$ is of a linear nature, and when the dimension of the system is not too high, it would be useful to directly calculate the actual values of $\boldsymbol{C}(\boldsymbol{\theta}_t^f, \boldsymbol{x}_t^f)$ and $\boldsymbol{A}(\boldsymbol{x}_t^f)$ in terms of the known quantities $\boldsymbol{Q}_t$, $\boldsymbol{R}_t$ and $\boldsymbol{H}_t$ using equations (6.17) and (6.21), instead of approximating them from the ensemble members.

Since the particle filter uses all the available information to draw samples instead of making prior forecasts for the trajectories and then updating them, the sampling step in the particle filter is basically equivalent to the aggregate of all the steps involved in the ensemble Kalman filter. Again, since both methods are attempts to solve the same integrals described earlier in chapter 3, the results are likely to be close when the problem dynamics are relatively well-defined, as was the case here. The particle filter, by using actual covariance quantities to generate means and covariances of the sampling distributions instead of approximating them from the sample, is in fact likely to provide a better result even if the final step involving importance weights is not used. Finally, by assigning importance weights on the trajectories according to their probability of closeness to reality, the particle filter prefers to choose the relatively 'better' trajectories, thereby further optimizing its solution.

An interesting consequence of this result is that this can be used to modify both the particle filter and the ensemble Kalman filter method, thereby reducing their computational burden, when the observations are linearly related to the state variables and the noises are Gaussian. It has been shown that the generation of ensemble members is methodologically an approximation of drawing samples for the particle filter under the said scenario. Depending on the dimensions and general nature of the system, it might be easier to draw samples from a priori populations or compute estimates using the ensemble Kalman gains. For instance, when the system dimension is significantly high, instead of inverting large matrices involved in the computations for the particle filter, it might be beneficial to use the ensemble Kalman filter approximations for the covariance terms. On the other hand, for low dimensional models, where matrix inversions would not be too critical an issue, trajectories should preferably be generated using the sequential importance sampling methods as used for the particle filter.

Since both methods would give approximately similar trajectories, one can make the two algorithms complement each other. That is, the particle filter may also use the ensemble members of the ensemble Kalman filter as its own trajectories and then compute the final estimate using the weighted averaging method. In general, though, since the particle filter can give at least as good a result as the ensemble

Kalman filter right after drawing the samples, it would be more efficient compared to its counterpart, and should therefore be used even if the final steps of importance weight and resampling are replaced by simple averaging to save computational burden. At all point, it is also to be remembered that the trajectories of the ensemble Kalman filter would approach the particle filter trajectories only when the ensemble size is sufficiently large and therefore complementary use of these schemes should be done with caution.

In this work a first step analysis of the two recursive algorithms has been done, i.e., we have established some relations between two trajectories of the two filters at some instant $t+1$ when it was given that the two corresponding estimates were equal at the previous instant t. It would be of interest to explore how the trajectories of the two filters relate to each other in the long run. It is intuitive that they should still be sufficiently close to each other, given that both are sub-optimal solutions for the problem chosen and are both based on similar principles; but it still would be an interesting problem to examine how far the trajectories would drift apart from each other as time goes by. Specifically, given that the estimate of a particle filter trajectory and that of an ensemble Kalman filter trajectory were equal at some instant t, we would like to find out how the two trajectories relate at a future instant $t + T$.

The result obtained in this study, namely the strong correspondence between the two methods, is only valid for a very restricted scenario, i.e., when the observations are linearly related to the states and the noises involved with the state evolution and observations are both zero-mean with known covariance matrices. However, in reality, these two methods are most extensively used in non-linear and non-Gaussian environments, where the solutions would take more involved forms for both the filters, and in general, it would not be easy to derive such simple yet strong relations. It would, nevertheless, be interesting to explore the relation between these two filters for a general case where there is no restriction on either the observation equation or the noises involved, or even for less relaxed system considerations; for instance when either the dynamics are Gaussian but entirely non-linear or when the dynamics are linear but non-Gaussian.

In a more general scenario, there would still be similarities owing to the fact that both are derived from similar first principles, but the two filters may not have as strong a correlation as was the case for the particular model chosen in this study. It would be of interest to explore their inter relations in such a generalized framework. It is intuitive that the particle filter would give more accurate results and a faster convergence under such a framework, given its generic nature as compared to the ensemble Kalman filter. The latter is an approximate model and only uses the first and second moments of the variables involved; a strategy that would make its accuracy doubtful in environments that are highly non-Gaussian. However, in

certain cases, it may still be chosen over the former because it may in fact involve lesser computational complexity.

Another interesting direction for future research can be to compare some of the other sub-optimal methods under different conditions with the particle filter or the ensemble Kalman filter. For instance, one might compare the results derived from the extended Kalman filter under a Gaussian assumption with those obtained from either of the two filters studied in this thesis. The grid-based methods may also be considered for comparison. Such comparative studies and analytical and experimental results derived from them would be helpful to determine the most efficient filter for a given problem under specific priorities.

APPENDIX

The MATLAB source code used for simulation has been presented here.

```
%Simulation of the estimation of the states of a system using the
%Particle Filter and the Ensemble Kalman Filter Algorithm

N = 400; %Defining number of data points to be simulated
Ns = 100; %Defining ensemble size/ particle size

%Initialization of the variables that would contain the state
%realizations and the observations

theta = zeros(3,N);
obs = zeros(2,N);

%Initializion of all variables used in the Particle Filter
%algorithm, including importance weights, means and variances of
% sampling distributions, variables involved in the development
%of trajectories and the final estimates

sum_w = 0;
var1 = zeros(3);
p=zeros(N,Ns);
mu = zeros(3,N,Ns);
w = (1/Ns)*ones(N,Ns);
w_new  = (1/Ns)*ones(N,Ns);
theta_hat = zeros(3,N,Ns);
f_theta_hat = zeros(3,N,Ns);
theta_hat_new = zeros(3,Ns);

theta_pf = zeros(3,N);

%Defining threshold/ minimum effective sample size, below which
%resampling will be done

N_min = 0.6*Ns;
%Initializion of all variables used in the Ensemble Kalman Filter
%algorithm, including the predicted ensemble values for the state
```

```
%and the observation set, variables involved in the development of
%trajectories and the final estimate

theta_hat1 = zeros(3,N,Ns);
theta_hat_f = zeros(3,N,Ns);
f_theta_hat1 = zeros(3,N,Ns);
obs_f = zeros(2,Ns);
P_theta_obs = zeros(3,2);
P_obs_obs = zeros(2,2);
E_theta= zeros(3,Ns);
E_obs = zeros(2,Ns);

theta_ekf = zeros(3,N);



%Simulation of System Dynamics and Observations

%Setting the initial conditions for all relevant variables including
%theta, the state variable and the variables that store the estimates

theta(:,1)= [12;10;16];
theta_hat_new(:,1)= [12;10;16];

for j= 1:Ns
f_theta_hat(:,1,j) = [12;10;16];
theta_hat(:,1,j) = [12;10;16];
theta_hat1(:,1,j) = [12;10;16];
theta_hat_f(:,1,j) = [12;10;16];
end

theta_pf(:,1)= [12;10;16];
theta_ekf(:,1)= [12;10;16];

%Defining the system parameters: H,Q and R
H = [4 5 1; 3 7 2];
Q = [1.0 0.5 0.2; 0.5 0.8 0.4; 0.2 0.4 0.9];
R= [1.0 0.6; 0.6 0.9];

%Evolution of the state dynamics using the state transition equation
%f(theta_n+1)=f(theta_n)+w_n

Q_chol = chol(Q);
for i = 1:N-1
   theta(:,i+1)= [theta(1,i)+cos(theta(2,i))+ 1.2*sin(theta(3,i));
   0.5*(theta(1,i)+theta(2,i));
   0.8*theta(3,i)+0.5;] + (randn(1,3)*Q_chol)';
end
```

```
%Simulation of observations using the equation x_n = H*theta_n + v_n

R_chol = chol(R);
for i=1:N
obs(:,i) = H*theta(:,i) + (randn(1,2)*R_chol)';
end


%Particle Filter Algorithm

var1 = inv(inv(Q)+H'*inv(R)*H);
sigma = R+H*Q*H';

for i = 1:N-1

        sum_w=0;

%Defining mu, the mean of the sampling distribution

        for j= 1:Ns

   f_theta_hat(1,i+1,j)=theta_hat(1,i,j)+cos(theta_hat(2,i,j))
                      + 1.2*sin(theta_hat(3,i,j));
   f_theta_hat(2,i+1,j)=0.5*(theta_hat(1,i,j)+theta_hat(2,i,j));
   f_theta_hat(3,i+1,j)=0.8*theta_hat(3,i,j)+0.5;

   mu(:,i+1,j) = var1*(inv(Q)*f_theta_hat(:,i+1,j)
         + H'*inv(R)*obs(:,i+1));

%Drawing samples from the importance distribution

    theta_hat(:,i+1,j)=  mu(:,i+1,j)+ (randn(1,3)*chol(var1))';

%Determination of importance weights using the relevant expression
  p(i+1,j)= (2*pi)^(-1*Ns/2)*sqrt(det(sigma))*exp(-0.5*(obs(:,i+1)
          -H*f_theta_hat(:,i+1,j))'*inv(sigma)*(obs(:,i+1)
          -H*f_theta_hat(:,i+1,j)));
        w(i+1,j)= p(i+1,j)*w_new(i,j);
        sum_w=sum_w+w(i+1,j);
        end

%Normalization of importance weights
        for j=1:Ns
            w_new(i+1,j)=w(i+1,j)/sum_w;
        end

%Provision for Resampling when N_effective < N_minimum

            if 1/(w_new(i+1,:)*(w_new(i+1,:))')< N_min
```

```
                index = discretesample(w_new(i+1,:),Ns);
                for j=1:Ns
                    theta_hat_new(:,j) = theta_hat(:,i+1,index(j));
                    w_new(i+1,j)=1/Ns;
                end
                  theta_hat(:,i+1,:) = theta_hat_new;
            end

%Final estimate: result obtaine as weighted sum of the different
%estimates on the different trajectories

  for j=1:Ns
      theta_pf(:,i+1)=theta_pf(:,i+1)+w_new(i+1,j)*theta_hat(:,i+1,j);
  end
end


%Ensemble Kalman Filter Algorithm
for i = 1:N-1
        obs_recent = obs(:,i+1);

%Prediction/Forecast of ensemble members for the state and the
%observations for the next instant

 for j= 1:Ns
     f_theta_hat1(1,i+1,j)= theta_hat1(1,i,j)+cos(theta_hat1(2,i,j))
                + 1.2*sin(theta_hat1(3,i,j));
     f_theta_hat1(2,i+1,j)= 0.5*(theta_hat1(1,i,j)+theta_hat1(2,i,j));
     f_theta_hat1(3,i+1,j)= 0.8*theta_hat1(3,i,j)+0.5;

     theta_hat_f(:,i+1,j)=f_theta_hat1(:,i+1,j)+(randn(1,3)*chol(Q))';
     obs_f(:,j) = H*theta_hat_f(:,i+1,j)+ (randn(1,2)*R_chol)';
 end



%Calculation of error quantities

 for j=1:Ns
     E_theta(:,j) = theta_hat_f(:,i+1,j)-mean(theta_hat_f(:,i+1,j));
     E_obs(:,j)= obs_f(:,j)-mean(obs_f,2);
 end

%Estimation of covariance matrices P_theta_obs and P_obs_obs using the
%error matricres

P_theta_obs=(1/(Ns-1))*E_theta*(E_obs)';
P_obs_obs= (1/(Ns-1))*E_obs*(E_obs)';
```

```
%Update of the current state using current measurement data
 for j=1:Ns
 theta_hat1(:,i+1,j) = theta_hat_f(:,i+1,j)+P_theta_obs*inv(P_obs_obs)
              *(obs_recent-H*theta_hat_f(:,i+1,j));
 end

%Final estimate as the arithmetic mean of the ensemble members
 theta_ekf(:,i+1) = mean(theta_hat1(:,i+1,:),3);
end
```

References

Thomas Bengtsson, Peter Bickel, and Bo Li. Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems. *IMS COLLECTIONS*, 2:316, 2008. URL doi:10.1214/193940307000000518. 7

M.D. Butala, Jonghyun Yun, Yuguo Chen, R.A. Frazin, and F. Kamalabadi. Asymptotic convergence of the ensemble kalman filter. *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pages 825–828, Oct. 2008. ISSN 1522-4880. doi: 10.1109/ICIP.2008.4711882. 4, 8

D. Crisan and A. Doucet. A survey of convergence results on particle filtering methods for practitioners. *Signal Processing, IEEE Transactions on*, 50(3):736–746, Mar 2002. ISSN 1053-587X. doi: 10.1109/78.984773. 4, 6

Arnaud Doucet, Simon Godsill, and Christophe Andrieu. On sequential monte carlo sampling methods for bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000. ISSN 0960-3174. doi: http://dx.doi.org/10.1023/A:1008935410038. 6, 15, 18, 19

Geir Evensen. The ensemble kalman filter: theoretical formulation and practical implementation. *Ocean Dynamics*, 53(4):343–367, 2003. doi: 10.1007/s10236-003-0036-9. URL http://dx.doi.org/10.1007/s10236-003-0036-9. 22

Geir Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics. *J. Geophys. Res*, 99:10143–10162, 1994. 4, 6

Reinhard Furrer and Thomas Bengtsson. Estimation of high-dimensional prior and posterior covariance matrices in kalman filter variants. *J. Multivar. Anal.*, 98(2):227–255, 2007. ISSN 0047-259X. doi: http://dx.doi.org/10.1016/j.jmva.2006.08.003. 8

S. Gillijns, O.B. Mendoza, J. Chandrasekar, B.L.R. De Moor, D.S. Bernstein, and A. Ridley. What is the ensemble kalman filter and how well does it

work? *American Control Conference, 2006*, pages 4448–4453, June 2006. doi: 10.1109/ACC.2006.1657419. 8, 22, 24

A. J. Haug. A tutorial on bayesian estimation and tracking techniques applicable to nonlinear and non-gaussian processes. Technical report, 2005. URL `http://www.mitre.org/work/tech_papers/tech_papers_05/05_0211/index.html`. 15

Xiao-Li Hu, T.B. Schon, and L. Ljung. A basic convergence result for particle filtering. *Signal Processing, IEEE Transactions on*, 56(4):1337–1348, April 2008. ISSN 1053-587X. doi: 10.1109/TSP.2007.911295. 7

R. E. Kalman. A new approach to linear filtering and prediction problems. 1960. URL `http://www.cs.unc.edu/~welch/kalman/media/pdf/Kalman1960.pdf`. 1, 22

Jan Mandel, Loren Cobb, and Jonathan D. Beezley. On the convergence of the ensemble kalman filter, 2009. URL `http://www.citebase.org/abstract?id=oai:arXiv.org:0901.2951`. 4, 8

S. Maskell and N. Gordon. A tutorial on particle filters for on-line nonlinear/non-gaussian bayesian tracking. volume Workshop, pages 2/1–2/15 vol.2, Oct. 2001. 2, 3, 11, 15, 16

P. Del Moral. Measure valued processes and interacting particle systems. application to non linear filtering problems. *Ann. Appl. Prob*, 8:438–495, 1996. 3, 6, 15

Kazuyuki Nakamura, Naoki Hirose, Byung Ho Choi, and Tomoyuki Higuchi. Particle filtering in data assimilation and its application to estimation of boundary condition of tsunami simulation model. In Seon K. Park and Liang Xu, editors, *Data Assimilation for Atmospheric, Oceanic and Hydrologic Applications*, pages 353–366. Springer Berlin Heidelberg, 2009. 8

Ilter Saygin. On the performance of particle filters. Master's thesis, Purdue University, 2004. 4, 7

Albrecht H. Weerts and Ghada Y. H. El Serafy. Particle filtering and ensemble kalman filtering for state updating with hydrological conceptual rainfall-runoff models. volume 42. American Geophysical Union, 2006. doi: 10.1029/2005WR004093. URL `http://www.agu.org/journals/wr/wr0609/2005WR004093/`. W09403. 9