UNIVERSITÄT POTSDAM
INSTITUT FÜR MATHEMATIK
AG NUMERISCHE MATHEMATIK

Time-Continuous State and Parameter Estimation with Application to Hyperbolic SPDEs

Dissertation
zur Erlangung des akademischen Grades

"doctor rerum naturalium"
(Dr. rer. nat.)

in der Wissenschaftsdisziplin
"Numerische Mathematik" eingereicht an der

Mathematisch-Naturwissenschaftlichen Fakultät
der Universität Potsdam

von:
Angwenyi Nyachae David

First Advisor:
Prof. Dr. Sebastian Reich

Second Advisor:
Prof. Dr. Claudia Stolle

September 23, 2019

Published online at the
Institutional Repository of the University of Potsdam:
https://doi.org/10.25932/publishup-43654
https://nbn-resolving.org/urn:nbn:de:kobv:517-opus4-436542

Acknowledgement

With most profound gratitude thank I my supervisors, Professor Dr. Sebastian Reich and Professor Dr. Claudia Stolle, for their patience, promptness, availability, many insights and unyielding support throughout the course of this study. I thank Professor Reich for introducing me to data assimilation—a field in which I have learnt invaluable ideas of practical use. I would also wish to thank Professor Brian Hunt for linking me up with Professor Reich.

For financial support, I thank the German Academic Exchange Service (DAAD) and the Kenya National Research Foundation (NRF), without which support this study could not have been possible. I thank the University of Potsdam for an ambient working environment with state-of-the-art facilities which made my stay and work very convenient. Masinde Muliro University of Science and Technology generously gave me a study leave, for which I am very grateful. My thanks are due also to the organizers of the project SFB 1294 for the frequent colloquia, lectures and seminars, which brought together a number of experts in data assimilation, from whom I derived a lot in terms of new research directions, skills and connections.

I am greatly indebted to the Numerical Analysis group of the University of Potsdam for the fun-filled moments we shared together. The numerous formal and informal meetings we had helped me learn a lot of things within and outside academia. The warm fellowship in the group, coffee times, lunch meetings and excursions made me feel at home away from home. I specifically thank Jana, Liv, Mark, Maria, Noa, Christian, Paul, Nawinda, Sahani, Kay, Gottfried and Walter. I thank Paul for the frequent meetings we had to discuss this work.

I would like also to acknowledge the fortune and pleasure I have had in sharing an office with Stefan, Milad and Abhishake, from whom I have been inspired in many ways. The discussions we often had on general matters of life will be gratefully remembered. Their words of encouragement and counsel, the light moments we shared to ease us up from the fatigue of study helped a great deal.

My finest acknowledgement goes to my family and friends to whom I am beholden for unconditional love, moral support, well wishes and prayers tending to my success and the happy issuance of my studies.

Last but not in any way least, I pay a special tribute to all those who, in one way or another, directly or indirectly, helped me during this study. This includes those who made our workplace and living quarters tidy, those who supplied us with material for use, those who, by their smiles and words, made this pursuit light—all of whom are too numerous to name here. This work is consequent upon their generosity.

Abstract

Data assimilation has been an active area of research in recent years, owing to its wide utility. At the core of data assimilation are filtering, prediction, and smoothing procedures. Filtering entails incorporation of measurements' information into the model to gain more insight into a given state governed by a noisy state space model. Most natural laws are governed by time-continuous nonlinear models. For the most part, the knowledge available about a model is incomplete; and hence uncertainties are approximated by means of probabilities. Time-continuous filtering, therefore, holds promise for wider usefulness, for it offers a means of combining noisy measurements with imperfect model to provide more insight on a given state.

The solution to time-continuous nonlinear Gaussian filtering problem is provided for by the Kushner-Stratonovich equation. Unfortunately, the Kushner-Stratonovich equation lacks a closed-form solution. Moreover, the numerical approximations based on Taylor expansion above third order are fraught with computational complications. For this reason, numerical methods based on Monte Carlo methods have been resorted to. Chief among these methods are sequential Monte-Carlo methods (or particle filters), for they allow for online assimilation of data. Particle filters are not without challenges: they suffer from particle degeneracy, sample impoverishment, and computational costs arising from resampling.

The goal of this thesis is to:— i) Review the derivation of Kushner-Stratonovich equation from first principles and its extant numerical approximation methods, ii) Study the feedback particle filters as a way of avoiding resampling in particle filters, iii) Study joint state and parameter estimation in time-continuous settings, iv) Apply the notions studied to linear hyperbolic stochastic differential equations.

The interconnection between Itô integrals and stochastic partial differential equations and those of Stratonovich is introduced in anticipation of feedback particle filters. With these ideas and motivated by the variants of ensemble Kalman-Bucy filters founded on the structure of the innovation process, a feedback particle filter with randomly perturbed innovation is proposed. Moreover, feedback particle filters based on coupling of prediction and analysis measures are proposed. They register a better performance than the bootstrap particle filter at lower ensemble sizes.

We study joint state and parameter estimation, both by means of extended state spaces and by use of dual filters. Feedback particle filters seem to perform well in both cases. Finally, we apply joint state and parameter estimation in the advection and wave equation, whose velocity is spatially varying. Two methods are employed: Metropolis Hastings with filter likelihood and a dual filter comprising of Kalman-Bucy filter and ensemble Kalman-Bucy filter. The former performs better than the latter.

Zusammenfassung

Die Datenassimilation war in den letzten Jahren aufgrund ihres breiten Nutzens ein aktives Forschungsgebiet. Im Zentrum der Datenassimilation stehen Filter-, Vorhersage- und Glättungsverfahren. Die Filterung beinhaltet die Einbeziehung von Messinformationen in das Modell, um einen besseren Einblick in einen gegebenen Zustand zu erhalten, der durch ein verrauschtes Zustandsraummodell gesteuert wird. Die meisten Naturgesetze werden von zeitkontinuierlichen nichtlinearen Modellen bestimmt. Das verfügbare Wissen über ein Modell ist größtenteils unvollständig; und daher werden Unsicherheiten mittels Wahrscheinlichkeiten angenähert. Die zeitkontinuierliche Filterung verspricht daher eine größere Nützlichkeit, denn sie bietet die Möglichkeit, verrauschte Messungen mit einem unvollkommenen Modell zu kombinieren, um mehr Einblick in einen bestimmten Zustand zu erhalten.

Das Problem der zeitkontinuierlichen nichtlinearen Gaußschen Filterung wird durch die Kushner-Stratonovich-Gleichung gelöst. Leider fehlt der Kushner-Stratonovich-Gleichung eine geschlossene Lösung. Darüber hinaus sind die numerischen Näherungen, die auf der Taylor-Erweiterung über der dritten Ordnung basieren, mit rechnerischen Komplikationen behaftet. Aus diesem Grund wurde auf numerische Methoden zurückgegriffen, die auf Monte-Carlo-Methoden basieren. Die wichtigsten dieser Methoden sind sequentielle Monte-Carlo-Methoden (oder Partikelfilter), da sie die Online-Assimilation von Daten ermöglichen. Partikelfilter sind nicht unproblematisch: Sie leiden unter Partikelentartung, Probenverarmung und Rechenkosten, die sich aus der Neuabtastung ergeben.

Das Ziel dieser Arbeit ist es, i) die Ableitung der Kushner-Stratonovich-Gleichung aus den ersten Prinzipien und ihre vorhandenen numerischen Approximationsmethoden zu überprüfen, ii) die Rückkopplungs-Partikelfilter zu untersuchen, um eine Neuabtastung in Partikelfiltern zu vermeiden, iii) Studieren Sie die Zustands- und Parameterschätzung in zeitkontinuierlichen Einstellungen, iv) Wenden Sie die untersuchten Begriffe auf lineare hyperbolische stochastische Differentialgleichungen an.

Die Verbindung zwischen Itô Integralen und stochastischen partiellen Differentialgleichungen und denen von Stratonovich wird in Erwartung von Rückkopplungs-Partikelfiltern eingeführt. Mit diesen Ideen und motiviert durch die Varianten von Kalman-Bucy-Filtern, die auf der Struktur des Innovationsprozesses gegründet, wird ein Feedback-Partikelfilter mit zufällig gestörter Innovation vorgeschlagen. Darüber hinaus werden Rückkopplungspartikelfilter basierend auf der Kopplung von Vorhersage- und Analysemaßnahmen vorgeschlagen. Diese Feedback-Partikelfiltern haben eine bessere Leistung als der Bootstrap-Partikelfilter bei niedrigeren Ensemble-Größen.

Wir untersuchen gemeinsame Zustands- und Parameterschätzungen, sowohl durch erweiterte Zustandsräume als auch durch Verwendung von Doppelfiltern. Rückkopplungs-Partikelfilter scheinen in beiden Fällen gut zu funktionieren. Schließlich wenden wir eine gemeinsame Zustands- und Parameterschätzung in der Advektions-und Wellengleichung an, deren Geschwindigkeit räumlich variiert. Es werden zwei Verfahren verwendet: Metropolis-Hastings mit Filterwahrscheinlichkeit und ein Doppelfilter bestehend aus Kalman-Bucy-Filter und Ensemble-Kalman-Bucy-Filter. Ersteres schneidet besser ab als letzteres.

To my family, friends and forebears.

Contents

List of Figures

List of Algorithms

List of Tables

CHAPTER 1

INTRODUCTION

1.1 Overview of the thesis

This thesis concerns the twin problems of state and parameter inference in time-continuous state space models (SSMs). From the outset, this study acknowledges the subdivision of SSMs into four parts: linear Gaussian state space models, linear non-Gaussian state space models, non-linear Gaussian state-space models and non-linear non-Gaussian state space models. We restrict ourselves to both linear and nonlinear models with an assumption that probability densities and conditional probability densities are Gaussian. Problems lacking closed-form solutions, which is the case when treating nonlinear models, are considered. We make approximations based on Taylor series expansion and Monte-Carlo methods.

In this introduction, we give a cursory view of the thesis and motivate the problem to be studied in the rest of this thesis. The achievements of the thesis and results are highlighted. Chapter 2 treats time-continuous SSMs. The setting is that of stochastic differential equations. We review the Itô and Stratonovich integrals and their interconversion, both integrals being useful in subsequent chapters. The notion of evolution of probability density and conditional probability densities is introduced, which comprises the forward Kolmogorov equation (also known as Fokker-Planck equation). Furthermore, the equations of evolution of moments is derived—which, in fact, arises naturally from weak formulation of the Fokker-Planck equation. It turns out that the equation of evolution of moments, in the nonlinear SSM, lack closed-form solution, except, perhaps in very few cases. For this reason, and in anticipation of applications in later chapters, we introduce numerical solutions for stochastic differential equations. An application to the Geometric Brownian motion forms the closing part of the chapter.

Introduction of the measurements to the equation of evolution of the state is handled in Chapter 3. This paves way to the definition of the filtering problem. We introduce the equation of evolution of the probability density of the state given all available measurements, which equation is called Kushner-Statonovich equation [Kushner (1962)]. The equation of evolution of moments—which is known as Kushner-Zakai equation—is derived by obtaining the weak formulation of the Kushner-Stratonovich equation. From the Kushner-Zakai equation, we obtain the equation of evolution of the mean and covariance—both for nonlinear scalar and vector signal and measurements equations. Now the equations governing the evolution of first and second moments—that is, the mean and covariance, respectively—in nonlinear settings depend on infinite higher moments, hence they are intractable. Which is why approximations are inevitable. We therefore review the derivation of the second and first order direct approximations to the equation of evolution of the mean and covariance using Taylor series expansion about the mean [Jazwinski (1970); Bass et al. (1966)]. What is more, we derive sequential Monte Carlo approximation to the solution to the nonlinear filtering problem. To conclude the chapter, we make applications to the Lorentz-63 model and offer critique to the numerical solution

of nonlinear filtering problem.

Time-continuous linear filtering comprises Chapter 4, where we introduce the model equations by specialising the nonlinear model equations to linear settings. We then arrive at the Kalman-Bucy filter (KBF) by substituting linear terms for the nonlinear terms in the equation of evolution of the mean and covariance obtained in Chapter 3. What follows is an introduction of two formulations of Ensemble Kalman Bucy filter (EnKBF), *viz.*, the Ensemble Kalman Bucy filter with a randomly perturbed innovation and that with deterministically perturbed innovation. The extensions to nonlinear setting, of both the KBF and the EnKBF to extended Kalman-Bucy filter (EKBF) and the extended ensemble Kalman-Bucy filter (EEnKBF), are spelt out. Thereafter, we demonstrate the consistency of EnKBF with KBF. An application example is supplied at the end of the chapter.

It is in Chapter 5 that we treat the feedback particle filter (FPF), in the most general nonlinear setting [Yang et al. (2013)]. For the first time, FPF with stochastically perturbed innovation is introduced. We show that the evolution of the density of each ensemble member given the measurements' history is consistent with the evolution of the true posterior under certain conditions. It turns out that the elliptic equation, the solving of which we obtain the gain, is the same in the FPF with stochastically perturbed innovation as that in the FPF with deterministically perturbed innovation. What follows is the numerical solution of the boundary value problem from which we obtain the gain. We then, for the first time, introduce two variants of FPF using the ideas of optimal transportation of measures as defined by Schrödinger. Finally, we close the chapter with a heuristic analysis of the performance of the filters.

Now thus far pertains state estimation. Parameter estimation is introduced in Chapter 6. We consider static, constant parameters, and describe statistical methods for parameter inference; that is, Maximum likelihood (ML) and maximum a posteriori (MAP) methods. The use of filters for parameter estimation is then described. We introduce for the first time the use of sequential Monte Carlo method paired with EnKBF and EnKBF paired with ensemble transform particle filter (ETPF) for simultaneous estimation of model state and parameters. Examples are used to illustrate the performance of the algorithms.

Chapter 7 comprises application of state and parameter estimation to stochastic partial differential equations (SPDEs). Two equations are considered: advection and wave equations with space-time white noise as the forcing term. The velocity is allowed to vary in time; so that the equations can be used to model waves travelling in heterogeneous media. To allow parameter estimation, we approximate the time varying velocity using a fourier series with a finite number of modes. For then estimation of spatially varying velocity reduces to estimation of static coefficients of the fourier series. Now to insure stability of the equation solutions, we use upwind discretisation in space for the advection equation and Störmer-Verlet discretisation method for the wave equation. We obtain results for parameter estimates using the likelihood method with Metropolis-Hastings steps. What is more, we obtain parameter estimates, and their respective errors, using EnKBF-KBF dual filters.

1.2 Motivation

The question of state—and parameter—estimation is, in many fields, very important and has a long history [Särkkä (2013); Bar-Shalom et al. (2001); Lewis et al. (2006)]. The central field that investigates and offers solution to this problem is data assimilation [Reich and Cotter (2015); Law et al. (2015)]. To describe it briefly, data assimilation is concerned with the incorporation of measurements of the state—which, mostly, come with measurement errors—into a model to improve the model's approximation of a given state. This can be accomplished by a number

of ways: least squares approach, statistical approach and by Bayesian inference. In this thesis, we take up the Bayesian inference approach to data assimilation.

The flourishing of data assimilation in recent years is informed by its wide, practical utility. Some of the areas which employ data assimilation include surveillance—detection, identification and tracking of, say, astrological objects; navigation which comprises dead reckoning, radio and celestial navigation; numerical weather prediction; robotics and in other control systems [Mohinder and Angus (2001); Tomera (2011); Lewis et al. (2006); Jazwinski (1970); Bar-Shalom et al. (2001)]. Most of these application areas have time-continuous, nonlinear, non-Gaussian governing equations.

The linear, Gaussian model equations have optimal estimates, thanks to the celebrated Kalman filter. The optimal performance of the Kalman filter has prompted extensions to nonlinear settings. These extensions, however, are not optimal and the user incurs extra computational costs owing to linearisation. Other exact filters for nonlinear setting have been proposed, although they are limited to certain model equations, hence they have a restricted use. Filter estimates—which are consistent, unbiased, efficient, robust and of minimal variance—for nonlinear models are highly required.

Sequential Monte-Carlo methods and ensemble-based filters have obtained prominence in recent years, owing to their better performance in nonlinear settings. Among these filters are the ensemble Kalman-Bucy filter and the feedback particle filters. Theoretical underpinnings of these filters are minimal [de Wiljes et al. (2016)]. Ensemble-based filters involving optimal coupling, for example, the ensemble transform particle filter, have been shown to perform well heuristically.

Whereas discrete filters have had a wide-spread treatment and application in data assimilation, their theoretical bases depend on the corresponding time-continuous counterparts. In time-continuous setting, for instance, the equation of evolution of the posterior density has been derived from first principles [Jazwinski (1970); Kushner (1962)]. To insure exactness of a filter, therefore, one has to check whether the filter posterior density matches the true posterior at all times.

Now this thesis is cast in time-continuous, nonlinear setting. Our guiding purpose is to study filtering in nonlinear systems and its application to state and parameter estimation. We treat linear filtering as a special case of nonlinear filtering. From the outset, we use limit in the mean convergence, which allows us to avoid technical measure theoretic terms.

1.3 Main achievements

In this section, we present, briefly, what has been attained in this thesis.

- A feedback particle filter with stochastically perturbed innovation is proposed. The exactness of this filter is demonstrated by showing that the filter posterior density evolves as the true posterior when the initial filter posterior is chosen to be the same as the initial true posterior. We show the performance of this filter with the use of the, highly nonlinear, Lorentz 63 model.

- We propose, for the first time, two formulations of feedback particle filters—the Sinkhorn particle filter (SPF) and the resampling Sinkhorn particle filter (RSPF)—where the controlled feedback is obtained by solving an optimal transport problem—the so-called Schrödinger problem. The optimal transport problem arising from the filter is solved iteratively using Sinkhorn method [Cuturi (2013)].

- The dual filter comprising of ensemble Kalman-Bucy filter and ensemble transform particle filter (EnKBF-ETPF), and bootstrap particle filter and ensemble Kalman-Bucy filter

(BPF-EnKBF) are proposed for the first time in this thesis. A portion of the work, which is part of this thesis, appears in [Angwenyi et al. (2017)]. The performance of the two dual filters is compared by means of a scalar example.

- Application of state and parameter estimation is done on the advection and wave equations driven by space-time white noise. We let the velocity be spatially varying—as would be the case when modelling a wave travelling in a heterogeneous media. To facilitate estimation of the wave, we approximate the velocity function by means of a Fourier series of finite modes. For then we obtain the approximation of the velocity function by estimating the static coefficients of the series.

1.4 Numerical findings

This study leads to following findings

- ⋆ The Sinkhorn particle filter and the resampling Sinkhorn particle filter are numerically shown to outperform the classic bootstrap particle filter at low ensemble sizes (see Figure 5.1).
- ⋆ EnKBF-ETPF dual filter yields a faster converging parameter estimates as compared to BPF-EnKBF dual filter in a scalar linear model (see Figure 6.3).
- ⋆ On the estimation of the spatially varying velocity, the use of the filter likelihood with Metropolis Hastings steps elicits a better performance as compared to the use of KBF-EnKBF dual filter in the wave equation (cref. Figures 7.10 and 7.14). Both methods seem to work well when applied to the advection equation (cref. Figures 7.8 and 7.12).

CHAPTER 2

STATE-SPACE MODELS

2.1 Introduction

Most models, which are evolutionary in nature, are fashioned in such a way that one can obtain an estimate of the state at a given time. It is also possible to obtain future estimates of the state by running the models forward in time. This is fine—in so far as the models are perfectly representing the desired dynamics, which, unfortunately, is not often the case. This imperfection of the models can be dealt with by quantifying the uncertainty. The uncertainties in the model are mostly represented as random variables, and are referred to as noise in the model. To be precise, a dynamical model in this context is as follows.

$$dx_t = f(x_t, t)dt + g(x_t, t)d\beta_t; \quad t_0 \leq t, \tag{2.1.1}$$

where:

Term	**Name**	**Dimension**
x_t	state vector	$n \times 1$
$f(x_t, t)$	drift function	$n \times 1$
$g(x_t, t)$	diffusion function	$n \times m$
$\{\beta_t,\, t > t_0\}$	Brownian motion process	$m \times 1$

The state vector, x_t, at a given time, t, evolves in time according to a function, f, of the state vector, x_t. The uncertainties and imperfections of the model are—as much as possible—captured in the additive term, $g(x_t, t)d\beta_t$, where $\{\beta_t,\, t > t_0\}$ is standard Brownian motion process. The model equation, eq. (2.1.1), is an expression of a continuous state space model. Equation (2.1.1) is said to be *linear* if the drift function, $f(x_t, t)$, and the diffusion function, $g(x_t, t)$, can be expressed as $F(t)x_t$ and $G(t)x_t$, respectively, where the $n \times n$ matrix function, $F(t)$, and the $n \times m$ matrix function, $G(t)$, are not functions of x_t; otherwise, eq. (2.1.1) is said to be *nonlinear*. Given the initial value of the state, x_{t_0}, the subsequent states in the model can be obtained by formal integration. This involves solving the model equation each time. That is to say, the state at a given time depends on the state at the previous time—and nothing else about the past states. Models whose states at a given time can be obtained this way are referred to as Markov models.

The model is said to be *deterministic* if it does not include randomness in any of its terms; otherwise, it is said to be *stochastic*. Generally, stochasticity in a model is induced through the forcing term, $g(x_t, t)d\beta_t$, or the parameters, or the initial conditions, or the boundary terms. The models studied in this thesis are stochastic. It is quite convenient to study various aspects of Markov models using probabilities. The reason for this, and this will become apparent in subsequent sections, is that probabilities, in some cases, summarise a lot of information about

the model using very few terms. Take Gaussian distribution for example. For a linear model equation, it is enough to know the evolution of the mean and the covariances of the model of interest; that is to say, the mean of the states and respective covariance are sufficient statistics for inferring the state of the system at a given time. It is partly for the ease and elegance in manipulations where Gaussian distributions are involved that we have Gaussian state space models. Furthermore, we are able, by means of probabilities, to make inferences about parameters given the states or observations.

We now introduce some well known terms and results of stochastic processes and probability theory, which will be useful in this study.

2.2 Stochastic Integral

Suppose, for the moment, that we would obtain a solution to eq. (2.1.1); which, in fact, is found by integrating both sides of eq. (2.1.1):

$$x_t = x_{t_0} + \int_{t_0}^{t} f(x_\tau, \tau) d\tau + \int_{t_0}^{t} g(x_\tau, \tau) d\beta_\tau. \tag{2.2.1}$$

The first integral on the right-hand-side of eq. (2.2.1) is the common integral in Calculus—the Riemann integral. Now the second integral is riddled with technicalities which arise from the fact that Brownian motion process, $\{\beta_t,\ t > t_0\}$, is of unbounded variation. The implication is that Brownian motion process is nowhere differentiable. Such integrals as this require a special treatment, due to inherent stochasticity, which is why they are known as *stochastic integrals*. Interpretation of these integrals began with the works of Norbert Wiener, in 1940s, on scalar stochastic differential equations and were later generalized by Kiyoshi Itô [Itô (1951)]. Ruslan Stratonovich, in 1960s, obtained another formulation of stochastic integrals giving rise to two ways of looking at stochastic differential equations [Stratonovich (1966); Schaffter (2010)].

Consider the stochastic integral (in scalar form)

$$\int_a^b g(x_\tau, \tau) d\beta_\tau. \tag{2.2.2}$$

In Itô's formulation, the stochastic integral, eq. (2.2.2), is defined by (see Jazwinski (1970); Schaffter (2010) for details)

$$\int_a^b g(x_\tau, \tau) d\beta_\tau = \underset{\delta t \to 0}{\text{l.i.m.}} \sum_{k=1}^{n} g(x_{t_k}, t_k)(\beta_{t_{k+1}} - \beta_{t_k}), \tag{2.2.3}$$

where $\delta t = t_{k+1} - t_k$ and the terms to be summed are evaluated at discretised times $a = t_0 < t_1 < \ldots < t_k < t_{k+1} < \ldots < t_n = b$. l.i.m. is limit-in-the-mean convergence[1] (see Doob (1953) for more details and for other forms of convergence).

The following interpretation of eq. (2.2.2) is also possible, and is due to Stratonovich:

$$\int_a^b g(x_\tau, \tau) \circ d\beta_\tau = \underset{\delta t \to 0}{\text{l.i.m.}} \sum_{k=1}^{n} g(\frac{x_{t_k} + x_{t_{k+1}}}{2}, t_k)(\beta_{t_{k+1}} - \beta_{t_k}), \tag{2.2.6}$$

[1] Let $\{x_{t_k}, k = 1, 2, 3, \ldots\}$ be a random sequence. Suppose a random sequence $\{x_{t_k}, k = 1, 2, 3, \ldots\}$ is such that $\mathbb{E}\left[x_{t_k}\right] < \infty$ and that there exists x such that $\mathbb{E}[x] < \infty$. Then the sequence is said to converge in mean square to x if

$$\lim_{k \to \infty} \mathbb{E}\left[|x_{t_k} - x|^2\right] = 0. \tag{2.2.4}$$

x is then referred to as the limit in the mean of the random sequence $\{x_{t_k}, k = 1, 2, 3, \ldots\}$; that is,

$$\underset{k \to \infty}{\text{l.i.m.}} x_{t_k} = x. \tag{2.2.5}$$

where $\circ$ is used to indicate a different formulation than that due to Itô. Equation (2.2.6) is known as Stratonovich stochastic integral.

In the following, we introduce an already established relationship between stochastic integral—and, subsequently, the stochastic differential equation—of Itô and that of Stratonovich; which is as follows:

$$\int_a^b g(x_\tau,\tau)\circ d\beta_\tau = \int_a^b g(x_\tau,\tau)d\beta_\tau + \frac{1}{2}\int_a^b \partial_x[g(x_\tau,\tau)]g(x_\tau,\tau)d\tau, \tag{2.2.7}$$

where $\partial_x[g] = \dfrac{\partial g}{\partial x}$. The integral in the left-hand-side of the equal sign is the Stratonovich integral; the first on the right-hand-side is the Itô integral; the last is a Riemann integral. From eq. (2.2.7), the relationships between Itô stochastic differential equation and Stratanovich stochastic differential equation is straightforward.

The scalar Itô stochastic differential equation

$$dx_t = f(x_t,t)dt + g(x_t,t)d\beta_t; \quad t_0 \le t, \tag{2.2.8a}$$

based on (2.2.7), has its Stratonovich equivalent

$$dx_t = f(x_t,t)dt - \frac{1}{2}g(x_t,t)\partial_x[g(x_t,t)]dt + g(x_t,t)\circ d\beta_t; \quad t_0 \le t. \tag{2.2.8b}$$

Now we turn to the vector form. The vector Itô SDE, eq. (2.1.1), has an equivalent vector Stratonovich SDE:

$$dx_t = q(x_t,t)dt + g(x_t,t)\circ d\beta_t; \quad t_0 \le t, \tag{2.2.9a}$$

where the $n\times 1$ vector q has elements given by

$$q_k(x,t) = f_k(x,t) - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{m} g_{ij}(x,t)\partial_{x_i}[g_{kj}(x,t)]. \tag{2.2.9b}$$

The import of these two different formulations will become apparent in later chapters. For the moment, however, we restrict ourselves to the Itô form.

2.3 Itô's formula

Consider the scalar differential form, eq. (2.2.8a), of the Itô stochastic differential equation, eq. (2.1.1). Let $h(x_t,t)$ be a function that is at least twice differentiable in both its arguments and has mixed derivatives. By x_t is meant that x is a function of time. Taylor expansion of $h(x_{t+\delta t}, t+\delta t)$ about (x_t,t) yields

$$\begin{aligned} h(x_{t+\delta t},t+\delta t) = {} & h(x_t,t) + \partial_t[h(x_t,t)]\delta t + \partial_x[h(x_t,t)]\delta x_t \\ & + \frac{1}{2}\left(\partial_{tt}[h(x_t,t)]\delta t^2 + 2\partial_{tx}[h(x_t,t)]\delta t\delta x_t + \partial_{xx}[h(x_t,t)]\delta x_t^2\right) + \ldots, \end{aligned} \tag{2.3.1}$$

where

$$x_{t+\delta t} := x_t + \delta x_t.$$

From eq. (2.2.8a) we have

$$\delta x_t = f(x_t,t)\delta t + g(x_t,t)\delta\beta_t, \tag{2.3.2a}$$

from which we get

$$\delta x_t^2 = f^2(x_t,t)\delta t^2 + 2f(x_t,t)\delta t g(x_t,t)\delta\beta_t + g^2(x_t,t)\delta\beta_t^2. \tag{2.3.2b}$$

Note that $\mathbb{E}[\delta x_t^2] = g^2(x_t,t)\delta t + o(\delta t)$, where $o(\delta t)$ represents terms containing second and higher order terms in δt. As $\delta t \to 0$, δx_t^2 attains its expected value, $\mathbb{E}[\delta x_t^2]$ (see Appendix 1 of Kushner (1962))—where $\mathbb{E}[.]$ stands for expectation operation. Substituting $\mathbb{E}[\delta x_t^2]$ for δx_t^2 and writing h in its differential form, gives

$$\delta h(x_t,t) = \partial_t[h(x_t,t)]\delta t + \partial_x[h(x_t,t)]\delta x_t + \frac{1}{2}\partial_{xx}[h(x_t,t)]g^2(x_t,t)\delta t + o(\delta t). \tag{2.3.3}$$

We substitute eq. (2.3.2a) into eq. (2.3.3) and, assuming higher order terms to be negligible, obtain

$$\begin{aligned}\delta h(x_t,t) &= \partial_t[h(x_t,t)]\delta t + \partial_x[h(x_t,t)]\left(f(x_t,t)\delta t + g(x_t,t)\delta\beta_t\right)\\ &\quad + \frac{1}{2}\partial_{xx}[h(x_t,t)]g^2(x_t,t)\delta t\\ &= \partial_t[h(x_t,t)]\delta t + \partial_x[h(x_t,t)]f(x_t,t)\delta t + \frac{1}{2}\partial_{xx}[h(x_t,t)]g^2(x_t,t)\delta t\\ &\quad + \partial_x[h(x_t,t)]g(x_t,t)\delta\beta_t.\end{aligned} \tag{2.3.4}$$

Extension of eq. (2.3.4) to vector form follows from substituting an n-dimensional column vector, x_t, in the place of the scalar, x_t, and using the differential operators

$$\nabla := \begin{pmatrix} \dfrac{\partial}{\partial x_1} \\ \vdots \\ \dfrac{\partial}{\partial x_n} \end{pmatrix}, \quad \text{and} \quad \triangle := \begin{bmatrix} \dfrac{\partial^2}{\partial x_1^2}, & \dfrac{\partial^2}{\partial x_1 \partial x_2}, & \cdots, & \dfrac{\partial^2}{\partial x_1 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial^2}{\partial x_n \partial x_1}, & \dfrac{\partial^2}{\partial x_n \partial x_2}, & \cdots, & \dfrac{\partial^2}{\partial x_n^2} \end{bmatrix}; \tag{2.3.5}$$

which yields

$$\begin{aligned}\delta h &= \partial_t[h]\delta t + \nabla[h]^{\mathrm{T}} f(x_t,t)\delta t + \frac{1}{2}\mathrm{tr}[g(x_t,t)g^{\mathrm{T}}(x_t,t)] \triangle [h]\delta t\\ &\quad + (\nabla[h]^{\mathrm{T}} g(x_t,t))\delta\beta_t;\end{aligned} \tag{2.3.6}$$

whereupon taking the limit-in-the-mean of δh as δt and δx tend to zero, we obtain

$$dh = \partial_t[h]dt + \nabla[h]^{\mathrm{T}} dx_t + \frac{1}{2}\mathrm{tr}[g(x_t,t)g^{\mathrm{T}}(x_t,t)] \triangle [h]dt. \tag{2.3.7}$$

Equation (2.3.7) is known as *Itô's formula* or *Itô's Lemma*. It is useful in stochastic calculus.

2.4 Fokker-Planck Equation

Consider the scalar Itô stochastic differential equation, eq. (2.2.8a). It can be shown [Jazwinski (1970)] that the process $\{x_t, t \in [t_0, T]\}$, generated by eq. (2.2.8a), is a Markov process. Of interest in Markov processes is the density function

$$\pi_t(x) = \pi(x,t), \qquad \forall t \in [t_0, T], \tag{2.4.1a}$$

together with the transition probability density function

$$\pi_{t|\tau}(x \mid z) = \pi(x, t \mid z, \tau), \qquad \forall \tau < t \in [t_0, T], \tag{2.4.1b}$$

which both characterise the process $\{x_t, t \in [t_0, T]\}$.

The Fokker-Planck equation is the equation of evolution of the density function, $\pi_t(x)$, and the conditional density, $\pi_{t|\tau}(x \mid z)$, for all $t > \tau \in [t_0, T]$. Since the process $\{x_t, t > 0\}$ generated by eq. (2.2.8a) is a Markov process, then, given $t_1 < t_2 < t_3$,

$$\pi_{t_3|t_1,t_2}(x \mid z, y) = \pi_{t_3|t_2}(x \mid y), \tag{2.4.2}$$

and the following equation is satisfied,

$$\pi_{t_3|t_1}(x \mid z) = \int \pi_{t_3|t_2}(x \mid y)\pi_{t_2|t_1}(y \mid z)dy. \tag{2.4.3}$$

Equation (2.4.3) is known as the Chapman-Kolmogorov equation, and it applies to every Markov process. Using Chapman-Kolmogorov equation and Taylor expansion, an equation for evolution of transition probability density function, $\pi_{t|\tau}(x \mid y) = \pi(x, t \mid y, \tau)$, can be obtained (see Jazwinski (1970) for the derivation), which equation is as follows.

$$\frac{\partial \pi_{t|\tau}(x \mid y)}{\partial t} = -\frac{\partial(\pi_{t|\tau}(x \mid y)f(x,t))}{\partial x} + \frac{1}{2}\frac{\partial^2(\pi_{t|\tau}(x \mid y)g^2(x,t))}{\partial x^2}, \tag{2.4.4}$$

which is the Fokker-Planck equation or the Kolmogorov's forward equation. Taking expectation in eq. (2.4.4) with respect to $\pi_t(y)$ and noting that,

$$\mathbb{E}_\tau[\pi_{t|\tau}(x \mid y)] = \int \pi_{t|\tau}(x \mid y)\pi_\tau(y)dy = \pi_t(x), \tag{2.4.5}$$

we obtain an equation for evolution of the probability density function, $\pi_t(x)$, that is,

$$\frac{\partial \pi_t(x)}{\partial t} = -\frac{\partial(\pi_t(x)f(x,t))}{\partial x} + \frac{1}{2}\frac{\partial^2(\pi_t(x_t)g^2(x,t))}{\partial x^2}. \tag{2.4.6}$$

For the vector equation, eq. (2.1.1), the corresponding Fokker-Planck equation is obtained by evolving each element according to eq. (2.4.6). This leads to

$$\begin{aligned}\frac{\partial \pi_t(x,t)}{\partial t} &= -\sum_{k=1}^{n}\frac{\partial(\pi_t(x,t)f_k(x,t))}{\partial x_k} + \frac{1}{2}\sum_{k,l=1}^{n}\frac{\partial^2(\pi_t(x,t)(g(x,t)g^{\mathrm{T}}(x,t))_{kl})}{\partial x_k \partial x_l} \\ &= \mathscr{L}\pi_t(x,t),\end{aligned} \tag{2.4.7}$$

where

$$\mathscr{L}\pi_t(x,t) = -\sum_{k=1}^{n}\frac{\partial(\pi_t(x,t)f_k(x,t))}{\partial x_k} + \frac{1}{2}\sum_{k,l=1}^{n}\frac{\partial^2(\pi_t(x,t)(g(x,t)g^{\mathrm{T}}(x,t))_{kl})}{\partial x_k \partial x_l}. \tag{2.4.8}$$

Solving the Fokker-Planck equation, eq. (2.4.7), yields the probability density function, $\pi_t(x)$. Analytical solutions have been provided for simple scalar systems [Spencer and Bergman (1993)], but most SDEs, especially those whose drift terms are nonlinear, are cumbersome to solve analytically. For this reason, numerical solutions are preferred. Finite element and finite difference methods, for example, have been used to solve the Fokker-Planck equation [Spencer and Bergman (1993); Kumar and Narayanan (2006)]. In the following, we outline the variational form of the Fokker-Planck equation; which forms the basis of a variety of numerical methods.

2.4.1 Weak form of the Fokker-Planck equation

For simplicity, consider the scalar Fokker-Planck equation, eq. (2.4.6), on the real line. Let $\phi(x) \in C_c^\infty(\mathbb{R},\mathbb{R})$; which is to say that $\phi(x)$ is an infinitely differentiable function from $\mathbb{R}$ to $\mathbb{R}$, whose compact support is the set $\mathbb{R}$.

Let the value of $\pi_t(x)$ at initial time, t_0, be given by

$$\pi_{t_0}(x) = \pi_0(x). \tag{2.4.9}$$

Now multiplying eq. (2.4.6) by $\phi(x)$ and integrating over the domain yields

$$\int \frac{\partial \pi_t(x)}{\partial t}\phi(x)dx = -\int \frac{\partial(\pi_t(x)f(x,t))}{\partial x}\phi(x)dx + \frac{1}{2}\int \frac{\partial^2(\pi_t(x)g^2(x,t))}{\partial x^2}\phi(x)dx. \tag{2.4.10}$$

Integrating eq. (2.4.10) by parts gives

$$\begin{aligned}\int \frac{\partial \pi_t(x)}{\partial t}\phi(x)dx &= \int \pi_t(x)f(x,t)\frac{\partial\phi(x)}{\partial x}dx - \frac{1}{2}\int \frac{\partial(\pi_t(x)g^2(x,t))}{\partial x}\frac{\partial\phi(x)}{\partial x}dx \\ &= \int \frac{\partial\phi(x)}{\partial x}\left(\pi_t(x)f(x,t) - \frac{1}{2}\frac{\partial(\pi_t(x)g^2(x,t))}{\partial x}\right)dx.\end{aligned} \tag{2.4.11}$$

Upon integrating the last term in the right-hand side of eq. (2.4.11) by parts we obtain,

$$\int \frac{\partial \pi_t(x)}{\partial t}\phi(x)dx = \int \frac{\partial\phi(x)}{\partial x}\pi_t(x)f(x,t)dx + \frac{1}{2}\int \frac{\partial^2\phi(x)}{\partial x^2}\pi_t(x)g^2(x,t)dx. \tag{2.4.12}$$

Finally, expressing eq. (2.4.12) using expected values,

$$\frac{\partial \mathbb{E}_t[\phi(x)]}{\partial t} = \mathbb{E}_t\left[\frac{\partial\phi(x)}{\partial x}f(x,t)\right] + \frac{1}{2}\mathbb{E}_t\left[\frac{\partial^2\phi(x)}{\partial x^2}g^2(x,t)\right]. \tag{2.4.13}$$

Now defining $\mathbb{E}_t[\phi]$ as

$$\begin{aligned}\mathbb{E}_t[\phi] &:= \pi_t[\phi] \\ &= \int \pi_t(x)\phi(x)dx,\end{aligned} \tag{2.4.14}$$

we then have

$$d\pi_t[\phi] = \pi_t[\mathscr{L}^*\phi]dt, \tag{2.4.15}$$

where the operator

$$\mathscr{L}^* := f\frac{\partial}{\partial x} + \frac{1}{2}g^2\frac{\partial^2}{\partial x^2}, \tag{2.4.16}$$

is expressible in vector form as follows

$$\mathscr{L}^* := f^{\mathrm{T}}\nabla + \frac{1}{2}\mathrm{tr}[gg^{\mathrm{T}}]\triangle, \tag{2.4.17}$$

where ∇ and $\triangle$ are as defined in eq. (2.3.5). Equation (2.4.15) is the *weak form* or the *variational form* of the Fokker-Planck equation. Different numerical methods arise from the choice of the *test function*, $\phi(x)$.

By definition, the operator $\mathscr{L}^*$ has an adjoint, $\mathscr{L}$, satisfying

$$\langle \mathscr{L}^*\phi, \pi_t\rangle = \langle \phi, \mathscr{L}\pi_t\rangle, \tag{2.4.18}$$

where $\langle\cdot,\cdot\rangle$ is the inner product. From eqs. (2.4.14) and (2.4.15),

$$\begin{aligned}\frac{d}{dt}\int \pi_t(x)\phi(x)dx &= \int \pi_t\mathscr{L}^*\phi(x)dx \\ &= \int \phi(x)\mathscr{L}\pi_t dx,\end{aligned} \tag{2.4.19}$$

which, upon dividing through by ϕ, integrals being common, yields the Fokker-Planck equation, eq. (2.4.7).

2.4.2 Equation of evolution of mean and covariance

The weak form of the Fokker-Planck equation, eq. (2.4.15), is the equation of evolution of moments, $\phi(x)$, with respect to the density, $\pi(x)$. By definition, the mean is given by the first moment and the covariance is given by the second central moment, respectively. We first consider the scalar forms of the mean and variance for ease in manipulation. To obtain the equation of evolution of the mean, $\hat{x}_t$, we substitute x_t in the place of $\phi(x)$ in the scalar form of eq. (2.4.15); which, in scalar form, is

$$d\hat{x}_t = \hat{f}_t dt, \tag{2.4.20}$$

where we have used the notation $\hat{x}_t = \pi_t[x] := \int x\pi_t(x)dx$ and $\hat{f}_t = \pi_t[f(x,t)] := \int f(x,t)\pi_t(x)dx$.

Let p_t denote the variance of the mean $\hat{x}_t$ at time t; that is,

$$p_t = \pi_t[x_t^2] - \hat{x}_t^2, \tag{2.4.21}$$

whence

$$dp_t = d\pi_t[x_t^2] - d\hat{x}_t^2. \tag{2.4.22}$$

Substituting x_t^2 in the place of $\phi(x)$ in the scalar form of eq. (2.4.15) yields

$$d\pi_t[x_t^2] = 2\pi_t[xf]dt + \pi_t[g^2]dt. \tag{2.4.23}$$

Note that by differentiation and the use of eq. (2.4.20)

$$d\hat{x}_t^2 = 2\hat{x}_t d\hat{x}_t = 2\hat{x}_t\hat{f}_t dt. \tag{2.4.24}$$

Rewriting eq. (2.4.22) with substitutions in eqs. (2.4.23) and (2.4.24) gives

$$dp_t = 2\hat{x}_t\hat{f}_t dt + 2\pi_t[xf]dt + \pi_t[g^2]dt, \tag{2.4.25}$$

which is the equation of evolution of the variance.

Now we extend the scalar forms of the mean and variance to vector form by noting that each element in the mean vector and covariance matrix will evolve according to eqs. (2.4.20) and (2.4.25), respectively. Therefore, the equations of evolution of mean vector, $\hat{x}_t$, and covariance matrix, P_t, are, respectively,

$$d\hat{x}_t = \hat{f}_t dt, \tag{2.4.26a}$$

$$dP_{tkl} = (\pi_t\left[x_k f_k\right] - \hat{x}_{kt}\hat{f}_{kt})dt + (\pi_t\left[f_l x_l\right] - \hat{f}_{lt}\hat{x}_l)dt + \pi_t\left[(gg^{\mathrm{T}})_{kl}\right]dt, \tag{2.4.26b}$$

where by the subscripts kl is signified an element in kth row and lth column, respectively.

In the next section, we briefly review some numerical methods for solving stochastic differential equations. An example is supplied at the end of the section.

2.5 Numerical Solution Methods for SDEs

Most SDEs do not have analytical solutions owing to the stochastic integrals involved and the nonlinearity of the drift and the diffusion functions. For this reason, numerical approximations provide estimates of the solutions. In this thesis, we shall consider numerical methods obtained by Taylor's series expansion, namely, Euler-Maruyama method, Milstein method, together with their variants.

2.5.1 Euler-Maruyama method

Consider the one-dimensional Itô stochastic differential equation

$$dx_t = f(x_t, t)dt + g(x_t, t)d\beta_t; \quad t_0 \le t, \tag{2.5.1a}$$

whose solution is

$$x_t = x_{t_0} + \int_{t_0}^{t} f(x_\tau, \tau)d\tau + \int_{t_0}^{t} g(x_\tau, \tau)d\beta_\tau; \qquad t \in [t_0, T]. \tag{2.5.1b}$$

Named after Leonard Euler and Gisiro Maruyama, Euler-Maruyama method is an extension of Euler method for approximating ordinary differential equations to stochastic differential equations. Euler-Maruyama approximation for eq. (2.5.1b) is obtained by first partitioning the time domain into, say, N subintervals $[t_n, t_{n+1}]$ of equal length, $\delta t = t_{n+1} - t_n$, $n = 0, 1, 2, 3, ..., N$, and then by approximating the functions f and g by using Taylor series expansion upto a constant term. The recursion

$$\tilde{x}_{t_{n+1}} = \tilde{x}_{t_n} + f(\tilde{x}_{t_n}, t_n)\delta t + g(\tilde{x}_{t_n}, t_n)\delta\beta_{t_n}, \tag{2.5.2}$$

where $\delta\beta_{t_n} = \int_{t_n}^{t_{n+1}} d\beta_\tau = \beta_{t_{n+1}} - \beta_{t_n} \approx \mathcal{N}(0, \delta t)$ and $\tilde{x}_{t_n}$ approximates x_{t_n} is called Euler-Maruyama method for eq. (2.5.1b).

It is of much interest to ascertain how best $\tilde{x}_{t_n}$ approximates x_{t_n} and, since this depends on the time step, convergence of $\tilde{x}_{t_n}$ to x_{t_n}. Note that $\{\tilde{x}_{t_n}\}$ is a random process. There are two ways of investigating convergence: strong convergence and weak convergence. Strong convergence is obtained by calculating the root-mean-square error (RMSE)

$$\sup_{0 \le t_n \le T} \mathbb{E}\left[\|x_{t_n} - \tilde{x}_{t_n}\|^2\right]^{1/2} \le K\delta t^p, \tag{2.5.3}$$

where $K > 0$ is a constant and p is the order of convergence.

Weak convergence, on the other hand, is obtained by means of distributions; that is, we obtain, say, M-sized ensemble $\{\tilde{x}^i_{t_N}\}_{i=1}^M$ of $\tilde{x}_{t_N}$, which is an approximation of x_{t_N}. Weak convergence is then ascertained by

$$|\mathbb{E}\left[\phi(\tilde{x}_{t_N})\right] - \mathbb{E}\left[\phi(x_{t_N})\right]| \le K\delta t^p, \tag{2.5.4}$$

where, as before, $K > 0$ is a constant, ϕ is a smooth test function, and p is the order of convergence. We approximate $\mathbb{E}\left[\phi(x_{t_N})\right]$ by

$$\mathbb{E}\left[\phi(x_{t_N})\right] \approx \frac{1}{M}\sum_{i=1}^{M} \phi(x^i_{t_N}). \tag{2.5.5}$$

Euler-Maruyama method has $p = 0.5$ strong order of convergence and $p = 1$ weak order of convergence [Lord et al. (2014); Schaffter (2010)].

2.5.2 Milstein method

Milstein method is a modification of the Euler-Maruyama method and is obtained by adding one more term in the Taylor series expansion of the function g. It is named after Grigori N. Milstein. Milstein method recursively approximates eq. (2.5.1b) as follows

$$\tilde{x}_{t_{n+1}} = \tilde{x}_{t_n} + f(\tilde{x}_{t_n}, t_n)\delta t + g(\tilde{x}_{t_n}, t_n)\delta\beta_{t_n} + \frac{1}{2}\frac{\partial g(\tilde{x}_{t_n}, t_n)}{\partial x} g(\tilde{x}_{t_n}, t_n)(\delta\beta^2_{t_n} - \delta t), \tag{2.5.6}$$

where $\delta\beta_{t_n} = \beta_{t_{n+1}} - \beta_{t_n} \approx \mathcal{N}(0, \delta t)$. With $p = 1$ strong and weak orders of convergence, Milstein method converges faster in the strong sense than Euler-Maruyama method.

2.5.3 θ-Euler-Maruyama method

This is an implicit variant of Euler-Maruyama method, obtained by introducing an implicitness parameter, $\theta \in [0, 1]$, in the drift term. The θ-Euler-Maruyama scheme is given by:

$$\tilde{x}_{t_{n+1}} = \tilde{x}_{t_n} + \big(\theta f(\tilde{x}_{t_{n+1}}, t_{n+1}) + (1-\theta) f(\tilde{x}_{t_n}, t_n)\big)\delta t + g(\tilde{x}_{t_n}, t_n)\delta\beta_{t_n}. \tag{2.5.7}$$

When $\theta = 0$, then $\theta-$Euler-Maruyama scheme becomes Euler-Maruyama scheme. The weak and strong orders of convergence of the $\theta-$Euler-Maruyama scheme when $\theta = 0.5$ are, respectively, $p = 1.0$ and $p = 0.5$ [Kloeden and Platen (1992); Kloeden et al. (1994)].

2.5.4 Stratonovich-Milstein method

The methods described heretofore pertain to the Itô stochastic differential equation, eq. (2.5.1a). In the following, extension of Euler-Maruyama and Milstein methods to the Stratonovich stochastic differential equation

$$dx_t = f(x_t, t)dt + g(x_t, t) \circ d\beta_t; \quad t_0 \le t, \tag{2.5.8a}$$

whose Itô equivalent is

$$dx_t = f(x_t)dt + \frac{1}{2}g(x_t)\partial_x[g(x_t)]dt + g(x_t)d\beta_t, \tag{2.5.8b}$$

the solution of which is

$$x_t = x_{t_0} + \int_{t_0}^{t}\left(f(x_\tau, \tau) + \frac{1}{2}g(x_\tau, \tau)\partial_x[g(x_\tau, \tau)]\right)d\tau + \int_{t_0}^{t} g(x_\tau, \tau)d\beta_\tau, \tag{2.5.8c}$$

are introduced.

The Stratonovich- Milstein method is obtained by using $f(\tilde{x}_{t_n}, t_n) + \frac{1}{2}g(\tilde{x}_{t_n}, t_n)\partial_x[g(\tilde{x}_{t_n}, t_n)]$ in the place of $f(\tilde{x}_{t_n}, t_n)$ in Milstein method above. The iterative equation becomes

$$\begin{aligned}\tilde{x}_{t_{n+1}} &= \tilde{x}_{t_n} + \left(f(\tilde{x}_{t_n}, t_n) + g(\tilde{x}_{t_n}, t_n)\delta\beta_{t_n} + \frac{1}{2}\frac{\partial g(\tilde{x}_{t_n}, t_n)}{\partial x}g(\tilde{x}_{t_n}, t_n)\right)\delta t \\ &\quad + \frac{1}{2}\frac{\partial g(\tilde{x}_{t_n}, t_n)}{\partial x}g(\tilde{x}_{t_n}, t_n)(\delta\beta^2 - \delta t) \\ &= \tilde{x}_{t_n} + f(\tilde{x}_{t_n}, t_n)\delta t + g(\tilde{x}_{t_n}, t_n)\delta\beta_{t_n} + \frac{1}{2}\frac{\partial g(\tilde{x}_{t_n}, t_n)}{\partial x}g(\tilde{x}_{t_n}, t_n)\delta\beta_{t_n}^2.\end{aligned} \tag{2.5.9}$$

Just like the Milstein method, the Stratonovich-Milstein method has $p = 1$ strong and weak orders of convergence.

2.5.5 Heun's method

Heun's approximation is an extension of Euler-Maruyama method to Stratonovich SDEs. It is obtained by replacing the last term in the Euler-Maruyama scheme—thus

$$\tilde{x}_{t_{n+1}} = \tilde{x}_{t_n} + f(\tilde{x}_{t_n}, t_n)\delta t + \frac{1}{2}\big(g(\bar{x}_{t_{n+1}}, t_{n+1}) + g(\tilde{x}_{t_n}, t_n)\big)\delta\beta_{t_n}, \tag{2.5.10a}$$

with $\bar{x}_{t_{n+1}}$ precomputed as follows:

$$\bar{x}_{t_{n+1}} = \tilde{x}_{t_n} + g(\tilde{x}_{t_n}, t_n)\delta\beta_{t_n}. \tag{2.5.10b}$$

Heun's method has $p = 0.5$ strong order of convergence. It is computationally more efficient compared to Milstein method because it does not involve computation of the derivative of the diffusion term.

Example 2.5.1: Geometric Brownian motion

Geometric Brownian motion is a continuous-time SDE used to model stock prices, denoted by x_t at a given time t. It is in the form of eq. (2.1.1) where, in the scalar case, the drift term is μx_t and the diffusion term is σx_t; that is to say,

$$dx_t = \mu x_t dt + \sigma x_t d\beta_t, \quad t \geq 0, \tag{2.5.11a}$$

in which case μ is the interest rate of an asset and σ is its volatility. By volatility is meant the measure of variability of returns of a given stock or security; so that higher volatility figures are indicative of a riskier return. The Itô equation, eq. (2.5.11a), has an analytical solution, (see Øksendal (2005) for details)

$$x_t = x_{t_0} \exp\left((\mu - \sigma^2/2)t + \sigma\beta_t\right), \tag{2.5.11b}$$

where $\beta(t)$ is Brownian motion process. The Stratonovich equation

$$dx_t = \mu x_t dt + \sigma x_t \circ d\beta_t, \quad t \geq 0, \tag{2.5.11c}$$

has an equivalent Itô representation

$$dx_t = \left(\mu + \frac{1}{2}\sigma^2\right) x_t dt + \sigma x_t d\beta_t, \tag{2.5.11d}$$

and its analytical solution is

$$x_t = x_{t_0} \exp\left(\mu t + \sigma\beta_t\right). \tag{2.5.11e}$$

The integral equation for eq. (2.5.11d), after dividing through by x_t, is

$$\begin{aligned} \int_{t_0}^{t} \frac{dx_t}{x_t} &= \left(\mu + \frac{1}{2}\sigma^2\right) \int_{t_0}^{t} dt + \sigma \int_{t_0}^{t} d\beta_t \\ &= \left(\mu + \frac{1}{2}\sigma^2\right) t + \sigma\beta_t, \end{aligned} \tag{2.5.12}$$

where $t_0 = 0$ and $\beta_{t_0} = 0$. By Itô's formula,

$$d(\ln x_t) = \frac{d}{dx}(\ln x_t) dx_t + \frac{1}{2}\frac{d^2}{dx^2}(\ln x_t)(dx_t)^2. \tag{2.5.13}$$

Now, in the limit $\delta t \to 0$ [Kushner (1962)],

$$(dx_t)^2 = \sigma^2 x_t^2 dt, \tag{2.5.14}$$

and we then, from eq. (2.5.13), have

$$\begin{aligned} d(\ln x_t) &= \frac{dx_t}{x_t} - \frac{\sigma^2 x_t^2}{2x_t^2} dt \\ &= \frac{dx_t}{x_t} - \frac{\sigma^2}{2} dt, \end{aligned} \tag{2.5.15}$$

and hence

$$\frac{dx_t}{x_t} = d(\ln x_t) + \frac{\sigma^2}{2} dt. \tag{2.5.16}$$

Substituting eq. (2.5.16) in eq. (2.5.12), we get

$$\int_{t_0}^{t} d(\ln x_t) + \int_{t_0}^{t} \frac{\sigma^2}{2} dt = \left(\mu + \frac{1}{2}\sigma^2\right) t + \sigma\beta_t, \tag{2.5.17}$$

from which we obtain

$$\ln\left(\frac{x_t}{x_{t_0}}\right) + \frac{1}{2}\sigma^2 t = \left(\mu + \frac{1}{2}\sigma^2\right) t + \sigma\beta_t, \tag{2.5.18}$$

and hence eq. (2.5.11e). Obtaining eq. (2.5.11b) from eq. (2.5.11a) follows the same argument.

In the following, we approximate strong convergence of the numerical methods described heretofore using the geometric Brownian motion. The approximation is done as follows:

$$\sup_{0\le t_n\le T} \mathbb{E}\left[\|x_{t_n} - \tilde{x}_{t_n}\|^2\right]^{1/2} \approx \left(\frac{1}{M}\sum_{k=1}^{M} \|x_{t_n}^k - \tilde{x}_{t_n}^k\|^2\right)^{1/2}, \tag{2.5.19}$$

where M is the number of samples. We obtain the root mean square errors for different time-steps, for different methods, and plot the results.

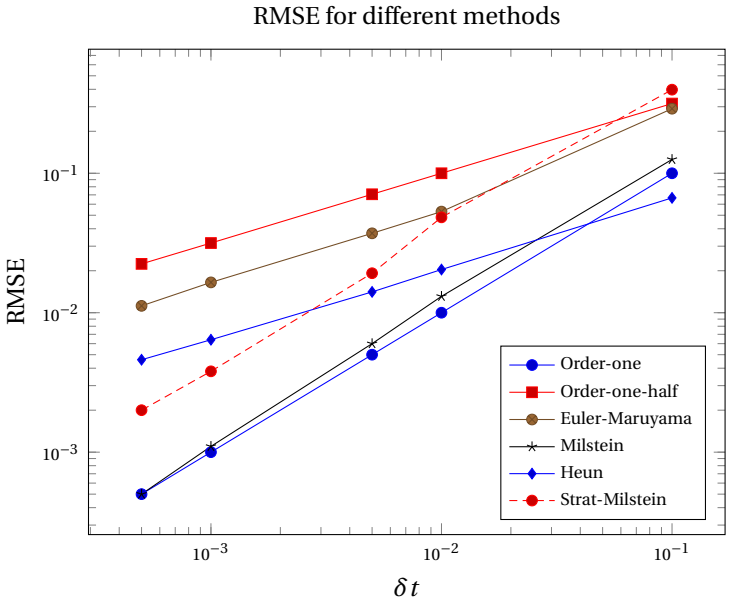


Figure 2.1: Loglog plot showing numerical approximation of strong order of convergence for different methods. The theoretical results are indicated by $o(\delta t)$ and $o(\delta t^{1/2})$, respectively. In each of the plots, $M = 10000$ samples are used. The reference solution is obtained with $\delta t = 0.0001$ while the approximate solutions are obtained with time steps 0.0005, 0.001, 0.005, 0.01 and 0.1, respectively. The numerically approximated errors agree remarkably with the theoretical values for all the methods considered.

CHAPTER 3

FILTERING PROBLEM

3.1 Introduction

As has been indicated in Chapter 2, the model equations, in so far as they are formulated to reflect the reality, usually fall short of this intended purpose on account of uncertainties. The implication is that the models cannot be entirely relied upon to give estimates of the state for all time—mainly because of the accumulation of model errors which eventually lead to unreliable estimates of the state. It is natural, therefore, to put in place a corrective mechanism to stem the accumulation of model errors. One of the ways of achieving this is by means of measurements of the state; that is, measurement implements are employed to collect data of the state and use this data, in one way or other, to check the findings from model dynamics. It would be hoped that the tools used for measuring the state are perfect, but this is unrealistic: the measuring skill is, as the model, subject to errors. The measurement model, thus, consists of a function of the state and time, say $h(x_t, t)$, and a quantification of the errors in measurements, say η_t—and this at any given time, t. This gives a time-continuous system comprising of the model dynamics, eq. (2.1.1), and that of measurements, both of which we now write for the convenience of the reader.

$$\text{Signal:} \quad dx_t = f(x_t, t)dt + g(x_t, t)d\beta_t; \quad t_0 \le t, \tag{3.1.1a}$$

$$\text{Measurement:} \quad dy_t = h(x_t, t)dt + R^{1/2}(t)d\eta_t; \quad t_0 \le t, \tag{3.1.1b}$$

where:

Term	Name	Dimension
y_t	output vector	$r \times 1$
$h(x_t, t)$	sensor function	$r \times 1$
$R(t)$	time-function matrix	$r \times r$
$\{\eta_t,\ t > t_0\}$	standard Brownian motion process	$r \times 1$

The terms appearing in the signal equation retain the same meaning and dimensions as indicated in Section 2.1.

This chapter is devoted to describing the procedure for combining the noisy measurements with the noisy dynamics with a view of obtaining the best estimate, $\hat{x}_t$, of the state, x_t, at any given time, t. A number of techniques have been proposed for doing this: the least squares approach [Lewis et al. (2006)], the variational approach [Mitter and Newton (2003)], and, among others, the Bayesian approach [Reich and Cotter (2015); Law et al. (2015)]. In this thesis, we take up the Bayesian approach.

3.2 Statement of the problem

Consider the signal process, which is obtained by integrating eq. (3.1.1a); that is,

$$x_t = x_{t_0} + \int_{t_0}^{t} f(x_\tau, \tau) d\tau + \int_{t_0}^{t} g(x_\tau, \tau) d\beta_\tau; \quad t_0 \le \tau \le t. \tag{3.2.1a}$$

It turns out, however, that we do not fully acces the signal. Suppose, instead, we can take measurements, the output of which is obtained by integrating (3.1.1b), that is,

$$y_t = y_{t_0} + \int_{t_0}^{t} h(x_\tau, \tau) d\tau + \int_{t_0}^{t} R^{1/2}(\tau) d\eta_\tau; \quad t_0 \le \tau \le t. \tag{3.2.1b}$$

Then the filtering problem is defined as follows:

Definition 3.2.1. ***Time-continuous filtering problem:—*** *What is the* current *best estimate of the state, x_t, given measurements $Y_t := y_{[t_0,t]}$?*

By $y_{[t_0,t]}$ we denote the continuum of measurements from the initial time t_0 to the time of interest t. The filtering problem is best solved by considering the conditional density of the state, x_t, given the measurements, Y_t; that is, $\pi_t(x \mid Y_t)$. Better still, it is preferable to consider the evolution of the conditional density, $\pi_t(x \mid Y_t)$. The evolution of the conditional density is provided for in the *Kushner-Stratonovich equation*, which we consider in the next section, together with a sketch of its proof.

Closely related to the filtering problem are the prediction and smoothing problems, the difference being the time reference of the estimate in question. We now state the prediction and smoothing problems.

Definition 3.2.2. ***Time-continuous prediction problem:—*** *What is the* future *best estimate of the state, x_t, given measurements $Y_\tau := y_{[t_0,\tau<t]}$? The prediction problem is characterised by finding the conditional density $\pi_t(x \mid Y_\tau)$, $\tau < t$.*

Definition 3.2.3. ***Time-continuous smoothing problem:—*** *What is the* past *best estimate of the state, x_t, given measurements $Y_\tau := y_{[t_0,\tau>t]}$? The conditional density $\pi_t(x \mid Y_\tau)$, $\tau > t$, is called the smoothing density.*

By the best estimate, we understand the estimate whose variance with the true value is minimal.

3.3 Kushner-Stratonovich Equation

The Kushner-Stratonovich equation can be intuitively comprehended as an additive perturbation of the Fokker-Planck equation, eq. (2.4.7), which perturbation consists of contributions arising from knowledge obtained from measurements. Use is made of Bayes' rule to quantify the additional information resulting from measurements.

Theorem 3.3.1. *Suppose that the functions f, g and h are real and that they have finite lower and upper bounds for finite values of x. Suppose, moreover, that the functions f, g and h satisfy uniform Lipschitz condition in x and $t \in [t_0, T]$. Finally, suppose that x_{t_0} is finite in the mean square sense and that it is independent of $\{d\beta_t\}$. For the system of eqs.* (3.1.1a) *and* (3.1.1b)*, the conditional density evolves as follows.*

$$\begin{aligned} \pi_t(x \mid Y_t) = \pi_{t_0}(x) &+ \int_{t_0}^{t} \mathscr{L}(\pi_\tau(x \mid Y_\tau)) d\tau \\ &+ \int_{t_0}^{t} \pi_\tau(x \mid Y_\tau)(h - \hat{h}_\tau)^{\mathrm{T}} R^{-1}(\tau)(dy_\tau - \hat{h}_\tau d\tau), \end{aligned} \tag{3.3.1}$$

where the operator $\mathscr{L}$ is the adjoint of $\mathscr{L}^*$, defined in eq. (2.4.17), and

$$\hat{h}_t = \int h(x,t)\pi_t(x \mid Y_t)dx.$$

Equation (3.3.1) is called *Kushner-Stratonovich equation*[2]. In the absence of observations, that is, when $R^{-1}(t) \equiv 0$, we obtain the *Fokker-Planck equation*.

Proof. We do not intend to give a complete proof; instead, we only provide here a sketch of the proof, and suggest Kushner (1962); Jazwinski (1970) for the full proofs. Now for convenience, we rewrite eq. (3.3.1) in its differential form—thus:

$$d\pi_t = \mathscr{L}(\pi_t)dt + \pi_t(h-\hat{h}_t)^{\mathrm{T}}R^{-1}(t)(dy_t - \hat{h}_t dt), \tag{3.3.2}$$

and recall that the conditional density, π_t, is the notationally convenient way of writing $\pi_t(x \mid Y_t)$ — by which is meant the conditional probability density function of the state, x_t, at time, t, given all the measurements from initial time, t_0, to the time of interest, t. After an infinitesimal change in time, δt, a corresponding infinitesimal measurement, δy_t, is acquired according to the measurements equation, eq. (3.1.1b); similarly, an infinitesimal change in the state, δx_t, is realised.

The idea of the proof, then, is to consider the infinitesimal change in the conditional density function, $\delta\pi_t$, as a result of the new information on both the signal and the measurements, which can be expressed as follows:

$$\begin{aligned}\delta\pi_t &= [\delta\pi_t]_{\text{signal}} + [\delta\pi_t]_{\text{measurement}},\\ &= [\pi_{t+\delta t}(x \mid Y_t + \delta y_t) - \pi_t(x \mid Y_t + \delta y_t)] + [\pi_t(x \mid Y_t + \delta y_t) - \pi_t(x \mid Y_t)].\end{aligned} \tag{3.3.3}$$

We notice that taking limits in eq. (3.3.3) as $\delta t \to 0$ yields

$$[d\pi_t]_{\text{signal}} = \mathscr{L}(\pi_t)dt, \tag{3.3.4a}$$

and

$$\begin{aligned}[d\pi_t]_{\text{measurement}} &= [\pi_t(x \mid Y_t + dy_t) - \pi_t(x \mid Y_t)]\\ &= \pi_t(h-\hat{h}_t)^{\mathrm{T}}R^{-1}(t)(dy_t - \hat{h}_t dt).\end{aligned} \tag{3.3.4b}$$

Since eq. (3.3.4a) is already provided for by the Fokker-Planck equation, it only remains to establish eq. (3.3.4b).

From the measurement's equation, eq. (3.1.1b), we have an expression for an infinitesimal measurement

$$\delta y_t = h(x_t)\delta t + R^{1/2}(t)\delta\eta_t, \quad t_0 \le t, \tag{3.3.5a}$$

and notice that

$$\mathbb{E}\left[\delta y_t \delta y_t^{\mathrm{T}}\right] = R(t)\delta t + o(\delta t). \tag{3.3.5b}$$

At the availability of infinitesimal measurement, δy_t, we get a new value for the transition density, $\pi_t(x \mid Y_t + \delta y_t)$, which is given by [Kushner (1962)]

$$\pi_t(x \mid Y_t + \delta y_t) = \frac{\pi_t(\delta y \mid x_t)\pi_t(x \mid Y_t)}{\int \pi_t(\delta y \mid x_t)\pi_t(x \mid Y_t)dx}, \tag{3.3.6a}$$

[2] It is named after the Mathematicians Harold J. Kushner and Ruslan Stratonovich.

where

$$\pi_t(\delta y \mid x_t) \sim \mathcal{N}\left(h(x_t)\delta t,\, R(t)\delta t\right). \tag{3.3.6b}$$

We define a function $q(\delta t, \delta y_t)$ as follows

$$q(\delta t, \delta y_t) := \frac{\pi_t(x \mid Y_t + \delta y_t)}{\pi_t(x \mid Y_t)}. \tag{3.3.7}$$

Then, we approximate $q(\delta t, \delta y_t)$ by Taylor series expansion, about $(0,0)$, up to the first derivative with respect to δt and the second derivative with respect to δy_t and neglect terms involving mixed derivatives. The terms $\delta y_t \delta y_t^{\mathrm{T}}$ are replaced with their expectation, which yields

$$q(\delta t, \delta y_t) = 1 + (h - \hat{h}_t)^{\mathrm{T}} R^{-1} (\delta y_t - \hat{h}_t \delta t) + o(\delta t). \tag{3.3.8}$$

From eqs. (3.3.7) and (3.3.8),

$$\pi_t(x \mid Y_t + \delta y_t) = \pi_t(x \mid Y_t) + (h - \hat{h}_t)^{\mathrm{T}} R^{-1} (\delta y_t - \hat{h}_t \delta t)\pi_t(x \mid Y_t) + o(\delta t). \tag{3.3.9}$$

By passing to the formal limit as $\delta t \to 0$ yields eq. (3.3.4b) as desired. □

3.3.1 Weak form of Kushner-Stratonovich Equation

Let $\phi(x)$ take up the assumptions of Section 2.4.1. By integration by parts, following the derivation of the weak formulation of the Fokker-Planck equation in Section 2.4.1, the variational form of the scalar Kushner-Stratonovich equation (henceforth K-S equation) is

$$\begin{aligned}\int \partial \pi_t \phi(x) \partial x = &- \int \frac{\partial(\pi_t f(x,t))}{\partial x} \phi(x) \partial t \partial x + \frac{1}{2} \frac{\partial^2(\pi_t g^2(x,t))}{\partial x^2} \phi(x) \partial t \partial x \\ &+ \int \phi(x) \pi_t (h - \hat{h}_t) R^{-1}(t) (\partial y_t - \hat{h}_t \partial t) \partial x.\end{aligned} \tag{3.3.10}$$

Integrating eq. (3.3.10) by parts, we get

$$\begin{aligned}\int \partial \pi_t \phi(x) \partial x = &\int \frac{\partial \phi(x)}{\partial x} \pi_t f(x,t) \partial t \partial x + \frac{1}{2} \int \frac{\partial^2 \phi(x)}{\partial x} \pi_t g^2(x,t) \partial t \partial x \\ &+ \int \pi_t (h - \hat{h}_t) R^{-1}(t) (\partial y_t - \hat{h}_t \partial t) \phi(x) \partial x.\end{aligned} \tag{3.3.11}$$

Expressing eq. (3.3.11) in terms of expectation yields

$$\begin{aligned}d\mathbb{E}[\phi(x)] = &\mathbb{E}\left[\frac{\partial \phi(x)}{\partial x} f(x,t)\right] dt + \frac{1}{2} \mathbb{E}\left[\frac{\partial^2 \phi(x)}{\partial x} g^2(x,t)\right] dt \\ &+ \left(\mathbb{E}[\phi(x) h] - \hat{\phi} \hat{h}_t\right) R^{-1}(t) (dy_t - \hat{h}_t dt).\end{aligned} \tag{3.3.12}$$

To obtain the vector form of eq. (3.3.12) we use the notations introduced in eqs. (2.4.15) to (2.4.17) and get

$$d\pi_t[\phi] = \pi_t[\mathscr{L}^* \phi] dt + (\pi_t[\phi(x) h] - \hat{\phi} \hat{h}_t)^{\mathrm{T}} R^{-1}(t) (dy_t - \hat{h}_t dt)], \tag{3.3.13}$$

which is the weak form of K-S equation.

3.4 Evolution of conditional mean and covariance

Mean and covariance correspond to the first and second moments, respectively. A moment, by definition, is the expectation of a function, say, $\phi(x)$, with respect to a given probability density. To specialise this to our scalar case, let $\phi(x)$ be scalar function of the state $x \in \mathbb{R}$, which function is twice differentiable with respect to the elements of the state vector. The moment with respect to the conditional density $\pi_t(x \mid Y_t)$ is

$$\pi_t[\phi(x)] = \int \phi(x)\pi_t(x \mid Y_t)dx, \tag{3.4.1}$$

whose evolution equation is given by the weak form of the K-S equation, eq. (3.3.13).

For convenience in deriving the equations of evolution of conditional mean and covariance, we use the scalar case of eq. (3.3.2); that is,

$$d\hat{\phi}_t(x) = \pi_t\left[f\frac{\partial\phi}{\partial x}\right]dt + \frac{1}{2}\pi_t\left[g^2\frac{\partial^2\phi}{\partial x^2}\right]dt + (\pi_t[\phi h] - \hat{\phi}_t\hat{h}_t)R^{-1}(t)(dy_t - \hat{h}_t dt), \tag{3.4.2}$$

where we have used the notation $\hat{\phi}_t(x) = \pi_t[\phi(x)]$. Now to obtain the evolution equation for the mean, we substitute $\phi(x) = x$ in eq. (3.4.2), which yields

$$d\hat{x}_t = \hat{f}_t dt + (\pi_t[xh] - \hat{x}_t\hat{h}_t)R^{-1}(t)(dy_t - \hat{h}_t dt). \tag{3.4.3}$$

With the mean at hand, we define conditional variance as follows [Jazwinski (1970)]

$$\begin{aligned} p_t &= \mathbb{E}\left[(x - \hat{x}_t)^2 \mid Y_t\right] \\ &= \pi_t[x^2] - \hat{x}_t^2. \end{aligned} \tag{3.4.4a}$$

It immediately follows that

$$dp_t = d\pi_t[x^2] - d\hat{x}_t^2. \tag{3.4.4b}$$

Now it remains to find the expressions for $d\pi_t[x^2]$ and $d\hat{x}_t^2$. Substituting $\phi(x) = x^2$ in eq. (3.4.2) yields

$$\begin{aligned} d\pi_t[x^2] = {} & 2\pi_t\left[xf\right]dt + \pi_t\left[g^2\right]dt \\ & + (\pi_t[x^2h] - \pi_t[x^2]\hat{h}_t)R^{-1}(t)(dy_t - \hat{h}_t dt). \end{aligned} \tag{3.4.5a}$$

Note that eq. (3.4.3) is stochastic, because of dy_t—which is given by the scalar form of eq. (3.1.1b). By inspection, the stochastic term in eq. (3.4.3) is

$$(\pi_t[xh] - \hat{x}_t\hat{h}_t)R^{-1/2}(t)d\eta_t.$$

We now use Itô's formula, eq. (2.3.7), to find the expression for $d\hat{x}_t^2$; that is

$$\begin{aligned} d\hat{x}_t^2 &= 2\hat{x}_t d\hat{x}_t + (\pi_t[xh] - \hat{x}_t\hat{h}_t)^2 R^{-1}(t)dt \\ &= 2\hat{x}_t\hat{f}dt + (2\hat{x}_t\pi_t[xh] - 2\hat{x}_t^2\hat{h}_t)R^{-1}(t)(dy_t - \hat{h}_t dt) \\ &\quad + (\pi_t[xh] - \hat{x}_t\hat{h}_t)^2 R^{-1}(t)dt, \end{aligned} \tag{3.4.5b}$$

where $d\hat{x}_t$ has been substituted according to eq. (3.4.3). Substituting eqs. (3.4.5a) and (3.4.5b) in eq. (3.4.4b) gives

$$\begin{aligned} dp_t = {} & (2\pi_t\left[xf\right] - 2\hat{x}_t\hat{f}_t)dt + \pi_t\left[g^2\right]dt - (\pi_t[xh] - \hat{x}_t\hat{h}_t)^2 R^{-1}(t)dt \\ & + (\pi_t[x^2h] - 2\hat{x}_t\pi_t[xh] - \pi_t[x^2]\hat{h}_t + 2\hat{x}_t^2\hat{h}_t)R^{-1}(t)(dy_t - \hat{h}_t dt). \end{aligned} \tag{3.4.6}$$

Equations (3.4.3) and (3.4.6) are the evolution equations for the conditional mean and variance, respectively, for the scalar time-continuous nonlinear model. To obtain the corresponding vector equations, we consider each element and notice that all the elements evolve according to eqs. (3.4.3) and (3.4.6). This gives

$$d\hat{x}_t = \hat{f}_t dt + (\pi_t[xh^{\mathrm{T}}] - \hat{x}_t\hat{h}_t^{\mathrm{T}})R^{-1}(t)(dy_t - \hat{h}_t dt), \tag{3.4.7a}$$

$$\begin{aligned} dP_{tkl} = {} & (\pi_t\left[x_k f_k\right] - \hat{x}_{kt}\hat{f}_{kt})dt + (\pi_t\left[f_l x_l\right] - \hat{f}_{lt}\hat{x}_l)dt + \pi_t\left[(gg^{\mathrm{T}})_{kl}\right]dt \\ & - (\pi_t[x_k h] - \hat{x}_{kt}\hat{h}_t)^{\mathrm{T}}R^{-1}(t)(\pi_t[hx_l] - \hat{h}_t\hat{x}_{lt})dt \\ & + (\pi_t[x_k x_l h] - \hat{x}_{kt}\pi_t[x_l h] - \hat{x}_{lt}\pi_t[x_k h] - \pi_t[x_k x_l]\hat{h}_t \\ & + 2\hat{x}_{kt}\hat{x}_{lt}\hat{h}_t)^{\mathrm{T}}R^{-1}(t)(dy_t - \hat{h}_t dt), \end{aligned} \tag{3.4.7b}$$

where the subscripts kl are understood to mean an element in kth row and lth column, respectively. The computation of expected values in eqs. (3.4.7a) and (3.4.7b) depends on higher order moments, and for this reason the exact solution to these equations is highly intractable. One can only approximate the solution by means of Taylor series expansion, with some restrictive assumptions—one of which is Gaussianity; for then, all odd central moments are zero. In the linear case, as shall be seen in the next chapter, however, eqs. (3.4.7a) and (3.4.7b) have a closed form solution.

3.5 Numerical Solution to the filtering problem

As we have noted above, the K-S equation provides a law for the evolution of the probability density function of the state conditioned on the available measurements. In essence, the solution of K-S equation is the solution to the filtering problem. The K-S equation, eq. (3.3.1), is an integro-differential equation whose solution is highly dependent on the dimension of the state variable. Solving the K-S equation requires the propagation of a conditional probability density, $\pi_t(x_t \mid Y_t)$, which is, in most cases, of infinite dimension. This phenomenon is called the *curse of dimensionality*. It has been established [Jazwinski (1970); Kushner (1962); Bain and Crisan (2009)] that the solution to the K-S equation is, generally, infinite—with an exception of the linear model and certain highly restrictive conditions. For the linear model scenario, the solution to the filtering problem is expressible in closed form. This, as is detailed in Chapter 4, is completely characterised by the evolution of the mean vector, $\hat{x}_t$, of the state and that of the covariance matrix, P_t; both of which equations can easily be computed. In other words, the mean and the covariance are sufficient statistics in linear filtering.

Nonlinear filtering, however, is riddled with difficulties, one of them being the involvement of higher order moments in the equation of evolution of the first and second order moments. It turns out, however, that there are attempts to obtain approximations to the nonlinear filter, which attempts involve making assumptions on the drift term and the sensor function. A very good example of a nonlinear filter is the Beneš' filter [Bain and Crisan (2009); Beneš (1981)], in which the sensor function is required linear, and the drift term is constrained to obey the so-called Beneš' conditions. The Beneš' filter gives an exact solution to the filtering problem. However, the Beneš' conditions are very limiting so that the Beneš' filter cannot be applied to many practical problems.

With the success and simplicity of application of the linear filter, extensions of the linear filter have been made to nonlinear filtering. This is the basis of the extended Kalman Bucy filter [Särkkä (2013)], more of which hereafter. Other numerical approximation methods used in nonlinear filtering include the partial differential method, the projection filter, and the particle method [Bain and Crisan (2009)].

3.5.1 Monte Carlo Methods

The filtering problem, as has heretofore been established, entails calculation of the posterior density, $\pi_t(x \mid Y_t)$, which turns out to be equivalent to obtaining a solution to the K-S equation, eq. (3.3.1). Monte Carlo methods provide an approximation to the conditional density by means of empirical averages of, say, M ensembles. To be precise, Monte Carlo methods provide an estimate, $\hat{\pi}_t$, of the posterior density, $\pi_t(x \mid Y_t)$, at time t—which density, in the parlance of Monte Carlo methods, is known as *the target density*—as follows

$$\pi_t(x \mid Y_t) \approx \hat{\pi}_t(x) = \sum_{i=1}^{M} w_t^i \delta(x_t - x_t^i). \tag{3.5.1}$$

$\{x_t^i\}_{i=1}^M$ are mutually independent and identically distributed point masses or particles and $\{w_t^i\}_{i=1}^M$ are associated weights.

To illustrate the power of Monte Carlo methods, suppose we seek the expected value,

$$\mathbb{E}\big[\psi(x)\big] = \int \psi(x)\pi(x)dx, \tag{3.5.2}$$

with respect to a target distribution $\pi(x)$ for some function ψ. Furthermore, let $\{x^i\}_{i=1}^M$ be M samples identically distributed according to the target density $\pi(x)$. We can obtain an approximation of eq. (3.5.2) by sample average

$$\begin{aligned} \hat{\psi} &= \int \psi(x) \sum_{i=1}^{M} w^i \delta(x - x^i) dx, \\ &= \frac{1}{M} \sum_{i=1}^{M} \psi(x^i), \end{aligned} \tag{3.5.3a}$$

where

$$\frac{1}{M} \sum_{i=1}^{M} \delta(x - x^i) = \hat{\pi}, \tag{3.5.3b}$$

is an empirical approximation of the target density $\pi(x)$ with uniform weights $\{w^i = 1/M\}_{i=1}^M$. For all practical purposes, especially when the integral in eq. (3.5.2) are intractable, the Monte Carlo approximation, eq. (3.5.3a), is very convenient. Monte Carlo approximation, $\hat{\psi}$, is an unbiased estimate of $\mathbb{E}\big[\psi(x)\big]$. What is more, the strong law of large numbers indicates that $\hat{\psi}$ converges almost surely to $\mathbb{E}\big[\psi(x)\big]$ as $N \to \infty$. It has been shown, by means of Central Limit Theorem, that the order of convergence of Monte Carlo methods is $\mathcal{O}(M^{-1/2})$; which applies in spite of the dimension of x. For proofs of convergence, see, for example, del Moral (2004) and Bain and Crisan (2009).

It is a common occurrence, however, that drawing independent and identically distributed samples from the target distribution, $\pi(x)$, is difficult. The target distribution may be known up to a normalizing constant. Samples drawn from such a distribution, which is not fully known, would potentially lead to an erroneous estimate. It is for this reason that sampling techniques have been developed, among which are rejection sampling and importance sampling [Robert and Casella (2004)].

While the sampling techniques are important and useful, they are limited to low dimensional integration. A very good example is in integrals with respect to time, which are common in state space models. In this context, techniques that involve updating the particles sequentially are highly desired. Two techniques useful to this end are *particle methods*—also known

as *Sequential Monte Carlo (SMC) methods*—and *Markov Chain Monte Carlo (MCMC) methods*. We now outline an approximate particle filter for time-continuous models.

Partition the time domain $[t_0, T]$ into, say, N subintervals, $[t_n, t_{n+1}]$, each of equal length, $\delta t = t_{n+1} - t_n$, $n = 0, 1, 2, 3, ..., N$. Let $\{x^i_{t_0:t_N}, w^i_{t_N}\}_{i=1}^M$ be a particle-weight system which we want to use to characterise the full posterior density

$$\pi_{t_0:t_N}\left(x_{t_0:t_N} \mid \delta y_{t_1:t_N}\right),$$

where $x_{t_0:t_N} = (x_{t_0}^{\mathrm{T}}, x_{t_1}^{\mathrm{T}}, ... x_{t_N}^{\mathrm{T}})^{\mathrm{T}}$ and $\delta y_{t_0:t_N} = (\delta y_{t_0}^{\mathrm{T}}, \delta y_{t_1}^{\mathrm{T}}, ... \delta y_{t_N}^{\mathrm{T}})^{\mathrm{T}}$. We assume that we have increments of observations, δy_{t_n}, at any time t_n obtained from eq. (3.1.1b); that is,

$$\delta y_{t_n} = h(x_{t_n})\delta t + R^{1/2}(t_n)\delta\eta_{t_n}, \qquad n = 0, 1, 2, ..., N. \tag{3.5.4}$$

The state, x_{t_n}, at each time, t_n, is drawn from a transition density $\pi_{t_n}(x_{t_n} \mid x_{t_{n-1}})$, that is, according to

$$x^i_{t_{n+1}} = x^i_{t_n} + f(x^i_{t_n})\delta t + g(x^i_{t_n})\delta\beta^i_{t_n}, \tag{3.5.5}$$

where $\delta\beta^i_{t_n} \approx \mathcal{N}(0, \delta t)$.

The weighted approximation of the true posterior is then given by

$$\pi_{t_0:t_N}\left(x_{t_0:t_N} \mid \delta y_{t_0:t_N}\right) \approx \sum_{i=1}^{M} w^i_{t_N}\delta(x_{t_0:t_N} - x^i_{t_0:t_N}). \tag{3.5.6}$$

Suppose we cannot draw samples, $\{x^i_{t_0:t_N}\}_{i=1}^M$, from the target distribution, $\pi_{t_0:t_N}\left(x_{t_0:t_N} \mid \delta y_{t_0:t_N}\right)$, with any convenience. Let $\gamma_{t_0:t_N}\left(x_{t_0:t_N} \mid \delta y_{t_0:t_N}\right)$ be a density—with semblance to the target density—from which we can conveniently draw samples. Furthermore, let the samples, $\{x^i_{t_0:t_N}\}_{i=1}^M$, have weights

$$w^i_{t_N} \propto \frac{\pi_{t_0:t_N}\left(x^i_{t_0:t_N} \mid \delta y_{t_0:t_N}\right)}{\gamma_{t_0:t_N}\left(x^i_{t_0:t_N} \mid \delta y_{t_0:t_N}\right)}; \qquad i = 1, 2, 3, ..., M. \tag{3.5.7}$$

The computation of weights, $w^i_{t_N}$, as shown in eq. (3.5.7) requires the entire posterior, $\pi_{t_0:t_N}(x^i_{t_0:t_N} \mid \delta y_{t_0:t_N})$, and a whole set of measurements, $\delta y_{t_0:t_N}$. To necessitate online (that is to say, at the receipt of each measurement) computation of weights, we take up a sequential approach in which, firstly, we so assemble the proposal that it factorizes as follows

$$\begin{aligned}\gamma_{t_0:t_N}\left(x^i_{t_0:t_N} \mid \delta y_{t_0:t_N}\right) &= \gamma_{t_N}\left(x^i_{t_N} \mid x^i_{t_{N-1}}, \delta y_{t_N}\right)\gamma_{t_0:t_{N-1}}\left(x^i_{t_0:t_{N-1}} \mid \delta y_{t_0:t_{N-1}}\right), \\ &= \gamma_{t_0}\left(x^i_{t_0} \mid \delta y_{t_0}\right)\prod_{n=1}^{N}\gamma_{t_n}\left(x^i_{t_n} \mid x^i_{n-1}, \delta y_{t_n}\right).\end{aligned} \tag{3.5.8a}$$

Notice that the weights can now be expressed as follows

$$\begin{aligned} w^i_{t_N} &\propto \frac{\pi_{t_0:t_N}\left(x^i_{t_0:t_N} \mid \delta y_{t_0:t_N}\right)}{\gamma_{t_0:t_N}\left(x^i_{t_0:t_N} \mid \delta y_{t_0:t_N}\right)}; \qquad i = 1, 2, 3, ..., M \\ &= \frac{\pi_{t_N}\left(\delta y_{t_N} \mid x^i_{t_N}\right)\pi_{t_N}\left(x^i_{t_N} \mid x^i_{t_{N-1}}\right)}{\gamma_{t_N}\left(x^i_{t_N} \mid x^i_{t_{N-1}}, \delta y_{t_N}\right)}\frac{\pi_{t_0:t_{N-1}}\left(x^i_{t_0:t_{N-1}} \mid \delta y_{t_0:t_{N-1}}\right)}{\gamma_{t_0:t_{N-1}}\left(x^i_{t_0:t_{N-1}} \mid \delta y_{t_0:t_{N-1}}\right)} \\ &= \frac{\pi_{t_N}\left(\delta y_{t_N} \mid x^i_{t_N}\right)\pi_{t_N}\left(x^i_{t_N} \mid x^i_{t_{N-1}}\right)}{\gamma_{t_N}\left(x^i_{t_N} \mid x^i_{t_{N-1}}, \delta y_{t_N}\right)} w^i_{t_{N-1}} \\ &= w^i_{t_0}\prod_{n=1}^{N} w^i_{t_n}, \end{aligned} \tag{3.5.8b}$$

where

$$w^i_{t_n} \propto \frac{\pi_{t_n}\left(\delta y_{t_n} \mid x^i_{t_n}\right)\pi_{t_n}\left(x^i_{t_n} \mid x^i_{t_{n-1}}\right)}{\gamma_{t_n}\left(x^i_{t_n} \mid x^i_{t_{n-1}}, \delta y_{t_n}\right)} w^i_{t_{n-1}}. \tag{3.5.8c}$$

Interestingly, if we choose a proposal density

$$\gamma_{t_n}\left(x^i_{t_n} \mid x^i_{t_{n-1}}, \delta y_{t_n}\right) = \pi_{t_n}\left(x^i_{t_n} \mid x^i_{t_{n-1}}\right),$$

we obtain

$$\begin{aligned} w^i_{t_n} &= \pi_{t_n}\left(\delta y_{t_n} \mid x^i_{t_n}\right) w^i_{t_{n-1}}, \\ &\propto \exp\left(-\frac{1}{2\delta t}\left[\delta y_{t_n} - h(x^i_{t_n})\delta t\right]^{\mathrm{T}} R^{-1}(t_n)\left[\delta y_{t_n} - h(x^i_{t_n})\delta t\right]\right) w^i_{t_{n-1}} \\ &= \exp\left(-\frac{1}{2\delta t}\delta y^{\mathrm{T}}_{t_n} R^{-1}(t_n)\delta y_{t_n}\right) \times \exp\Big(-\frac{1}{2\delta t}(-\delta y^{\mathrm{T}}_{t_n} R^{-1}(t_n) h(x^i_{t_n})\delta t \\ &\quad - h^{\mathrm{T}}(x^i_{t_n})R^{-1}(t_n)\delta y_{t_n}\delta t + h^{\mathrm{T}}(x^i_{t_n})R^{-1}(t_n)h(x^i_{t_n})\delta t^2)\Big) w^i_{t_{n-1}} \\ &\propto \exp\Big(-\frac{1}{2\delta t}(-2\delta y^{\mathrm{T}}_{t_n} R^{-1}(t_n)h(x^i_{t_n})\delta t + h^{\mathrm{T}}(x^i_{t_n})R^{-1}(t_n)h(x^i_{t_n})\delta t^2)\Big) w^i_{t_{n-1}}, \end{aligned} \tag{3.5.9}$$

which forms the weights of the most basic particle filter known as the *Bootstrap particle filter*.

It turns out, however, that the weight particle system, $\{w^i_{t_n}, x^i_{t_n}\}$, generated from the densities, $\pi_{t_n}\left(y_{t_n} \mid x^i_{t_n}\right)$ and $\pi_{t_n}\left(x^i_{t_n} \mid x^i_{t_{n-1}}\right)$, respectively, have a drawback arising from deterioration of importance weights after a few iterations. Before long, the system is sustained by only one particle, or a few, leading to an exponential increase in variance [Doucet and Johansen (2011); Lindsten (2013)]. This is mitigated by use of resampling techniques, among which are multinomial resampling, systematic resampling and residual resampling (see, for example, Doucet and Johansen (2011) for description). The idea behind resampling is to revive the particles by replicating those of appreciable weights and culling off those of relatively lesser weight. Resampling is achieved by drawing a set $\{x^{ia}_{t_n}\}$ from the discrete approximation of the posterior

$$\pi_{t_n}(x_{t_n} \mid y_{t_n:t_0}) \approx \sum_{i=1}^{M} w^i_{t_n}\delta(x_{t_n} - x^i_{t_n}), \tag{3.5.10}$$

such that $\mathbb{P}(x^{ia}_{t_n} = x^i_{t_n}) = w^i_{t_n}$ [Arulampalam et al. (2002)].

The demerit of resampling, which is explained in more detail in Section 3.5.6.1, is that—useful as it is—it introduces additional covariances into the estimate thus undermining the accuracy of the filter; especially when resampling is applied frequently. To lessen this effect, resampling is applied only when weights deteriorate to a certain threshold. One common measure of deterioration of the weights is the effective sample size (ESS) approximated as follows

$$\mathrm{ESS}_{t_n} = \frac{1}{\sum_{i=1}^{M}(\tilde{w}^i_{t_n})^2}, \tag{3.5.11}$$

where $\{\tilde{w}^i_{t_n}\}_{i=1}^{M}$ are normalized weights.

Algorithm 3.5.1 Bootstrap particle filter

Input: δt, α, M, N, π_{t_0}, π_{t_n} and x_{t_0}.
Output: $\{x^i_{t_n}\}^{N,M}_{n,i=1}$.

1: Draw $x^i_{t_0} \sim \pi_{t_0}\left(x^i_{t_0} \mid x_{t_0}\right)$
2: Compute initial weights $w^i_{t_0} = \dfrac{1}{M}$
3: **for** $n = 1$ **to** N, $\delta t > 0$ **do**
4: Draw $x^i_{t_n} \sim \pi_{t_n}\left(x^i_{t_n} \mid x^i_{t_{n-1}}\right)$
5: Compute weights $w^i_{t_n} \sim \pi_{t_n}\left(\delta y_{t_n} \mid x^i_{t_n}\right) w^i_{t_{n-1}}$
6: Normalise the weights to obtain $\{\tilde{w}^i_{t_n}\}^M_{i=1}$
7: Compute $\text{ESS}_{t_n} = \dfrac{1}{\sum^M_{i=1}(\tilde{w}^i_{t_n})^2}$
8: **if** $\text{ESS}_{t_n} \leq \alpha$ **then**
9: Resample the particles
10: Set the weights to $w^i_{t_n} = \dfrac{1}{M}$
11: **end if**
12: **end for**

[3]

3.5.2 Second-order approximate nonlinear filter (SoAF)

The aim of approximate filters is to provide an approximation to the equations of evolution of mean, eq. (3.4.7a), and covariance, eq. (3.4.7b), by means of Taylor series expansion—about the conditional mean, $\hat{x}_t$—of the conditional expectation of functions on the right-hand-side of eqs. (3.4.7a) and (3.4.7b), respectively. What is more, an assumption is made—a Gaussian assumption—which eliminates all odd central moments, for then the approximation is simplified a great deal. In a more general way, we begin with a second-order approximation of the equations of evolution of conditional mean and covariance by Taylor series expansion of nonlinear terms about the mean estimate, $\hat{x}_t$. By second order approximation we mean approximation involving at most second order terms in the Taylor series expansion of functions about the mean estimate. Terms containing third order derivatives and above are eliminated. This leads to second-order approximate nonlinear filter (SoAF) equations. Then we shall only eliminate all the terms with second order derivatives from the second-order approximation equations to arrive at the first-order approximate filter equations. To simplify the exposition, use is made of the scalar case of the system of eqs. (3.1.1a) and (3.1.1b),

$$\text{Signal:} \quad dx_t = f(x_t)dt + g(x_t)d\beta_t; \quad x_{t_0} = x(0), \quad t_0 \leq t, \tag{3.5.12a}$$

$$\text{Measurement:} \quad dy_t = h(x_t)dt + R^{1/2}(t)d\eta_t; \quad t_0 \leq t, \tag{3.5.12b}$$

and the scalar equations of evolution of conditional mean and covariance, eqs. (3.4.3) and (3.4.6), respectively.

We summarise the result below and proceed to show how we arrived at it.

Theorem 3.5.1 (Second-order approximate nonlinear filter (SoAF))**.** *Suppose $f(x)$ and $h(x)$ are continuous functions, whose first and second order derivatives, $\partial_x[f]$, $\partial_{xx}[f]$, $\partial_x[h]$, $\partial_{xx}[h]$, respectively, exist. Then the second-order approximation equations for the exact filter equations,*

[3] α is a threshold to be chosen by the user. The precise value of the threshold is still an open question.

eqs. (3.4.3) *and* (3.4.6)*, neglecting the third- and higher–order derivatives and moments, are:*

$$\begin{aligned} d\hat{x}_t &= f(\hat{x}_t)dt + \frac{1}{2}p_t\partial_{xx}[f](\hat{x}_t)dt \\ &\quad + p_t\partial_x[h](\hat{x}_t)R^{-1}(t)(dy_t - (h(\hat{x}_t) + \frac{1}{2}p_t\partial_{xx}[h](\hat{x}_t))dt), \end{aligned} \tag{3.5.13a}$$

$$\begin{aligned} dp_t &= 2p_t\partial_x[f](\hat{x}_t)dt + g^2(\hat{x}_t)dt + p_t\partial_x[g]^2(\hat{x}_t)dt \\ &\quad + g(\hat{x}_t)\partial_{xx}[g](\hat{x}_t)p_t dt - (p_t\partial_x[h](\hat{x}_t))^2R^{-1}(t)dt \\ &\quad + \frac{1}{2}p_t^2\partial_{xx}[h](\hat{x}_t)R^{-1}(t)(dy_t - (h(\hat{x}_t) + \frac{1}{2}p_t\partial_{xx}[h](\hat{x}_t))dt). \end{aligned} \tag{3.5.13b}$$

See Appendix A for the proof. Second-order approximate filter for the vector case have been developed in Bass et al. (1966); Jazwinski (1970). They are obtained by extending the scalar equations, eqs. (3.5.13a) and (3.5.13b), into the vector form by considering the evolution of each element. We first introduce notations, following Bass et al. (1966).

We recall that the dimensions for the function vectors $f(\hat{x}_t)$ and $h(\hat{x}_t)$ are, respectively, $n\times 1$ and $r\times 1$. The term $g(\hat{x}_t)$ is of dimension $n\times m$. Besides, $\hat{x}_t$ is of dimension $n\times 1$. $R^{-1}(t)$ is an $r\times 1$ positive definite matrix. By $\nabla[h] = \dfrac{\partial h_i}{\partial x_j}$ is signified the Jacobian of the vector function $h(x)$, and is of dimension $r\times r$. Similarly, $\nabla\big[f\big] = \dfrac{\partial f_i}{\partial x_j}$ is an $n\times n$-dimensional Jacobian of the vector function $f(x)$. The Hessian of an ith function of the vector function $f(\hat{x}_t)$ is denoted by $\triangle[f_i] = \dfrac{\partial^2 f_i}{\partial x_i\partial x_j}$, an $n\times n$ matrix, so that $\triangle[f] = \dfrac{\partial^2 f}{\partial x_i\partial x_j}$ is an $n\times n\times n$ whilst $\triangle[h] = \dfrac{\partial^2 h}{\partial x_i\partial x_j}$ is an $n\times n\times r$. The covariance of the $n\times 1$ vector, $x_t - \hat{x}_t$, is denoted by an $n\times n$ matrix

$$P_t = (x_t - \hat{x}_t)(x_t - \hat{x}_t)^{\mathrm{T}}. \tag{3.5.14}$$

By $\triangle[f] : P$ we signify an $n\times 1$ vector with elements

$$(\triangle[f] : P)_i = \mathrm{tr}(\triangle[f_i]P), \tag{3.5.15}$$

and whose transpose is $P : \triangle[f]$. Similarly, $\triangle[h] : P$ is an $r\times 1$ vector whose transpose is $P : \triangle[h]$. Moreover, for any matrix $A(x)$, $\triangle[A(x)] : P$ signifies a matrix with elements

$$(\triangle[A(x)] : P)_{ij} = \mathrm{tr}(\triangle[A_{ij}] : P). \tag{3.5.16}$$

Considering each element in an $n-$dimensional vector, $\hat{x}_t$, and allowing it to evolve according to eqs. (3.5.13a) and (3.5.13b), aided by the just defined notations, we get

$$\begin{aligned} d\hat{x}_t &= f(\hat{x}_t)dt + \frac{1}{2}\triangle[f](\hat{x}_t) : P_t dt \\ &\quad + P_t\nabla[h]^{\mathrm{T}}(\hat{x}_t)R^{-1}(t)(dy_t - (h(\hat{x}_t) + \frac{1}{2}\triangle[h](\hat{x}_t) : P_t)dt), \end{aligned} \tag{3.5.17a}$$

$$\begin{aligned} dP_t &= P_t\nabla\big[f\big]^{\mathrm{T}}(\hat{x}_t)dt + \nabla\big[f\big](\hat{x}_t)P_t dt \\ &\quad + g(\hat{x}_t)g^{\mathrm{T}}(\hat{x}_t)dt + \nabla\big[g(\hat{x}_t)\big]\nabla\big[g(\hat{x}_t)\big]^{\mathrm{T}}P_t dt \\ &\quad + (\triangle[g](\hat{x}_t) : P_t)g^{\mathrm{T}}(\hat{x}_t)dt - P_t\nabla[h]^{\mathrm{T}}(\hat{x}_t)R^{-1}\nabla[h](\hat{x}_t)P_t dt \\ &\quad + \frac{1}{2}P_t : \triangle[h]^{\mathrm{T}}(\hat{x}_t)R^{-1}(t)\Big(dy_t - (h(\hat{x}_t) + \frac{1}{2}\triangle[h](\hat{x}_t) : P_t)dt\Big)P_t. \end{aligned} \tag{3.5.17b}$$

3.5.3 Second-order extended Kalman-Bucy filter (SoEKBF)

In a similar manner, as in the derivation of the SoAF, we can obtain the second order extended Kalman-Bucy filter (SoEKBF) by substituting $g(x,t)$ with $g(t)$ in the derivation of the second order approximate filter discussed in Section 3.5.2. This yields

$$\begin{aligned} d\hat{x}_t &= f(\hat{x}_t)dt + \frac{1}{2}\triangle[f](\hat{x}_t) : P_t dt \\ &\quad + P_t \nabla[h]^{\mathrm{T}}(\hat{x}_t) R^{-1}(t)(dy_t - (h(\hat{x}_t) + \frac{1}{2}\triangle[h](\hat{x}_t) : P_t)dt), \end{aligned} \tag{3.5.18a}$$

$$\begin{aligned} dP_t &= P_t \nabla\big[f\big]^{\mathrm{T}}(\hat{x}_t)dt + \nabla\big[f\big](\hat{x}_t)P_t dt \\ &\quad + g(t)g^{\mathrm{T}}(t)dt + -P_t \nabla[h]^{\mathrm{T}}(\hat{x}_t)R^{-1}\nabla[h](\hat{x}_t)P_t dt \\ &\quad + \frac{1}{2}P_t : \triangle[h]^{\mathrm{T}}(\hat{x}_t)R^{-1}(t)\Big(dy_t - (h(\hat{x}_t) + \frac{1}{2}\triangle[h](\hat{x}_t) : P_t)dt\Big)P_t. \end{aligned} \tag{3.5.18b}$$

Although the SoAF is more accurate compared to the first order approximate filter, it utilizing second order terms in the Taylor expansion, the additional complexity hampers its usefulness. For this reason, higher order approximate filters are seldom used and the first order approximate filter—more of which presently—has been extensively used.

3.5.4 First-order approximate nonlinear filter (FoAF)

The equations for the first-order approximation of the exact filter are summarised in the following Theorem.

Theorem 3.5.2 (First-order approximate filter (FoAF))**.** *Suppose $f(x)$, $h(x)$ and $g(x)$ are continuous functions whose first order derivatives, $\partial_x[f]$, $\partial_x[h]$ and $\partial_x[g]$, respectively, exist. Then the first-order approximation equations for the exact filter equations, eqs.* (3.4.3) *and* (3.4.6)*, omitting the second and higher central moments, are:*

$$d\hat{x}_t = f(\hat{x}_t)dt + p_t \partial_x[h](\hat{x}_t)R^{-1}(t)(dy_t - h(\hat{x}_t)dt), \tag{3.5.19a}$$

$$dp_t = 2p_t \partial_x[f](\hat{x}_t)dt + (g^2(\hat{x}_t) + p_t \partial_x[g]^2(\hat{x}_t))dt - (p_t \partial_x[h](\hat{x}_t))^2 R^{-1}(t)dt. \tag{3.5.19b}$$

Proof. Set to zero the second order derivatives of $f(x)$, $h(x)$ and $g(x)$ in the equations for second-order approximate filter. □

The equations

$$d\hat{x}_t = f(\hat{x}_t)dt + P_t \nabla[h](\hat{x}_t)R^{-1}(t)(dy_t - h(\hat{x}_t)dt), \tag{3.5.20a}$$

$$\begin{aligned} dP_t &= P_t \nabla\big[f\big]^{\mathrm{T}}(\hat{x}_t)dt + \nabla\big[f\big](\hat{x}_t)P_t dt + g(\hat{x}_t)g^{\mathrm{T}}(\hat{x}_t)dt \\ &\quad + \nabla\big[g(\hat{x}_t)\big]\nabla\big[g(\hat{x}_t)\big]^{\mathrm{T}}P_t dt - P_t \nabla[h]^{\mathrm{T}}(\hat{x}_t)R^{-1}\nabla[h](\hat{x}_t)P_t dt, \end{aligned} \tag{3.5.20b}$$

constitute the vector form of the first-order approximate nonlinear filter.

3.5.5 First-order extended Kalman-Bucy filter (FoEKBF)

The first-order extended Kalman-Bucy filter (FoEKBF) can also be arrived at by setting the diffusion term to be only a function of time in the derivation of first-order approximate filter. The FoEKBF equations,—obtained by substituting $g(x,t)$ and its derivative with respect to x with

$g(t)$ and its corresponding derivatives with respect to x being zero, respectively,—are [Jazwinski (1970)]:

$$d\hat{x}_t = f(\hat{x}_t)dt + P_t\nabla[h](\hat{x}_t)R^{-1}(t)(dy_t - h(\hat{x}_t)dt), \tag{3.5.21a}$$

$$\begin{aligned} dP_t = {} & P_t\nabla\big[f\big]^{\mathrm{T}}(\hat{x}_t)dt + \nabla\big[f\big](\hat{x}_t)P_t dt + g(t)g^{\mathrm{T}}(t)dt \\ & - P_t\nabla[h]^{\mathrm{T}}(\hat{x}_t)R^{-1}\nabla[h](\hat{x}_t)P_t dt. \end{aligned} \tag{3.5.21b}$$

The approximate filters discussed heretofore approximate the posterior $\pi_t(x \mid Y_t)$ to be Gaussian. If the true posterior is Gaussian, then the approximate filters return a fairly good approximation. If, however, the true posterior is skewed or bimodal or multi-modal, the approximate filters give an approximation that does not match well with the truth. For this reason, particle filters present a better alternative [Arulampalam et al. (2002)].

Let us now consider an application example with nonlinear signal and measurements.

Example 3.5.1: Lorenz 63 model

Named after Edward Lorenz, Lorenz 63 model [Lorentz (1963)] is a system of deterministic equations which are idealizations of hydrodynamical equations. Such a system of equations is peculiar in that it has nonperiodic solutions—solutions which do not repeat their past history and, if any, approximate repetitions are transient. The stochastic Lorenz 63 model, obtained by adding noise to the deterministic Lorenz 63 model, is given by

$$dx_t = f(x_t)dt + G^{1/2}d\beta_t; \qquad t \in [t_0, T], \tag{3.5.22}$$

where x is a 3-dimensional column vector $[x_1, x_2, x_3]^{\mathrm{T}}$, $G = \alpha \mathrm{I}_{3\times 3}$, α is a constant, $\mathrm{I}_{3\times 3}$ is a 3-dimensional identity matrix,

$$f(x) = \begin{pmatrix} a(x_2 - x_1) \\ bx_1 - x_1x_3 - x_2 \\ x_1x_2 - cx_3 \end{pmatrix},$$

with $a = 10$, $b = 8/3$ and $c = 28$. $\{\beta_t, t \geq t_0\}$ is a 3-dimensional standard Brownian motion process. We use the following initial values

$$x_{t_0} = [-5.91652, -5.52332, 24.5723]^{\mathrm{T}}. \tag{3.5.23}$$

For the purpose of this study, we introduce synthetic measurements

$$dy_t = h(x_t)dt + R^{1/2}d\eta_t, \tag{3.5.24}$$

where $\{\eta_t, t \geq t_0\}$ is a 3-dimensional standard Brownian motion process, $R = \sigma \mathrm{I}_{3\times 3}$, σ is a constant,

$$h(x) = \begin{pmatrix} a(x_2 - x_1) \\ bx_1 - x_1x_3 - x_2 \\ x_1x_2 - cx_3 \end{pmatrix}.$$

The implementation of FoAF in Example (3.5.1) requires the computation of $\nabla[h](\hat{x}_t)$ and $\nabla\big[f\big](\hat{x}_t)$ in eq. (3.5.20a):

$$\nabla[h](\hat{x}_t) = \nabla\big[f\big](\hat{x}_t) = \begin{bmatrix} -a & a & 0 \\ b - x_3 & -1 & -x_1 \\ x_2 & x_1 & -c \end{bmatrix}.$$

Similarly, to implement the SoAF requires the computation of the Hessians of the functions f and h, which are

$$\triangle[f_1] = \triangle[h_1] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

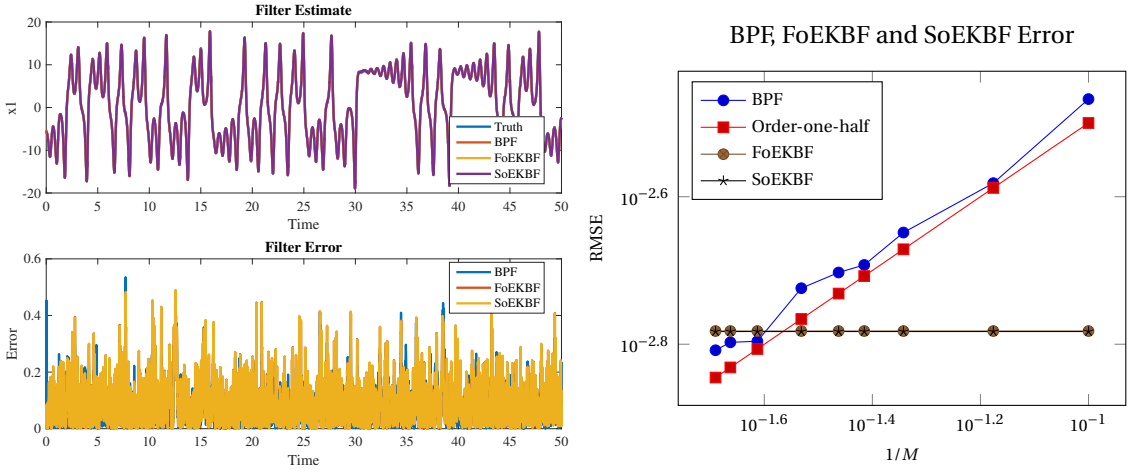
$$\triangle[f_2] = \triangle[h_2] = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix},$$

and

$$\triangle[f_3] = \triangle[h_3] = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix};$$

which can then be used in eqs. (3.5.17a) and (3.5.17b).

The following plots show filter results.



(a) (b)

Figure 3.1: (a) is solution plot for Bootstrap particle filter estimates with $M = 1000$ of the first variable in the stochastic Lorenz 63 model. The time-step used is $\delta t = 0.01$. Used also are the constants $\alpha = 0.20$ and $\sigma = 0.17$. Multinomial resampling is done when ESS$\leq 0.7M$ and the filter ran for 5000 iterations. Below the filter estimate is shown the absolute error of the estimate with respect to the true values obtained from Euler-Maruyama approximation of the stochastic Lorenz 63 model. Alongside are plotted the FoEKBF and SoEKBF estimates and their respective estimate errors. (b) is a plot of the root mean square errors of the Bootstrap particle filter estimate for the ensemble sizes $M = 10$, 15, 22, 26, 29, 34, 41, 46 and 49 against the reciprocal of ensemble sizes for 15000 iterations. The root mean square errors for FoEKBF and SoEKBF are, respectively, 0.0017 and 0.0016.

3.5.6 Critique of nonlinear filters

The EKBF and its discrete variant—the Extended Kalman Filter (EKF)—have been extensively used in application areas such as weather prediction, communication, financial engineering, navigation, tracking, multisensor data fusion and robotics. Their application, however, come with the cost of computing the Jacobians of the nonlinear parts in the model. The fact that the EKBF requires Gaussian assumption greatly hampers its application: the assumption is quite

restrictive. Integration of the Riccati equation to get covariances is costly especially with the increase in the dimension of the model. The EKBF, however, performs comparatively better than the particle filter.

The particle filters, on the other hand, have been used in simultaneous localisation and mapping (SLAM), tracking, and, among others, robotics. The application of particle filters is, however, hampered by a number of challenges which we now consider in turn.

3.5.6.1 Particle degeneracy

Degeneracy is a term used to describe the loss of weights of particles. It is common for some particles to occupy places in the state space where their contribution to the posterior is negligible. This leads to wastage. This well-known problem is called particle degeneracy. A litany of remedial measures have been proposed in literature [Daum and Huang (2011); Rebeschini and van Handel (2015); Reich and Cotter (2015); van Leeuwen (2015)]:

1. **Resampling.** This is an artificial way of culling away particles with less weights and supplying the loss by replicating particles with appreciable weights. This process renders the particle filter unparallelizable owing to the fact that all particles are resampled jointly. Repeated replication of particles with more weights during resampling occasions lack of diversity of the samples leading to the so-called *sample impoverishment*. What is more, resampling introduces extraneous covariances into the filter, thus undermining the accuracy of the filter.

2. **Localization.** Spatial correlations in high dimensional models, for example, those used in numerical weather prediction (of order 10^7) have been found to hinder application of particle filters. The error bound is well known to increase exponentially with the increase in the dimension of the model [Rebeschini and van Handel (2015)]; which phenomenon is called *curse of dimensionality*. Although finding out how to cure the curse of dimensionality is an open problem, initial studies suggest controlling the influence of neighbouring grid points on a reference grid points—a process called *localization* [Reich and Cotter (2015)]. The idea is to allow those grid points closer to the reference grid point in a state space have the most influence whilst the influence of those that are further away is rendered negligible. This way, the spatial correlations are minimized and, as a result, the performance of particle filters in high dimensional models is improved.

3. **Sampling new particles from the likelihood.** In most filters, the prior is assigned to represent the importance density. However, it may chance that the posterior is closely similar to the likelihood than the prior. The importance density can be chosen to be represented by the likelihood. This has been shown to improve filter performance [Arulampalam et al. (2002)].

4. **Sampling from a good proposal.** By a good proposal, we mean an optimal one; that is, a proposal which is a minimal variance estimate of the posterior density. This can be attained, in the linear Gaussian setting, by using the Kalman Bucy filter (KBF). This defeats the purpose of the particle filter because the KBF can be used instead. It is quite difficult to construct a good proposal [van Leeuwen (2015)].

5. **Moving the particles to a desired position in the state space.** This is the basis of exact nonlinear filters which include the Daum's filter [Daum and Huang (2011)], ensemble transform particle filter (ETPF) [Reich (2013, 2011)], the square root filter and the feedback particle filter [Yang et al. (2011*a*); Amirhossein et al. (2016)]. The latter will be studied extensively in Chapter 4.

CHAPTER 4

ENSEMBLE KALMAN-BUCY FILTER

4.1 Kalman-Bucy Filter

With the equations of evolution of the mean and covariance for the nonlinear case in hand, the corresponding equations for the linear case only require specializing. To do this, we begin with the linear model equation corresponding to the nonlinear equations, eqs. (3.1.1a) and (3.1.1b). To specialize nonlinear state space model to a linear one, we substitute as follows: the drift function $f(x_t) = F(t)x_t$, where $F(t)$ is an $n \times n$ time function valued matrix, and we let $g(x_t)$ be independent of the state variable, x_t. Then from eq. (3.1.1a),

$$\text{Signal:} \quad dx_t = F(t)x_t dt + G(t)d\beta_t; \quad t_0 \le t, \tag{4.1.1a}$$

where $G(t)$ is an $n \times m$ continuous time-function matrix. $\{\beta_t,\ t_0 \le t\}$ is an $m-$dimensional standard Brownian motion vector process. Furthermore, let $h(x_t, t) = H(t)x_t$, an $r \times n$ continuous time-function matrix. From (3.1.1b),

$$\text{Measurement:} \quad dy_t = H(t)x_t dt + R^{1/2}(t)d\eta_t; \quad t_0 \le t, \tag{4.1.1b}$$

where $\{\eta_t,\ t_0 \le t\}$ is an $r-$dimensional standard Brownian motion vector process and $R(t)$, as before, is an $r \times r$ time-continuous matrix valued function.

Note that the expected values in eqs. (3.4.7a) and (3.4.7b) now take a simplified form; that is,

$$\hat{f}_t = F(t)\hat{x}_t, \tag{4.1.2a}$$
$$\hat{h}_t = H(t)\hat{x}_t, \tag{4.1.2b}$$
$$\pi_t[x_t f^{\mathrm{T}}] = \pi_t[x_t x_t^{\mathrm{T}}]F^{\mathrm{T}}(t), \tag{4.1.2c}$$
$$\pi_t[x_t h^{\mathrm{T}}] = \pi_t[x_t x_t^{\mathrm{T}}]H^{\mathrm{T}}(t), \tag{4.1.2d}$$
$$\hat{x}_t\hat{f}_t^{\mathrm{T}} = (\hat{x}_t\hat{x}_t^{\mathrm{T}})F^{\mathrm{T}}(t), \tag{4.1.2e}$$
$$\hat{x}_t\hat{h}_t^{\mathrm{T}} = (\hat{x}_t\hat{x}_t^{\mathrm{T}})H^{\mathrm{T}}(t). \tag{4.1.2f}$$

The conditional density for the linear case is Gausssian. Therefore, all the odd central moments are zero. Substituting eqs. (4.1.2a) to (4.1.2f) in eqs. (3.4.7a) and (3.4.7b) and noting that the odd central moments are zero, yields

$$d\hat{x}_t = F(t)\hat{x}_t dt + P_t H^{\mathrm{T}}(t)R^{-1}(t)(dy_t - H(t)\hat{x}_t dt), \tag{4.1.3a}$$
$$dP_t = F(t)P_t dt + P_t F^{\mathrm{T}}(t)dt + G(t)G^{\mathrm{T}}(t)dt - P_t H^{\mathrm{T}}(t)R^{-1}(t)H(t)P_t dt. \tag{4.1.3b}$$

The system of eqs. (4.1.3a) and (4.1.3b) are the continuous time linear filtering equations, and constitute the Kalman-Bucy filter.[4] The theory of the Kalman-Bucy filter was first documented in an article by Kalman and Bucy (1961). A comprehensive coverage of various aspects of linear filtering are found in a book by Kailath et al. (2000). Equation (4.1.3a) is the evolution equation for the mean while eq. (4.1.3b) is for the covariance. The covariance equation is a matrix Riccati equation. The Kalman-Bucy filter gives the minimum variance estimate, $\hat{x}_t$, at a time, t, and the error covariance in the estimate. The Kalman-Bucy filter, hereafter KBF, gives the optimal estimate of the state given the measurements, $Y_t := \{y_\tau; t_0 \le \tau \le t\}$.

With meaningless measurements, that is, when $R^{-1}(t) \equiv 0$,

$$d\hat{x}_t = F(t)\hat{x}_t dt, \tag{4.1.4a}$$

$$dP_t = F(t)P_t dt + P_t F^{\mathrm{T}}(t)dt + G(t)G^{\mathrm{T}}(t)dt. \tag{4.1.4b}$$

This, in geoscience, is known as the *prediction step*. Integration of eq. (4.1.4a) gives a predicted estimate of the state—also called the *a priori* estimate or, simply, the prior—at time t, whilst eq. (4.1.4b) provides the error in that estimate. The additive term

$$P_t H^{\mathrm{T}}(t)R^{-1}(t)(dy_t - H(t)\hat{x}_t dt),$$

summarises the contribution of the new measurement in the predicted estimate, leading to an analysed estimate of the state—also known as the *a posteriori* estimate or the posterior—at time t, and hence the procedure that adds this term into the predicted outcome is known as the *analysis step*. The analysis step can be interpreted as a weighted linear addition of the new measurement to the predicted estimate. The weight,

$$P_t H^{\mathrm{T}}(t)R^{-1}(t),$$

is called the Kalman gain. The term

$$dy_t - H(t)\hat{x}_t dt,$$

is commonly referred to as the *innovation*. The innovation is the residual in the measurement equation, eq. (4.1.1b), and is equivalent to the measurement noise. Although KBF is easy to implement in a digital computer, and has elegant features such as stability and optimality, it however comes with a computational cost especially when the dimension of the state is high. The reason for this is mainly in the computation of the covariance matrix. Ensemble Kalman Filters attempt to circumvent this challenge by way of approximating the covariance using Monte Carlo methods. What is more, KBF is limited to linear models and hence has a limited practical utility, seeing that most practical models are nonlinear. There have been studies to extend KBF to nonlinear models in what is commonly referred to as the extended Kalman-Bucy filters introduced in Chapter 3.

4.2 Ensemble Kalman-Bucy Filter

Solving matrix Riccati equation, eq. (4.1.3b), is increasingly made costly with the increase in the dimension of the state. Ensemble Kalman-Bucy filter, and almost all the ensemble filters, are designed to mitigate this challenge. Like the particle filters introduced in Section 3.5.1 ensemble filters use Monte Carlo techniques. Ensemble filters, however, use uniform weights and the covariance is estimated using Monte Carlo methods. Ensemble Kalman Filter, henceforth EnKF, which is the first ensemble filter designed for discrete models, was introduced by

[4] The Kalman-Bucy filter is named after Rudolf Emil Kálmán (1930-2016) and Richard Snowden Bucy (1935-)

Evensen in 1994 [Evensen (1994, 2006); Burgers et al. (1998)]. Ensemble Kalman-Bucy filter, hereafter EnKBF, designed for time-continuous state space models, was introduced much later. The underlying ideas, however, are the same.

To begin with, EnKBF involves drawing, say, M independent and identically distributed ensemble members $\{x^i_{t_0}\}^M_{i=1}$ according to the initial probability density, π_{t_0}. This forms an initial ensemble, which is an M-columned matrix. Each ensemble member, x^i, is propagated according to the equation of evolution of the mean, eq. (4.1.3a), in the KBF:

$$dx^i_t = F(t)x^i_t dt + G(t)d\beta^i_t + P^M_t H^{\mathrm{T}}(t)R^{-1}(t)(dy_t + R^{1/2}(t)\eta^i_t - H(t)x^i_t dt), \tag{4.2.1}$$

where $\{\eta_t,\ t_0 \le t\}$ and $\{\beta_t,\ t_0 \le t\}$ are, respectively, $r-$dimensional and $m-$dimensional standard Brownian motion vector processes. This leads to an EnKBF with stochastically perturbed innovation [Law et al. (2015); Reich and Cotter (2015); Bergemann and Reich (2012)]. The ensemble average,

$$\bar{x}_t = \frac{1}{M}\sum_{i=1}^{M} x^i_t, \tag{4.2.2}$$

gives a Monte Carlo estimate of the conditional mean, $\hat{x}_t$, at an arbitrary time, t. The covariance, P^M_t, is estimated thus:

$$P^M_t = \frac{1}{M-1}\sum_{i=1}^{M}(x^i_t - \bar{x}_t)(x^i_t - \bar{x}_t)^{\mathrm{T}}. \tag{4.2.3}$$

The division by $M-1$ is to ensure that the Monte Carlo approximation of covariance is unbiased. Equation (4.2.1) has an associated mean-field continuous time equation, of the interacting process $\{\tilde{x}^i_t\}^M_{i=1}$, given by

$$d\bar{x}_t = F(t)\bar{x}_t dt + G(t)d\bar{\beta}_t + \mathscr{P}_t H^{\mathrm{T}}(t)R^{-1}(t)(dy_t + R^{1/2}(t)\bar{\eta}_t - H(t)\bar{x}_t dt), \tag{4.2.4}$$

where $\bar{x}_t$ is a mean-field process whose posterior distribution given Y_t is equal to the posterior distribution of x_t, $\bar{\beta}_t$ and $\bar{\eta}_t$ are, respectively, independent copies of β_t and η_t. Moreover,

$$\tilde{x}_t = \bar{\pi}_t[x], \tag{4.2.5a}$$

and

$$\mathscr{P}_t = \bar{\pi}_t[(x-\tilde{x})(x-\tilde{x})^{\mathrm{T}}], \tag{4.2.5b}$$

where $\bar{\pi}_t$ denotes the probability density of $\bar{x}_t$ given Y_t[del Moral (2013); de Wiljes et al. (2016); Taghvaei and Mehta (2018)].

Another variant of EnKBF—with deterministically pertubed innovation [Bergemann and Reich (2012)]—is

$$dx^i_t = F(t)x^i_t dt + G(t)d\beta^i_t + P^M_t H^{\mathrm{T}}(t)R^{-1}(t)(dy_t - \frac{1}{2}H(t)(x^i_t + \bar{x}_t)dt), \tag{4.2.6}$$

where $\bar{x}_t$ is the ensemble mean, obtained by eq. (4.2.2). The covariance matrix is approximated by means of eq. (4.2.3). In anticipation of the analysis of the consistency of EnKBF, we replace the noise term in eq. (4.2.6) with the term

$$\frac{1}{2}G(t)G^{\mathrm{T}}(t)(P^M_t)^{-1}(x^i_t - \bar{x}_t)dt; \tag{4.2.7}$$

and hence *eq*. (4.2.6) becomes

$$\begin{aligned} dx_t^i = F(t)x_t^i dt + \frac{1}{2}G(t)G^{\mathrm{T}}(t)(P_t^M)^{-1}(x_t^i - \bar{x}_t)dt \\ + P_t^M H^{\mathrm{T}}(t)R^{-1}(t)(dy_t - \frac{1}{2}H(t)(x_t^i + \bar{x}_t)dt). \end{aligned} \tag{4.2.8}$$

Notice that

$$\frac{1}{2M}\sum_i^M G(t)G^{\mathrm{T}}(t)(P_t^M)^{-1}(x_t^i - \bar{x}_t)dt = 0$$

and

$$\frac{1}{M-1}\sum_i^M G(t)G^{\mathrm{T}}(t)(P_t^M)^{-1}(x_t^i - \bar{x}_t)(x_t^i - \bar{x}_t)^{\mathrm{T}}dt = 2G(t)G^{\mathrm{T}}(t)dt, \tag{4.2.9}$$

where we have supposed that P_t^M is invertible. Let $\{\tilde{x}_t^i\}_{i=1}^M$ be M independent solutions to which the ensemble $\{x_t^i\}_{i=1}^M$ converges. Then the mean-field equation associated to eq. (4.2.8) is

$$\begin{aligned} d\bar{x}_t = F(t)\bar{x}_t^i dt + \frac{1}{2}G(t)G(t)^{\mathrm{T}}\mathscr{P}_t(\bar{x}_t^i - \bar{\pi}_t[x])dt \\ + \mathscr{P}_t H^{\mathrm{T}}(t)R^{-1}(t)(dy_t - \frac{1}{2}H(t)(\bar{x}_t^i + \bar{\pi}_t[x])dt), \end{aligned} \tag{4.2.10}$$

where $\bar{\pi}_t = \mathrm{Law}(\bar{x}_t)$ and $\mathscr{P}_t$ is given by eq. (4.2.5b) .

4.3 Extension of EnKBF to nonlinear filtering

Seldom are linear models to be met with in nature. To expand the utility of linear filters in solving practical nonlinear problems, therefore, modifications of the filters is inevitable. As has been indicated in the foregoing discussion, the KBF offers a closed form solution to the filtering problem. The EnKBF, on the other hand, is consistent with the KBF at the limit $M \to \infty$. Extension of the KBF to nonlinear filtering problem involves certain approximations that render the resultant filter, albeit useful, sub-optimal (see the discussion in Chapter 3). The extension of EnKBF to nonlinear filtering takes the following form: Each particle of the state, x_t^i, is propagated and updated, all at once, as follows.

$$dx_t^i = f(x_t^i, t)dt + G(t)d\beta_t^i + D_t^M R^{-1}(t)\left(dy_t - \frac{1}{2}(h(x_t^i) + \bar{h}_t)dt\right), \tag{4.3.1}$$

where

$$D_t^M = \frac{1}{M-1}\sum_{i=1}^M (x_t^i - \bar{x}_t)(h(x_t^i) - \bar{h}_t)^{\mathrm{T}},$$

$$\bar{h}_t = \frac{1}{M}\sum_{i=1}^M h(x_t^i), \quad \text{and} \quad \bar{x}_t = \frac{1}{M}\sum_{i=1}^M x_t^i.$$

The mean field equation for the EnKBF extended to nonlinear dynamics, eq. (4.3.1), is

$$d\tilde{x}_t^i = f(\tilde{x}_t^i, t)dt + G(t)d\beta_t^i + \mathscr{D}_t R_t^{-1}\left(dy_t - \frac{1}{2}(h(\tilde{x}_t^i) + \hat{h}_t)dt\right), \tag{4.3.2}$$

where $\{\tilde{x}_t^i\}_{i=1}^M$ are M independent solutions to which $\{x_t^i\}_{i=1}^M$ converge as $M \to \infty$,

$$\mathscr{D}_t = \tilde{\pi}_t[(x - \tilde{\pi}_t[x])(h(x) - \hat{h}_t)^{\mathrm{T}}]$$

and

$$\hat{h}_t = \tilde{\pi}_t[h].$$

Similarly, the extension of EnKBF with stochastically perturbed innovation to nonlinear filtering is expressed as follows:

$$dx_t^i = f(x_t^i, t)dt + G(t)d\beta_t^i + D_t^M R_t^{-1}\left(dy_t + R^{1/2}(t)\eta_t^i - h(x_t^i))dt\right), \tag{4.3.3}$$

4.3.1 Consistency of the EnKB filter

We now show the properties of ensemble Kalman-Bucy filter. The ensemble Kalman-Bucy filter can be extended to correspond to the nonlinear system, and this is given by

$$\begin{aligned} dx_t^i = f(x_t^i)dt + \frac{1}{2}G(t)G^T(t)(P_t^M)^{-1}(x_t^i - \bar{x}_t)dt \\ + D_t^M R^{-1}(t)(dy_t - \frac{1}{2}(h(x_t^i) + \bar{h}_t)dt); \quad t > t_0, \end{aligned} \tag{4.3.4}$$

Notice that eq. (4.3.4) is a stochastic differential equation, owing to the term dy_t.

The equation of evolution of ensemble mean is obtained empirically from eq. (4.3.4); that is,

$$d\bar{x}_t = \bar{f}_t dt + D_t^M R^{-1}(t)(dy_t - \bar{h}_t dt); \quad t > t_0, \tag{4.3.5}$$

where

$$\bar{f}_t = \frac{1}{M}\sum_{i=1}^{M} f(x_t^i). \tag{4.3.6}$$

Similarly, the equation of evolution of covariance is

$$\begin{aligned} dP_t^M = \frac{1}{M-1}\sum_{i=1}^{M}(f(x_t^i) - \bar{f}_t)(x_t^i - \bar{x}_t)^{\mathrm{T}} + \frac{1}{M-1}\sum_{i=1}^{M}(x_t^i - \bar{x}_t)(f(x_t^i) - \bar{f}_t)^{\mathrm{T}} \\ + \frac{1}{2}G(t)G^T(t)(P_t^M)^{-1}P_t^M dt + P_t^M(P_t^M)^{-1}\frac{1}{2}G(t)G^T(t)dt + D_t^M(D_t^M)^{\mathrm{T}}. \end{aligned} \tag{4.3.7}$$

We recover the linear ensemble Kalman-Bucy equation, eq. (4.2.6), by substituting $F(t)x_t^i$ and $H(t)x_t^i$ in the place of $f(x_t^i)$ and $h(x_t^i)$, respectively, in eq. (4.3.4). This is evident by inspection. The equations of evolution of the empirical mean and covariance in the linear case then are

$$d\bar{x}_t = F(t)\bar{x}_t dt + P_t^M H^{\mathrm{T}}(t)R^{-1}(t)(dy_t - H(t)\bar{x}_t dt), \tag{4.3.8a}$$

$$dP_t^M = F(t)P_t^M dt + P_t^M F^{\mathrm{T}}(t)dt + G(t)G^{\mathrm{T}}(t)dt - P_t^M H^{\mathrm{T}}(t)R^{-1}H(t)P_t^M dt; \tag{4.3.8b}$$

which equations are in the form of the Kalman-Bucy filter equations, eqs. (4.1.3a) and (4.1.3b).

It can be shown that the ensemble Kalman Bucy filter converges to a Kalman Bucy filter, in the linear setting, as the ensemble size attains a high value. This we now demonstrate (see de Wiljes et al. (2016) for the full proof from which we obtain a summary).

Let $\hat{x}_t$ and P_t be the solutions of the Kalman-Bucy eqs. (4.1.3a) and (4.1.3b), respectively. Let, moreover, $\{x_{t_0}^i\}_{i=1}^M \sim \pi_{t_0}$ be an initial ensemble in the ensemble Kalman-Bucy filter. We now show that

$$\underset{M\to\infty}{\text{l.i.p.}}\ \bar{x}_t = \hat{x}_t, \tag{4.3.9}$$

where $\bar{x}_t$ is the solution to the ensemble Kalman-Bucy filter equation, eq. (4.3.8a). By l.i.p.[5] is signified limit in probability. The following relation follows from eqs. (4.1.3a) and (4.3.8a).

$$\begin{aligned} d(\bar{x}_t - x_t) = F(t)(\bar{x}_t - x_t)dt - P_t^M H^{\mathrm{T}}(t)R^{-1}(t)H(t)\bar{x}_t dt \\ + P_t H^{\mathrm{T}}(t)R^{-1}(t)H(t)x_t dt + (P_t^M - P_t)H^{\mathrm{T}}(t)R^{-1}(t)dy_t. \end{aligned} \tag{4.3.12}$$

Taking integrals yields,

$$\begin{aligned} \bar{x}_t - x_t = \bar{x}_{t_0} - x_{t_0} + \int_{t_0}^{t} F(\tau)(\bar{x}_\tau - x_\tau)d\tau - \int_{t_0}^{t} P_\tau^M H^{\mathrm{T}}(\tau)R^{-1}(s)H(\tau)\bar{x}_s d\tau \\ + \int_{t_0}^{t} P_\tau H^{\mathrm{T}}(\tau)R^{-1}(\tau)H(\tau)x_\tau d\tau + \int_{t_o}^{t} (P_\tau^M - P_\tau)H^{\mathrm{T}}(\tau)R^{-1}(\tau)dy_\tau. \end{aligned} \tag{4.3.13}$$

Taking the norm of the vectors and matrices results in the following

$$\begin{aligned} \|\bar{x}_t - x_t\| &= \|\bar{x}_{t_0} - x_{t_0}\| + \int_{t_0}^{t} \|F(\tau)\|\|(\bar{x}_\tau - x_\tau)\|d\tau \\ &\quad - \int_{t_0}^{t} \|P_\tau^M\|\|H(\tau)\|^2\|R^{-1}(\tau)\|\|\bar{x}_s\|d\tau \\ &\quad + \int_{t_0}^{t} \|P_\tau\|\|H(\tau)\|^2\|R^{-1}(\tau)\|\|x_\tau\|d\tau \\ &\quad + \|\int_{t_o}^{t} (P_\tau^M - P_\tau)H^{\mathrm{T}}(\tau)R^{-1}(\tau)dy_\tau\| \\ &\leq \|\bar{x}_{t_0} - x_{t_0}\| + \int_{t_0}^{t} (\|F(\tau)\| + \|H(\tau)\|^2\|R^{-1}(\tau)P_\tau)\|(\bar{x}_\tau - x_\tau)\|d\tau \\ &\quad - \int_{t_0}^{t} \|P_\tau^M - P_\tau\|\|H(\tau)\|^2\|R^{-1}(s)\|\|\bar{x}_s\|d\tau \\ &\quad + \|\int_{t_o}^{t} (P_\tau^M - P_\tau)H^{\mathrm{T}}(\tau)R^{-1}(\tau)dy_\tau\|. \end{aligned} \tag{4.3.14a}$$

We now take expectation with respect to the distribution π_t and get

$$\begin{aligned} \pi_t[\|\bar{x}_t - x_t\|] \leq \pi_t[\|\bar{x}_{t_0} - x_{t_0}\|] + \int_{t_0}^{t} (\|F(\tau)\| + \|H(\tau)\|^2\|R^{-1}(\tau)P_\tau)\pi_t[\|(\bar{x}_\tau - x_\tau)\|]d\tau \\ - \int_{t_0}^{t} \pi_t[\|P_\tau^M - P_\tau\|]\|H(\tau)\|^2\|R^{-1}(s)\|\|\bar{x}_s\|d\tau \\ + \pi_t[\|\int_{t_o}^{t} (P_\tau^M - P_\tau)H^{\mathrm{T}}(\tau)R^{-1}(\tau)dy_\tau\|]. \end{aligned} \tag{4.3.15}$$

By almost sure convergence of P_t^M to P_t as $M \to \infty$

$$\lim_{M\to\infty} \pi_t[\|P_t^M - P_t\|] = 0, \tag{4.3.16}$$

[5] Let $\{x_{t_n}, n = 1, 2, 3, ...\}$ be a random sequence and $\mathbb{P}(x > 0) = a$ denote a probability that $x > 0$, whose realisation is a. If

$$\lim_{n\to\infty} \mathbb{P}(|x_{t_n} - x|^2 \geq \epsilon) = 0 \tag{4.3.10}$$

then the random sequence $\{x_{t_n}, n = 1, 2, 3, ...\}$ is said to converge to x in probability. This is precisely written as

$$\pi \lim x_{t_n} = x. \tag{4.3.11}$$

from which we have that the limit of the last term in eq. (4.3.15) as $M \to \infty$ is zero. With these considerations, eq. (4.3.15) becomes

$$\pi_t[\|\bar{x}_t - x_t\|] \leq \pi_t[\|\bar{x}_{t_0} - x_{t_0}\|] + \int_{t_0}^{t} (\|F(\tau)\| + \|H(\tau)\|^2 \|R^{-1}(\tau)P_\tau)\pi_t[\|(\bar{x}_\tau - x_\tau)\|]d\tau. \tag{4.3.17}$$

Finally, we invoke Gronwall's Lemma to attain

$$\lim_{M\to\infty} \pi_t[\|\bar{x}_t - x_t\|] = 0. \tag{4.3.18}$$

Almost sure convergence of ensemble covariance to the optimal covariance, eq. (4.3.16), remains to be shown. From eqs. (4.1.3b) and (4.3.8b),

$$\begin{aligned}
d\|P_t^M - P_t\|^2 \leq\ & 2\langle F(t)(P_t^M - P_t), P_t^M - P_t\rangle dt + 2\langle P_t^M - P_t, (P_t^M - P_t)F_t^{\mathrm{T}}\rangle dt \\
& + 2\langle G(t)G^{\mathrm{T}}(t)(P_t^M - P_t), P_t^M - P_t\rangle dt \\
& - 2\langle P_t^M H^{\mathrm{T}}(t)R^{-1}H(t)P_t^M - P_t H^{\mathrm{T}}(t)R^{-1}H(t)P_t, P_t^M - P_t\rangle dt \\
=\ & 2\langle F(t)(P_t^M - P_t), P_t^M - P_t\rangle dt + 2\langle P_t^M - P_t, (P_t^M - P_t)F_t^{\mathrm{T}}\rangle dt \\
& + 2\langle G(t)G^{\mathrm{T}}(t)(P_t^M - P_t), P_t^M - P_t\rangle dt \\
& - 2\langle P_t^M H^{\mathrm{T}}(t)R^{-1}H(t)(P_t^M - P_t), P_t^M - P_t\rangle dt \\
& - 2\langle (P_t^M - P_t)H^{\mathrm{T}}(t)R^{-1}H(t)P_t, P_t^M - P_t\rangle dt \\
\leq\ & \left(4\|F(t)\| + 2\|G(t)G^{\mathrm{T}}(t)\| + 2\|R^{-1}\|\|H(t)\|^2(\|P_t^M\| + \|P_t\|)\right)\|P_t^M - P_t\|^2 dt.
\end{aligned} \tag{4.3.19}$$

By separation of variables,

$$\frac{d\|P_t^M - P_t\|^2}{\|P_t^M - P_t\|^2} \leq \left(4\|F(t)\| + 2\|G(t)G^{\mathrm{T}}(t)\| + 2\|R^{-1}\|\|H(t)\|^2(\|P_t^M\| + \|P_t\|)\right) dt. \tag{4.3.20}$$

Integrating both sides of eq. (4.3.20) yields

$$\begin{aligned}
\|P_t^M - P_t\|^2 \leq\ & \exp\left((2\|F(t)\| + \|G(t)G^{\mathrm{T}}(t)\|)2t\right)\|P_{t_0}^M - P_{t_0}\|^2 \\
& \times \exp\left(2t\|R^{-1}\|\|H(t)\|^2 \int_{t_0}^{t} (\|P_\tau^M\| + \|P_\tau\|)d\tau\right).
\end{aligned} \tag{4.3.21}$$

It remains to obtain a uniform in M upper bound on $\|P_t^M\|$ [de Wiljes et al. (2016)]. This requires a bound on the deviation of the ensemble, x_t^i, from the mean estimate, $\bar{x}_t$, which deviation we define as follows:

$$e_t := \frac{1}{M-1}\sum_{i=1}^{M} \|x_t^i - \bar{x}_t\|^2. \tag{4.3.22}$$

To obtain the equation of evolution of e_t, we make the following simplifications: assume, in eq. (4.3.4), that $h(x) = x$, $R = \sigma I_{r\times r}$, where σ is a constant and that $G(t)G^T(t) = 2\Pi(t)$ is of full rank. Then the equation of evolution of the ensembles becomes

$$\begin{aligned}
dx_t^i = & F(t)x_t^i dt + \Pi(t)(P_t^M)^{-1}(x_t^i - \bar{x}_t)dt \\
& + \frac{D_t^M}{\sigma}(dy_t - \frac{1}{2}(x_t^i + \bar{x}_t)dt); \quad t > t_0.
\end{aligned} \tag{4.3.23}$$

From eq. (4.3.22) and eqs. (4.3.4) and (4.3.5), we obtain that

$$\begin{aligned} de_t = &\frac{2}{M-1}\sum_{i=1}^{M}\langle x_t^i - \bar{x}_t, F(t)(x_t^i - \bar{x}_t)\rangle dt \\ &+ \frac{1}{M-1}\sum_{i=1}^{M} 2\langle x_t^i - \bar{x}_t, \Pi(P_t^M)^{-1}(x_t^i - \bar{x}_t)\rangle dt \\ &- \frac{1}{\sigma(M-1)}\sum_{i=1}^{M}\langle x_t^i - \bar{x}_t, P_t^M(x_t^i - \bar{x}_t)\rangle dt \\ &= 2\mathrm{tr}(F(t)P_t^M)dt + 2\mathrm{tr}(\Pi)dt - \frac{\mathrm{tr}([P_t^M]^2))}{\sigma}dt. \end{aligned} \tag{4.3.24}$$

Now the second term on the right hand side of eq. (4.3.24) has the bounds

$$2d_{\min}dt \le 2\mathrm{tr}(\Pi)dt \le 2d_{\mathrm{sum}}dt, \tag{4.3.25}$$

where $d_{\min}$ denotes the minimum element in the diagonal of Π whilst d_{sum} is the sum of all diagonal elements of Π.

The term,

$$\begin{aligned} \frac{\mathrm{tr}([P_t^M]^2)}{\sigma} &= \frac{\|P_t^M\|^2}{\sigma} \\ &= \frac{1}{\sigma(M-1)^2}\sum_{i,j}^{M}\langle x_t^i - \bar{x}_t, x_t^j - \bar{x}_t\rangle^2, \end{aligned} \tag{4.3.26}$$

has bounds

$$\frac{e_t^2}{\sigma M} \le \frac{\mathrm{tr}([P_t^M]^2)}{\sigma} \le \frac{e_t^2}{\sigma}. \tag{4.3.27}$$

To see this, notice that

$$\begin{aligned} \frac{1}{\sigma(M-1)^2}\sum_{i,j}^{M}\langle x_t^i - \bar{x}_t, x_t^j - \bar{x}_t\rangle^2 &\le \frac{1}{\sigma(M-1)^2}\sum_{i,j}^{M}\|x_t^i - \bar{x}_t\|^2\|x_t^j - \bar{x}_t\|^2 \\ &= \frac{1}{\sigma}\left[\frac{1}{M-1}\sum_{i=1}^{M}\|x_t^i - \bar{x}_t\|\right]^2 \\ &= \frac{e_t^2}{\sigma}; \end{aligned} \tag{4.3.28a}$$

and that

$$\begin{aligned} \frac{1}{\sigma(M-1)^2}\sum_{i,j}^{M}\langle x_t^i - \bar{x}_t, x_t^j - \bar{x}_t\rangle^2 &\ge \frac{1}{\sigma(M-1)^2}\sum_{i=1}^{M}\|x_t^i - \bar{x}_t\|^4 \\ &\ge \frac{1}{\sigma M}\left[\frac{1}{M-1}\sum_{i=1}^{M}\|x_t^i - \bar{x}_t\|^2\right]^2 \\ &= \frac{e_t^2}{\sigma M}. \end{aligned} \tag{4.3.28b}$$

From eq. (4.3.27), and taking $f_{\min}$ and f_{sum} to be, respectively, the minimum element in the diagonal of F and the sum of all the diagonal elements of F, it follows that

$$2f_{\min}\frac{e_t}{\sqrt{M}}dt \le 2\mathrm{tr}(F(t)P_t^M)dt \le 2f_{\mathrm{sum}}e_t dt, \tag{4.3.29}$$

which gives the bounds for the first term on the right-hand-side of eq. (4.3.24).

Equations (4.3.25), (4.3.27) and (4.3.29) substituted in eq. (4.3.24) yield

$$de_t \geq \left(2f_{\min}\frac{e_t}{\sqrt{M}} + 2d_{\min} - \frac{e_t^2}{\sigma}\right)dt \tag{4.3.30a}$$

for the lower bound and

$$de_t \leq \left(2f_{\text{sum}}e_t + 2d_{\text{sum}} - \frac{e_t^2}{\sigma M}\right)dt \tag{4.3.30b}$$

for the upper bound. As $M \to \infty$ the last term on the right hand side of eq. (4.3.30b) becomes negligible. Then the upper bound can be approximated by

$$de_t \leq \left(2f_{\text{sum}}e_t + d_{\text{sum}}\right)dt \tag{4.3.31}$$

which yields

$$e_t \leq (e_{t_0} + \frac{d_{\text{sum}}}{2f_{\text{sum}}})\exp\left(2f_{\text{sum}}t\right). \tag{4.3.32}$$

From eqs. (4.3.27) and (4.3.32), we obtain

$$\|P_t^M\| \leq e_t \leq (e_{t_0} + \frac{d_{\text{sum}}}{2\|F\|})\exp(2\|F\|t), \tag{4.3.33}$$

where we have used $f_{\text{sum}} \leq \|F\|$. Finally, we note that the solution, P_t, of eq. (4.1.3b) is continuous and hence locally bounded. Equation (4.3.21) can then be approximated by

$$\begin{aligned}\|P_t^M - P_t\|^2 \leq \exp\big((2\|F(t)\| + \|G(t)G^{\mathrm{T}}(t)\|)2t\big) \\ \times \exp\left(2t\|R^{-1}\|\|H(t)\|^2\left((e_{t_0} + \frac{d_{\text{sum}}}{2\|F\|})\exp(2t\|F\|) + \max_{t_0<\tau<t}\|P_\tau\|\right)\right),\end{aligned} \tag{4.3.34}$$

which completes the assertion that

$$\lim_{M\to\infty} P_t^M = P_t \text{ a. s. for } t > 0. \tag{4.3.35}$$

As mentioned in Section 3.5.6, one of the challenges affecting particle filters, of which belongs EnKBF, is the presence of spurious correlations evident especially when treating spatio-temporal models. The remedy to this undesirable effect is ensemble inflation and localization, which we now outline.

4.3.2 Ensemble inflation and covariance localisation

Particle filters, among which is EnKBF, are known to be afflicted with spurious correlations, a phenomenon common with spatial-temporal models, arising from undersampling. This has been addressed extensively (e.g. in Reich and Cotter (2015) p. 242-257). To circumvent this challenge, covariance localisation has been proposed, the leading idea of which is as follows: the effect of observations at a grid point is limited to contributions from few neighbouring grid points. On the other hand, there is a possibility, in the course of running the filter, for ensemble members to drastically reduce the essential correlation, which process dramatically affects the performance of the filter. The counter-measure is to increase the sample spread, a phenomenon called ensemble inflation.

We now describe, in the briefest manner, these two phenomena after which we shall implement them. In the first place, inflation: we require that each ensemble x_t^i $i = 1, 2, 3, ..., M$ be inflated as follows

$$x_t^i := \hat{x}_t^M + \rho(x_t^i - \hat{x}_t^M) \tag{4.3.36}$$

where the inflation factor $\rho \geq 1$.

In the second place, we describe localisation. We require that each grid point be influenced by information from other grid points, say, within a radius r. We further require that those grid points closer to the reference point provide more influence than those geographically further away. This necessitates the following criteria. We define a matrix L whose entries are given by

$$L_{ij} = f(\frac{|\, x_i - x_j \,|}{r}), \tag{4.3.37}$$

where i, j represent the ith and jth gridpoints, $f(s)$ is a funtion describing the degree of influence of the neighbouring grid points. How to derive these functions is described in Cohn (1999). For example, use can be made of the following tapered function.

$$f(s) = \begin{cases} 1 - \frac{80}{33}s^2 + \frac{20}{11}s^3 + \frac{8}{11}s^4 - \frac{28}{33}s^5, & 0 \leq s \leq \frac{1}{2}, \\ -\frac{7}{44s} + \frac{51}{32} - \frac{45}{11}s + \frac{100}{33}s^2 - \frac{16}{11}s^4 + \frac{20}{33}s^5, & \frac{1}{2} \leq s \leq 1, \\ \frac{115}{132s} - \frac{61}{22} + 52 - \frac{100}{33}s^2 - \frac{10}{11}s^3 + \frac{16}{11}s^4 - \frac{4}{11}s^5, & 1 \leq s \leq \frac{3}{2}, \\ -\frac{32}{33s} + \frac{64}{11} - \frac{80}{11}s + \frac{80}{33}s^2 + \frac{10}{11}s^3 - \frac{8}{11}s^4 + \frac{4}{33}s^5, & \frac{3}{2} \leq s \leq 2, \\ 0, & \text{Otherwise.} \end{cases} \tag{4.3.38}$$

We then have a modified Kalman gain; that is,

$$K_t^{loc} = (L \circ P_t^M) H^T(t) R^{-1}(t), \tag{4.3.39}$$

where $\circ$ denotes the Schur product.

4.4 Time-continuous linear smoother

For the purpose of the following example, and to illustrate prediction, smoothing and filtering, we briefly introduce time-continuous linear smoothing. The derivation can be found in Rauch et al. (1965); Einicke (2012).

For a linear system governed by eqs. (4.1.1a) and (4.1.1b), the smoothed estimate, $\hat{x}_t^s$, and the smoothed covariance, P_t^s, satisfy the following equations.

$$d\hat{x}_t^s = F(t)\hat{x}_t^s dt + G(t)G^{\mathrm{T}}(t)P_t^{-1}(\hat{x}_t^s dt - \hat{x}_t dt), \tag{4.4.1a}$$

$$dP_t^s/dt = (F(t) + G(t)G^{\mathrm{T}}(t)P_t^{-1})P_t^s + P_t^s(F^{\mathrm{T}}(t) + P_t^{-1}GG^{\mathrm{T}}) - G(t)G^{\mathrm{T}}(t). \tag{4.4.1b}$$

The system of eqs. (4.4.1a) and (4.4.1b)—known as the Rauch-Tung-Striebel smoother—is the equation of evolution of the first and second moments under the smoothing distribution $\pi_t(x \mid Y_\tau)$, $\tau > t$. The Rauch-Tung-Striebel smoother is a backward recursion beginning with an initial smoothed estimate set equal to the filter estimate; that is,

$$\hat{x}_T^s = \hat{x}_T. \tag{4.4.2}$$

We now turn to a simple scalar example.

Example 4.4.1: Scalar SDE

Consider the following linear Gaussian Itô state space model.

$$dx_t = (0.2 - 0.2x_t)dt + Q^{1/2}dv_t; \qquad t_0 \le t, \tag{4.4.3a}$$

$$dy_t = 1.01x_t dt + R^{1/2}dw_t; \qquad t_0 \le t, \tag{4.4.3b}$$

where $\{v_t\}$ and $\{w_t\}$ are Brownian motion processes with, respectively, $\mathbb{E}\{dv_t dv_t^T\} = dt$ and $\mathbb{E}\{dw_t dw_t^T\} = dt$. Let the x_t at time t_0 be $x_{t_0} \sim \mathcal{N}(0, 0.001)$. Let, moreover, x_{t_0}, $\{v_t,\ t \ge t_0\}$ and $\{v_t,\ t \ge t_0\}$ be uncorrelated. Set $Q = 0.001$ and $R = 0.0001$. We seek an estimate of x_t.

The following panels show the results.

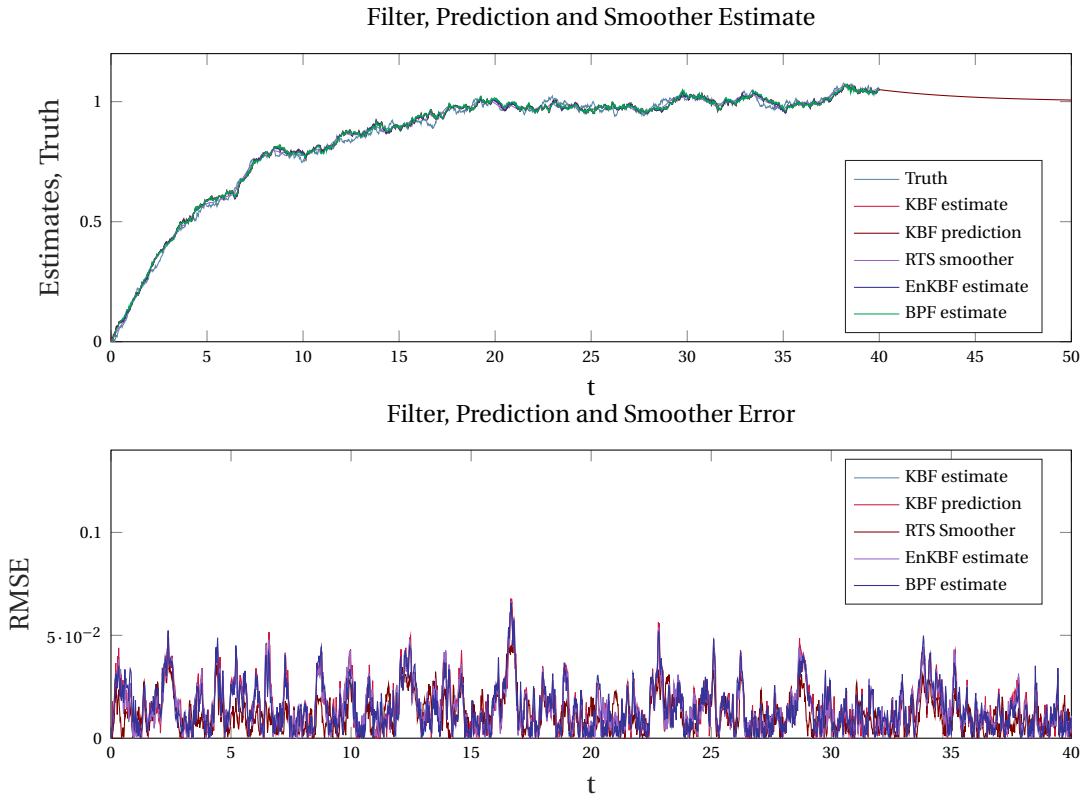


Figure 4.1: Filtering, prediction and smoothing estimates for Example (4.4.1). The ensemble size used in both the EnKBF and the BPF is $M = 1000$. Prediction is shown beyond $T = 40$. A uniform time step-size of $\delta t = 0.02$ is used. The prior used is $\pi_{t_0} = \mathcal{N}(0, 0.001)$.

	KBF pred.	**KBF est.**	**RTS est.**	**EnKBF est.**	**BPF est.**
Time-averaged RMSE	0.0129	0.0127	0.0104	0.0127	0.0131

Table 4.1: A table showing time-averaged RMSE for the prediction and estimate of KBF, the RTS smoother estimate, EnKBF and BPF estimates shown in Figure 4.1 after a burn-in of 1000 iterations.

The RMSE for the smoother is obtained by

$$\text{RMSE}_t = \sqrt{(\hat{x}_t^s - x_t)^2}. \tag{4.4.4}$$

From Figure 4.1, the prediction, filtering and smoothing estimates have no wide excursions from the truth. In fact, the errors of the smoother, prediction and filter estimates do not grow with change in t. And so are the errors in the EnKBF estimate, the ensemble size being large ($M = 1000$). The EnKBF registers a stable outcome in this setting; that is, the estimate forgets the error due to the initial condition before long. As is evident from Table 4.1, EnKBF filter estimate has the same accuracy as the KBF filter estimate, both of which estimates are of a lesser accuracy than the RTS smoother estimate—as anticipated. That the error of EnKBF matches that of the optimal filter (KBF) indicates that EnKBF is consistent with KBF at high ensemble sizes, which corroborates the theoretical results discussed and obtained in Section 4.3.1. What is more, that the error in the EnKBF remains bounded and does not grow indicates that EnKBF gives a robust filter estimate; that is, the small excursions from the optimal filter values are cancelled off and do not build up over time to compromise filter performance.

CHAPTER 5

FEEDBACK PARTICLE FILTER

5.1 Motivation

The feedback particle filter (FPF) [Yang et al. (2011*a*,*b*, 2012, 2013); Yang (2014); Amirhossein et al. (2016); Taghvaei and Mehta (2018)] arose from the need to overcome the challenges that come with application of existing filters to nonlinear state-space models. Extended Kalman filters, although meant to exploit the optimal performance of the Kalman filters in nonlinear systems, are greatly hampered by the need to compute the Jacobians. This challenge is more prevalent in approximate second-order filters, what with the need to compute the Hessians besides. Moreover, the computational cost in the aforementioned filters is compounded by the need to integrate the equation for evolution of covariance. Whereas particle filters hold promise for overcoming the challenge faced in using approximate filters in nonlinear settings, they, too, have their own challenges. As was indicated in Chapter 3, particle filters are sequential sampling Monte Carlo methods with particle degeneracy mitigating resampling steps. Although they are easy to implement, they suffer from the so-called sample impoverishment owing to frequent resampling—which is why resampling is done when the effective sample size has deteriorated to a given threshold. Despite the improvisations tending to the improvement of particle filters, the particle filters retain a comparatively high error covariance.

The idea behind feedback particle filters is designing the posterior density of each particle so that it matches, optimally, the true posterior. As will be seen in the ensuing argument, if one begins by setting the prior density of each particle to be the same as the true prior density, FPF yields the same posterior density as the true posterior at all subsequent times. This matching of the posterior density of the particles with the true posterior is arrived at by minimizing a cost function. To the present time, the cost function used is the Kullback-Leibler divergence between the two posteriors. Minimization of the cost function, in the first variation, yields a Poisson equation, the solution of which results in an optimal control input. The FPF is characterized by a controlled innovation process and the gain. It turns out, and this will be apparent shortly, that the FPF gain is obtained by differentiating the optimal control input with respect to the particles.

Much like in the deterministic variant of the EnKBF, the FPF—for the vector nonlinear system, eqs. (3.1.1a) and (3.1.1b),

$$\text{Signal:} \quad dx_t = f(x_t,t)dt + g(x_t,t)d\beta_t; \quad t_0 \le t, \tag{5.1.1a}$$

$$\text{Measurement:} \quad dy_t = h(x_t,t)dt + R^{1/2}(t)d\eta_t, \quad t_0 \le t, \tag{5.1.1b}$$

—consists of the evolution of a mean-field process, $\bar{x}_t$, whose posterior distribution (given the measurements' history) is the same as the true posterior, which mean-field process evolves as

follows.

$$d\bar{x}_t = f(\bar{x}_t, t)dt + g(\bar{x}_t, t)d\bar{\beta}_t + K(\bar{x}_t, t) \circ (dy_t - 0.5[h(\bar{x}_t, t) + \bar{\pi}_t[h]]dt). \tag{5.1.2}$$

where $\bar{\pi}_t = \text{Law}(\bar{x}_t)$ and $\bar{\beta}_t$ is an independent copy of β_t. Then a batch of hypotheses of the state, $X_t := \{x_t^i\}_{i=1}^M$, each propagated using the *controlled Stratonovich SDE*

$$dx_t^i = f(x_t^i, t)dt + g(x_t^i, t)d\beta_t^i + K(x_t^i, t) \circ (dy_t - 0.5[h(x_t^i, t) + \hat{h}_t]dt), \tag{5.1.3}$$

—where $\{\beta_t^i\}_{i=1}^M$ are the independent copies of the process noise β_t and $\hat{h}_t = \frac{1}{M}\sum_{i=1}^M h(x_t^i)$— is used to empirically estimate the distribution of $\bar{x}_t$. Equations (5.1.2) and (5.1.3) are called McKean-Vlasov stochastic differential equations and their analysis is known as propagation of chaos [Taghvaei and Mehta (2018)]. For ease in subsequent analyses we assume that the measurements equation, eq. (5.1.1b), is scalar. Equation (5.1.3), dropping the notation of the particles for convenience, has an equivalent vector Itô form:

$$dx_t = f(x_t, t)dt + g(x_t, t)d\beta_t + q(x_t, t)dt + K(x_t, t)(dy_t - 0.5[h(x_t, t) + \hat{h}_t]dt), \tag{5.1.4}$$

where

$$q_j(x_t, t) = \frac{R}{2}\sum_{k=1}^n K_k(x, t)\frac{\partial K_j}{\partial x_k}, \tag{5.1.5}$$

and $K = \nabla\phi$, the gain, where $\phi(x)$ satisfies the following Poisson equation

$$\nabla \cdot \big(\pi_t(x \mid Y_t)\nabla\phi(x)\big) = -(h(x_t, t) - \hat{h})\pi_t(x \mid Y_t)R^{-1}(t), \tag{5.1.6a}$$

$$\int \phi(x_t)\pi_t(x \mid Y_t)dx = 0, \tag{5.1.6b}$$

where $\pi_t(x \mid Y_t)$ is the conditional density of the particle, x_t^i, given measurements $Y_\tau = \{y_\tau : t_0 \le \tau \le t\}$.

We recall that the equation of evolution of the conditional density, $\pi_t^*(x \mid Y_t)$, for the state-space model governed by eqs. (3.1.1a) and (3.1.1b), is given by the K-S equation,

$$d\pi_t^* = \mathscr{L}(\pi_t^*)dt + \pi_t^*(h - \hat{h}_t)^{\mathrm{T}}R^{-1}(t)(dy_t - \hat{h}_t dt), \tag{5.1.7}$$

where $\mathscr{L}$ is as expressed in eq. (2.4.8). The FPF establishes an optimal control on the equation of evolution of the particles, eq. (5.1.3), via a cost function defined by the Kullback-Leibler divergence between $\pi_t^*(x \mid Y_t)$ and $\pi_t(x \mid Y_t)$, thus yielding an exact filter when the prior filter density is chosen equal to the prior true density. This renders the FPF robust in that, choosing $\pi_{t_0} = \pi_{t_0}^*$ at initial time, $t = t_0$, the estimated posterior density remains the same as the true posterior at all times; that is,

$$\pi_t(x \mid Y_t) = \pi_t^*(x \mid Y_t), \qquad t \ge t_0. \tag{5.1.8}$$

Notice that eq. (5.1.4) is an SDE comprising of the signal equation and the control terms where

$$q(x_t, t)dt + K(x_t, t)dy_t - 0.5K(x_t, t)[h(x_t, t) + \hat{h}_t]dt$$

forms the drift and $R^{1/2}K(x_t, t)$ forms the diffusion term—as a result of the noise in the measurements. We then apply the Fokker-Planck equation, eq. (2.4.7), to obtain the equation of evolution of the filtering density. The conditional density of each particle, given the measurements $Y_\tau = \{y_\tau : t_0 \le \tau \le t\}$, evolves as follows [Yang et al. (2012)]:

$$\begin{aligned} d\pi_t = \mathscr{L}(\pi_t)dt - \nabla \cdot (\pi_t K)dy_t - \nabla \cdot \big(\pi_t \psi\big)dt \\ + \frac{R}{2}\sum_{k,l=1}^n \frac{\partial^2}{\partial x_k \partial x_l}(\pi_t(KK^{\mathrm{T}})_{kl})dt, \end{aligned} \tag{5.1.9}$$

where K is a solution of

$$\nabla \cdot (\pi_t K) = -\pi_t (h - \hat{h}_t)^{\mathrm{T}} R^{-1}(t) \tag{5.1.10}$$

and ψ is given by

$$\psi = -\frac{1}{2} K(x_t, t)(h + \hat{h}_t) + q(x_t, t). \tag{5.1.11}$$

5.2 Exactness of the FPF

The FPF is an exact filter under certain conditions. To demonstrate this, it is enough to show that the equation of evolution of the posterior density of the particles given measurements' history and the equation of evolution of the true posterior are the same—when the prior densities are the same. In other words, the exactness of the FPF is demonstrated by showing that eqs. (5.1.7) and (5.1.9), are equivalent. This has been done in Amirhossein et al. (2016); Yang et al. (2012); we only repeat it here for completeness. The gain function, we recall, is

$$K(x) = \nabla \phi(x), \tag{5.2.1}$$

where $\phi(x)$ is the solution of eqs. (5.1.6a) and (5.1.6b). The task at hand is executed sufficiently should it be shown that $d\pi_t = d\pi_t^*$ given eqs. (5.1.10) and (5.1.11), the solution of which yield K.

The product of eq. (5.1.11) with $-\pi$ is

$$\begin{aligned} -\pi\psi &= \pi \frac{1}{2} K(h + \hat{h}_t) - \pi q(x_t, t) \\ &= \pi \frac{1}{2} K(h - \hat{h}_t) + \pi K \hat{h}_t - \pi q(x_t, t), \end{aligned} \tag{5.2.2a}$$

whereupon substituting eq. (5.1.10) we get

$$-\pi\psi = -\frac{1}{2} KR\left[\nabla \cdot (\pi_t K)\right] + \pi K \hat{h}_t - \pi q(x_t, t). \tag{5.2.2b}$$

Notice, by product rule, that

$$\nabla \cdot \left(\pi \left[KRK^{\mathrm{T}}\right]\right) = \pi KR \nabla \cdot (K) + KR \nabla \cdot (\pi K). \tag{5.2.3}$$

Substituting eq. (5.2.3) in eq. (5.2.2b) and using eq. (5.1.5), we get

$$-\pi\psi = -\frac{1}{2} \nabla \cdot \left(\pi \left[KRK^{\mathrm{T}}\right]\right) + \pi K \hat{h}_t, \tag{5.2.4}$$

whereupon taking divergence on both sides yields

$$-\nabla \cdot (\pi\psi) = -\frac{1}{2} \sum_{l,k=1}^{n} \frac{\partial^2}{\partial x_l \partial x_k} \left(\pi \left[KRK^{\mathrm{T}}\right]_{lk}\right) + \nabla \cdot \left(\pi K \hat{h}_t\right). \tag{5.2.5}$$

Finally, we substitute eqs. (5.1.10) and (5.2.5) in eq. (5.1.9), yielding

$$\begin{aligned} d\pi_t &= \mathscr{L}(\pi_t) dt - \nabla \cdot (\pi K) dy_t + \nabla \cdot (\pi K) \hat{h}_t dt, \\ &= \mathscr{L}(\pi_t) dt + \pi_t (h - \hat{h}_t)^{\mathrm{T}} R^{-1}(t)(dy_t - \hat{h}_t dt), \end{aligned} \tag{5.2.6}$$

as desired.

The FPF comes with many advantages: that it is exact when initialized with the true prior ensures optimal estimates; that the FPF does not require resampling, which is characteristic of

importance sampling filters, makes the FPF parallelizable and hence can be used in scenarios involving models of high dimensions; and, that the FPF has a controlled gain insures robustness. The downside of the FPF, however, is that a boundary value problem (BVP), eq. (5.1.10), needs to be solved at every time to obtain the gain, K. It turns out that an analytical solution for the BVP does not, as yet, exist. Numerical methods are resorted to to obtain an estimate of the gain. This forms the subject of the next section. But before that, we introduce the stochastic feedback particle filter.

5.3 FPF with stochastically perturbed innovation

Motivated by the two variants of EnKBF based on the perturbation of the innovation process, we obtain a stochastically-perturbed-innovation variant of the feedback particle filter by replacing the deterministic perturbation term to the innovation in eq. (5.1.3) with a stochastic term. The McKean-Vlasov stochastic differential equations for the proposed feedback particle filter, for the state-space model eqs. (5.1.1a) and (5.1.1b), consists of the equation of the mean-field process,

$$d\bar{x}_t = f(\bar{x}_t)dt + g(\bar{x}_t)d\bar{\beta}_t + K(\bar{x}_t)\circ(dy_t + R^{1/2}(t)d\bar{\eta}_t - h(\bar{x}_t)dt), \tag{5.3.1}$$

and the finite M system of equations for the evolution of interacting hypotheses of the state, $X_t := \{x_t^i\}_{i=1}^M$; that is,

$$dx_t^i = f(x_t^i)dt + g(x_t^i)d\beta_t^i + K(x_t^i)\circ(dy_t + R^{1/2}(t)d\eta_t^i - h(x_t^i)dt), \tag{5.3.2}$$

where $\bar{\beta}_t$ and $\{\beta_t\}_{i=1}^M$ are independent copies of β_t and $\bar{\eta}_t$ and $\{\eta_t\}_{i=1}^M$ are independent copies of η_t; such that, for a given function $q(x)$,

$$\pi_t[q(x)\mid Y_t] = \bar{\pi}_t[q(x)\mid Y_t] \approx \frac{1}{M}\sum_{i=1}^M q(x_t^i),$$

where $\pi_t = \text{Law}(x_t)$ and $\pi_t = \text{Law}(\bar{x}_t)$.

For notational convenience, we rewrite the *controlled Stratonovich SDE* eq. (5.3.2) as

$$dx_t = f(x_t)dt + g(x_t)d\beta_t + K(x_t)\circ(dy_t + R^{1/2}(t)d\eta_t - h(x_t)dt), \tag{5.3.3}$$

where $K = \nabla\phi(x)$ is the gain and $\phi(x)$ satisfies the Poisson equation, eqs. (5.1.6a) and (5.1.6b). $\{\eta_t\}$ is an r-dimensional vector standard Brownian motion process.

Written in Itô form, eq. (5.3.3) becomes

$$\begin{aligned} dx_t = {} & f(x_t,t)dt + g(x_t,t)d\beta_t + 2q(x_t,t)dt \\ & + K(x_t,t)(dy_t + R^{1/2}(t)d\eta_t - h(x_t,t)dt), \end{aligned} \tag{5.3.4}$$

where $q(x_t,t)$ is given by eq. (5.1.5). As in the FPF with a deterministically perturbed innovation, the equation of evolution of the filtering density $\pi(x_t\mid Y_t)$ in the FPF with stochastically perturbed innovation is given by the Fokker-Planck equation to the SDE, eq. (5.3.4); that is,

$$\begin{aligned} d\pi_t = {} & \mathscr{L}(\pi_t)dt - \nabla\cdot(\pi_t K)dy_t - \nabla\cdot(\pi_t U)dt \\ & + \sum_{k,l=1}^n \frac{\partial^2}{\partial x_k \partial x_l}(\pi_t(KRK^{\mathrm{T}})_{kl})dt, \end{aligned} \tag{5.3.5}$$

where $K = \nabla\phi$ is the solution of

$$\nabla\cdot(\pi_t K) = -\pi_t(h - \hat{h}_t)R^{-1}(t) \tag{5.3.6}$$

and U is then defined by

$$U = -Kh + 2q. \tag{5.3.7}$$

5.3.1 Exactness of FPF with stochastically perturbed innovation

We now show, that beginning with the filter posterior

$$\pi_{t_0}(x \mid Y_{t_0}) = \pi^*_{t_0}(x \mid Y_{t_0}), \tag{5.3.8}$$

where $\pi^*_{t_0}(x \mid Y_{t_0})$ is the true posterior at initial time, t_0, the filter posterior matches the true posterior at all times, t; for then we demonstrate that the filter defined by eq. (5.3.3) is exact. It suffices to show that the equations of evolution of the true posterior and filter posterior are the same; that is, eq. (5.3.5) and eq. (5.1.7) are the same.

Multiplying U in eq. (5.3.7) with $-\pi_t$ yields

$$\begin{aligned} -\pi_t U &= -\pi_t K h + 2\pi_t q \\ &= -\pi_t K(h - \hat{h}_t) - \pi_t K \hat{h}_t + 2\pi_t q, \end{aligned} \tag{5.3.9}$$

where in the second equality we introduce $\pi_t K\hat{h}_t - \pi_t K\hat{h}_t$. From eq. (5.3.6) we have

$$-\pi_t(h - \hat{h}_t) = R(t)\nabla\cdot(\pi_t K_k). \tag{5.3.10}$$

Substituting eq. (5.3.10) in eq. (5.3.9) gives

$$-\pi_t U = KR(t)\nabla\cdot(\pi_t K) - \pi_t K\hat{h}_t + 2\pi_t q. \tag{5.3.11}$$

Using eq. (5.2.3) in eq. (5.3.11) and noting the expression of $q(x_t, t)$ in eq. (5.1.5), we get

$$-\pi U = -\nabla\cdot\left(\pi\left[KRK^{\mathrm{T}}\right]\right) + \pi K\hat{h}_t. \tag{5.3.12}$$

whereupon taking divergence on both sides of eq. (5.3.12) yields

$$-\nabla\cdot(\pi U) = -\sum_{l,k=1}^{n} \frac{\partial^2}{\partial x_l \partial x_k}\left(\pi\left[KRK^{\mathrm{T}}\right]_{lk}\right) + \nabla\cdot\left(\pi K\hat{h}_t\right). \tag{5.3.13}$$

Finally, we substitute eqs. (5.3.6) and (5.3.13) for

$$-\nabla\cdot(\pi_t U) + \sum_{l,k=1}^{n} \frac{\partial^2}{\partial x_l \partial x_k}\left(\pi\left[KRK^{\mathrm{T}}\right]_{lk}\right)$$

and $\nabla\cdot(\pi_t K)$ in eq. (5.3.5) to obtain

$$d\pi_t = \mathscr{L}(\pi_t)dt + \pi_t(h - \hat{h}_t)^{\mathrm{T}} R^{-1}(t)(dy_t - \hat{h}_t dt), \tag{5.3.14}$$

which proves exactness. We now summarise the result from the preceding discussion in the following Theorem.

Theorem 5.3.1 (Exactness of FPF with stochastic innovation)**.** *Let initial filter posterior, π_{t_0}, be equal to the true posterior $\pi^*_{t_0}$ at time t_0. Then the filter governed by eq. (5.3.3) is exact.*

Remark 5.3.1 (The gain)**.** *The equation from which the gain is obtained is the same for both FPF with deterministically perturbed innovation and that with stochastically perturbed innovation.*

5.4 Approximation of the gain function

The gain, K, is obtained by solving eq. (5.3.6), with boundary condition

$$\lim_{x_k \to \pm\infty} K_{kl}\pi_t = 0. \tag{5.4.1}$$

The scalar form of eq. (5.3.6) is

$$\frac{\partial}{\partial x}(\pi_t K) = -\frac{1}{R(t)}\pi_t(h - \hat{h}_t), \tag{5.4.2}$$

the integration of which yields

$$K = -\frac{1}{\pi_t R(t)}\int_{-\infty}^{x} \pi_t(h - \hat{h}_t)dz. \tag{5.4.3}$$

Extension of the integration in eq. (5.4.3) is involving [Sahoo (2010)]. A number of approximation methods have been developed. We introduce a few here.

5.4.1 Direct approximation

For a scalar signal, the gain as expressed in eq. (5.4.3) can be estimated by means of empirical estimate [Yang (2014)],

$$K \approx -\frac{1}{MR(t)\tilde{\pi}_t}\sum_{i=1}^{M}(h(x_t^i) - \hat{h}_t)H(x - x_t^i), \tag{5.4.4}$$

where

$$\hat{h}_t \approx \frac{1}{M}\sum_{i=1}^{M} h(x_t^i), \tag{5.4.5}$$

$H(.)$ is the Heaviside function and

$$\tilde{\pi}_t = \frac{1}{M}\sum_{i=1}^{M}\pi_t^i \tag{5.4.6}$$

where

$$\pi_t^i = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{1}{2\sigma^2}(x - x_t^i)^{\mathrm{T}}(x - x_t^i)\right),$$

and σ can be chosen such that $\sigma \to 0$ as $M \to \infty$. The following algorithm gives a summary of the approximation.

Algorithm 5.4.1 Direct approximation of the gain

1: Approximate $\hat{h}_t$ as follows: $\hat{h}_t \approx \frac{1}{M}\sum_{i=1}^{M} h(x_t^i)$
2: Approximate the mean posterior: $\tilde{\pi}_t = \frac{1}{M}\sum_{i=1}^{M}\pi_t^i$
3: Compute the gain: $K = -\frac{1}{MR(t)\tilde{\pi}_t}\sum_{i=1}^{M}(h(x_t^i) - \hat{h}_t)H(x - x_t^i)$

5.4.2 Weak formulation of the Poisson BVP

Consider the Poisson equation, eqs. (5.1.6a) and (5.1.6b), which we rewrite here for convenience,

$$\nabla\cdot\left(\pi_t\nabla\phi(x)\right) = -\frac{1}{R}(h(x) - \hat{h}_t)\pi_t, \tag{5.4.7a}$$

$$\int \phi(x)\pi_t dx = 0, \tag{5.4.7b}$$

with homogenous boundary condition

$$\lim_{x\to\pm\infty} \phi(x)\pi_t = 0. \tag{5.4.7c}$$

Let $\psi \in C_c^\infty([-\infty,\infty])$; that is, $\psi(x)$ is an infinitely differentiable function with a compact support in $\mathbb{R}$. Upon multiplying eq. (5.4.7a) with $\psi(x)$ and integrating, we get

$$\int_{-\infty}^{\infty} \psi(x)\nabla\cdot\big(\pi_t\nabla\phi(x)\big)dx = -\frac{1}{R}\int_{-\infty}^{\infty}\psi(x)(h(x)-\hat{h}_t)\pi_t dx, \tag{5.4.8}$$

Integrating eq. (5.4.8) by parts yields

$$\big[\psi(x)\pi_t\nabla\phi(x))\big]_{-\infty}^{\infty} - \int_{-\infty}^{\infty}\nabla\psi(x)\pi_t\nabla\phi(x)dx = -\frac{1}{R}\int_{-\infty}^{\infty}\psi(x)(h(x)-\hat{h}_t)\pi_t dx. \tag{5.4.9}$$

From the boundary condition eq. (5.4.7c), the first term in eq. (5.4.9) is zero, and so we have

$$\int_{-\infty}^{\infty}\nabla\psi(x)\pi_t\nabla\phi(x)dx = \frac{1}{R}\int_{-\infty}^{\infty}\psi(x)(h(x)-\hat{h}_t)\pi_t dx, \tag{5.4.10}$$

which we can rewrite, by means of expectation, as

$$\pi_t\left[\nabla\psi(x)\nabla\phi(x)\right] = \frac{1}{R}\pi_t\left[\psi(x)(h(x)-\hat{h}_t)\right]. \tag{5.4.11}$$

Therefore, any solution ϕ to eqs. (5.4.7a) to (5.4.7c) satisfies the weak formulation

$$a(\psi,\phi) = l(\psi), \tag{5.4.12a}$$

where

$$a(\psi,\phi) := \pi_t\left[\nabla\psi(x)\nabla_x\phi(x_t)\right], \tag{5.4.12b}$$

and

$$l(\psi) := \frac{1}{R}\pi_t\left[\psi(x)(h(x)-\hat{h}_t)\right]. \tag{5.4.12c}$$

The gain term is obtained from eq. (5.4.11) by substituting $K = \nabla\phi(x)$.

5.4.2.1 Constant gain approximation

Choosing the basis functions to be the canonical coordinate vectors, then the test function becomes $\psi(x) = x$. Substituting this in eq. (5.4.11) yields

$$\pi_t[K] = \frac{1}{R}\pi_t\left[x(h(x)-\hat{h}_t)\right]. \tag{5.4.13}$$

Supposing that we approximate the gain, K, empirically,

$$\begin{aligned}\tilde{K} &= \pi_t[K]\\ &\approx \frac{1}{R}\frac{1}{M}\sum_{i=1}^{M} x^i(h(x^i)-\hat{h}_t),\end{aligned} \tag{5.4.14}$$

we obtain the constant approximation of the gain. Recall that the two formulations of the EnKBF, eqs. (4.3.1) and (4.3.3), have a gain

$$\begin{aligned}K &= R^{-1}\frac{1}{M-1}\sum_{i=1}^{M}(x^i-\bar{x})(h(x^i)-\hat{h}_t)^{\mathrm{T}}\\ &\approx R^{-1}\frac{1}{M}\sum_{i=1}^{M}x^i(h(x^i)-\hat{h}_t)^{\mathrm{T}},\end{aligned} \tag{5.4.15}$$

which is equivalent to eq. (5.4.14); and hence EnKBF is a FPF with the gain obtained by constant approximation.

Constant approximation of the gain is a popular method in practice [Amirhossein et al. (2016)]. The summary of constant gain approximation is given in the following algorithm

Algorithm 5.4.2 Constant gain approximation

1: Approximate $\hat{h}_t$ as follows: $\hat{h}_t \approx \frac{1}{M}\sum_{i=1}^{M} h(x^i)$
2: Compute the gain: $K = \frac{1}{R}\frac{1}{M}\sum_{i=1}^{M} x^i(h(x_t^i) - \hat{h}_t)$

5.4.2.2 Galerkin approximation

In the Galerkin approximation approach, the solution to the gain, K, is approximated using basis functions $\{\psi_l\}_{l=1}^{L}$ as [Amirhossein et al. (2016); Yang et al. (2012)]

$$\tilde{K}(x) = \sum_{l=1}^{L} c_l \nabla\psi_l(x), \tag{5.4.16}$$

where $\{c_l\}_{l=1}^{L}$ are constants chosen such that

$$\pi_t\left[\nabla\psi(x)\tilde{K}(x)\right] = \frac{1}{R}\pi_t\left[\psi(x)(h(x) - \hat{h}_t)\right]. \tag{5.4.17}$$

Galerkin approximation entails solving an algebraic equation,

$$Ac = b, \tag{5.4.18a}$$

for a vector of constants, $c = [c_1, c_2, c_3, ..., c_n]^{\mathrm{T}}$. A is an $n \times n$ matrix, whose components are approximated as

$$A_{kl} \approx \frac{1}{M}\sum_{i=1}^{M} \nabla\psi_k(x^i).\nabla\psi_l(x^i); \tag{5.4.18b}$$

b, on the other hand, is a vector of length n, and whose components are approximated as

$$b_l \approx \frac{1}{R}\frac{1}{M}\sum_{i=1}^{M} \psi_k(x^i)(h(x^i) - \hat{h}_t). \tag{5.4.18c}$$

With the vector of constants, c, at hand, we can now obtain an approximation of the gain by means of eq. (5.4.16). Galerkin approximation is summarised in the following algorithm.

Algorithm 5.4.3 Galerkin approximation of the gain

1: Approximate $\hat{h}_t$ as follows: $\hat{h}_t \approx \frac{1}{M}\sum_{i=1}^{M} h(x^i)$
2: **for** $l = 1$ **to** n **do**
3: Compute the components of b: $b_l \approx \frac{1}{R}\frac{1}{M}\sum_{i=1}^{M} \psi_k(x^i)(h(x^i) - \hat{h}_t)$
4: **for** $k = 1$ **to** n **do**
5: Compute the components of A: $A_{kl} \approx \frac{1}{M}\sum_{i=1}^{M} \nabla\psi_k(x^i).\nabla\psi_l(x^i)$
6: **end for**
7: **end for**
8: Solve for c in $Ac = b$
9: Compute the gain $K \approx \sum_{l=1}^{L} c_l \nabla\psi_l(x)$

The disadvantage of Galerkin approximation is the need to invert an $n \times n$ matrix, A, in eq. (5.4.18a) every time in solving for c. It is for this reason that data based methods—for example semigroup method and Karhunen Loeve expansion method—are preferred.

5.4.3 Kernel-based approximation

Consider the Poisson equation, eq. (5.4.7a), whose solution, for any fixed $\epsilon > 0$, is given by the fixed-point representation [Amirhossein et al. (2016)],

$$\phi = \exp\left(-\frac{\epsilon}{\pi_t}\nabla\cdot(\pi_t\nabla)\right)\phi + \int_0^\epsilon \exp\left(-\frac{s}{\pi_t}\nabla\cdot(\pi_t\nabla)\right)\frac{1}{R}(h-\hat{h}_t)\,ds, \tag{5.4.19}$$

where $\exp\left(-\frac{\epsilon}{\pi_t}\nabla\cdot(\pi_t\nabla)\right)$ is a Markov semigroup generated by the linear operator, $-\frac{1}{\pi_t}\nabla\cdot(\pi_t\nabla)$. Approximation of eq. (5.4.19) is necessitated by an approximation of the Markov semigroup, $\exp\left(-\frac{\epsilon}{\pi_t}\nabla\cdot(\pi_t\nabla)\right)$, using an operator

$$T_\epsilon f(x) := \sum_{i=1}^N k_\epsilon(x,x^i)f(x^i), \tag{5.4.20}$$

where

$$k_\epsilon(x,y) = \frac{n_\epsilon^{-1}g_\epsilon(x-y)}{\sqrt{\frac{1}{M}\sum_{i=1}^M g_\epsilon(x-x^i)}\sqrt{\frac{1}{M}\sum_{i=1}^M g_\epsilon(y-x^i)}}, \tag{5.4.21}$$

g_ϵ is an n-dimensional Gaussian function and n_ϵ is a normalization factor, chosen so that $T_\epsilon 1 = 1$.

For small ϵ,

$$\int_0^\epsilon \exp\left(-\frac{s}{\pi_t}\nabla\cdot(\pi_t\nabla)\right)ds \approx \epsilon\frac{1}{R}(h-\hat{h}_t). \tag{5.4.22}$$

This leads to an approximation of ϕ; that is,

$$\phi_\epsilon = T_\epsilon\phi_\epsilon + \epsilon\frac{1}{R}(h-\hat{h}_t). \tag{5.4.23}$$

It now remains to obtain the gradient of ϕ_ϵ, yielding an approximation of the gain. The gain is approximated as follows:

$$K = \nabla T_\epsilon\phi_\epsilon + \epsilon\frac{1}{R}\nabla T_\epsilon(h-\hat{h}), \tag{5.4.24}$$

where the operator ∇T_ϵ is approximated as follows

$$\begin{aligned}\nabla T_\epsilon f(x) &:= \sum_{i=1}^N \nabla k_\epsilon(x,x^i)f(x^i)\\ &= \frac{1}{2\epsilon}\sum_{i=1}^M k_\epsilon(x,x^i)f(x^i)\left(x^i - \sum_{j=1}^M k_\epsilon(x,x^i)x^j\right).\end{aligned} \tag{5.4.25}$$

From eq. (5.4.24), $K(x^i)$ can be written as

$$K(x^i) = \sum_{j=1}^M a_{ij}x^j, \tag{5.4.26}$$

where

$$a_{ij} := \frac{1}{2\epsilon}k_\epsilon(x^i,x^j)\left(r_j - \sum_{l=1}^M k_\epsilon(x^i,x^l)r_l\right), \tag{5.4.27a}$$

and

$$r_i := \phi_\epsilon(x^i) + \epsilon\frac{1}{R}(h(x^i)-\hat{h}_t). \tag{5.4.27b}$$

Algorithm 5.4.4 Kernel-based gain approximation

for $i = 1$ **to** M **do**
  for $j = 1$ **to** M **do**
    Compute $g_{ij} := \exp\left(\frac{-|x^i - x^j|^2}{4\epsilon}\right)$
    Compute $k_{ij} := \frac{g_{ij}}{\sqrt{\sum_{l=1}^M g_{il}}\sqrt{\sum_{l=1}^M g_{jl}}}$
    Compute $T_{ij} := \frac{k_{ij}}{\sqrt{\sum_{l=1}^M k_{il}}}$
  end for
  Approximate $\hat{h}_t$ as follows: $\hat{h}_t \approx \frac{1}{M}\sum_{i=1}^M h(x_t^i)$
  Compute $\phi_i = \sum_{j=1}^M T_{ij}\phi_j + \epsilon\frac{1}{R}(h(x^i) - \hat{h})$
  Compute $\phi_i = \phi_i - \frac{1}{M}\sum_{j=1}^M \phi_j$
  Compute $r_i = \phi_i + \epsilon\frac{1}{R}(h(x^i) - \hat{h})$
  Compute $a_{ij} = \frac{1}{2\epsilon}T_{ij}(r_j - \sum_{l=1}^M T_{il}r_l)$
  Compute $K^i = \sum_{j=1}^M a_{ij}x^j$
end for

5.4.4 Gain approximation based on optimal coupling

Consider two marginal distributions, say, π and π_ϵ. It concerns optimal coupling to obtain a joint distribution of the two distributions. Optimal coupling entails obtaining an optimal transport map, T_ϵ, by solving the following optimal transport problem [Amirhossein et al. (2016); Reich and Cotter (2015)]

$$\text{Objective:} \qquad \min_{T_\epsilon} \mathbb{E}\left[|T_\epsilon(x) - x|^2\right] \tag{5.4.28a}$$

$$\text{Constraints:} \qquad x \sim \pi, \;\; T_\epsilon(x) \sim \pi_\epsilon, \tag{5.4.28b}$$

where

$$\pi_\epsilon(x) := \pi(x)(1 + \epsilon\frac{1}{R}(h(x) - \hat{h})) \tag{5.4.29}$$

is a family of densities parametrized by $\epsilon > 0$ sufficiently small.

The ensemble transform algorithm [Reich and Cotter (2015)] provides an approximation to T_ϵ given an ensemble $\{x^i\}_{i=1}^M$ drawn from π. In the framework of ensemble transform, the optimal transport problem, eqs. (5.4.28a) and (5.4.28b), is recast as follows:

$$\text{Objective:} \qquad \min_{t_{ij}} \sum_{i=1}^M \sum_{j=1}^M t_{ij}|x^i - x^j|^2 \tag{5.4.30a}$$

$$\text{Constraints:} \qquad \sum_{i=1}^M t_{ij} = \frac{1}{M}, \quad \sum_{j=1}^M t_{ij} = \frac{R + \epsilon(h(x^j) - \hat{h})}{RM}, \quad t_{ij} \geq 0 \quad \forall\, i \neq j, \tag{5.4.30b}$$

the solution, t_{ij}^*, of which is called the optimal coupling. The objective to be minimized can, for example, be the distance between the two measures, π and π_ϵ—the well-known Wasserstein distance.

Let

$$a_{ij} = \frac{M t_{ij}^* - \delta_{ij}}{\epsilon},$$

where δ_{ij} is the Kronecker delta. Then, the gain

$$K(x^i) := \sum_{j=1}^{M} a_{ij} x_t^j. \tag{5.4.31}$$

The following algorithm summarises the foregoing discussion.

Algorithm 5.4.5 Optimal coupling-based gain approximation

for $i = 1$ **to** M **do**
  Approximate $\hat{h}_t$ as follows: $\hat{h}_t \approx \frac{1}{M}\sum_{i=1}^{M} h(x_t^i)$
  Compute $\pi_i = \frac{1}{M}$ and $\pi_{\epsilon i} = \frac{R + \epsilon(h(x^i) - \hat{h}_t)}{RM}$
  for $j = 1$ **to** M **do**
    Compute $d_{ij} := |x^i - x^j|^2$
    Solve for t_{ij}^* in eqs. (5.4.30a) and (5.4.30b)
    Compute $a_{ij} = \frac{M t_{ij}^* - \delta_{ij}}{\epsilon}$
  end for
  Compute $K(x^i) = \sum_{j=1}^{M} a_{ij} x_t^j$
end for

We now turn to feedback particle filters whose feedback structure is obtained using optimal transport algorithms.

5.5 Ensemble Transform Particle Filter

The ensemble transform particle filter (ETPF) [Reich (2013); Reich and Cotter (2015)] is, as the FPF, characterised by a control law, in the sense of optimal transportation, which moves the particles to the convenient position in the state space, thus improving upon the performance of the filter. The notable distinction between ETPF and particle filters with resampling is that the resampling step is replaced with a linear transformation. The transformation seeks to establish an optimal coupling between the prior ensemble and the posterior ensemble.

Most precisely, the linear transport problem is as follows:

$$T^* = \underset{T \in \mathscr{R}^{M\times M}}{\arg\min} \sum_{i,j=1}^{M} t_{ij} \|x_{t_n}^i - x_{t_n}^j\|^2, \tag{5.5.1}$$

under the constraints

$$\sum_{i=1}^{M} t_{ij} = \frac{1}{M}, \ \sum_{j=1}^{M} t_{ij} = w_{t_n}^i, \ \text{and } t_{ij} > 0, \tag{5.5.2}$$

where $\{w_{t_n}^i\}_{i=1}^M$ are weights at time $t_n = n\delta t$ and t_{ij} represents the element in row i and column j of the $M \times M$ matrix T. The weights are propagated as follows:

$$dw_t = \frac{1}{R} w_t^i (h(x_t^i) - \hat{h}_t)^{\mathrm{T}} (dy_t - \hat{h}_t dt), \tag{5.5.3}$$

where

$$\hat{h}_t = \frac{1}{M}\sum_{i=1}^{M} h(x_t^i).$$

To avoid negative weights, eq. (5.5.3) is approximated by

$$w_{t_{n+1}}^i \approx w_{t_n}^i \exp\left(-\frac{1}{2R}(h(x_{t_n}^i))^T h(x_{t_n}^i)\delta t - (h(x_{t_n}^i))^T \delta y_{t_n}\right), \tag{5.5.4}$$

and initialized with

$$w_{t_0}^i = \frac{1}{M}.$$

For the ETPF, the importance weights of the prior samples $\{x^i\}_{i=1}^M$ are given by $W_1 = \{w_1^i\}_{i=1}^M$ and $W_2 = \{w_2^i = 1/M\}_{i=1}^M$. Moreover, $X_1 = X_2 = \{x^i\}_{i=1}^M$. The posterior samples are given by

$$\tilde{x}_{t_n}^j = \frac{1}{M}\sum_{i=1}^M x_{t_n}^i t_{ij}^*. \tag{5.5.5}$$

5.6 Feedback formulation based on the Schrödinger problem

The Schrödinger problem [Schrödinger (1931)] arises in the context of boundary value problems involving stochastic differential equations. Filtering problem can be cast as a Schrödinger problem, by means of Bayes' rule, and in the space of probability measures. In the ensuing argument, we invoke the algorithms used in optimal transportation [Peyré and Cuturi (2018)] to solve the Schrödinger problem, which solution provides the estimates of the filtering distribution.

Definition 5.6.1. ***Schrödinger problem:***— *Find two functions $\hat{\phi}_{t_n}(x_{t_n})$ and $\phi_{t_{n+1}}(x_{t_{n+1}})$, satisfying the following equations*

$$\pi_{t_n}(x_{t_n} \mid Y_{t_n}) = \pi_{t_n}^{\phi}(x_{t_n} \mid Y_{t_n})\hat{\phi}_{t_n}(x_{t_n}), \tag{5.6.1a}$$

$$\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_{n+1}}) = \pi_{t_{n+1}}^{\phi}(x_{t_{n+1}} \mid Y_{t_{n+1}})\phi_{t_{n+1}}(x_{t_{n+1}}), \tag{5.6.1b}$$

$$\pi_{t_{n+1}}^{\phi}(x_{t_{n+1}} \mid Y_{t_{n+1}}) = \int \pi_{t_{n+1}}(x_{t_{n+1}} \mid x_{t_n})\pi_{t_n}^{\phi}(x_{t_n} \mid Y_{t_n})dx_{t_n}, \tag{5.6.1c}$$

$$\hat{\phi}_{t_n}(x_{t_n}) = \int \pi_{t_{n+1}}(x_{t_{n+1}} \mid x_{t_n})\phi_{t_{n+1}}(x_{t_{n+1}})dx_{t_{n+1}}, \tag{5.6.1d}$$

where $\pi_{t_n}(x_{t_n} \mid Y_{t_n})$ and $\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_{n+1}})$ are the marginal filtering distributions at time t_n and t_{n+1}, respectively.

When solved, eqs. (5.6.1a) to (5.6.1d) yield transition density

$$\pi_{t_{n+1}}^{\phi}(x_{t_{n+1}} \mid x_{t_n}) := \frac{\phi_{t_{n+1}}(x_{t_{n+1}})}{\hat{\phi}_{t_n}(x_{t_n})}\pi_{t_{n+1}}(x_{t_{n+1}} \mid x_{t_n}), \tag{5.6.2}$$

such that

$$\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_{n+1}}) := \int \pi_{t_{n+1}}^{\phi}(x_{t_{n+1}} \mid x_{t_n})\pi_{t_n}(x_{t_n} \mid Y_{t_n})dx_{t_n}, \tag{5.6.3}$$

where $\pi_{t_{n+1}}^{\phi}(x_{t_{n+1}} \mid x_{t_n})$ can be intuitively understood to define a coupling between the probability density function at time t_n, $\pi_{t_n}(x_{t_n} \mid Y_{t_n})$, and the filtering density function at time t_{n+1}, $\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_{n+1}})$.

How do these notions fit in to filtering? Recall that, in the discrete setting, Bayes' Theorem gives an expression of the filtering density, $\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_{n+1}})$, at time t_{n+1} as

$$\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_{n+1}}) = \frac{\pi_{t_{n+1}}(y_{t_{n+1}} \mid x_{t_{n+1}})\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_n})}{\int \pi_{t_{n+1}}(y_{t_{n+1}} \mid x_{t_{n+1}})\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_n})dx_{t_n}}, \tag{5.6.4}$$

where

$$\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_n}) = \int \pi_{t_{n+1}}(x_{t_{n+1}} \mid x_{t_n})\pi_{t_n}(x_{t_n} \mid Y_{t_n})dx_{t_n} \tag{5.6.5}$$

is the prediction density at time t_{n+1} [Särkkä (2013); Jazwinski (1970)]. It then becomes apparent that eqs. (5.6.3) and (5.6.4) are equivalent. Notice that the Schrödinger problem reduces to eq. (5.6.3) where, beginning with the filtering distribution $\pi_{t_n}(x_{t_n} \mid Y_{t_n})$, we directly obtain the filtering distribution $\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_{n+1}})$ at time t_{n+1} via the twisted transition density $\pi^{\phi}_{t_{n+1}}(x_{t_{n+1}} \mid x_{t_n})$.

We use empirical estimates of the densities to solve eqs. (5.6.1a) to (5.6.1d) governing the Schrödinger problem; that is, given an ensemble $\{x^i_{t_n}\}_{i=1}^M$ at time t_n, we obtain an estimate of the prediction density, eq. (5.6.5), by

$$\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_n}) = \frac{1}{M}\sum_{i=1}^{M} \pi_{t_{n+1}}(x_{t_{n+1}} \mid x^i_{t_n}). \tag{5.6.6}$$

The filtering density, eq. (5.6.5), is then empirically estimated as follows:

$$\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_{n+1}}) = \frac{\frac{1}{M}\sum_{i=1}^{M} \pi_{t_{n+1}}(y_{t_{n+1}} \mid x_{t_{n+1}})\pi_{t_{n+1}}(x_{t_{n+1}} \mid x^i_{t_n})}{\int \pi_{t_{n+1}}(y_{t_{n+1}} \mid x_{t_{n+1}})\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_n})dx_{t_n}}. \tag{5.6.7}$$

To solve the Schrödinger problem, eqs. (5.6.1a) to (5.6.1d), we make the following proposition

$$\pi^{\phi}_{t_n}(x_{t_n} \mid Y_{t_n}) = \frac{1}{M}\sum_{i=1}^{M} \alpha^i \delta(x_{t_n} - x^i_{t_n}), \tag{5.6.8a}$$

$$\hat{\phi}_{t_n}(x_{t_n}) = \frac{1}{M}\sum_{i=1}^{M} \frac{1}{\alpha^i}, \tag{5.6.8b}$$

where

$$\frac{1}{M}\sum_{i=1}^{M} \alpha^i = 1. \tag{5.6.8c}$$

Substituting eqs. (5.6.8a) and (5.6.8b) in to eqs. (5.6.1a) to (5.6.1d) yields,

$$\pi_{t_n}(x_{t_n} \mid Y_{t_n}) = \frac{1}{M}\sum_{i=1}^{M} \delta(x_{t_n} - x^i_{t_n}), \tag{5.6.9a}$$

$$\pi^{\phi}_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_{n+1}}) = \frac{1}{M}\sum_{i=1}^{M} \alpha^i \pi_{t_{n+1}}(x_{t_{n+1}} \mid x^i_{t_n}), \tag{5.6.9b}$$

$$\phi_{t_{n+1}}(x_{t_{n+1}}) = \frac{N}{\int (N) dx_{t_n} \frac{1}{M}\sum_{i=1}^{M} \alpha^i \pi_{t_{n+1}}(x_{t_{n+1}} \mid x^i_{t_n})}, \tag{5.6.9c}$$

where

$$N = \pi_{t_{n+1}}(y_{t_{n+1}} \mid x_{t_{n+1}}) \sum_{i=1}^{M} \pi_{t_{n+1}}(x_{t_{n+1}} \mid x^i_{t_n}).$$

Therefore, one sets $\hat{\phi}_{t_n}(x_{t_n}) = \frac{1}{M}\sum_{i=1}^{M} \frac{1}{\alpha^i}$ and defines $\phi_{t_{n+1}}(x_{t_{n+1}})$ by eq. (5.6.9c) in order to solve the Schrödinger problem; for then, given a twisted prediction density

$$\begin{aligned} \pi^{\phi}_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_n}) &= \frac{1}{M}\sum_{i=1}^{M} \pi^{\phi}_{t_{n+1}}(x_{t_{n+1}} \mid x^i_{t_n}) \\ &= \frac{1}{M}\sum_{i=1}^{M} \frac{\phi_{t_{n+1}}(x_{t_{n+1}})}{\hat{\phi}_{t_n}(x^i_{t_n})} \pi_{t_{n+1}}(x_{t_{n+1}} \mid x^i_{t_n}), \end{aligned} \tag{5.6.10}$$

by eq. (5.6.2), the relationship between the prediction density, $\pi^{\phi}_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_n})$, and the twisted prediction density, $\pi^{\phi}_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_n})$, is

$$\frac{\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_n})}{\pi^{\phi}_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_n})} = \frac{\frac{1}{M}\sum_{i=1}^{M} \pi_{t_{n+1}}(x_{t_{n+1}} \mid x^i_{t_n})}{\frac{1}{M}\sum_{i=1}^{M} \frac{\phi_{t_{n+1}}(x_{t_{n+1}})}{\hat{\phi}_{t_n}(x^i_{t_n})} \pi_{t_{n+1}}(x_{t_{n+1}} \mid x^i_{t_n})}; \tag{5.6.11}$$

from which, if we draw samples from the twisted distribution

$$x^i_{t_n} \sim \pi^{\phi}_{t_{n+1}}(x_{t_{n+1}} \mid x^i_{t_n}), \tag{5.6.12}$$

then we can approximate the prediction density, $\pi^{\phi}_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_n})$, by

$$\pi^{\phi}_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_n}) \approx \frac{1}{M} w^i \delta(x_{t_n} - x_{t_n}), \tag{5.6.13a}$$

where

$$w^i = \frac{\pi_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_n})}{\pi^{\phi}_{t_{n+1}}(x_{t_{n+1}} \mid Y_{t_n})}. \tag{5.6.13b}$$

Now, suppose that we have $L = kM$, where $k \in \mathbb{N}$, samples $\{x^j_{t_n}\}_{j=1}^{L}$ drawn from $\pi_{t_n}(x_{t_n} \mid Y_{t_n})$. We obtain a bi-stochastic matrix $Q \in \mathbb{R}^{L\times M}$, which approximates the Markov process defined by $\pi(x_{t_{n+1}} \mid x_{t_n})$. The Schrödinger problem, eqs. (5.6.1a) to (5.6.1d), can then be recast as follows: Find two non-negative vectors $u \in \mathbb{R}^L$ and $v \in \mathbb{R}^M$ so that,

$$P^* = \mathrm{diag}(u) Q \mathrm{diag}(v)^{-1}, \tag{5.6.14}$$

given that P^* belong to a polytope

$$U := \left\{ P \in \mathbb{R}^{L\times M} : P \geq 0, \sum_{j=1}^{L} p_{ji} = p_1, \sum_{i=1}^{M} p_{ji} = p_0 \right\}.$$

P^* is also a solution to the optimization problem defined by minimizing the distance between all possible bi-stochastic matrices P and Q; that is,

$$P^* = \arg\min_{P\in U} \mathrm{KL}(P\|Q), \tag{5.6.15}$$

where KL is the Kullback Leibler divergence between $P \in U$ and Q; that is,

$$\mathrm{KL}(P\|Q) := \sum_{j,i=1}^{L,M} p_{ji} \log \frac{p_{ji}}{q_{ji}}, \tag{5.6.16}$$

where p_{ji} and q_{ji} are the elements of, respectively, matrices P and Q in row j and column i.

Now the feedback particle filter in the Schrödinger formulation is as follows: we first obtain an M-sized ensemble of states $\{\bar{x}^i_{t_n}\}_{i=1}^{M}$ using a forecast distribution, $\pi_{t_n}(\bar{x}^i_{t_n} \mid Y_{t_n})$, with respect to which each particle evolves according to the weak form of the Fokker-Planck equation, eq. (2.4.15); that is

$$\bar{x}^i_{t_n} = \bar{x}^i_{t_{n-1}} + f(\bar{x}^i_{t_{n-1}})\delta t. \tag{5.6.17}$$

Secondly, $L = kM$ ensembles are obtained as follows:

$$x^j_{t_n} = \bar{x}^j_{t_n} + g(t_n) d\beta^j_{t_n}, \tag{5.6.18}$$

where $\{\bar{x}^j_{t_n}\}_{j=1}^L$ are obtained by replicating each particle $\bar{x}^i_{t_n}$ k times. The weights are obtained using

$$w^j_{t_n} = \exp\Big(-\frac{1}{2\delta t}(-2\delta y^{\mathrm{T}}_{t_n} h(x^j_{t_n})\delta t + h^{\mathrm{T}}(x^j_{t_n})h(x^j_{t_n})\delta t^2)\Big) w^j_{t_{n-1}}. \tag{5.6.19}$$

This gives a particle weight system, $\{x^j_{t_n}, \tilde{w}^j_{t_n}\}_{j=1}^L$, where $\tilde{w}^j_{t_n}$ signifies the normalised $w^j_{t_n}$—normalisation done according to

$$\tilde{w}^j_{t_n} = \frac{w^j_{t_n}}{\sum_{j=1}^L w^j_{t_n}}.$$

By $Q \in \mathbb{R}^{L\times M}$, we denote and understand a matrix whose elements are

$$q_{ji} = \exp\Big(-\frac{1}{2g^2\delta t}\|x^j_{t_n} - \bar{x}^i_{t_n}\|^2\Big). \tag{5.6.20}$$

Then we solve the Schrödinger problem defined by eq. (5.6.15), from whence we obtain P^*. P^* can be obtained, iteratively, by means of the Sinkhorn scheme stipulated in the following algorithm, whose implementation details are stipulated in [Cuturi (2013)].

Algorithm 5.6.1 Sinkhorn iteration

Input: p_0 and p_1.
Output: P.

1: Compute Q and set $P^0 = Q$
2: **while** k > 1 **do**
3: Compute: $u^{k+1} = \mathrm{diag}(\sum_{m=1}^M P^k)^{-1} p_1$.
4: Compute: $v^{k+1} = \mathrm{diag}(\sum_{m=1}^M p_0)^{-1} \sum_{l=1}^L (\mathrm{diag}(u^{k+1})P^k)^{\mathrm{T}}$.
5: Compute: $P^{k+1} = \mathrm{diag}(u^{k+1})P^k \mathrm{diag}(v^{k+1})^{-1}$.
6: **end while**

Filtered particles are eventually obtained thus

$$\tilde{x}^i_{t_n} = \sum_{j=1}^L x^j_{t_n} p^*_{ji} + g(\delta t)^{1/2}\xi^i_{t_n}, \qquad \xi^i_{t_n} \sim \mathcal{N}(\bar{0}_{n\times 1}, \mathrm{I}_{n\times n}), \tag{5.6.21}$$

where $\bar{0}_{n\times 1}$ and $\mathrm{I}_{n\times n}$ are, respectively, the null vector and the identity matrix of sizes as indicated in the subscripts. This yields the proposed Sinkhorn particle filter (SPF), whose summary is in Algorithm 5.6.2.

Alternatively to obtaining the particles by means of eq. (5.6.21), we can resample $\{x^j_{t_n}\}_{j=1}^L$ such that

$$\mathbb{P}(x^i_{t_n} = x^j_{t_n}) = p^*_{ji}, \qquad \forall i = 1, 2, ..., M. \tag{5.6.22}$$

This results in the resampling Sinkhorn Particle filter (RSPF), the summary of which is in Algorithm 5.6.3.

Notice that all these formulations in this section are done on a discrete setting. Time-continuous settings are arrived at by passing to the formal limit as $\delta t \to 0$. We now consider an example to test the performance of the SPF and RSPF in comparison to the BPF.

Example 5.6.1: Lorentz 63 model

We return to the stochastic Lorentz 63 model introduced in Example (3.5.1), but with $G = 0.1\mathrm{I}_{3\times 3}$ and $R = \mathrm{I}_{3\times 3}$, where $\mathrm{I}_{3\times 3}$ is an identity matrix of order 3.

Algorithm 5.6.2 Sinkhorn particle filter

Input: $x^i_{t_0}$, $w^i_{t_0} = 1/M \; \forall i \in \{1, 2, ..., M\}$, k, and $\delta y_{[t_0,t_T]}$.
Output: $\hat{x}_{[t_0,t_T]}$.

for $n = 1$ **to** $N, \delta t > 0$ **do**
  for $i = 1$ **to** M **do**
    Obtain $\bar{x}^i_{t_n}$ using eq. (5.6.17)
    for $j = 1$ **to** L **do**
      Replicate $\bar{x}^i_{t_n}$ k times and obtain $x^j_{t_n}$ using eq. (5.6.17)
      Compute weights $w^j_{t_n}$ using eq. (5.6.19)
      Compute q_{ji} using eq. (5.6.20)
    end for
    Calculate P^* by solving the the Schrödinger problem via Algorithm 5.6.1
    Compute the filtered particles $\tilde{x}^i_{t_n}$ using eq. (5.6.21)
  end for
  Compute $\hat{x}_{t_n} = \frac{1}{M}\sum_{i=1}^{M} \tilde{x}^i_{t_n}$
end for

Algorithm 5.6.3 Resampling Sinkhorn particle filter

Input: $x^i_{t_0}$, $w^i_{t_0} = 1/M \; \forall i \in \{1, 2, ..., M\}$, k, and $\delta y_{[t_0,t_T]}$.
Output: $\hat{x}_{[t_0,t_T]}$.

for $n = 1$ **to** $N, \delta t > 0$ **do**
  for $i = 1$ **to** M **do**
    Obtain $\bar{x}^i_{t_n}$ using eq. (5.6.17)
    for $j = 1$ **to** L **do**
      Replicate $\bar{x}^i_{t_n}$ k times and obtain $x^j_{t_n}$ using eq. (5.6.17)
      Compute weights $w^j_{t_n}$ using eq. (5.6.19)
      Compute q_{ji} using eq. (5.6.20)
    end for
    Calculate P^* by solving the the Schrödinger problem via Algorithm 5.6.1
    Resample the particles according to eq. (5.6.22) to obtain the filtered particles $\tilde{x}^i_{t_n}$
  end for
  Compute $\hat{x}_{t_n} = \frac{1}{M}\sum_{i=1}^{M} \tilde{x}^i_{t_n}$
end for

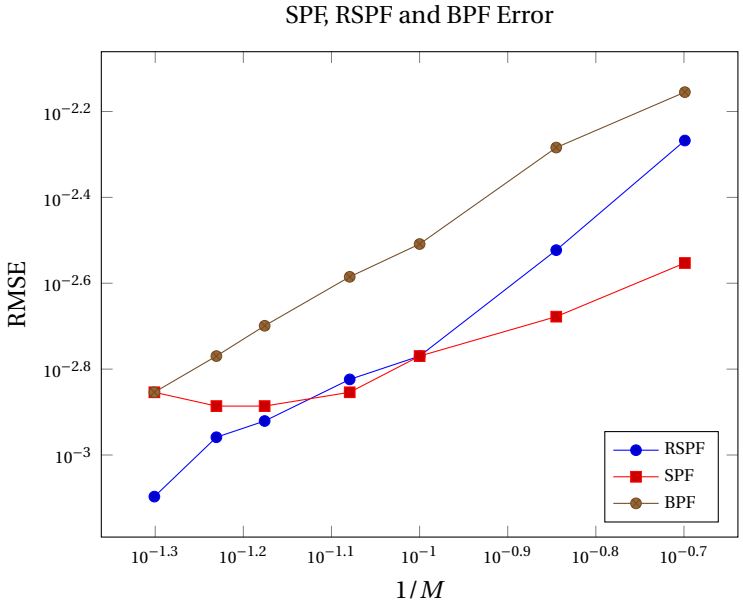


Figure 5.1: Plot of RMSE resulting from the estimates of Lorentz 63 using optimal filters. The settings are:– $dt = 0.01$, $P_{t_0} = \mathrm{I}_{3\times 3}$ and $T = 4{,}000{,}000$. The experiment is repeated for different ensemble member, *viz.*, $M = 5, 7, 10, 12, 15, 17, 20$.

The plot Figure 5.1 on shows the errors due to the filter estimates obtained using optimal particle filters and that of Bootstrap particle filter. The performance of the optimal filters, as shown in Figure 5.1, is better than that of BPF, when a small ensemble size is used. All the three filters show an improvement in performance with the increase in ensemble size. Whereas RSPF registers a better performance for all the ensemble sizes used, SPF elicits a better performance than the BPF only when the ensemble sizes are small. SPF and RSPF could be good candidate filters when ensemble sizes are small.

CHAPTER 6

COMBINED STATE AND PARAMETER ESTIMATION

6.1 Introduction

Most state-space models are characterized by, among other things, parameters—which can be constant or varying. By a parameter is comprehended and signified a measurable factor which defines a model and influences its operation. As the parameter changes, so does the model; expressed differently, a parameter is unique to a model it characterises. The choice of a certain model, therefore, is achieved by choosing the right parameters. It occurs more often than not—in hidden Markov models, for instance—that measurements are available but the underlying signal is not readily apparent. This forms an example where parameter estimation is paramount: measurements are used to learn the model parameters, which, in turn, are used to fit the model.

For the ensuing discussion, we modify the signal and measurement equations, eqs. (3.1.1a) and (3.1.1b) of Section 3.1, to include parameters, now and henceforth signified by a $d-$dimensional vector, θ, thus

$$\text{Signal:} \quad dx_t = f(x_t,\theta)dt + g(x_t)d\beta_t; \quad t_0 \le t, \tag{6.1.1a}$$

$$\text{Measurement:} \quad dy_t = h(x_t,\theta)dt + R^{1/2}(t)d\eta_t; \quad t_0 \le t, \tag{6.1.1b}$$

where the dimensions and nomenclature of other terms remain as indicated in Section 3.1.

Parameter estimation problem concerns finding the optimal parameter so that the signal best fits the data [Einicke (2012); Särkkä (2013)]. This is, classically, achieved by optimization procedure where a cost function is minimized [Lewis et al. (2006)]. The cost function mostly defines the discrepancy between the state and the measurements. Intuitively, parameter estimation can be seen as a procedure for seeking a parameter value that gives the least discrepancy between the state and the corresponding measurements (also known as the algebraic distance or the residual). The method of least squares has been extensively used to define an objective function. Given the increment in measurement, dy_t, of the state, x_t, at time t, the objective function, $\mathscr{J}(\theta)$, in the least squares sense, is given by

$$\mathscr{J}(\theta) = \int_{t_0}^{t} w_t \| dy_t - h(x_t,\theta)dt \|^2, \tag{6.1.2}$$

where w_t is the weighting function.

Most commonly used procedures in the framework of least-squares include: linear least-squares, orthogonal least-squares, gradient weighted least-squares, bias corrected renormalization. This thesis, however, attends not to the study of least-squares approaches. Suffice it to only direct the interested reader to the article Zhang (1997) for an elaborate explanation and application of least-squares methods in computer vision. Instead, we study parameter

estimation by means of filtering. But before that, we mention a few merits and demerits of least-squares and other methods defined by algebraic distances.

The use of algebraic distances in defining a cost function is computationally efficient and closed-form solutions are possible. The end result, however, is not satisfactory. This is due, in one part, to the fact that the objective function is mostly not invariant with respect to Euclidean transformations, for example, translation. This limits the coordinates systems to be used. In the other part, outliers may not contribute to the parameters the same way as inliers [Zhang (1997)]. Other more satisfactory parameter estimation methodologies are highly desired. We consider, in this thesis, the use of Bayesian inference techniques.

Estimation of parameters by means of a filter can be achieved in a number of ways; one of them being the use of the filter evidence, or its near approximation, and selection criteria for parameters which give a reasonable estimate of the evidence. The second method involves updating the parameters and the state at the same time. This is known as dual estimation, which further subdivides into joint estimation and a dual filter. Joint estimation entails subjoining the vector of parameters to the state vector to form an extended state-space. The filter is then implemented and run forward in time with the hope of filter convergence to the optimal state and parameter values. A dual filter, on the other hand, involves implementing a filter for the state and that of parameters simultaneously. The filter provides a self-correcting mechanism which may lead to convergence of state and parameter estimates.

The rest of this chapter is arranged as follows. We first consider the evidence-based parameter estimation approach, in which we define the evidence approximations for different filters in time-continuous framework. The second part is devoted to dual estimation where extended state-space formulation and the dual filter mechanism are elaborated. Application of the foregoing ideas to estimating constant parameters forms the closing part of this chapter.

6.2 Bayesian parameter inference

In Bayesian inference of parameters, the parameters are treated as a random variable. The parameter is assigned a prior, $\pi_{t_0}(\theta)$, based on some initial belief. Let t_n such that $t_{n+1} > t_n \; \forall n = 0, 1, 2, \ldots N$ be a partition of the interval $[t_0, T]$ and let $\delta t = t_{n+1} - t_n$. Bayes' rule gives the joint posterior of parameters and the states;

$$\begin{aligned}\pi_{[t_0,T]}(x, \theta \mid Y_T) &\approx \operatorname*{l.i.m.}_{\substack{\delta t \to 0 \\ N \to \infty}} \pi_{t_0:t_N}(x_{t_0:t_N}, \theta \mid y_{t_0:t_N}) \\ &= \operatorname*{l.i.m.}_{\substack{\delta t \to 0 \\ N \to \infty}} \frac{\pi_{t_0:t_N}(y_{t_0:t_N} \mid x_{t_0:t_N}, \theta)\pi_{t_0:t_N}(x_{t_0:t_N} \mid \theta)\pi_{t_0}(\theta)}{\pi_{t_0:t_N}(y_{t_0:t_N})},\end{aligned} \tag{6.2.1a}$$

where $Y_T = y_{[t_0,T]}$,

$$\pi_{t_0:t_N}(y_{t_0:t_N} \mid x_{t_0:t_N}, \theta) = \prod_{n=1}^{N} \pi_{t_n}(y_{t_n} \mid x_{t_n}, \theta), \tag{6.2.1b}$$

and

$$\pi_{t_0:t_N}(x_{t_0:t_N} \mid \theta) = \pi_{t_0}(x_{t_0} \mid \theta) \prod_{n=1}^{N} \pi_{t_n}(x_{t_n} \mid x_{t_{n-1}}, \theta). \tag{6.2.1c}$$

Now to arrive at the marginal posterior of parameters, we integrate out the states from the joint posterior of states and parameters, eq. (6.2.1a):

$$\pi_{t_0:t_N}(\theta \mid y_{t_0:t_N}) = \int \frac{\pi_{t_0:t_N}(y_{t_0:t_N} \mid x_{t_0:t_N}, \theta)\pi_{t_0:t_N}(x_{t_0:t_N} \mid \theta)\pi_{t_0}(\theta)}{\pi_{t_0:t_N}(y_{t_0:t_N})} dx_{t_0:t_N}. \tag{6.2.2}$$

It turns out that direct computation of the integral in eq. (6.2.2) is difficult, especially with the increase in measurements [Särkkä (2013)]. This challenge is circumvented through the use of recursive techniques which include the use of filters and smoothers, maximum a posteriori (MAP) estimates, and drawing samples from the posterior using Markov Chain Monte Carlo (MCMC) methods.

To use recursive methods, we begin with the following parameter posterior

$$\pi(\theta \mid Y_T) \approx \underset{\substack{\delta t \to 0 \\ N \to \infty}}{\text{l.i.m.}}\, \pi(\theta \mid y_{t_0:t_N}) \propto \underset{\substack{\delta t \to 0 \\ N \to \infty}}{\text{l.i.m.}}\, \pi_{t_0:t_N}(y_{t_0:t_N} \mid \theta)\pi_{t_0}(\theta). \tag{6.2.3a}$$

where

$$\begin{aligned} \pi_{t_0:t_N}(y_{t_0:t_N} \mid \theta) &= \prod_{n=1}^{N} \pi_{t_n}(y_{t_n} \mid y_{t_1:t_{n-1}}, \theta) \\ &= \prod_{n=1}^{N} \pi_{t_n}(y_{t_n} \mid x_{t_n}, \theta)\pi_{t_n}(x_{t_n} \mid y_{t_1:t_{n-1}}, \theta) dx_{t_n} \end{aligned} \tag{6.2.3b}$$

The state's predictive distribution, $\pi_{t_n}(x_{t_n} \mid y_{t_1:t_{n-1}}, \theta)$, can be computed recursively as follows [Särkkä (2013)]:

$$\pi_{t_n}(x_{t_n} \mid y_{t_1:t_{n-1}}, \theta) = \int \pi_{t_n}(x_{t_n} \mid x_{t_{n-1}}, \theta)\pi_{t_{n-1}}(x_{t_{n-1}} \mid y_{t_1:t_{n-1}}, \theta) dx_{t_{n-1}}. \tag{6.2.4a}$$

Instead of working with the posterior, $\pi(\theta \mid Y_T)$, it is quite convenient to use the negative log-posterior obtained by expressing the posterior as follows

$$\pi(\theta \mid Y_T) \approx \underset{\substack{\delta t \to 0 \\ N \to \infty}}{\text{l.i.m.}}\, \pi(\theta \mid y_{t_0:t_N}) \propto \underset{\substack{\delta t \to 0 \\ N \to \infty}}{\text{l.i.m.}}\, \exp\big(-\psi_T(\theta)\big), \tag{6.2.5}$$

where the *energy function*, $\psi_T(\theta)$, is given by

$$\psi_T(\theta) = -\log\big(\pi_{t_0:t_N}(y_{t_0:t_N} \mid \theta)\big) - \log\big(\pi_{t_0}(\theta)\big). \tag{6.2.6}$$

The maximum a posteriori estimate (MAP) can then be obtained by the mode of the posterior distribution, or, equivalently, the minimum of the energy function, the latter of which is easier to compute; that is,

$$\begin{aligned} \hat{\theta}_{\text{MAP}} &= \arg\max_{\theta} \pi(\theta \mid Y_T) \\ &= \arg\min_{\theta} \psi_T(\theta). \end{aligned} \tag{6.2.7}$$

One demerit of MAP estimate is that it yields a point estimate of the parameter posterior, and therefore ignores the dispersion of the estimate. Setting the prior, $\pi_{t_0}(\theta)$, to be a uniform density then eq. (6.2.7) yields a Maximum Likelihood estimate.

6.3 Metropolis-Hastings method

Metropolis-Hastings[6] [Robert and Casella (2004)] is a Markov-Chain Monte Carlo sampling algorithm with optimal convergence. It is premised on detailed balance and ergodicity. Given a

[6]Named after Nicholas Constantine Metropolis (1915-1999) and Wilfred Keith Hastings (1930-2016)

probability density, say, $\pi(\theta)$, from which it is difficult to sample—for instance if the said distribution is known to a normalisation constant—, and given another distribution $\rho(\theta)$, say, from which it is easy to sample, then detailed balance is the condition

$$\pi(\theta_k)\rho(\theta_k \mid \theta_{k+1}) = \pi(\theta_{k+1})\rho(\theta_{k+1} \mid \theta_k), \tag{6.3.1}$$

where $\rho(\theta_k \mid \theta_{k+1})$ is a transition distribution. The detailed balance condition is necessary for any random walk to asymptotically converge to a stationary distribution. By ergodicity is meant that there is a non-zero probability in moving from a state to any other state in a Markov-Chain.

The following algorithm summarises the Metropolis-Hastings procedure

Algorithm 6.3.1 Metropolis-Hastings

1: Draw $\tilde{\theta}$ from $\rho(\tilde{\theta} \mid \theta_k)$

2: Set $\theta_{k+1} \leftarrow \tilde{\theta}$ with probability $\alpha = \min\left(1, \dfrac{\pi(\tilde{\theta})\rho(\theta_k \mid \tilde{\theta})}{\pi(\theta_k)\rho(\tilde{\theta} \mid \theta_k)}\right)$

3: Otherwise set $\theta_{k+1} \leftarrow \theta_k$

6.4 Dual estimation

Dual estimation comprehends simultaneous estimation of state and parameters by means of an appropriate filter. The self-correcting mechanism of the filter is taken advantage of to converge to both the true state and the true parameters. Depending on the initial parameter, the filter sooner or later converges to the true parameter value. Dual estimation can be achieved in two ways: joint estimation and by a dual filter [Lü et al. (2011); Lint et al. (2008); Moradkhani et al. (2005*b*)]. In this section, and the rest of the thesis, we shall assume that the parameters are static; that is, time-invariant.

6.4.1 Joint estimation (augmented state-space)

In joint estimation, the state vector is augmented with the vector of parameters to form an extended state-space and then the filter is run forward in time for an update of both the state and the parameters. The parameters are induced with artificial dynamics, or are made to assume a random walk; that is, respectively,

$$dz_t = \zeta_t; \quad t_0 \le t, \tag{6.4.1}$$

where

$$dz_t = \begin{pmatrix} dx_t \\ d\theta_t \end{pmatrix} \text{ and } \zeta_t = \begin{pmatrix} f(x_t,\theta)dt + g(x_t)d\beta_t \\ 0 \end{pmatrix}, \tag{6.4.2a}$$

or where

$$\zeta_t = \begin{pmatrix} f(x_t,\theta)dt + g(x_t)d\beta_t \\ \sigma d\chi_t \end{pmatrix}, \tag{6.4.2b}$$

in which $\{\chi_t,\, t > t_0\}$ is a $d-$dimensional standard Brownian motion vector process and σ is a small constant. A filter is then implemented with the augmented state z_t in the place of x_t. The demerit of this method is that the extended state-space has an increased degree of freedom owing to many unknowns, of both the state and the parameters, which renders the filter unstable and intractable, especially in nonlinear models [Moradkhani et al. (2005*a*)].

Example 6.4.1: Scalar SDE

Consider the following linear Gaussian Itô state space model.

$$dx_t = (ax_t + b)dt + Q^{1/2}dv_t; \qquad t_0 \le t, \tag{6.4.3a}$$

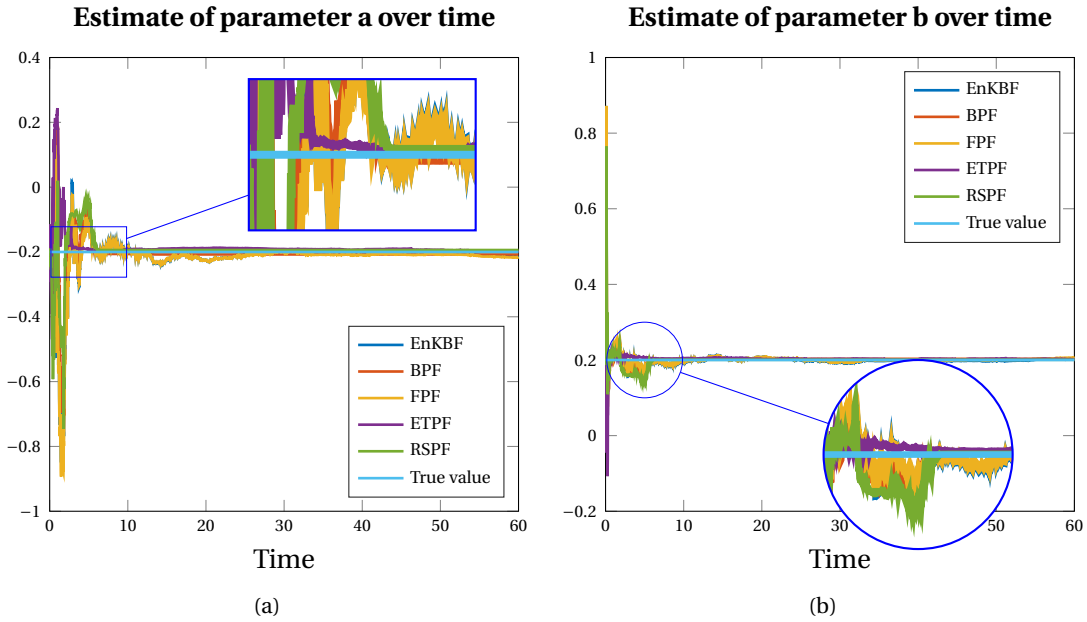
$$dy_t = cx_t dt + R^{1/2}dw_t; \qquad t_0 \le t, \tag{6.4.3b}$$

where $\{v_t\}$ and $\{w_t\}$ are standard Brownian motion processes with, respectively,

$$\mathbb{E}\{dv_t dv_t^T\} = dt \text{ and } \mathbb{E}\{dw_t dw_t^T\} = dt.$$

Let the state, x_t, at time t_0 be $x_{t_0} \sim \mathcal{N}(0, 0.001)$. Let, moreover, x_{t_0}, $\{v_t, t \ge t_0\}$ and $\{v_t, t \ge t_0\}$ be uncorrelated. We take $a = -0.2$, $b = 0.2$, $c = 1.01$, $Q = 0.001$, $R = 0.0001$ and proceed to simultaneously estimate the state, x_t, and the parameters, a and b using different filters.

The following panels show the results obtained using EnKBF, BPF, FPF (feedback particle filter with kernel-based gain approximation), ETPF—ensemble transform particle filter [Reich and Cotter (2015)] (introduced in Section 6.4.2.2) and RSPF.



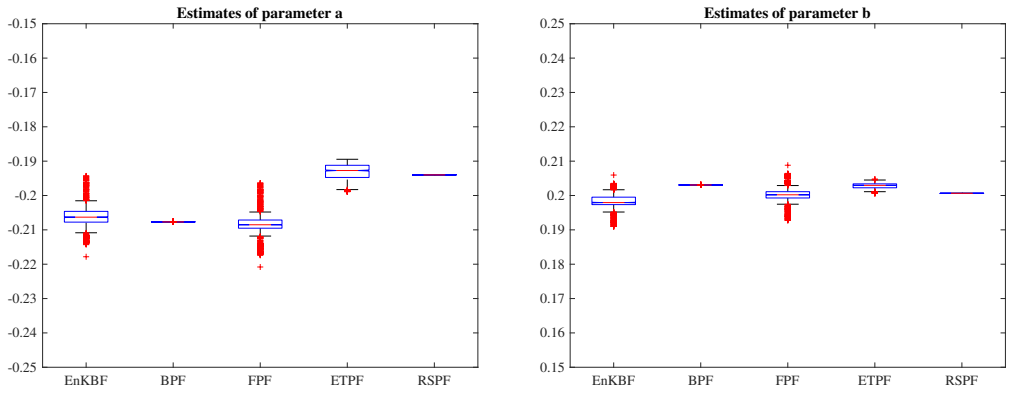
(a) (b)

Figure 6.1: Plots showing:- (a) estimates of parameter a and (b) estimates of parameter b over time using EnKBF, BPF, FPF, ETPF and RSPF. The true parameter values are, respectively, $a = -0.2$ and $b = 0.2$. The plots indicate that all the filters converge to the true parameter values. The time step used is $\delta t = 0.02$.

The EnKBF, BPF, FPF, ETPF and RSPF yield converging results to the true parameter values but with some margin of error. From Figures 6.1(a) and 6.1(b), it is evident that the ETPF yields a faster converging results compared to the other filters. Furthermore, the FPF takes more time before a steady value is obtained as compared to other filters.

We now plot the box-plots, the better to see the distribution of parameter estimates in the results shown in Figure 6.1, beyond time 30.

From Table 6.1, the BPF is faster compared to other methods. FPF is slower whilst ETPF is the slowest, seeing that the number of particles used is 100. The gain in the FPF is com-



(a) (b)

Figure 6.2: Box-plots (a) and (b) showing, respectively, the distribution of estimates of parameters a and b obtained using EnKBF, BPF, FPF, ETPF and RSPF. The gain in the FPF is computed using the kernel based gain approximation method, Algorithm 5.4.4. In both cases, the EnKBF and FPF register more dispersive results with more outliers than the BPF, ETPF and RSPF.

	EnKBF	**BPF**	**FPF**	**ETPF**	**RSPF**
M	1000	1000	1000	100	1000
Time in seconds	16.00	2.40	194.60	139.59	569.90

Table 6.1: A table showing run-time and number of particles for results shown in Figure 6.2 for different filters. This is an output of 2.3 GHz Intel Core i5 processor.

puted using the kernel based gain approximation method, Algorithm 5.4.4. The reason for this is that FPF involves a lengthy procedure for gain computation, Algorithm 5.4.4, at each iteration. On the other hand, The ETPF involves frequent solution of optimal transport problem. In this example, earth movers distance (EMD) algorithms were used in the ETPF. As a remedy, faster optimal transport algorithms, for example the Sinkhorn iteration, Algorithm 5.6.1, can be viable [Cuturi (2013)]. Although the RSPF takes much time, the result, as seen in Figure 6.2, is more accurate compared to that obtained from other methods.

6.4.2 Dual filter

Dual filtering of the state and parameters is attained by use of two filters, one for state update and another for updating parameters, both run simultaneously. The two filters interact symbiotically in that one provides the update of the state to be used by the other, while the other provides an update of the parameters to be used by the former. A very good example in literature is the dual extended Kalman filter [Wan and Nelson (1997)] used for estimating neural network models and the weights. In this case, the state is the model signal and the weights are parameters. Another example appears in Angwenyi et al. (2017) where a dual filter comprising of the ensemble transform particle filter (ETPF) and the feedback particle filter (FPF) is used for simultaneous estimation of the state of a wave equation and its speed. This latter work forms part of this thesis. We intend to enlarge upon the arguments set forth in Angwenyi et al. (2017) and include other formulations of a dual filter and make comparisons.

The model for the dual filter of our consideration comprises of a $d-$dimensional vector equation of artificial dynamics of parameters together with the state space model of Sec-

tion 3.1:

$$\text{Parameter:} \quad d\theta_t = 0, \qquad t_0 \le t, \tag{6.4.4a}$$

$$\text{Signal:} \quad dx_t = f(x_t,\theta_t)dt + g(x_t)d\beta_t, \quad t_0 \le t, \tag{6.4.4b}$$

$$\text{Measurement:} \quad dy_t = h(x_t)dt + R^{1/2}(t)d\eta_t, \quad t_0 \le t, \tag{6.4.4c}$$

where the nomenclature and dimensions remain as stipulated in Section 3.1.

In the following, we propose different formulations of a dual filter.

6.4.2.1 Sequential Monte Carlo Methods and EnKBF

In this *ansatz*, the state is propagated using sequential Monte Carlo methods and the parameters are updated using the EnKBF.

Propagation and update of the state. We recall, from Chapter 3, that sequential Monte Carlo methods—or, to use another name, particle filters—involve propagation of, say M, hypotheses of the state, which we denote as

$$X_t^j := \{x_t^{i,j}\}_{i=1}^M,$$

at a given time, t. For optimal solutions, the hypotheses or particles are drawn from the posterior distribution. It turns out, however, that the full posterior is—in most cases, especially for nonlinear and high-dimensional models—inaccessible. The posterior might be known up to a normalising constant, which constant may involve intractable integrals. To circumvent this problem, the particles are drawn from a distribution—known as the importance distribution—with highest semblance to the posterior. The estimate of the state, $\hat{x}_t$, at a time, t, is obtained by means of a weighted mean of the particles.

Commonly, a simple filter involves drawing hypotheses from the model signal; that is, each particle, $x_t^{i,j}$, for every parameter $\theta_t^j \in \{\theta_t^j\}_{j=1}^L$, is drawn and propagated as follows.

$$dx_t^{i,j} = f(x_t^{i,j},\theta_t^j)dt + g(x_t^{i,j})d\beta_t^{i,j}; \quad t_0 \le t. \tag{6.4.5a}$$

The normalized weights,

$$W_t := \{w_t^i\}_{i=1}^M,$$

on the other hand, are propagated thus.

$$dw_t^i = (h(x_t^{i,j}) - \hat{h}_t^j)R^{-1}(t)(dy_t - \hat{h}_t^j dt)w_t^i; \quad t_0 \le t, \tag{6.4.5b}$$

where

$$\hat{h}_t^j = \sum_{i=1}^M h(x_t^{i,j}). \tag{6.4.6}$$

Notice that the parameter vector, θ_t^j, is now time dependent and evolves according to the artificial dynamics, eq. (6.4.4a). It happens, however, that this leads to a circumstance where most of the particles bear almost negligible weights—hence undermining the performance of the filter. To correct this anomaly, resampling—in which the particles with more weights are replicated to make up for those of small weights—is carried out. This helps, albeit with additional covariances and high computational costs as trade-offs. More discussions on this are found, for example, in Reich and Cotter (2015); Carrassi et al. (2016).

Given the complications that come with resampling, we propose resampling to be carried out only when necessary; or, as it has been proposed in literature, when the effective sample size, $\text{ESS}_t = (\sum_{i=1}^M (w_t^i)^2)^{-1}$, surpasses a given threshold.

The estimate of the state, $\hat{x}_t^j$, at time t is obtained by:

$$\hat{x}_t^j = \sum_{i=1}^{M} x_t^{i,j} w_t^i. \tag{6.4.7}$$

Update of the parameters. This is accomplished by means of an EnKBF. The idea behind ensemble filters, in this formulation, as we now recall from Chapter 4, is to propagate, say L, hypotheses of the parameters, $\Theta_t := \{\theta_t^j\}_{j=1}^L$. Since the parameter dynamics, eq. (6.4.4a), are artificial, the prediction step is static; that is to say, forward propagation in time leaves the parameters as before. In the prediction step, each parameter hypothesis, θ_t^j, is updated with the innovation process from the propagation of the state; that is.

$$d\theta_t^j = D_t^L R^{-1}(t)(dy_t - 0.5(h(\hat{x}_t^j) + \hat{h}_t)dt); \quad t_0 \le t, \tag{6.4.8}$$

where $\hat{x}_t^j$ is the sequential Monte Carlo state estimate, eq. (6.4.7), obtained for each parameter particle, θ_t^j. Moreover,

$$\hat{\theta}_t = \frac{1}{L}\sum_{j=1}^{L} \theta_t^j; \qquad t_0 \le t, \tag{6.4.9a}$$

$$D_t^L = \frac{1}{L-1}\sum_{i=1}^{L} (\theta_t^j - \hat{\theta}_t)(h(\hat{x}_t^j) - \hat{h}_t)^{\mathrm{T}}; \qquad t_0 \le t, \tag{6.4.9b}$$

where

$$\hat{x}_t^j = \frac{1}{M}\sum_{i=1}^{M} x_t^{i,j} \text{ and } \hat{h}_t = \frac{1}{L}\sum_{j=1}^{L} h(\hat{x}_t^j). \tag{6.4.9c}$$

For the purpose of computation using a digital computer, the gain term in eq. (6.4.8) is computed as follows.

$$D_t^L R^{-1}(t) \approx \lim_{dt\to 0} \frac{D_t^L}{(\mathscr{D}_t + R(t)/\delta t)\delta t}, \tag{6.4.10}$$

where δt is the time step and

$$\mathscr{D}_t = \frac{1}{L-1}\sum_{i=1}^{L} (h(\hat{x}_t^j) - \hat{h}_t)(h(\hat{x}_t^j) - \hat{h}_t)^{\mathrm{T}}; \qquad t_0 \le t. \tag{6.4.11}$$

6.4.2.2 EnKBF and ETPF

In this formulation, the state is propagated and updated using the EnKBF whilst the parameters are updated using the ETPF, which we shall describe here briefly—as an adaptation of the discussion introduced in Section 5.5—and direct the reader to Reich and Cotter (2015) for an extensive explanation.

Propagation and update of the state. This is accomplished by means of EnKBF introduced in Section 4.2. We now specialize it to attain our intended end. Each particle of the state, $x_t^{i,j} \in \{x_t^{i,j}\}_{i,j}^{M,L}$, for every parameter $\theta_t^j \in \{\theta_t^j\}_{j=1}^L$, is propagated and updated, all at once, as follows.

$$dx_t^{i,j} = f(x_t^{i,j}, \theta_t^j, t)dt + g(x_t^{i,j}, t)d\beta_t^i + P_t^i R_t^{-1}\left(dy_t - \frac{1}{2}(h(x_t^{i,j}, t) + \hat{h}_t^j)dt\right) \tag{6.4.12}$$

Algorithm 6.4.1 BPF-EnKBF dual filter

Input: $x_{t_0}^{i,j}$, $\theta_{t_0}^{j}$, $w_{t_0}^{i} = 1/M \; \forall i \in \{1, 2, ..., M\}$ $j \in \{1, 2, ..., L\}$, P_{t_0} and $\delta y_{[t_0,t_T]}$.
Output: $\hat{x}_{[t_0,t_T]}$, $\hat{\theta}_{[t_0,t_T]}$.

for $n = 1$ **to** N, $\delta t > 0$ **do**
  for $j = 1$ **to** L **do**
    for $i = 1$ **to** M **do**
      Calculate $x_{t_n}^{i,j}$ using eq. (6.4.5a)
      Compute weights $w_{t_n}^{i}$ using eq. (6.4.5b)
      if $\mathrm{ESS}_{t_n} \leq \alpha$ **then**
        Resample the particles
        Set weights $w_{t_n}^{i} := 1/M$
      end if
    end for
    Update $\hat{x}_{t_n}^{j}$ using eq. (6.4.7)
    Update parameters $\theta_{t_n}^{j}$ using eq. (6.4.8)
  end for
  Compute $\hat{\theta}_{t_n} = \frac{1}{L}\sum_{j=1}^{L} \theta_{t_n}^{j}$
  Compute $\hat{x}_{t_n} = \frac{1}{L}\sum_{i=1}^{L} \hat{x}_{t_n}^{j}$
end for

where

$$P_t^i = \frac{1}{M-1}\sum_{i=1}^{M}(x_t^{i,j} - \hat{x}_t^j)(h(x_t^{i,j}) - \hat{h}_t^j)^{\mathrm{T}},$$

$$\hat{h}_t^j = \frac{1}{M}\sum_{i=1}^{M} h(x_t^{i,j}), \quad \text{and} \quad \hat{x}_t^j = \frac{1}{M}\sum_{i=1}^{M} x_t^{i,j}.$$

Update of the parameters. This is achieved via the ETPF. The ETPF is characterised by a control law, in the sense of optimal transportation, which moves the particles to the convenient position in the state space, thus improving upon the performance of the filter. The notable distinction between ETPF and particle filters with resampling is that the resampling step is replaced with a linear transformation. The transformation seeks to establish an optimal coupling between the prior ensemble and the posterior ensemble.

Most precisely, the linear transport problem, in which parameters play a part, is as follows.

$$T^* = \underset{T \in \mathscr{R}^{L\times L}}{\arg\min} \sum_{i,j=1}^{L} t_{ij}\|\theta_t^i - \theta_t^j\|^2 \tag{6.4.13}$$

under the constraints

$$\sum_{i=1}^{L} t_{ij} = \frac{1}{M}, \; \sum_{j=1}^{M} t_{ij} = w_{t_n}^{j}, \; \text{and } t_{ij} > 0, \tag{6.4.14}$$

where $\{w_{t_n}^{j}\}_{j=1}^{L}$ are weights at time $t_n = n\delta t$. The weights are propagated as follows

$$dw_t^j = \frac{1}{R} w_t^j (\hat{h}_t^j - \hat{h}_t)^{\mathrm{T}}(dy_t - \hat{h}_t dt), \tag{6.4.15}$$

where

$$\hat{h}_t^j = \frac{1}{M}\sum_{i=1}^{M} h(x_t^{i,j}), \;\; \hat{h}_t = \frac{1}{L}\sum_{i=1}^{L} \hat{h}_t^j.$$

To avoid negative weights, eq. (6.4.15) is approximated by

$$w^j_{t_{n+1}} \approx w^j_{t_n} \exp\left(-\frac{1}{2R}(\hat{h}^j_{t_n})^T \hat{h}^j_{t_n} \delta t - (\hat{h}^j_{t_n})^T \delta y_{t_n}\right) \tag{6.4.16}$$

and initialized with

$$w^j_{t_0} = \frac{1}{L}.$$

The parameters are updated as follows

$$\hat{\theta}^j_{t_n} = \sum_{i=1}^{M} \theta^j_{t_n} t^*_{ij}. \tag{6.4.17}$$

Finally, the state and parameter estimates are obtained empirically by

$$\hat{x}_{t_n} = \frac{1}{L}\sum_{j=1}^{L} \hat{x}^j_{t_n}, \tag{6.4.18a}$$

$$\hat{\theta}_{t_n} = \frac{1}{L}\sum_{j=1}^{L} \hat{\theta}^j_{t_n}, \tag{6.4.18b}$$

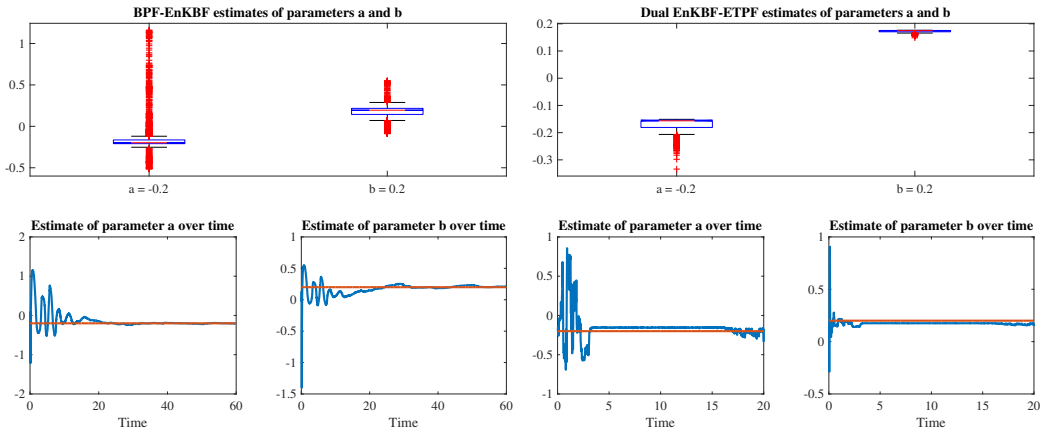
respectively. The following algorithm gives a summary of the dual filter.

Algorithm 6.4.2 EnKBF-ETPF dual filter

Input: $x^{i,j}_{t_0}, \theta^j_{t_0}, w^j_{t_0} = 1/L \;\; \forall i \in \{1, 2, ..., M\}\; j \in \{1, 2, ..., L\}$, P_{t_0} and $\delta y_{[t_0,t_T]}$.
Output: $\hat{x}_{[t_0,t_T]}, \hat{\theta}_{[t_0,t_T]}$.

for $n = 1$ **to** $N, \delta t > 0$ **do**
  for $j = 1$ **to** L **do**
    for $i = 1$ **to** M **do**
      Calculate $x^{i,j}_{t_n}$ using eq. (6.4.12)
    end for
    Compute weights $w^j_{t_n}$ using eq. (6.4.16)
    if $\text{ESS}_{t_n} \leq \alpha$ **then**
      Compute T^* by solving the optimal transports problem eq. (6.4.13)
      Update parameters: $\hat{\theta}^j_{t_n} = \sum_{i=1}^{M} \theta^j_{t_n} t^*_{ij}$
      Set weights $w^j_{t_n} := 1/L$
    end if
    Compute: $\hat{x}^j_{t_n} = \frac{1}{M}\sum_{i=1}^{M} x^{i,j}_{t_n}$
  end for
  Compute $\hat{\theta}_{t_n} = \frac{1}{L}\sum_{i=1}^{L} \hat{\theta}^j_{t_n}$
  Compute $\hat{x}_{t_n} = \frac{1}{L}\sum_{j=1}^{L} \hat{x}^j_{t_n}$
end for

We implement the BPF-EnKBF and EnKBF-ETPF dual filters on Example (6.4.1) with the aim of estimating parameters a and b. The results are shown in the following panels.



(a) (b)

Figure 6.3: Plots showing:- (a) top, the box-plot of parameters a and b obtained with the BPF-EnKBF dual filter; (a) bottom—the plots of a and b with time using BPF-EnKBF dual filter; (b) top, the box-plot of parameters a and b obtained by means of EnKBF-ETPF dual filter; (b) bottom—the plots of parameters a and b with time using EnKBF-ETPF dual filter. Both dual filters are run with 1000 state ensembles, 120 parameter ensembles and time-step $\delta t = 0.02$. The plots indicate that EnKBF-ETPF filter performs better than the BPF-EnKBF—based on the many outliers in the output of BPF-EnKBF and the faster convergence of the EnKBF-ETPF.

CHAPTER 7

APPLICATION TO HYPERBOLIC SPDES

7.1 Introduction

In the following we introduce the equations to which we later apply concepts derived in the preceding chapters.

7.1.1 Advection Equation

We take up an advection equation, excited with space-time white noise process, with some diffusion term added to it, and on a periodic spatial domain of length L, which equation we write as follows:

$$\frac{\partial u}{\partial t} = \frac{\partial (C(x)u)}{\partial x} + \mu \frac{\partial^2 u}{\partial x^2} + \sigma \dot{\beta}_{x,t}, \quad 0 \le t \le T_t \times 0 \le x \le L, \tag{7.1.1}$$

where $C(x)$ is a spatially varying velocity (of which constant velocity is a special case), σ is a constant, and $u(x,t)$ is the function to be obtained, which function describes the state of the signal. μ is a constant whilst $\dot{\beta}_{x,t}$ is space-time white noise process where the dot denotes the singularity of the noise process.

Equation (7.1.1) needs a remark owing to the roughness of the stochastic forcing term $\dot{\beta}_{x,t}$, which is a mixed distributional derivative of Brownian sheet. As is well known (see Stuart (2007) for details), the Brownian sheet is nowhere differentiable. We, however, use the method introduced in Allen et al. (1998); that is, we approximate the noise term as follows. Let the domain $0 \le t \le T_t \times 0 \le x \le L$ be tessellated into rectangles $[t_n, t_{n+1}] \times [x_i, x_{i+1}]$ of dimensions $\delta t \times \delta x$ each, for $n = 1, 2, 3, \ldots, T$ and $i = 1, 2, 3, \ldots, N$ so that $\delta t = T_t/T$ and $\delta x = L/N$. Then,

$$\dot{\beta}_{x,t} \approx \frac{1}{\delta x \delta t} \sum_{i=1}^{N} \sum_{n=1}^{T} \omega_{i,n} (\delta x \delta t)^{1/2} \chi_i(x) \chi_n(t), \tag{7.1.2}$$

where $\{\omega_{i,n}\}_{i=1}^{N}$ are independent and identically distributed random variables of mean 0 and unit variance. $\chi_i(x)$ and $\chi_n(t)$ are characteristic functions, and are given by

$$\chi_n(t) = \begin{cases} 1, & \text{if } t_n \le t \le t_{n+1}, \\ 0, & \text{otherwise,} \end{cases}$$

$$\chi_i(x) = \begin{cases} 1, & \text{if } x_i \le x \le x_{i+1}, \\ 0, & \text{otherwise.} \end{cases}$$

By three-point upwind scheme in space [Courant et al. (1952)], we discretise eq. (7.1.1) and arrive at the following

$$\begin{aligned}\frac{du_i}{dt} \approx & \frac{3C_i u_i - 4C_{i-1}u_{i-1} + C_{i-2}u_{i-2}}{2\delta x} \\ & + \mu \frac{3u_{i+2} - 16u_{i+1} + 26u_i - 16u_{i-1} + 3u_{i-2}}{4\delta x^2} + \sigma \frac{1}{\sqrt{\delta x}}\dot{\omega}_i,\end{aligned} \tag{7.1.3}$$

where δx is the spatial step size and $\{\omega_{i,t}, t > t_0\}$ is standard Brownian motion process. The ith grid point is represented by $x_i = i\delta x$. With this notation, $u_{i,n}$ is understood to mean the value of u at the ith grid point at time t_n.

Time discretisation, by means of Euler-Maruyama scheme, leads to

$$\begin{aligned}u_{i,t_{n+1}} = u_{i,t_n} + & \frac{3C_i u_{i,t_n} - 4C_{i-1}u_{i-1,t_n} + C_{i-2}u_{i-2,t_n}}{2\delta x}\delta t \\ & + \mu \frac{3u_{i+2,t_n} - 16u_{i+1,t_n} + 26u_{i,t_n} - 16u_{i-1,t_n} + 3u_{i-2,t_n}}{4\delta x^2}\delta t + \sigma \frac{\delta t^{1/2}}{\sqrt{\delta x}}\omega_{i,t_n},\end{aligned} \tag{7.1.4}$$

where ω_{i,t_n} is a random variable of mean 0 and variance 1. The time increment, $\delta t > 0$, is such that the limit of δu_i as $\delta t \to 0$ is du_i. Furthermore, $n = 1, 2, 3, \ldots, T$. We use the following initial value.

$$u(x, t_0) = \sin(x). \tag{7.1.5}$$

Moreover, $C(x) = e^{\lambda(x)}$ where

$$\lambda(x) = \sin(2\pi x).$$

Considering every grid point in eq. (7.1.4) leads to a vector representation of the signal u. To do so requires the following shorthand definition of operations,

$$(D_1 u)_i := \frac{3u_i - 4u_{i-1} + u_{i-2}}{2\delta x}, \qquad \forall i = 1, 2, 3, \ldots, N,$$

and

$$(D_1^{\mathrm{T}} u)_i := \frac{u_{i+2} - 4u_{i+1} + 3u_i}{2\delta x}, \qquad \forall i = 1, 2, 3, \ldots, N,$$

so that

$$\begin{aligned}(D_1 D_1^{\mathrm{T}} u)_i := & \frac{3u_{i+2} - 12u_{i+1} + 9u_i}{4\delta x^2} + \frac{-4u_{i+1} + 16u_i - 12u_{i-1}}{4\delta x^2} \\ & + \frac{u_i - 4u_{i-1} + 3u_{i-2}}{4\delta x^2} \\ = & \frac{3u_{i+2} - 16u_{i+1} + 26u_i - 16u_{i-1} + 3u_{i-2}}{4\delta x^2}, \qquad \forall i = 1, 2, 3, \ldots, N.\end{aligned} \tag{7.1.6}$$

We finally have

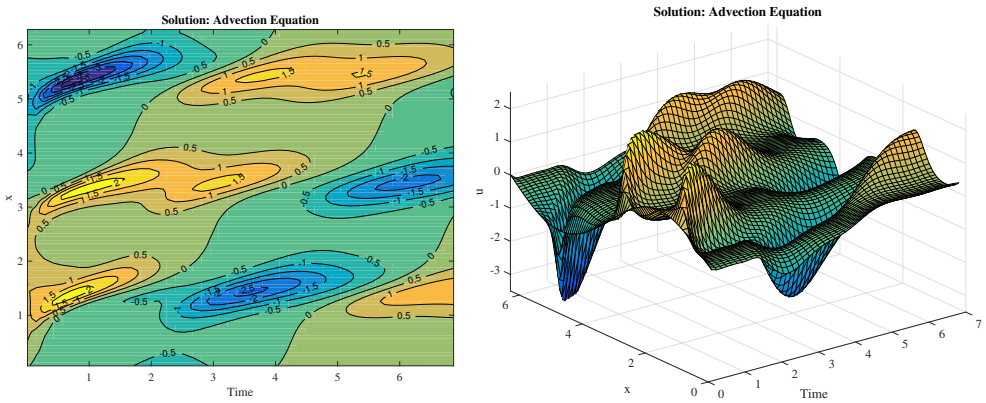
$$u_{t_{n+1}} = u_{t_n} + F(t_n)u_{t_n}\delta t + G(t_n)\omega_{t_n}, \tag{7.1.7}$$

where u_{t_n} is an $N-$dimensional column vector at time t_n comprising of elements u_{i,t_n}, $i = 1, 2, 3, \ldots, N$ and

$$F(t_n) = \left[D_1 C_{\mathrm{diag}} - \mu D_1 D_1^{\mathrm{T}}\right], \; G(t_n) = \left[\sigma \frac{\delta t^{1/2}}{\sqrt{\delta x}} I_{N\times N}\right],$$

in which C_{diag} is a diagonal matrix made of the elements of C, and $I_{N\times N}$ is the Nth order identity matrix.

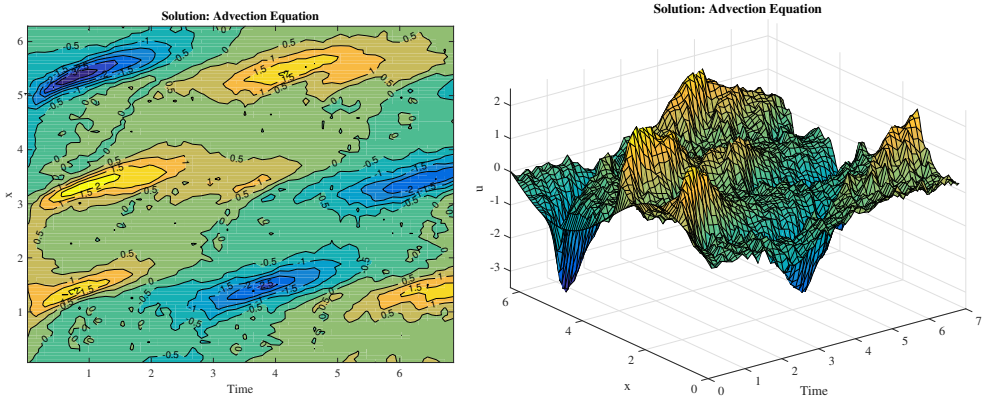
The surface and contour plots for the advection equation are shown below when $\sigma = 0$; that is, without noise. This is to show the underlying dynamics against which we shall compare the numerical solution to the stochastic advection equation. Of note in Figure 7.1 is the



(a) (b)

Figure 7.1: Contour and surface plots for the solution to the deterministic advection equation. This is under the following setting: $L = 2\pi$, $N = 100$, $\delta x = L/N$, $\delta t = 0.007$, $T = 1000$, $\sigma = 0$, $\mu = 0.01$, $C(x) = e^{\lambda(x)}$ where $\lambda(x) = \sin(2\pi x)$, and $u_{t_0} = \sin(x)$.

smoothness of the solution of advection equation—both in contours and in the surface plot. This is contrasted with the solution of the stochastic advection equation where noise is added. The surface and contour plots for the stochastic advection equation are shown below; that is, when $\sigma = 0.1$. The ruggedness evident in Figure 7.2 as opposed to the smoothness manifest in



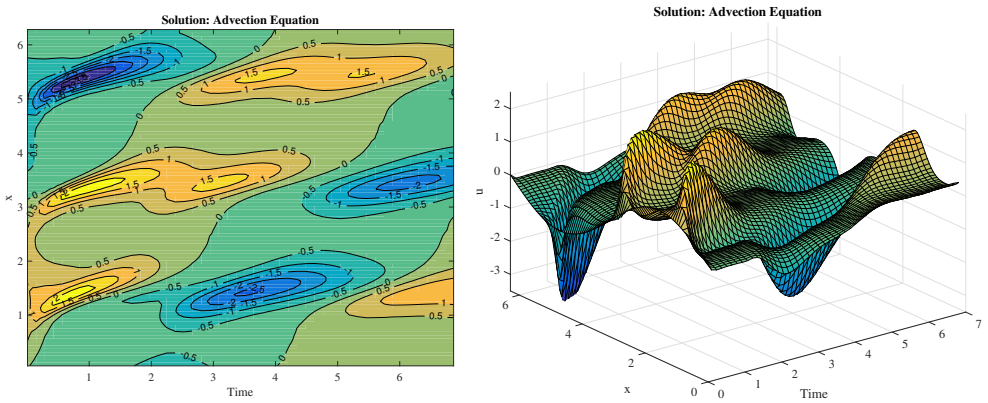
(a) (b)

Figure 7.2: Contour and surface plots for a single realisation of the stochastic advection equation under the following setting: $L = 2\pi$, $N = 100$, $\delta x = L/N$, $\delta t = 0.007$, $T = 1000$, $\sigma = 0.1$, $\mu = 0.01$, $C(x) = e^{\lambda(x)}$ where $\lambda(x) = \sin(2\pi x)$, and $u_{t_0} = \sin(x)$.

Figure 7.1 is consequent upon the addition of noise to the underlying dynamics.

The mean of, say, M, realizations of the stochastic advection equation should match the solution of the deterministic advection equation [Tuckwell (2015)]. This is because the noise term is Gaussian, that is, of mean zero. The surface and the contour plots for the the average of $M = 1000$ realizations of the stochastic advection equation are shown below. As was anticipated, the realisation of the mean of a fairly large ensemble of solution paths is smooth, as seen in Figure 7.3—and is as if noise was not added to the dynamics.

In the next subsection, we introduce the wave equation.



(a) (b)

Figure 7.3: Plots for the average of the realizations of the advection equation. This is under the following setting: $L = 2\pi$, $N = 100$, $\delta x = L/N$, $\delta t = 0.007$, $T = 1000$, $\sigma = 0.1$, $\mu = 0.01$, $C(x) = e^{\lambda(x)}$ where $\lambda(x) = \sin(2\pi x)$ and $u_{t_0} = \sin(x)$.

7.1.2 Wave Equation

The wave equation—for our consideration—is given by:

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial(C(x)\partial u/\partial x)}{\partial x} + \mu\frac{\partial^3 u}{\partial x^2 \partial t} + \sigma\dot{\beta}_{x,t}, \quad 0 \le t \le T_t \times 0 \le x \le L, \tag{7.1.8}$$

where $C(x) = e^{\lambda(x)}$ is the wave velocity—and is for a wave travelling in a heterogeneous medium—, and $u(x,t)$ is the function to be obtained, which function, as in the previous subsection, describes the state of the signal. Moreover, σ is a constant and $\dot{\beta}_{x,t}$, as before, is the space-time white noise.

We employ mixed difference schemes to discretise eq. (7.1.8) in space, so that we have:

$$\frac{du_i}{dt} \approx p_i, \tag{7.1.9a}$$

$$\frac{dp_i}{dt} \approx \frac{C_{i+1}w_{i+1} - C_i w_i}{\delta x} + \mu\frac{p_{i+1} - 2p_i + p_{i-1}}{\delta x^2} + \frac{1}{\sqrt{\delta x}}\dot{\omega}_i, \tag{7.1.9b}$$

where $w_i := \dfrac{u_i - u_{i-1}}{\delta x}$, and δx is the spatial step size. For time integration we use Verlet's method, because of its geometric properties; namely, volume preservation, symplecticity, conservation of first integrals and reversibility [Hairer et al. (2003); Reich (1999)]—which method, applied to the deterministic part of eqs. (7.1.9a) and (7.1.9b), yields

$$u_{i,t_{n+1}} = u_{i,t_n} + p_{i,t_{n+1/2}}\delta t, \tag{7.1.10a}$$

$$p_{i,t_{n+1/2}} = p_{i,t_n} + \frac{C_{i+1}w_{i+1,t_n} - C_i w_{i,t_n}}{\delta x}\frac{\delta t}{2} + \mu\frac{p_{i+1,t_n} - 2p_{i,t_n} + p_{i-1,t_n}}{\delta x^2}\frac{\delta t}{2}, \tag{7.1.10b}$$

$$\begin{aligned} p_{i,t_{n+1}} = p_{i,t_{n+1/2}} &+ \frac{C_{i+1}w_{i+1,t_{n+1}} - C_i w_{i,t_{n+1}}}{\delta x}\frac{\delta t}{2} \\ &+ \mu\frac{p_{i+1,t_{n+1}} - 2p_{i,t_{n+1}} + p_{i-1,t_{n+1}}}{\delta x^2}\frac{\delta t}{2} + \sigma\frac{\delta t^{1/2}}{\sqrt{\delta x}}\omega_{i,t_n}, \end{aligned} \tag{7.1.10c}$$

where δt is the time step and $w_{i,t_n} := \dfrac{u_{i,t_n} - u_{i-1,t_n}}{\delta x}$. Substituting eq. (7.1.10b) into eqs. (7.1.10a)

and (7.1.10c), we get

$$\begin{aligned} u_{i,t_{n+1}} &= u_{i,t_n} + p_{i,t_n} + \frac{C_{i+1}w_{i+1,t_n} - C_i w_{i,t_n}}{\delta x}\frac{\delta t}{2} \\ &\quad + \mu\frac{p_{i+1,t_n} - 2p_{i,t_n} + p_{i-1,t_n}}{\delta x^2}\frac{\delta t}{2}, \end{aligned} \tag{7.1.11a}$$

$$\begin{aligned} p_{i,t_{n+1}} &= p_{i,t_n} + \frac{C_{i+1}w_{i+1,t_n} - C_i w_{i,t_n}}{\delta x}\frac{\delta t}{2} \\ &\quad + \mu\frac{p_{i+1,t_n} - 2p_{i,t_n} + p_{i-1,t_n}}{\delta x^2}\frac{\delta t}{2} + \frac{C_{i+1}w_{i+1,t_{n+1}} - C_i w_{i,t_{n+1}}}{\delta x}\frac{\delta t}{2} \\ &\quad + \mu\frac{p_{i+1,t_{n+1}} - 2p_{i,t_{n+1}} + p_{i-1,t_{n+1}}}{\delta x^2}\frac{\delta t}{2} + \sigma\frac{\delta t^{1/2}}{\sqrt{\delta x}}\omega_{i,t_n}. \end{aligned} \tag{7.1.11b}$$

We use the following initial values:

$$u(x,0) = \exp\left(-4(x-0.5L).^2\right), \tag{7.1.12a}$$

$$p(x,0) = 0, \tag{7.1.12b}$$

where L is the length of the domain. Now $C(x) = e^{\lambda(x)}$ where

$$\lambda(x) = \sin(x).$$

Considering every grid point leads to a vector representation of the signal u. To do so requires the following shorthand definition of operations,

$$(D_2 u)_i := \frac{u_i - u_{i-1}}{\delta x}, \qquad \forall i = 1,\, 2,\, 3,\, ...,\, N.$$

Equations (7.1.11a) and (7.1.11b) then become

$$\underline{u}_{t_{n+1}} = \underline{u}_{t_n} + F t_n \underline{u}_{t_n} + G(t_n)\underline{\omega}_{t_n}, \tag{7.1.13}$$

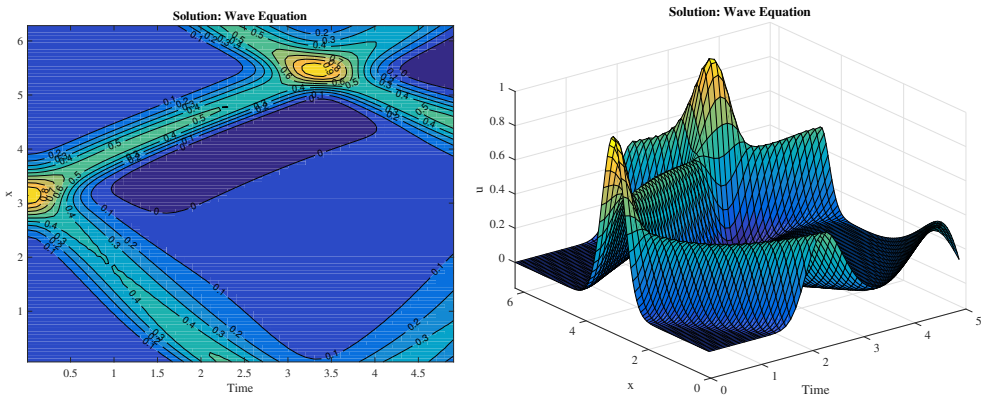
where

$$\begin{aligned} F(t_n) = -I_{2N\times 2N} &+ \begin{bmatrix} I_{N\times N} - \frac{\delta t^2}{2}D_2(C_{\text{diag}}(x)D_2^{\mathrm{T}} & \delta t I_{N\times N} - \frac{\delta t^2}{2}\mu D_2 D_2^{\mathrm{T}} \\ -\frac{\delta t}{2}D_2(C_{\text{diag}}(x)D_2^{\mathrm{T}} & I_{N\times N} - \frac{\delta t}{2}\mu D_2 D_2^{\mathrm{T}} \end{bmatrix} \\ &\times \begin{bmatrix} I_{N\times N} & 0_{N\times N} \\ \frac{\delta t}{2}D_2(C_{\text{diag}}(x)D_2^{\mathrm{T}}) & I_{N\times N} - \frac{\delta t}{2}\mu D_2 D_2^{\mathrm{T}} \end{bmatrix}^{-1}, \end{aligned}$$

$$\underline{u} = \begin{pmatrix} p \\ u \end{pmatrix},\ G(t_n) = \begin{bmatrix} 0_{N\times N} & 0_{N\times N} \\ 0_{N\times N} & \sigma\frac{\delta t^{1/2}}{\sqrt{\delta x}} I_{N\times N} \end{bmatrix},\ \text{and}\ \underline{\omega} = \begin{pmatrix} \omega \\ \omega \end{pmatrix},$$

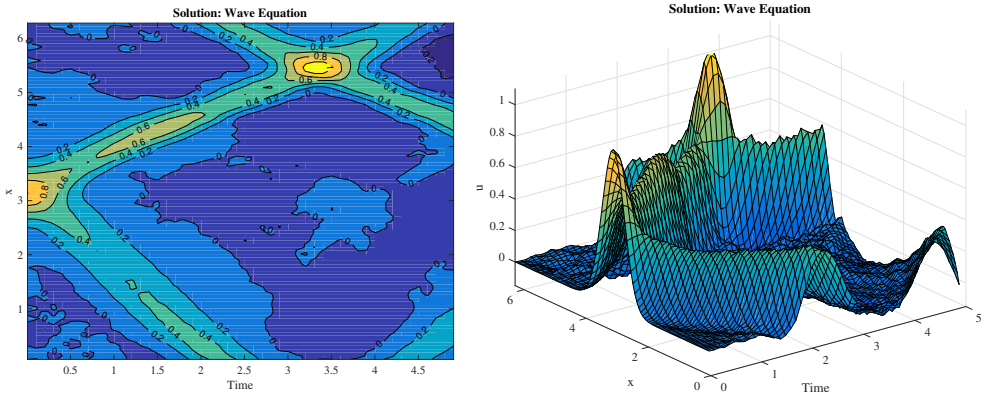
where $I_{2N\times 2N}$ is the identity matrix of order $2N$ while $0_{N\times N}$ is an Nth order null matrix. 0_N is an Nth dimensional null vector. $F(t_n) \in \mathbb{R}^{2N\times 2N}$, $G(t_n) \in \mathbb{R}^{2N\times 2N}$ and $\underline{\omega} \in \mathbb{R}^N$.

The contour and surface plots for the deterministic wave equation are shown below. The solution for the deterministic wave equation, Figure 7.4, is smooth. We next add a Gaussian noise and obtain a realisation of stochastic wave equation. The contour and surface plots for a single realisation of stochastic wave equation are shown below. Although the realisation in Figure 7.5 is rugged, owing to the addition of randomness, the form of the underlying dynamics—shown in Figure 7.4—is preserved. To see this more evidently, we now obtain an ensemble of realisations of stochastic wave equation and obtain an average. The surface and the contour



(a) (b)

Figure 7.4: Plots for the solution to the deterministic wave equation. The settings are: $L = 2\pi$, $N = 100$, $\delta x = L/N$, $\delta t = 0.005$, and $T = 1000$, $\mu = 0.01$, $\sigma = 0$, $C(x) = e^{\lambda(x)}$ where $\lambda(x) = \sin(x)$ and $u_{t_0} = \exp\left(-4(x - 0.5L).^2\right)$.



(a) (b)

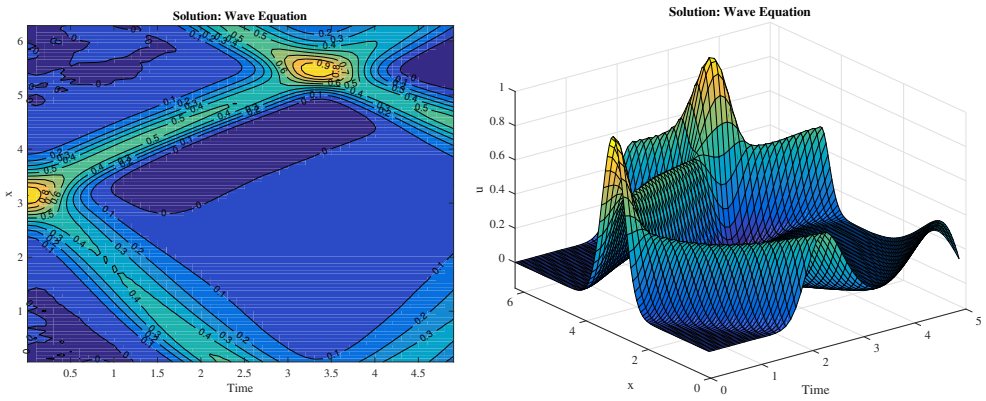
Figure 7.5: Plots for a single realisation of the stochastic wave equation. The settings are: $L = 2\pi$, $N = 100$, $\delta x = L/N$, $\delta t = 0.005$, and $T = 1000$, $\mu = 0.01$, $\sigma = 0.2$, $C(x) = e^{\lambda(x)}$ where $\lambda(x) = \sin(x)$ and $u_{t_0} = \exp\left(-4(x - 0.5L).^2\right)$.

plots for the average of 1000 realizations of the stochastic wave equation are shown in Figure 7.6. The smoothness in the average of the realisation, as shown in Figure 7.6, compared with the dynamics without noise—Figure 7.4(b)—shows that averaging "eliminates" the noise, it being Gaussian.

For the purpose of estimating the speed of the wave, which varies spatially, we give a little prelude to estimation of spatially varying parameters in the next section.

7.2 Estimating spatially varying parameters

Varying parameters exist naturally, an example being in the speed of a wave moving in a heterogeneous media. Varying parameters present a challenge owing to their varying nature as opposed to static (in time) parameters. There are three broad categories of varying parameters: spatially varying parameters, parameters that vary with time and parameters that vary both with time and space. Estimation of time-varying parameters, albeit for deterministic models, and application to estimation of parameters in HIV/AIDS model, appears in Chen and Wu (2008)—in which least-squares methods have been used. Although consistent estimates are



(a) (b)

Figure 7.6: Plots for the solution to the wave equation. The settings are: $L = 2\pi$, $N = 100$, $\delta x = L/N$, $\delta t = 0.005$, and $T = 1000$, $\mu = 0.01$, $\sigma = 0.2$, $C(x) = e^{\lambda(x)}$ where $\lambda(x) = \sin(x)$ and $u_{t_0} = \exp\big(-4(x - 0.5L).^2\big)$.

realized, extension to non-linear models remains to be realised. In this chapter, we study spatially varying parameters using filtering algorithms. We also provide applications in order to evaluate the performance of the algorithms introduced here.

The strategy employed in this study comprises of three steps:

Step 1: Express the parameter $\theta(x)$ as a Fourier series with a given number of modes, say, N_m and collect all the coefficients in a vector λ.

Step 2: By means of an appropriate filtering algorithm, estimate the vector of hyper-parameters, λ.

Step 3: Substitute the estimated constant coefficients back in the Fourier series to obtain an estimate of the parameter $\theta(x)$.

To illustrate this method, we employ it to estimate the velocity of a wave travelling in a heterogeneous media. Let the parameter, $C(x)$, $x \in \mathbb{R}^N$, where N is the number of dimensions, be given by

$$C(x) = \exp\big(f(x)\big), \tag{7.2.1}$$

where

$$f(x) = a_0 + \sum_{k=1}^{N_m} \frac{a_k}{k^2} \sin(kx) + \sum_{k=1}^{N_m} \frac{b_k}{k^2} \cos(kx), \;\; k = 1,\, 2,\, 3,\, ...,\, N_m, \tag{7.2.2}$$

where the coefficients a_0, a_k and b_k are drawn randomly from a normal distribution of mean 0 and variance 1. This way, the parameter $C(x)$ will be positive for all values of x. The aim of this section is to obtain an estimate $C(x)_{est}$ of the wave velocity $C(x)$ by means of KBF and EnKBF. To this end, we use a function

$$g(x) = A_0 + \sum_{k=1}^{\aleph_m} \frac{A_k}{k^2} \sin(kx) + \sum_{k=1}^{\aleph_m} \frac{B_k}{k^2} \cos(kx), \;\; k = 1,\, 2,\, 3,\, ...,\, \aleph_m, \tag{7.2.3}$$

where $\aleph_m \leq N_m$ so that

$$C(x)_{est} = \exp\big(g(x)\big). \tag{7.2.4}$$

Let $\lambda \in \mathbb{R}^{2\aleph_m+1}$ be a vector whose elements are the coefficients of the function $g(x)$. That is,

$$\lambda = \begin{pmatrix} A_0 \\ A_1 \\ B_1 \\ A_2 \\ B_2 \\ \vdots \\ A_{\aleph_m} \\ B_{\aleph_m} \end{pmatrix}. \tag{7.2.5}$$

Estimation of the spatially varying parameter, $C(x)$, is equivalent to estimating the parameters that form the vector λ. We consider two methods: using the likelihood with Metropolis-Hastings method and using a dual filter. Let us now consider each method in turn.

7.3 Using the Likelihood with Metropolis-Hastings method

We now adapt Algorithm 6.3.1 to estimating the vector of parameters, λ. We let each parameter λ_i have artificial dynamics with the transition density

$$\zeta_{k+1}(\lambda_{i,k+1} \mid \lambda_{i,k}) = \mathcal{N}\left(\lambda_{i,k}\cos(\phi), \frac{\omega}{\aleph}\sin(\phi)\right), \tag{7.3.1}$$

where $\aleph$ is the mode number and the constants ϕ and ω are to be chosen. The density π is defined by the filter likelihood,

$$\pi(\delta y_{[t_0,T]} \mid \tilde{u}_{[t_0,T]}, \lambda_k) = \underset{\substack{\delta t\to 0 \\ N\to\infty}}{\text{l.i.m.}}\, \pi_{t_0:t_N}(\delta y_{t_0:t_N} \mid \tilde{u}_{t_0:t_N}, \lambda_k),$$

where $\tilde{u}_{t_n}$ is the filter prediction of the state at time t_n. Notice that the parameters, λ, are implicitly contained in the dynamics.

Example 7.3.1: Advection Equation

We take the advection equation, of Section 7.1.1, and the given initial conditions. Furthermore, let there be time-continuous measurements of the state, u, given by,

$$dy_t = H(t)u(x,t)dt + Q(t)d\eta_{x,t}, \tag{7.3.2}$$

where $\{\eta_{t,x}\}$ is standard space-time Brownian motion process. The initial value of u_t, $\{\beta_t\}$ and $\{\eta_{t,x}\}$ are uncorrelated. The aim is to estimate the spatially varying velocity, $C(x)$, by means of filter likelihood and Metropolis-Hastings algorithm.

We follow the discretisation described in Section 7.1.1 for the advection equation.

The measurements' equation, eq. (7.3.2), is discretised in time using Euler-Maruyama scheme to yield

$$\delta y_{t_n} = H(t_n)u(x,t_n)\delta t + R^{1/2}(t_n)\delta\eta_{t_n}, \tag{7.3.3}$$

upon substituting $R = QQ^{\mathrm{T}}/\delta x$ and where $\mathbb{E}[\delta\eta_{t_n}\delta\eta_{t_n}^{\mathrm{T}}] = \mathbf{I}_{r\times r}\delta t$.

Remark 7.3.1. *The observation likelihood pdf for the KBF is Gaussian since the initial condition and the observation errors are Gaussian. So is the posterior pdf. With observation increments expressed as in eq. (7.3.3), the observation increment likelihood pdf is*

$$\pi(\delta y_{t_0:t_N} | \tilde{u}_{t_N}, \lambda_k) \propto \prod_{n=0}^{N} \exp\left(-\frac{1}{2} \parallel \delta y_{t_n} - H(t_n)\tilde{u}_{t_n}\delta t \parallel^2_{\delta t R(t_n)}\right). \tag{7.3.4}$$

Similarly, the filter forecast pdf is

$$\pi(u_{t_N}|\delta y_{t_{N-1}:t_0}) \propto \prod_{n=0}^{N} \exp\left(-\frac{1}{2} \parallel u_{t_n} - \tilde{u}_{t_n} \parallel^2_{P_{t_n}}\right). \tag{7.3.5}$$

The next step is to implement a KBF and EnKBF and to obtain the likelihood at each time steps. This is followed by implementing a Metropolis-Hastings algorithm.

Algorithm 7.3.1 KBF likelihood with MH

Input: $\delta t, \aleph_m, N, u_{t_0}, \pi_{t_n}$ and λ_k.
Output: $\{\lambda_k\}_{k=1}^{T}$.

1: **for** $k = 1$ **to** N **do**
2: Draw $\tilde{\lambda} \sim \mathcal{N}\left(\lambda_{i,k}\cos(\phi), \frac{\omega}{\aleph_m}\sin(\phi)\right) \quad \forall i = 1, 2, ..., 2\aleph_m + 1.$
3: Compute $C_k(x) = \exp\big(g(x, \lambda_k)\big)$.
4: **for** $n = 1$ **to** $T, \delta t > 0$ **do**
5: Run a single step KBF prediction mean $\tilde{u}_{t_{n+1}} = u_{t_n} + F(t_n, \lambda_k)u_{t_n}\delta t$
6: Run a single step KBF prediction covariance $\tilde{P}_{t_{n+1}} = P_{t_n} + F(t_n, \lambda_k)P_{t_n}\delta t + P_{t_n}F^{\mathrm{T}}(t_n, \lambda_k)\delta t + G(t_n)G^{\mathrm{T}}(t_n)\delta t$
7: Run a single step KBF analysis mean $u_{t_{n+1}} = \tilde{u}_{t_{n+1}} + P_{t_n}H^{\mathrm{T}}(t_n)R^{-1}(t_n)(dy_{t_n} - H(t_n, \lambda_k)\tilde{u}_{t_n}\delta t)$
8: Run a single step KBF analysis covariance $P_{t_{n+1}} = \tilde{P}_{t_{n+1}} + G(t_n)G^{\mathrm{T}}(t_n)\delta t - \tilde{P}_{t_n}H^{\mathrm{T}}(t_n)R^{-1}(t_n)H(t_n)P_{t_n}\delta t$
9: **end for**
10: Metropolis Hastings
11: Compute $\alpha_{\text{ratio}} = \dfrac{\pi(\delta y_{t_N:t_0}|\tilde{u}_{t_N}, \lambda_k)}{\pi(\delta y_{t_N:t_0}|\tilde{u}_{t_N}, \lambda_{k-1})} \dfrac{\rho_\lambda(\tilde{\lambda})}{\rho_\lambda(\lambda_k)}$
12: Compute $\alpha = \min(1, \alpha_{\text{ratio}})$
13: **if** $\alpha > U(0,1)$ **then**
14: $\tilde{\lambda} = \lambda_k$
15: **else**
16: $\lambda_k = \tilde{\lambda}$
17: **end if**
18: **end for**

The same is repeated but with EnKBF in the place of KBF. The following pseudo code shows the basic steps.

Results for the first 2 parameters are shown in Figure 7.7. The results in Figure 7.7 show that the EnKBF performs like the KBF filter. This agrees with the theory (see the findings of Section 4.3.1), that the EnKBF yields optimal results in the limit $M \to \infty$. It is also noteworthy that Metropolis-Hastings algorithm converges to the true parameter estimate. This can be seen in Figure 7.7(a), for example, where the filter estimate converges after about 100 parameter draws.

We now look at the errors in the parameter estimates, the better to see the performance of the filters for the 21 hyper-parameters.

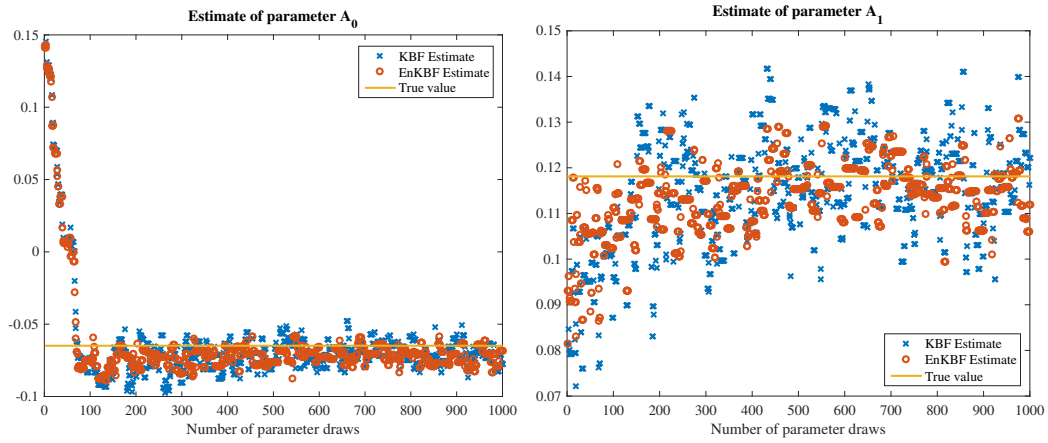
In Figure 7.8(b) panels are plotted the box-plots showing the dispersion of parameter estimates resulting from the use of EnKBF and the root mean square errors for parameter estimates for both the EnKBF and KBF. The RMSE values are computed as follows.

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(\lambda_{i,k} - \lambda_i^{\text{true}})^2}, \tag{7.3.6}$$

Algorithm 7.3.2 EnKBF likelihood with MH

Input: $\delta t, \aleph_m, M, N, \{u^i_{t_0}\}^M_{i=1}, \pi_{t_n}$ and λ_0.
Output: $\{\lambda_k\}^T_{k=1}$.

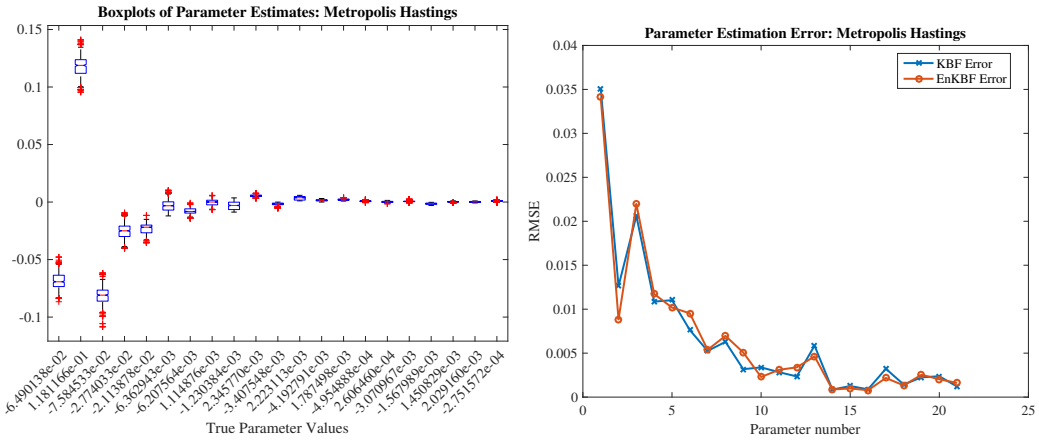
for $k = 1$ **to** N **do**
  Draw $\tilde{\lambda} \sim \mathcal{N}\left(\lambda_{i,k}\cos(\phi), \frac{\omega}{\aleph_m}\sin(\phi)\right) \quad \forall i = 1, 2, ..., 2\aleph_m + 1.$
  Compute $C_k(x) = \exp\big(g(x, \lambda_k)\big)$.
  for $n = 1$ **to** $T, \delta t > 0$ **do**
    for $i = 1$ **to** M **do**
      Run a single step EnKBF prediction ensemble $\tilde{u}^i_{t_{n+1}} = u^i_{t_n} + F(t_n, \lambda_k)u^i_{t_n}\delta t$
      Run a single step EnKBF analysis ensemble $u^i_{t_{n+1}} = \tilde{u}^i_{t_{n+1}} + P_{t_n}H^{\mathrm{T}}(t_n)R^{-1}(t_n)(dy_{t_n} + \epsilon_i - H(t_n)\tilde{u}^i_{t_n}\delta t)$
    end for
    Compute prediction ensemble mean: $\bar{u}_{t_n} = \frac{1}{M}\sum_{i=1}^M \tilde{u}^i_{t_n}$.
    Compute analysis ensemble mean: $\hat{u}_{t_n} = \frac{1}{M}\sum_{i=1}^M u^i_{t_n}$.
    Compute covariance: $P_{t_n} = \frac{1}{M-1}\sum_{i=1}^M (u^i_{t_n} - \hat{u}_{t_n})(u^i_{t_n} - \hat{u}_{t_n})^{\mathrm{T}}$.
  end for
  Metropolis Hastings
  Compute $\alpha_{\text{ratio}} = \dfrac{\pi(\delta y_{t_N:t_0}|\bar{u}_{t_N}, \lambda_k)}{\pi(\delta y_{t_N:t_0}|\bar{u}_{t_N}, \lambda_{k-1})}\dfrac{\rho_\lambda(\tilde{\lambda})}{\rho_\lambda(\lambda_k)}$
  Compute $\alpha = \min(1, \alpha_{\text{ratio}})$
  if $\alpha > U(0,1)$ **then**
    $\tilde{\lambda} = \lambda_k$
  else
    $\lambda_k = \tilde{\lambda}$
  end if
end for



(a) (b)

Figure 7.7: Plots for velocity parameters for the first three parameters in λ obtained using KBF and EnKBF and run for 1000 time steps and for 1000 Metropolis-Hastings cycles. The number of particles used for EnKBF is $M = 1000$, the time step size used in both filters is $dt = 0.01$, $\mu = 0.001$ and 100 grid points are used. The plots indicate that the estimates, for both filters, converge after about 100 iterations.

where $\lambda_{i,k}$ is the estimate of the ith parameter at Metropolis-Hastings cycle k and λ_i^{true} the true ith parameter. The RMSE for both the KBF and EnKBF, as shown in Figure 7.8(b), indicate



(a) (b)

Figure 7.8: (a) Box plot for the 21 velocity parameters for the EnKBF run for 1000 time steps with 1000 Metropolis-Hastings cycles. A burn-in of 500 parameter draws is discarded. The stochastic advection equation model is used with the following settings: $L = 2\pi$, 100 grid points, $\delta t = 0.01$, $M = 1000$ particles, $\mu = 0.001$ and localization radius of 10 grid points. (b) A plot of the root mean square error for the 21 hyper-parameter estimates obtained using EnKBF and KBF, respectively. The plot indicates that the performance of EnKBF matches that of KBF in this setting.

that the performance of EnKBF matches that of the KBF for the 21 parameters. These heuristic results corroborate the theoretical findings. The boxplot, Figure 7.8(a), shows the dispersion of parameter samples in the case when EnKBF is used. The result indicates that the estimates matches the true parameter values, with not so many outliers. This is indicative not only of the performance of EnKBF but also the Metropolis-Hastings procedure in locating the true parameter values and ensuring that no large excursions are made from the true parameter values.

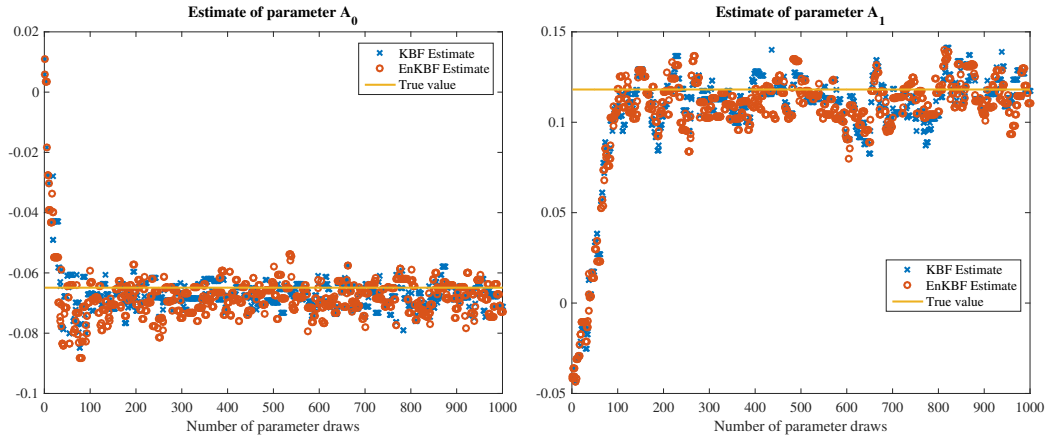
We now implement Algorithms 7.3.1 and 7.3.2 for the discretised wave equation, eq. (7.1.13).

Example 7.3.2: Wave Equation

We take the wave equation of Section 7.1.2 and the associated initial conditions, eqs. (7.1.12a) and (7.1.12b). The measurements are given by eq. (7.3.2). The initial value of u_t, $\{\beta_t\}$ and $\{\eta_t\}$ are uncorrelated. The aim is to estimate the spatially varying velocity, $C(x)$, by means of filter likelihood and Metropolis-Hastings algorithm.

The discretisation of the wave equation is as shown in Section 7.1.2. The panels in Figure 7.9 show the results.

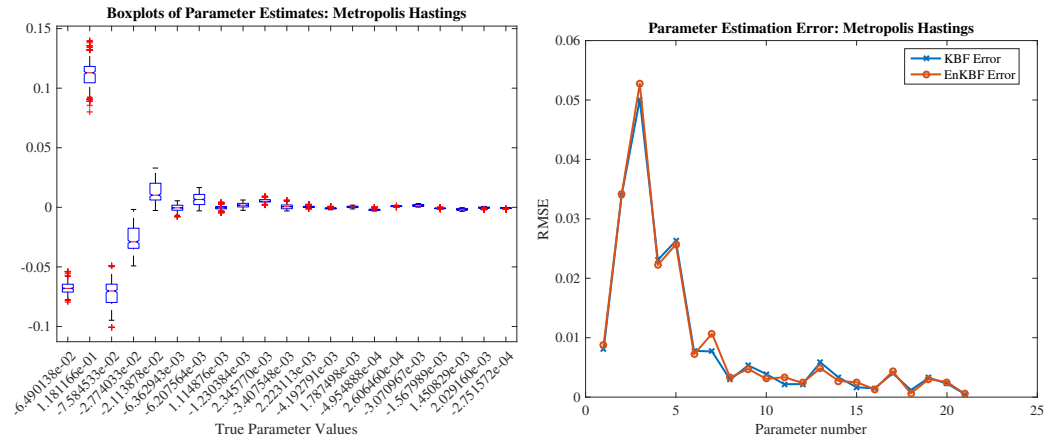
The results in Figure 7.9 indicate a close match in the performance of the EnKBF and KBF. This is as was anticipated in the theoretical findings of Section 4.3.1. Notice also the the two filters converge to the true parameter values after a few parameter draws (about 50 in Figure 7.9(a) and 100 in Figure 7.9(b))—which is indicative of the robustness of the Metropolis-Hastings algorithm atop the EnKBF and KBF filters. The results also show that there are no wide excursions from the true parameter values, which testifies of the good performance of Algorithms 7.3.1 and 7.3.2.



(a) (b)

Figure 7.9: Plots for velocity parameters for the first three parameters in λ obtained using KBF and EnKBF and run for 1000 time steps and 1000 Metropolis-Hastings cycles. The number of particles used for EnKBF is $M = 1000$, the time step size used in both filters is $dt = 0.01$, $\mu = 0.001$ and 100 grid points are used. The plots indicate that the estimates, for both filters, converge after about 100 iterations.

In Figure 7.10 are plotted the box-plots showing the dispersion of the 21 hyper-parameter estimates resulting from the use of EnKBF and the root mean square errors for parameter estimates for both the EnKBF and KBF. The EnKBF elicits an optimal performance as can be seen



(a) (b)

Figure 7.10: (a) Box plot for velocity parameters for the EnKBF run for 1000 time steps and 1000 Metropolis-Hastings cycles. A burn-in of 500 parameter draws is discarded. The stochastic wave equation model is used with the following settings: $L = 2\pi$, 100 grid-points, $\delta t = 0.01$, $M = 1000$ particles, $\mu = 0.001$ and localization radius of 10 grid points. (b) A plot of the root mean square error for the parameter estimates obtained using EnKBF and KBF, respectively. The plot indicates that the performance of EnKBF matches that of KBF in this setting.

in Figure 7.10(b) where the RMSEs for both the EnKBF and the optimal filter (KBF) match for all the 21 hyper-parameters. This also, as in the advection equation above, is in agreement with the theoretical findings that the EnKBF attains an optimal estimate in the limit $M \to \infty$. The boxplot, Figure 7.10(a), shows that the mean of EnKBF parameter estimates matches the true parameter values. What is more, there are not many outliers in the estimates. All this show that the EnKBF-Metropolis-Hastings algorithm is robust.

In the next section, we apply the concepts on dual estimation of Chapter 6 to stochastic hyperbolic PDEs, which, in this case, are the advection and the wave equations.

7.4 Simultaneous estimation of the state and spatially varying parameters

The dual filter (see Section 6.4.2 for details) can be adapted to allow for estimating both the state and spatially varying parameters, contemporaneously. The idea here is to replace the parameters in the dual filter with hyper-parameters of the varying parameters to be approximated. The hyper-parameters are then updated simultaneously, and in parallel, with the state at each iteration, where one filter estimates the state and the other filter updates the hyper-parameters—with each filters making use of the outcome of the other. To illustrate this argument, we turn to the advection and wave equation described in Examples 7.3.1 and 7.3.2, respectively, and use the KBF-EnKBF dual filter—in which the state is propagated and updated by means of the KBF whilst the hyper-parameters are updated using EnKBF—and ENKBF dual filter—where both the state and the hyper-parameters are propagated and updated using two EnKBFs running in parallel. The spatially varying velocity is as shown in Section 7.2.

In the KBF-EnKBF dual filter, we update, for every jth parameter particle $\lambda_t^j \in \{\lambda_t^j\}_{j=1}^L$ the state estimate, $\hat{u}_t^j$, using the KBF equations, eqs. (4.1.3a) and (4.1.3b); that is,

$$d\hat{u}_t^j = F(t)\hat{u}_t^j dt + P_t H^{\mathrm{T}}(t)R^{-1}(t)(dy_t - H(t)\hat{u}_t^j dt), \tag{7.4.1a}$$

$$dP_t = F(t)P_t dt + P_t F^{\mathrm{T}}(t)dt + G(t)G^{\mathrm{T}}(t)dt - P_t H^{\mathrm{T}}(t)R^{-1}(t)H(t)P_t dt. \tag{7.4.1b}$$

The parameters are updated using the EnKBF as described in Section 6.4.2.1, which we now adapt to suit the settings in this chapter; that is, each parameter hypothesis, λ_t^j, is updated using

$$d\lambda_t^j = D_t^L H(t)R^{-1}(t)(dy_t - 0.5(H(t)\hat{x}_t^j + H(t)\hat{x}_t)dt); \quad t_0 \le t, \tag{7.4.2}$$

where

$$\hat{\lambda}_t = \frac{1}{L}\sum_{j=1}^{L}\lambda_t^j; \qquad t_0 \le t, \tag{7.4.3a}$$

$$D_t^L = \frac{1}{L-1}\sum_{i=1}^{L}(\lambda_t^j - \hat{\lambda}_t)(\hat{u}_t^j - \hat{u}_t)^{\mathrm{T}}; \qquad t_0 \le t, \tag{7.4.3b}$$

where

$$\hat{u}_t^j = \frac{1}{M}\sum_{i=1}^{M} u_t^{i,j}, \tag{7.4.3c}$$

$$\hat{u}_t = \frac{1}{L}\sum_{i=1}^{L}\hat{u}_t^j. \tag{7.4.3d}$$

Algorithm 7.4.1 gives a summary of the KBF-EnKBF dual filter.

The EnKBF dual filter consists of an update of M particles of the state, $u_t^{i,j} \in \{u_t^{i,j}\}_{i,j=1}^{M,L}$, for every parameter particle, $\lambda_t^j \in \{\lambda_t^j\}_{j=1}^L$, using the EnKBF, eq. (4.2.1); that is,

$$du_t^{i,j} = F(t)u_t^{i,j}dt + G(t)d\beta_t^{i,j} + P_t^M H^{\mathrm{T}}(t)R^{-1}(t)(dy_t + R^{1/2}(t)\eta_t^{i,j} - H(t)u_t^{i,j}dt), \tag{7.4.4}$$

where $\{\eta_t^{i,j},\ t_0 \le t\}$ and $\{\beta_t^{i,j},\ t_0 \le t\}$ are, respectively, standard Brownian motion vector processes. The parameters are updated using the EnKBF given by eq. (7.4.2). The summary of the EnKBF dual filter is given in Algorithm 7.4.2.

Algorithm 7.4.1 KBF-EnKBF dual filter

Input: $u^j_{t_0}$, $\lambda^j_{t_0}$, $w^j_{t_0} = 1/L \;\; \forall j \in \{1, 2, ..., L\}$, P_{t_0} and $\delta y_{[t_0,t_T]}$.
Output: $\hat{u}_{[t_0,t_N]}$, $\hat{\lambda}_{[t_0,t_N]}$.

for $n = 1$ **to** $N, \delta t > 0$ **do**
  for $j = 1$ **to** L **do**
    Update $\hat{u}^j_{t_n}$ using eqs. (7.4.1a) and (7.4.1b)
    Update parameters $\lambda^j_{t_n}$ using eq. (7.4.2)
  end for
  Compute $\hat{\lambda}_{t_n} = \frac{1}{L}\sum_{j=1}^{L} \lambda^j_{t_n}$
  Compute $\hat{u}_{t_n} = \frac{1}{L}\sum_{i=1}^{L} \hat{u}^j_{t_n}$
end for

Algorithm 7.4.2 EnKBF dual filter

Input: $u^{i,j}_{t_0}$, $\lambda^j_{t_0}$, $w^j_{t_0} = 1/L \;\; \forall i \in \{1, 2, ..., M\}\; j \in \{1, 2, ..., L\}$, P_{t_0} and $\delta y_{[t_0,t_T]}$.
Output: $\hat{u}_{[t_0,t_N]}$, $\hat{\lambda}_{[t_0,t_N]}$.

for $n = 1$ **to** $N, \delta t > 0$ **do**
  for $j = 1$ **to** L **do**
    for $i = 1$ **to** M **do**
      Calculate $u^{i,j}_{t_n}$ using eq. (7.4.4)
    end for
    Update $\hat{u}^j_{t_n}$ using eq. (7.4.3c)
    Update parameters $\lambda^j_{t_n}$ using eq. (7.4.2)
  end for
  Compute $\hat{\lambda}_{t_n} = \frac{1}{L}\sum_{j=1}^{L} \lambda^j_{t_n}$
  Compute $\hat{u}_{t_n} = \frac{1}{L}\sum_{i=1}^{L} \hat{u}^j_{t_n}$
end for

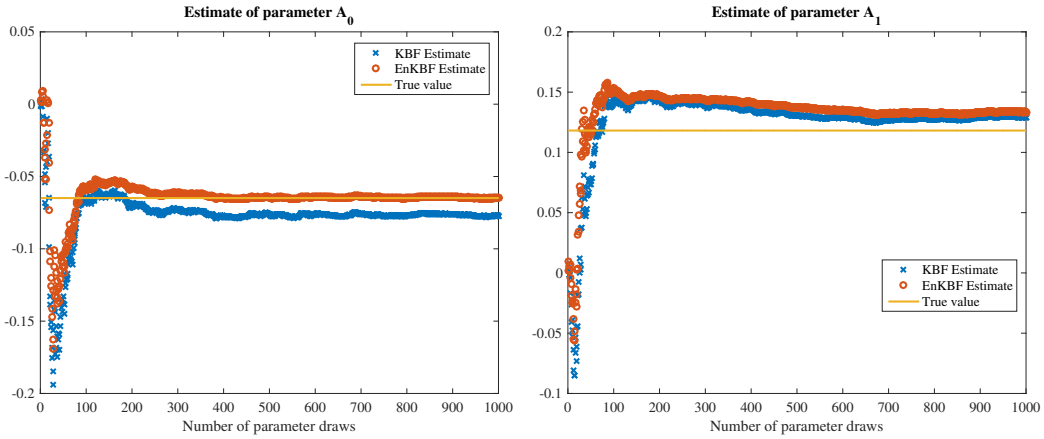
The panels in Figure 7.11 show the results for the first two parameters in eq. (7.2.5) when the dual filters are applied to the advection equation.

Evidently, from Figure 7.11, the performance of the EnKBF and KBF-EnKBF dual filters almost match. Both filters converge to the true estimate after a few iterations (about 100 in Figure 7.11(a)). This is another testament to the fact that EnKBF attains optimal estimates at high ensemble values. Moreover, both the EnKBF and KBF-EnKBF parameter estimates are not much spread as compared to the previous case where Metropolis-Hastings was used. This is more evident in the following results for the 21 hyper-parameters estimated.

In the following panels are plotted the root mean square errors for parameter estimates for both the EnKBF and KBF-EnKBF dual filters and the box-plots showing the dispersion of parameter estimates resulting from the use of EnKBF dual filter applied to the advection equation.

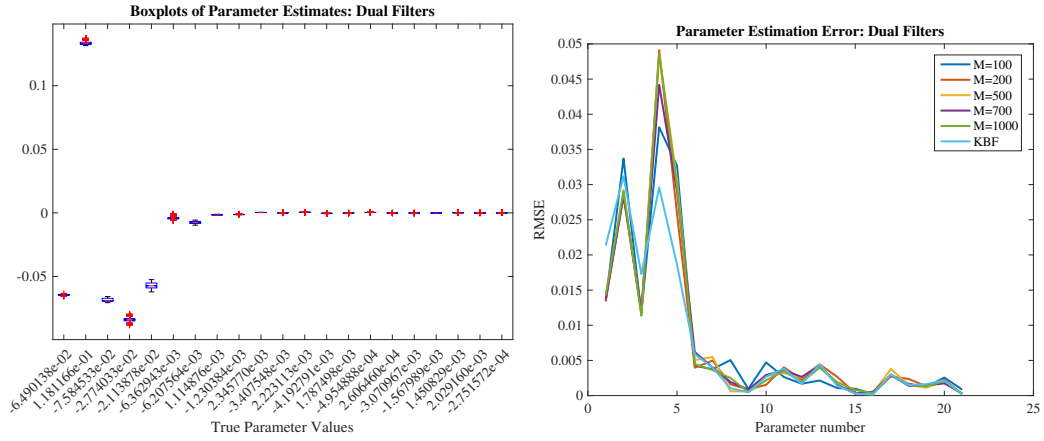
That the boxplots of EnKBF dual filter parameter estimates have short whiskers (see Figure 7.12(a)) and few outliers and that the estimates match the true parameter values indicates that EnKBF dual filter is robust in this setting. The EnKBF dual filter registers a slight variation in RMSE from that of the KBF-EnKBF dual filter. This indicates that the EnKBF, in this setting, performs optimally.

We now repeat the same procedure but with the wave equation described in Section 7.1.2 in the place of the advection equation. The following panels show the results. The results shown in Figure 7.13 are indicative of a dismal performance of the two dual filters—KBF-EnKBF and



(a) (b)

Figure 7.11: (a), (b) and (c) are plots for velocity parameters for the first three parameters in λ obtained using KBF-EnKBF and EnKBF dual filters, applied to the advection equation, both run for 1000 time steps. The number of particles, for the state and hyper-parameters, used in EnKBF is $M = 1000$ and $L = 1000$; the time step size used in both filters is $dt = 0.01$. $\mu = 0.001$ and 100 grid points are used. Compared to the results obtained using Metropolis-Hastings algorithm (Section 7.3), the dual filters register a better performance, at least in this example.

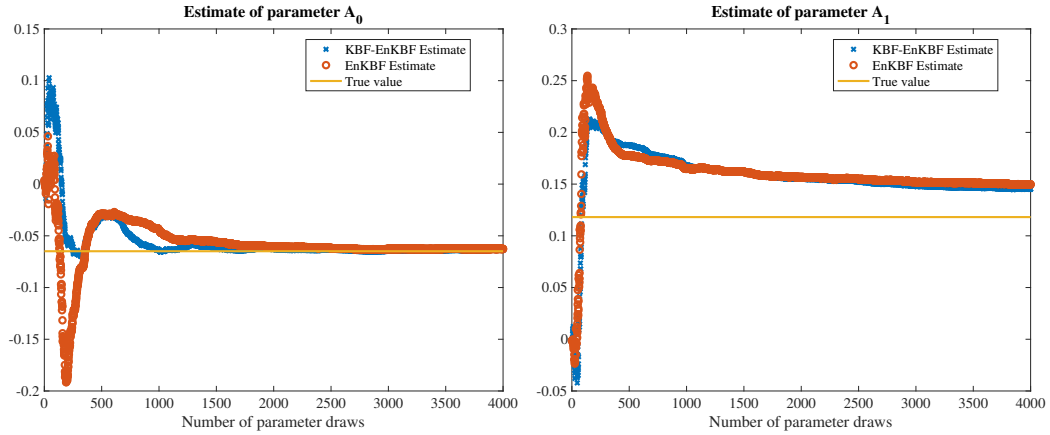


(a) (b)

Figure 7.12: (a) and (b) are, respectively, the box-plots showing the distribution of hyper-parameter estimates of EnKBF dual filter, applied to the advection equation, after a burn-in of 500 iterations and the plots of the root mean square error for the hyper-parameter estimates obtained using KBF-EnKBF and EnKBF (with different ensemble sizes) dual filters.

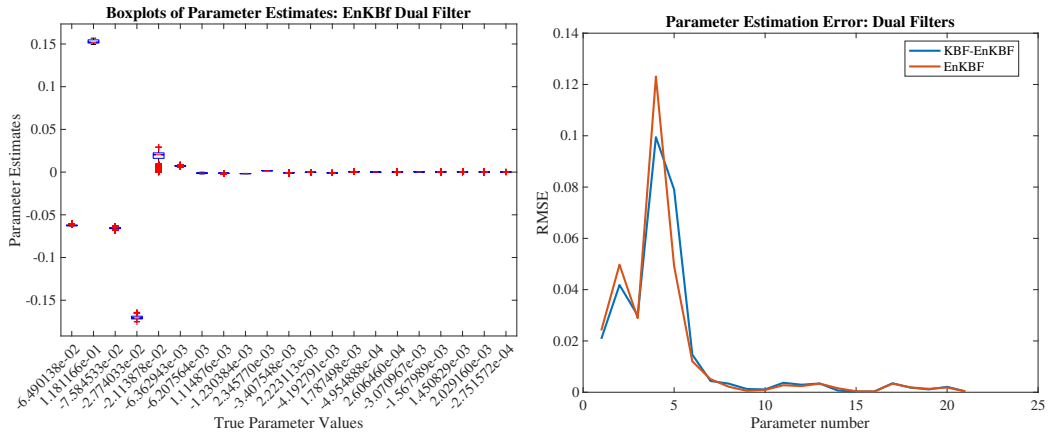
EnKBF dual filters—when applied to the wave equation as compared to the results obtained when the dual filters are applied to the advection equation (see Figure 7.11). We note that the wave equation is partially observed; that is, u only is observed in the discretised wave equation, eq. (7.1.13), whereas the advection equation is fully observed. Furthermore, the number of unknowns in the state-parameter system of the wave equation is 300 whilst that in the advection equation is 200. These account for the dismal performance of the KBF-EnKBF and EnKBF dual filters when applied to the wave equation.

In Figure 7.14 are plotted the root mean square errors for parameter estimates for both the EnKBF and KBF and the box-plots showing the dispersion of parameter estimates resulting from the use of EnKBF.



(a) (b)

Figure 7.13: (a), (b) and (c) are plots for velocity parameters for the first three parameters in λ obtained using KBF-EnKBF and EnKBF dual filters, applied to the wave equation (Example 7.3.1), both filters run for 1000 time steps. The number of particles, for the state and hyper-parameters, used in EnKBF filter is $M = 1000$; the time step size used in both filters is $dt = 0.01$. $\mu = 0.001$ and 100 grid points are used. Compared to the results obtained using Metropolis-Hastings algorithm (Section 7.3), the dual filters register a dismal performance, at least in this example.



(a) (b)

Figure 7.14: (a) and (b) are, respectively, the box-plots showing the distribution of the 21 hyper-parameter estimates of EnKBF dual filter after a burn-in of 500 iterations and plots of the root mean square error for the hyper-parameter estimates obtained using KBF-EnKBF and EnKBF (with different ensemble sizes) dual filters, applied to the wave equation (Example (7.3.2)).

From the foregoing, Metropolis-Hastings with the filter evidence performs well in estimation of parameters compared to the dual filters—especially when the number of unknowns is large. This is is indicative of the robustness of the Metropolis-Hastings algorithm in searching, and remaining in the, high probability region of the state-space.

CHAPTER 8

CONCLUSION

8.1 Summary and discussion

This thesis is established on time-continuous setting, where both the signal and measurements evolve in a temporal continuum. This is motivated by the fact that the time-continuous setting, as opposed to the discrete setting, offers a good basis for studying the filtering problem especially when nonlinear state space models are involved, because the equation of evolution of probability densities is already established. For the most part, the filter variants studied in this thesis obtain inspiration from the Kushner-Stratonovich equation, which equation is a solution to the filtering problem—assuming that the true posterior density is Gaussian.

The solution to the filtering problem lacks closure in most nonlinear state space models, for the Taylor series expansion of expected values of functions involve infinitely many higher-order moments. What is more, the Gaussian assumption both in the derivation of the Kushner-Sratonovich equation and in Taylor expansion of nonlinear functions renders the solution to the filtering problem highly restrictive. The Gaussian assumption presupposes that the true posterior is Gaussian. In the likely case where the true posterior is non-Gaussian, or multi-modal, then Kushner-Stratonovich equation underestimates the solution to the filtering problem.

In this thesis, a review is made of the first and second order approximations to the equations of evolution of mean and covariance with the Gaussian assumption. The first order approximate filter, second order approximate filter, first order extended Kalman-Bucy filter and the second order extended Kalman-Bucy filter arise naturally from these approximations. Although higher order approximate filters and extensions of Kalman-Bucy filters are promising more accuracy, they are, however, accompanied with the problem of complexity—which renders them unfavourable for practical applications. For this reason, filters based on Monte Carlo methods have gained prominence—among them being the particle filters.

The sampling importance resampling variant of particle filters, as is well known, is characterised by a resampling step equivalent to the analysis step where information from measurements is assimilated. Now the information coming from the measurements is incorporated into the dynamics by drawing from an empirical approximation of the true posterior where the measurements contribute to the weights used. Useful as it is, the resampling step in particle filters comes with the challenge of rendering the filter unfit for parallel operations; for the particles are resampled jointly. Moreover, resampling is known to induce a lack of variability of the particles, it being the case that particles of more weight are chosen and multiplied frequently to replace those of lesser weights. This undermines filter performance. Partly as a result, filtering algorithms that utilize Monte Carlo methods but avoid resampling are being sought for, which filters include feedback particle filters.

What sets feedback particle filters apart is the control structure at the measurements' as-

similation step. One variant of the feedback particle filters establishes a control in the calculation of the gain, which determines the weights given to the innovation process. This variant of feedback particle filters proposes Stratonovich stochastic differential equation for the evolution of the particles. The SDE comprises of the signal equation and a linearly added control characterised by an innovation process and the gain. Now the gain is obtained by solving an elliptic boundary value problem arising from an underlying optimization problem. The problem to be optimized is defined by Kullback-Leibler divergence between the density of the particles conditioned on all available measurements and the true posterior. The feedback particle filter is shown to be exact when the initial conditional density is exactly known.

The boundary value problem, the solving of which yields the gain, lacks analytical solution. A number of numerical methods have been devised to solve for the gain, among which is the constant gain approximation. It turns out that the constant gain approximated feedback particle filter is equivalent to the ensemble Kalman-Bucy filter. Feedback particle filters, therefore, can be considered as a large class of filters where ensemble Kalman-Bucy filter falls under.

Inspired by the two formulations of the ensemble Kalman-Bucy filter, *viz.*, with deterministically perturbed innovation and that with stochastically perturbed innovation, this thesis proposes a feedback particle filter with randomly perturbed innovation. A feedback particle filter with deterministically perturbed innovation is well known. It turns out, as is demonstrated in Chapter 5, that both formulations of the feedback particle filter are exact and have the same boundary value problem to be solved for the gain. The ensemble Kalman-Bucy filter with stochastically perturbed innovation is a consequent of the feedback particle filter with randomly perturbed innovation and with constant approximation of the gain. We show that, setting the initial probability density of the particles equal to the initial true posterior, the feedback particle filter with randomly perturbed innovation is exact.

We also propose feedback particle filters based on optimal transport of measures. The idea behind these filters is establishment of an optimal coupling between the probability density of the predicted particles and the density of the particles after assimilation step. The underlying optimal control problem is analogous to the Schrödinger problem. For practical implementation of the numerical control problem, we use Sinkhorn iteration as opposed to the earth movers distance algorithms—the former being faster and robust. For this cause, we call the resulting filter Sinkhorn particle filter (SPF). Another filter is proposed in which the ensemble transformation step is replace by drawing particles from an discretely approximated posterior where the weights are given by the solution to the Schrödinger problem defined by the prediction and analysis measures. This filter we call it resampling Sinkhorn particle filter (RSPF). The performance of the two filters are heuristically investigated by means of the stochastic Lorentz equation. It turns out that the two filters perform better than the bootstrap particle filter in the lower ensemble sizes. As the ensemble size increases, the RSPF elicits a continued better performance compared to the bootstrap particle filter. On the other hand, the SPF registers a better performance compared to the bootstrap particle filter but only for lower ensemble sizes.

This study also attends to the question of parameter estimation. We review the least squares, maximum a posteriori (MAP) and maximum likelihood (ML) methods of parameter estimation. Dual estimation approaches, which involve extended state space and use of dual filters for state and parameter estimation are also reviewed. We propose sequential Monte Carlo cum ensemble Kalman-Bucy dual filter and ensemble Kalman-Bucy filter cum ensemble transform particle filter dual filter. The latter filter is heuristically shown to perform better than the former. This is done in Chapter 6 of this thesis.

In Chapter 7, application of joint state and parameter estimation to advection and wave equation is done. We consider a special case of parameter estimation where the parameter to be estimated is spatially varying. Such a case arises, for example, in the velocity of a wave travelling through an inhomogeneous media. We propose and study two approaches: the use of

filter likelihood and Metropolis Hastings procedure and joint estimation of state and parameters. The parameter is expressed as a Fourier series with constant coefficients. The coefficients are approximated and then substituted back to the Fourier series to obtain an approximation of the velocity. Kalman-Bucy filter and the ensemble Kalman-Bucy filter are used. The filter likelihood with Metropolis Hastings procedure register a better performance compared to the joint estimation procedure in both advection and wave equations.

Overally, feedback particle filters offer a promising competence in solving the nonlinear filtering problem. The reason for this is that the feedback particle filters are shown to be exact under certain conditions. The downside is that the boundary value problem need to be solved for the gain each time. With the development of faster optimal control algorithms, feedback particle filters will be the filters of choice where robust filters are needed for practical use.

8.2 Future directions

The performance of the filters reviewed and proposed in this thesis can be enhanced in several ways. The following is a listing of possible directions to take in bolstering the performance of the feedback particle filters both for state and parameter estimation.

- ⋆ Study of the theoretical underpinnings of the SPF and the RSPF would be useful in showing why the filters work. This far, we have only shown the numerical performance of the SPF and RSPF filter as applied to stochastic Lorentz 63 model. The theory behind the local and global convergence of these filters forms part of our future study.

- ⋆ With the proposed feedback particle filters at hand, one would want to extend the idea behind their derivation to solving the smoothing problem. In future we shall study feedback particle smoothers and their applications.

- ⋆ It would be also interesting to study the forward-backward feedback particle filters, where the forward move is accomplished by means of a feedback particle filter, and a backward move is attained via a feedback particle smoother; for then are obtained accurate approximations for both state and parameter estimation.

- ⋆ Application of proposed filters in high-dimensional nonlinear practical problems is yet to be done. We would like to investigate how the feedback particle filter fare when applied to problems in which curse of dimensionality has been an hindrance.

- ⋆ With the successful application of filter likelihood with Metropolis Hastings procedure in estimating spatially varying parameters as done in Chapter 7 of this thesis, albeit in the linear hyperbolic pdes, we would, in future, consider extending this notion to high-dimensional nonlinear problems.

Bibliography

Allen, E., Novosel, S. and Zhang, Z. (1998), *Finite Element and Difference Approximation of Some Linear Stochastic Partial Differential Equations*, arXiv:1612.06065v1.

Amirhossein, T., de Wiljes, J., Prashant, G. and Reich, S. (2016), 'Kalman Filter and its Modern Extensions for the Continuous-time Nonlinear Filtering Problem', *Psychological Review* .

Angwenyi, D., de Wiljes, J. and Reich, S. (2017), 'Interacting Particle Filters for Simultaneous State and Parameter Estimation', *arXiv: 1709.09199v1* .

Arulampalam, S., Maskell, S., Gordon, N. and Clapp, T. (2002), 'A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking', *IEEE Transactions on Signal Processing* **50**(2), 174 – 188.

Bain, A. and Crisan, D. (2009), *Fundamentals of Stochastic Filtering*, Vol. 60 of *Stochastic Modelling and Applied Probability*, Springer-Verlag, New York.

Bar-Shalom, Y., Rong, X. and Kirubarajan, T. (2001), *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software*, John Wiley & Sons, Inc.

Bass, R., Norum, V. and Schwartz, L. (1966), 'Optimal Multichannel Nonlinear Filtering', *J. Math. Anal. Appl.* **16**, 152 – 164.

Beneš, V. E. (1981), 'Exact Finite-dimensional Filters for Certain Diffusions with Nonlinear Drift', *Stochastics* **5**(1-2), 65 – 92.

Bergemann, K. and Reich, S. (2012), 'An Ensemble Kalman-Bucy Filter for Continuous Data Assimilation', *Meteorolog. Zeitschrift* **21**, 213 – 219.

Brown, R. and Hwang, P. (1997), *Introduction to Random Signals and Applied Kalman Filtering with Matlab Exercises and Solutions*, third edition edn, John Wiley & Sons, Inc.

Burgers, G., van Leeuwen, P. and Evensen, G. (1998), 'Analysis Scheme in the Ensemble Kalman filter', *Monthly Weather Review* **126**, 1719–1724.

Carrassi, A., Bocquet, M., Hannart, A. and Ghil, M. (2016), 'Estimating Model Evidence Using Data Assimilation', *Quarterly Journal of the Royal Meteorological Society* **00**, 1–22.

Chen, J. and Wu, H. (2008), 'Estimating of Time-varying Parameters in Deterministic Dynamic Models', *Statistica Sinica* **18**, 987–1006.

Cohn, G. G. E. (1999), 'Construction of Correlation Functions in Two and Three Dimensions', *Q. J. Roy. Meteor. Soc.* **125**, 723–757.

Courant, R., Isaacson, E. and Rees, M. (1952), 'On the Solution of Nonlinear HyperbolicDifferential Equations by Finite Differences', *Communications on Pure and Applied Mathematics* **v**, 243–255.

Cuturi, M. (2013), 'Sinkhorn Distances: Lightspeed Computation of Optimal Transportation Distances', *arXiv:1306.0895v1 [stat.ML]* .

Daum, F. and Huang, J. (2011), 'Particle Degeneracy: Root Cause and Solution', *Proceedings of SPIE Conference, Orlando Florida, April* .

de Wiljes, J., Reich, S. and Stannat, W. (2016), 'Long-time Stability and Accuracy of the Ensemble Kalman-Bucy Filter for Fully Observed Processes and Small Measurement Noise', *arXiv:1612.06065v1* .

del Moral, P. (2004), *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*, Springer-Verlag, New York.

del Moral, P. (2013), *Mean Field Simulation for Monte Carlo Integration*, Chapman and Hall/CRC Monographs on Statistics and Applied Probability, Taylor and Francis.

Doob, J. L. (1953), *Stochastic Processes*, Wiley, New York.

Doucet, A. and Johansen, A. (2011), 'A tutorial on Particle Filtering and Smoothing: Fifteen Years Later'. In Crisan D., and Rozovskii B., editors, The Oxford Handbook of Nonlinear Filtering.

Einicke, G. A. (2012), *Smoothing, Filtering and Prediction: Estimating the Past, Present and Future*, InTech, Janeza Trdine 9, 51000 Rijeka, Croatia.

Evensen, G. (1994), 'Sequential Data Assimilation with a Nonlinear Quasi-geostrophic Model Using Monte Carlo Methods to Forecast Error Statistics', *J. Geophys. Res.* **99**(10), 143–162.

Evensen, G. (2006), *Data Assimilation: The Ensemble Kalman Filter*, Springer-Verlag, New York.

Hairer, E., Lubich, C. and Wanner, G. (2003), 'Geometric Numerical Integration Illustrated by the Störmer-Verlet Method', *Acta Numerica* pp. 399–450.

Itô, K. (1951), 'On Stochastic Differential Equations', *Mem. Amer. Math. Soc.* (4).

Jazwinski, A. (1970), *Stochastic Processes and Filtering Theory*, Academic Press, New York.

Kailath, T., Sayed, A. H. and Hassibi, B. (2000), *Linear Estimation*, Prentice Hall, New York.

Kalman, R. and Bucy, R. (1961), 'New Result in Linear Filtering and Prediction Theory', *Journal of Basic Engineering* .

Kloeden, P. and Platen, E. (1992), *Numerical Solution of Stochastic Differential Equations*, Vol. 23 of *Applications of Mathematics: Stochastic Modelling*, Springer.

Kloeden, P., Platen, E. and Schurz, H. (1994), *Numerical Solution of SDE through Computer Experiments*, Springer.

Kumar, P. and Narayanan, S. (2006), 'Solution of Fokker-Planck Equation by Finite Element and Finite Difference Methods for Nonlinear Systems', *Sādhanā* **31**(4), 445–461.

Kushner, H. (1962), 'On the Differential Equations Satisfied by Conditional Probability Densities of Markov Processes, with Applications', *J. SIAM Control* **2**(1).

Law, K., Stuart, A. and Zygalakis, K. (2015), *Data Assimilation: A Mathematical Introduction*, Springer.

Lewis, J., Lakshmivarahan, S. and Dhall, S. (2006), *Dynamic Data Assimilation: A Least Squares Approach*, Cambridge University Press.

Lindsten, F. (2013), Particle Filters and Markov Chains for Learning of Dynamical Systems, PhD thesis, Linköping Studies in Sciences and Technology. No. 1530.

Lint, J. W. C. V., Hoogendoorn, S. P. and Hagyi, A. (2008), Dual EKF State and Parameter Estimation in Multi-Class First-order Traffic Models, *in* 'Proceedings of the17th World Congress', The International Federation of Automatic Control, Seoul, Korea.

Lord, G., Powell, C. and Shardlow, T. (2014), *An Introduction to Computational Stochastic PDEs*, Cambridge University Press.

Lorentz, E. (1963), 'Deterministic Nonperiodic Flow', *Journal of Atmospheric Sciences* **20**, 130–141.

Lü, H., Yu, Z., Zhu, Y., Drake, S., Hao, Z. and Sudicky, E. A. (2011), 'Dual State-Parameter Estimation of Root Zone Soil Moisture by Optimal Parameter Estimation and Extended Kalman Filter Data Assimilation', *Advances in Water Resources* **34**, 395–406.

Mitter, S. K. and Newton, N. J. (2003), 'A Variational Approach to Nonlinear Estimation', *SIAM J. Control Optimization* **42**(5), 1813–1833.

Mohinder, G. S. and Angus, A. P. (2001), *Kalman Filtering: Theory and Practice Using MATLAB*, 2 edn, John Wiley & Sons, Inc.

Moradkhani, H., Sorooshian, S., Gupta, H. and Houser, P. (2005*a*), 'Dual State-Parameter Estimation of Hydrological Models Using Ensemble Kalman Filter', *Advances in Water Resources* **28**, 135–147. Elsevier.

Moradkhani, H., Sorooshian, S., Gupta, H. V. and Houser, P. R. (2005*b*), 'Dual State-Parameter Estimation of Hydrological Models Using Ensemble Kalman Filter', *Advances in Water Resources* **28**, 135–147.

Øksendal, B. (2005), *Stochastic Differential Equations: An introduction with Applications*, 6 edn, Springer.

Peyré, G. and Cuturi, M. (2018), *Computational Optimal Transport*, arXiv: 1803.00567v1 [stat.ML].

Rauch, H. E., Tung, F. and Striebel, C. T. (1965), 'Maximum Likelihood Estimates of Linear Dynamic Systems', *AIAA Journal* **3**(8), 1445–1450.

Rebeschini, P. and van Handel, R. (2015), 'Can Local Particle Filters Beat the Curse of Dimensionality', *The Annals of Applied Probability* **25**(5), 2809–2866.

Reich, S. (1999), 'Backward Error Analysis for Numerical Integrators', *SIAM Journal of Numerical Analysis* **36**, 1549–1570.

Reich, S. (2011), 'A Dynamical Systems Framework for Intermittent Data Assimilation', *BIT Numerical Mathematics* **51**, 235–249.

Reich, S. (2013), 'A Nonparametric Ensemble Transform Method for Bayesian Inference', *SIAM Journal of Scientific Computing* **35**, A2013–A2024.

Reich, S. and Cotter, C. (2015), *Probabilistic Forecasting and Bayesian Data Assimilation*, Cambridge University Press.

Robert, C. and Casella, G. (2004), *Monte Carlo Statistical Methods*, Springer-Verlag, New York.

Sahoo, S. (2010), 'Inverse vector operators', *arXiv: 0804.2239v3* .

Särkkä, S. (2013), *Bayesian Filtering and Smoothing*, Cambridge University Press, Cambridge.

Schaffter, T. (2010), 'Numerical Integration of SDEs: A Short Tutorial'. Swiss Federal Institute of Technology in Lausanne (EAPFL), Switzerland, Unpublished manuscript.

Schrödinger, E. (1931), 'Über die Umkehrung der Naturgesetze', *Sitzungberichte der Preußischen Akademie der Wissenschaften, Physikalisch-mathematische Klasse* pp. 144–153.

Spencer, B. J. and Bergman, L. (1993), 'On the Numerical Solution of the Fokker-Planck Equation for Nonlinear Stochastic Systems', *Nonlinear Dynamics* **4**, 357–372.

Stratonovich, R. (1966), 'A New Representation for Stochastic Integrals and Equations', *J. Siam Control* **4**, 362–371.

Stuart, M. (2007), 'Finite Difference Approximation for Linear Stochastic Partial Differential Equations with Method of Lines'. Munich Personal RePEc Archive, Paper No. 3983.

Taghvaei, A. and Mehta, P. G. (2018), 'Error analysis of the stochastic linear feedback particle filter', *arXiv: 1809.07892v1* .

Tomera, M. (2011), 'Kalman-Bucy Filter Design for Multivariable Ship Motion Control', *International Journal on Marine Navigation and Safety of Sea Transportation* **5**(3), 345–355.

Tuckwell, H. C. (2015), 'Numerical Solutions of some Hyperbolic Stochastic Partial Differential Equations with Mixed Derivatives Including Sine-Gordon Equation ', *arXiv:1508.01756v1* .

van Leeuwen, P. J. (2015), *Nonlinear Data Assimilation for High-dimensional Systems*, Vol. 2 of *Frontiers in Applied Dynamical Systems: Reveiws and Tutorials*, Springer-Verlag, New York, chapter 1, pp. 1–73.

Wan, E. and Nelson, T. (1997), 'Dual Kalman Filtering Methods for Nonlinear Prediction, Estimation and Smoothing'.

Yang, T. (2014), Feedback Particle Filter and its Applications, PhD thesis, Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign.

Yang, T., Laugesen, R., Mehta, P. and Meyn, S. (2012), Multivariable feedback particle filter, *in* 'Proceedings of the 51st IEEE Conference on Decision Control', pp. 4063–4070. Maui, HI.

Yang, T., Mehta, P. and Meyn, S. (2011*a*), A Mean-field Control-oriented Approach to Particle Filtering, *in* 'Proceedings of American Control Conference', pp. 2037–2043.

Yang, T., Mehta, P. and Meyn, S. (2011*b*), Feedback Particle Filter with Mean-field Coupling, *in* 'Proceedings of IEEE 50th Conference on Decision and Control', pp. 7909–7916. Orlando, FL.

Yang, T., Mehta, P. and Meyn, S. (2013), 'Feedback Particle Filter', *IEEE Transactions in Automatic Control* **58**(10), 2465–2480.

Zhang, Z. (1997), 'Parameter Estimation Techniques: A Tutorial with Applications to Conic Fitting', *Image and Vision Computing* . Elsevier.

APPENDIX A

ADDENDUM TO CHAPTER 3

A.1 Proof of Theorem 3.5.1

We follow the proof in Bass et al. (1966); Jazwinski (1970) and specialize to the scalar case.

Proof. By Taylor expansion about $\hat{x}_t$, we have:

$$f(x) \approx f(\hat{x}_t) + (x-\hat{x}_t)\partial_x[f](\hat{x}_t) + \frac{1}{2}(x-\hat{x}_t)^2\partial_{xx}[f](\hat{x}_t), \tag{A.1.1a}$$

$$h(x) \approx h(\hat{x}_t) + (x-\hat{x}_t)\partial_x[h](\hat{x}_t) + \frac{1}{2}(x-\hat{x}_t)^2\partial_{xx}[h](\hat{x}_t), \tag{A.1.1b}$$

$$g^2(x) \approx \left(g(\hat{x}_t) + (x-\hat{x}_t)\partial_x[g](\hat{x}_t) + \frac{1}{2}(x-\hat{x}_t)^2\partial_{xx}[g](\hat{x}_t)\right)^2. \tag{A.1.1c}$$

Taking expectations and noticing that $p = \pi_t[(x-\hat{x}_t)^2] = \pi_t[x(x-\hat{x}_t)]$, and that $\pi_t[(x-\hat{x}_t)] = 0$ leads to

$$\hat{f} \approx f(\hat{x}_t) + \frac{1}{2}p\partial_{xx}[f](\hat{x}_t), \tag{A.1.2a}$$

$$\hat{h} \approx h(\hat{x}_t) + \frac{1}{2}p\partial_{xx}[h](\hat{x}_t), \tag{A.1.2b}$$

$$\pi_t[g^2] \approx g^2(\hat{x}_t) + p\partial_x[g]^2(\hat{x}_t) + g(\hat{x}_t)\partial_{xx}[g](\hat{x}_t)p, \tag{A.1.2c}$$

$$\begin{aligned}\pi_t[xf] - \hat{x}_t\hat{f}_t &\approx \hat{x}_t f(\hat{x}_t) + p\partial_x[f](\hat{x}_t) + \frac{1}{2}\hat{x}_t p\partial_{xx}[f](\hat{x}_t) \\ &\quad - \hat{x}_t f(\hat{x}_t) - \frac{1}{2}\hat{x}_t p\partial_{xx}[f](\hat{x}_t) \\ &= p\partial_x[f](\hat{x}_t).\end{aligned} \tag{A.1.2d}$$

It now remains to obtain the approximation of the term

$$\pi_t[x^2h] - \pi_t[x^2]\hat{h}_t - 2\hat{x}_t\pi_t[xh] + 2\hat{x}_t^2\hat{h}_t.$$

By Taylor expansion

$$\pi_t[x^2] \approx \hat{x}_t^2 + (x-\hat{x}_t)2\hat{x}_t + (x-\hat{x}_t)^2. \tag{A.1.3}$$

Using eq. (A.1.1b) and eq. (A.1.3),

$$\begin{aligned}\pi_t[x^2h] &\approx \pi_t[(\hat{x}_t^2 + (x-\hat{x}_t)2\hat{x}_t + (x-\hat{x}_t)^2)(h(\hat{x}_t)] \\ &\quad + \pi_t[(x-\hat{x}_t)\partial_x[h](\hat{x}_t) + \frac{1}{2}(x-\hat{x}_t)^2\partial_{xx}[h](\hat{x}_t))] \\ &= \hat{x}_t^2h(\hat{x}_t) + \frac{1}{2}\hat{x}_t^2p_t\partial_{xx}[h](\hat{x}_t) + 2\hat{x}_tp_t\partial_x[h](\hat{x}_t) + p_th(\hat{x}_t).\end{aligned} \tag{A.1.4}$$

Similarly,

$$\begin{aligned}\pi_t[x^2]\hat{h}_t &\approx \pi_t[(\hat{x}_t^2 + (x-\hat{x}_t)2\hat{x}_t + (x-\hat{x}_t)^2)] \\ &\quad \times \pi_t[h(\hat{x}_t) + (x-\hat{x}_t)\partial_x[h](\hat{x}_t) + \frac{1}{2}(x-\hat{x}_t)^2\partial_{xx}[h](\hat{x}_t)] \\ &= (\hat{x}_t^2 + p_t)(h(\hat{x}_t) + (x-\hat{x}_t)\partial_x[h](\hat{x}_t) + \frac{1}{2}(x-\hat{x}_t)^2\partial_{xx}[h](\hat{x}_t)) \\ &= \hat{x}_t^2 h(\hat{x}_t) + \frac{1}{2}\hat{x}_t^2 p_t \partial_{xx}[h](\hat{x}_t) + p_t h(\hat{x}_t) + \frac{1}{2}p_t^2\partial_{xx}[h](\hat{x}_t).\end{aligned} \tag{A.1.5}$$

The remaining term, $-2\hat{x}\pi[xh] + 2\hat{x}_t^2\hat{h}_t$, can be written in the form of eq. (A.1.2d), with h replacing f, which simplifies it a great deal. That is,

$$\begin{aligned}-2\hat{x}_t\pi_t[xh] + 2\hat{x}_t^2\hat{h}_t &= -2\hat{x}_t(\pi_t[xh] - \hat{x}_t\hat{h}_t) \\ &\approx -2\hat{x}_t p_t \partial_x[h](\hat{x}_t).\end{aligned} \tag{A.1.6}$$

Now collecting like terms together in eqs. (A.1.4) to (A.1.6), we get:

$$\pi_t[x^2 h] - \pi_t[x^2]\hat{h}_t - 2\hat{x}_t\pi_t[xh] + 2\hat{x}_t^2\hat{h}_t = \frac{1}{2}p_t^2\partial_{xx}[h](\hat{x}_t). \tag{A.1.7}$$

The approximate filter for the exact filter eqs. (3.4.6) and (4.2.6) is obtained by replacing the expected values of the terms with their approximations, which have been developed above. □

APPENDIX B

ADDENDUM TO CHAPTER 4

B.1 Derivation of the KBF: minimum variance method

Given the Ito differential equations for the state and the observations; that is,

$$dx = F(t)xdt + G^{1/2}(t)d\beta_t, \quad t \geq t_0, \quad x_{t_0} \sim N(\hat{x}_{t_0}, P_{t_0}), \tag{4.1.1a revisited}$$

$$dy = H(t)xdt + R^{1/2}(t)d\eta_t, \quad t \geq t_0, \tag{4.1.1b revisited}$$

where the input $d\beta$ and the observation error $d\eta$ are white noise processes with statistics Brown and Hwang (1997)

$$\mathbb{E}\{d\beta\} = 0, \tag{B.1.2a}$$

$$\mathbb{E}\{d\beta d\beta^{\mathrm{T}}\} = I\delta(t-\tau), \tag{B.1.2b}$$

$$\mathbb{E}\{d\eta\} = 0, \tag{B.1.2c}$$

$$\mathbb{E}\{d\eta d\eta^{\mathrm{T}}\} = I\delta(t-\tau), \tag{B.1.2d}$$

we can now proceed to derive the equations governing the evolution of the state estimate and the error covariance matrix.

The solution to 4.1.1a is given by

$$x = \exp(F(t_0)(t-t_0))x_{t_0} + \int_{t_0}^{t} \exp(F(\tau)(t-\tau))G^{1/2}(\tau)d\beta d\tau. \tag{B.1.3}$$

At time t_{n+1} we have,

$$x_{t_{n+1}} \approx (I + F(t_n)\delta t)x_{t_n} + \int_{t_n}^{t_{n+1}} (I + F(\tau)(t_{n+1}-\tau))G^{1/2}(\tau)d\beta d\tau. \tag{B.1.4}$$

Next, we derive the discrete Kalman Filter with equivalent dynamics.

B.1.1 Kalman filter

suppose we have a discrete model for the evolution of an N-dimensional state, x, given by

$$x_{t_{n+1}} = \Pi(t_n)x_{t_n} + G^{1/2}(t_n)v_{t_n}, \tag{B.1.5}$$

and increments observations given by

$$\delta y_{t_n} = H(t_n)x_{t_n}\delta t + \delta t^{1/2}R^{1/2}(t_n)w_{t_n}, \tag{B.1.6}$$

with the following statistics

$$\mathbb{E}\{v_{t_n}\} = 0, \tag{B.1.7a}$$
$$\mathbb{E}\{v_{t_n} v_{t_n}^{\mathrm{T}}\} = I, \tag{B.1.7b}$$
$$\mathbb{E}\{w_{t_n}\} = 0, \tag{B.1.7c}$$
$$\mathbb{E}\{w_{t_n} w_{t_n}^{\mathrm{T}}\} = I; \tag{B.1.7d}$$

that is to say, v_{t_n} and w_{t_n} are i.i.d sequences of zero mean and identity covariances. Let $\bar{x}_{t_n}$ be the state estimate before assimilation of data. The corresponding error, $\tilde{e}_{t_n}$, is given by

$$\tilde{e}_{t_n} = x_{t_n} - \bar{x}_{t_n}. \tag{B.1.8}$$

Similarly, let $\hat{x}_{t_n}$ be the state estimate after incorporating observations, and let the observations be incorporated thus

$$\hat{x}_{t_n} = \bar{x}_{t_n} + K_{t_n}(\delta y_{t_n} - H(t_n)\bar{x}_{t_n}\delta t). \tag{B.1.9}$$

The error at time t_n is given by

$$e_n = x_{t_n} - \hat{x}_{t_n}. \tag{B.1.10}$$

The error covariance matrices before and after incorporation of observation are, respectively,

$$\bar{P}_{t_n} = \mathbb{E}\{\tilde{e}_{t_n}\tilde{e}_{t_n}^{\mathrm{T}}\}, \tag{B.1.11a}$$

$$\begin{aligned}
P_{t_n} &= \mathbb{E}\{e_n e_n^{\mathrm{T}}\} = \mathbb{E}\{[x_{t_n} - \hat{x}_{t_n}][x_{t_n} - \hat{x}_{t_n}]^{\mathrm{T}}\} \\
&= \mathbb{E}\{[x_{t_n} - \bar{x}_{t_n} - K_{t_n}(\delta y_{t_n} - H(t_n)\bar{x}_{t_n}\delta t)\bar{x}_{t_n})] \\
&\quad \times [x_{t_n} - \bar{x}_{t_n} - K_{t_n}(\delta y_{t_n} - H(t_n)\bar{x}_{t_n}\delta t]^{\mathrm{T}}\} \\
&= \mathbb{E}\{[x_{t_n} - \bar{x}_{t_n} - K_{t_n}(H(t_n)x_{t_n}\delta t + \delta t^{1/2}R^{1/2}(t_n)w_{t_n} - H(t_n)\bar{x}_{t_n}\delta t)] \\
&\quad \times [x_{t_n} - \bar{x}_{t_n} - K_{t_n}(H(t_n)x_{t_n}\delta t + \delta t^{1/2}R^{1/2}(t_n)w_{t_n} - H(t_n)\bar{x}_{t_n}\delta t)]^{\mathrm{T}}\} \\
&= \mathbb{E}\{[x_{t_n} - \bar{x}_{t_n} - \delta t K_{t_n}H(t_n)(x_{t_n} - \bar{x}_{t_n}) - \delta t^{1/2}K_{t_n}R^{1/2}(t_n)w_{t_n}] \\
&\quad \times [x_{t_n} - \bar{x}_{t_n} - \delta t K_{t_n}H(t_n)(x_{t_n} - \bar{x}_{t_n}) - \delta t^{1/2}K_{t_n}R^{1/2}(t_n)w_{t_n}]^{\mathrm{T}}\} \\
&= \mathbb{E}\{[x_{t_n} - \bar{x}_{t_n}][x_{t_n} - \bar{x}_{t_n}]^{\mathrm{T}}\} - \delta t K_{t_n}H(t_n)\mathbb{E}\{[x_{t_n} - \bar{x}_{t_n}][x_{t_n} - \bar{x}_{t_n}]^{\mathrm{T}}\} \\
&\quad - \delta t\mathbb{E}\{[x_{t_n} - \bar{x}_{t_n}][x_{t_n} - \bar{x}_{t_n}]^{\mathrm{T}}\}H(t_n)^{\mathrm{T}}K_{t_n}^{\mathrm{T}} \\
&\quad + \delta t^2 K_{t_n}H(t_n)\mathbb{E}\{[x_{t_n} - \bar{x}_{t_n}][x_{t_n} - \bar{x}_{t_n}]^{\mathrm{T}}\}H(t_n)^{\mathrm{T}}K_{t_n}^{\mathrm{T}} \\
&\quad + \delta t^{3/2}K_{t_n}H(t_n)\mathbb{E}\{[x_{t_n} - \bar{x}_{t_n}]w_{t_n}^{\mathrm{T}}\}R^{1/2}(t_n)^{\mathrm{T}}K_{t_n}^{\mathrm{T}} \\
&\quad - \delta t^{1/2}K_{t_n}R^{1/2}(t_n)\mathbb{E}\{w_{t_n}[x_{t_n} - \bar{x}_{t_n}]^{\mathrm{T}}\} \\
&\quad + \delta t^{3/2}K_{t_n}R^{1/2}(t_n)\mathbb{E}\{w_{t_n}[x_{t_n} - \bar{x}_{t_n}]^{\mathrm{T}}H(t_n)^{\mathrm{T}}K_{t_n}^{\mathrm{T}}\} \\
&\quad - \delta t^{1/2}\mathbb{E}\{[x_{t_n} - \bar{x}_{t_n}]w_{t_n}^{\mathrm{T}}\}R^{1/2}(t_n)^{\mathrm{T}}K_{t_n}^{\mathrm{T}} \\
&\quad + \delta t K_{t_n}R^{1/2}(t_n)\mathbb{E}\{w_{t_n}w_{t_n}^{\mathrm{T}}\}R^{1/2}(t_n)^{\mathrm{T}}K_{t_n}^{\mathrm{T}} \\
&= \bar{P}_{t_n} - \delta t K_{t_n}H(t_n)\bar{P}_{t_n} - \delta t\bar{P}_{t_n}H(t_n)^{\mathrm{T}}K_{t_n}^{\mathrm{T}} + \delta t^2 K_{t_n}H(t_n)\bar{P}_{t_n}H(t_n)^{\mathrm{T}}K_{t_n}^{\mathrm{T}} \\
&\quad + \delta t K_{t_n}R^{1/2}(t_n)R^{1/2}(t_n)^{\mathrm{T}}K_{t_n}^{\mathrm{T}};
\end{aligned} \tag{B.1.11b}$$

where we have assumed that the filter errors and the observation errors are uncorrelated. To realise an optimal filter, we need a value of K_{t_n} that will minimise the error covariance, P_{t_n}. This is obtained by differentiating the trace of P_{t_n} with respect to K_{t_n} and solve the resultant derivative after equating it to zero. More formally,

$$\begin{aligned}
\frac{d\mathrm{tr}(P_{t_n})}{dK_{t_n}} = 0 &= -\delta t[H(t_n)\bar{P}_{t_n}]^{\mathrm{T}} - \delta t\bar{P}_{t_n}H(t_n)^{\mathrm{T}} \\
&\quad + 2\delta t^2 K_{t_n}H(t_n)\bar{P}_{t_n}H(t_n)^{\mathrm{T}} + 2\delta t K_{t_n}R(t_n) \\
&= -2\delta t\bar{P}_{t_n}H(t_n)^{\mathrm{T}} + 2\delta t^2 K_{t_n}H(t_n)\bar{P}_k H(t_n)^{\mathrm{T}} + 2\delta t K_{t_n}R(t_n),
\end{aligned} \tag{B.1.12}$$

in which case we have used the fact that $\bar{P}_{t_n}$ is symmetric. Solving for K_{t_n} in eq. (B.1.12) yields

$$K_{t_n} = \bar{P}_{t_n} H(t_n)^{\mathrm{T}} [H(t_n)\bar{P}_{t_n} H(t_n)^{\mathrm{T}} + R(t_n)/\delta t]^{-1}/\delta t. \tag{B.1.13}$$

We revert back to eq. (B.1.9) and, for an optimal filter, substitute eq. (B.1.13) to obtain

$$\begin{aligned} \hat{x}_{t_n} = \bar{x}_{t_n} + \bar{P}_{t_n} H(t_n)^{\mathrm{T}} [H(t_n)\bar{P}_{t_n} H(t_n)^{\mathrm{T}} \\ + R(t_n)/\delta t]^{-1} (\delta y_{t_n} - H(t_n)\bar{x}_{t_n}\delta t)/\delta t. \end{aligned} \tag{B.1.14}$$

Similarly, substituting the value of K_{t_n} in the error covariance equation, eq. (B.1.11b), which we rewrite as

$$\begin{aligned} P_{t_n} = \bar{P}_{t_n} - \delta t K_{t_n} H(t_n)\bar{P}_{t_n} - \delta t \bar{P}_{t_n} H(t_n)^{\mathrm{T}} K_{t_n}^{\mathrm{T}} \\ + \delta t^2 K_{t_n} [H(t_n)\bar{P}_{t_n} H(t_n)^{\mathrm{T}} + R(t_n)/\delta t] K_{t_n}^{\mathrm{T}}, \end{aligned} \tag{B.1.15}$$

gives

$$P_{t_n} = \bar{P}_{t_n} - \bar{P}_{t_n} H(t_n)^{\mathrm{T}} [H(t_n)\bar{P}_{t_n} H(t_n)^{\mathrm{T}} + R(t_n)/\delta t]^{-1} H(t_n)\bar{P}_{t_n}. \tag{B.1.16}$$

Equations (B.1.14) and (B.1.16) are, collectively, the equations for the classic Kalman filter.

B.1.2 Extension to the continuous case

In the next time step, the estimate before assimilation is given by

$$\bar{x}_{n+1} = \Pi(t_n)\hat{x}_{t_n}. \tag{B.1.17}$$

The corresponding error covariance matrix, then, is

$$\begin{aligned} \bar{P}_{n+1} &= \mathbb{E}\{[x_{t_{n+1}} - \bar{x}_{n+1}][x_{t_{n+1}} - \bar{x}_{n+1}]^{\mathrm{T}}\} \\ &= \mathbb{E}\{[\Pi(t_n)x_{t_n} + G^{1/2}(t_n)\upsilon_{t_n} - \Pi(t_n)\hat{x}_{t_n}] \\ &\quad \times [\Pi(t_n)x_{t_n} + G^{1/2}(t_n)\upsilon_{t_n} - \Pi(t_n)\hat{x}_{t_n}]^{\mathrm{T}}\} \\ &= \mathbb{E}\{[\Pi(t_n)(x_{t_n} - \hat{x}_{t_n}) + G^{1/2}(t_n)\upsilon_{t_n}][\Pi(t_n)(x_{t_n} - \hat{x}_{t_n}) + G^{1/2}(t_n)\upsilon_{t_n}]^{\mathrm{T}}\} \\ &= \Pi(t_n)\mathbb{E}\{[x_{t_n} - \hat{x}_{t_n}][x_{t_n} - \hat{x}_{t_n}]^{\mathrm{T}}\} + \Pi(t_n)\mathbb{E}\{[x_{t_n} - \hat{x}_{t_n}]\upsilon_n^{\mathrm{T}}\} G^{1/2}(t_n)^{\mathrm{T}} \\ &\quad + G^{1/2}(t_n)\mathbb{E}\{\upsilon_{t_n}(x_{t_n} - \hat{x}_{t_n})^{\mathrm{T}}\}\Pi(t_n)^{\mathrm{T}} + G^{1/2}(t_n)\mathbb{E}\{\upsilon_{t_n}\upsilon_{t_n}^{\mathrm{T}}\} G^{1/2}(t_n)^{\mathrm{T}} \\ &= \Pi(t_n)P_{t_n}\Pi(t_n)^{\mathrm{T}} + G(t_n). \end{aligned} \tag{B.1.18}$$

We now seek to link eqs. (B.1.5) and (B.1.6) with eqs. (3.1.1a) and (3.1.1b), respectively, and their noise statistics. With a small increment in time, $\delta t = t_{n+1} - t_n$, we have, from eq. (B.1.5),

$$\begin{aligned} \delta x_{t_n} &= x_{t_{n+1}} - x_{t_n} \\ &= [\Pi(t_n) - I]x_{t_n} + G^{1/2}(t_n)\upsilon_{t_n}, \end{aligned} \tag{B.1.19}$$

whose limit, as $\delta t \to 0$, is

$$dx_{t_n} = [\Pi(t_n) - I]x_{t_n} + G^{1/2}(t_n)\upsilon_{t_n}. \tag{B.1.20}$$

Comparing equations eq. (B.1.20) and eq. (4.1.1a) it is evident that

$$\Pi(t_n) \approx I + F(t_n)\delta t. \tag{B.1.21}$$

Moreover, from eq. (B.1.4)

$$\begin{aligned} G^{1/2}(t_n)\upsilon_{t_n} &\approx \int_{t_n}^{t_{n+1}} (I + F(\tau)(t_{n+1} - \tau)) G^{1/2}(\tau) d\beta d\tau \\ &= \int_{t_n}^{t_{n+1}} \Pi(\tau) G^{1/2}(\tau) d\beta d\tau. \end{aligned} \tag{B.1.22}$$

Furthermore, we have the following expression for the model noise.

$$\begin{aligned}\mathbb{E}\{[G^{1/2}(t_n)v_{t_n}][G^{1/2}(t_n)v_{t_n}]^{\mathrm{T}}\} = \mathbb{E}\{[\int_{t_n}^{t_{n+1}} \Pi(\tau)G^{1/2}(\tau)d\beta d\tau] \\ \times [\int_{t_n}^{t_{n+1}} \Pi(\tau)G^{1/2}(\tau)d\beta d\tau]^{\mathrm{T}}\},\end{aligned} \tag{B.1.23}$$

which we simplify further as follows, noting that $\Pi(\tau) \approx I$ for very small δt,

$$\begin{aligned}G^{1/2}(t_n)\mathbb{E}\{v_{t_n}v_{t_n}^{\mathrm{T}}\}G^{1/2}(t_n)^{\mathrm{T}} &= \int\int G^{1/2}(\tau_1)\mathbb{E}\{d\beta_{\tau_1}d\beta_{\tau_2}^{\mathrm{T}}\}G^{1/2}(\tau_2)^{\mathrm{T}}\delta(\tau_1-\tau_2)d\tau_1 d\tau_2 \\ &= \int G^{1/2}(\tau_1)\mathbb{E}\{d\beta_{\tau_1}d\beta_{\tau_1}^{\mathrm{T}}\}G^{1/2}(\tau_1)^{\mathrm{T}}d\tau_1 \\ &= G^{1/2}(t)\mathbb{E}\{d\beta d\beta^{\mathrm{T}}\}G^{1/2}(t)^{\mathrm{T}}\delta t,\end{aligned} \tag{B.1.24}$$

whence we get

$$\mathbb{E}\{v_{t_n}v_{t_n}^{\mathrm{T}}\} = \mathbb{E}\{d\beta d\beta^{\mathrm{T}}\}\delta t. \tag{B.1.25}$$

Next, we consider the observation equation, eq. (B.1.6). The increment in observations, δy_{t_n} is given by

$$\delta y_{t_n} = H(t_n)x_{t_n}\delta t + \delta t^{1/2}R^{1/2}(t_n)w_{t_n}, \tag{B.1.6 revisited}$$

Taking the limit of δy_{t_n} as $\delta t \to 0$ gives

$$dy_{t_n} = \int_{t_n}^{t_n+\delta t} H(\tau)x\delta t d\tau + \int_{t_k}^{t_k+\delta t} \delta t^{1/2}R^{1/2}(\tau)w d\tau; \tag{B.1.26}$$

which, when compared with eq. (4.1.1b), shows that

$$R^{1/2}(t_n)d\eta_{t_n} \approx \int_{t_n}^{t_n+\delta t} \delta t^{1/2}R^{1/2}(\tau)w d\tau. \tag{B.1.27}$$

The covariances are then given by

$$\begin{aligned}\mathbb{E}\{[R^{1/2}(t_n)d\eta_{t_n}][R^{1/2}(t_n)d\eta_{t_n}]^{\mathrm{T}} \approx \delta t\mathbb{E}\{[\int_{t_n}^{t_n+\delta t} R^{1/2}(\tau)w d\tau] \\ \times [\int_{t_n}^{t_n+\delta t} R^{1/2}(\tau)w d\tau]^{\mathrm{T}}\}.\end{aligned} \tag{B.1.28}$$

Following the same argument as in the model noise covariance equation above leads to

$$R^{1/2}(t)\mathbb{E}\{d\eta_{t_n}d\eta_{t_n}^{\mathrm{T}}\}R^{1/2}(t)^{\mathrm{T}} = R^{1/2}(t_n)\mathbb{E}\{w_{t_n}w_{t_n}^{\mathrm{T}}\}\delta t^2 R^{1/2}(t_n). \tag{B.1.29}$$

We now revert back to the classic Kalman filter equations. subtracting $\hat{x}_{t_{n-1}}$ from eq. (B.1.14) and using the equivalent values of $\bar{x}_{t_n}$ as indicated in eq. (B.1.17) gives

$$\begin{aligned}\hat{x}_{t_n} - \hat{x}_{t_{n-1}} = \Pi(t_{n-1})\hat{x}_{t_{n-1}} - \hat{x}_{t_{n-1}} + \bar{P}_{t_n}H(t_n)^{\mathrm{T}}[H(t_n)\bar{P}_{t_n}H(t_n)^{\mathrm{T}} + R(t_n)/\delta t]^{-1} \\ \times (\delta y_{t_n} - H(t_n)\Pi(t_{n-1})\hat{x}_{t_{n-1}}\delta t)/\delta t.\end{aligned} \tag{B.1.30}$$

Using the approximation eq. (B.1.21) we get

$$\begin{aligned}\hat{x}_{t_n} - \hat{x}_{t_{n-1}} = F(t_{n-1})\hat{x}_{t_{n-1}}\delta t + \bar{P}_{t_n}H(t_n)^{\mathrm{T}}[H(t_n)\bar{P}_{t_n}H(t_n)^{\mathrm{T}} + R(t_n)/\delta t]^{-1} \\ \times (\delta y_{t_n} - H(t_n)\hat{x}_{t_{n-1}}\delta t - F(t_{n-1})\hat{x}_{t_{n-1}}\delta t^2)/\delta t.\end{aligned} \tag{B.1.31}$$

We drop the values of order $O(\delta t^2)$ and get

$$\begin{aligned}\delta \hat{x}_{t_{n-1}} = F(t_{n-1})\hat{x}_{t_{n-1}}\delta t + \bar{P}_{t_n} H(t_n)^{\mathrm{T}}[H(t_n)\bar{P}_{t_n}H(t_n)^{\mathrm{T}} \\ + R(t_n)/\delta t]^{-1}(\delta y_{t_n} - H(t_n)\hat{x}_{t_{n-1}}\delta t)/\delta t.\end{aligned} \tag{B.1.32}$$

From eqs. (B.1.13) and (B.1.29) we get

$$\bar{P}_{t_n} H(t_n)^{\mathrm{T}}[H(t_n)\bar{P}_{t_n}H(t_n)^{\mathrm{T}} + R(t_n)/\delta t]^{-1}/\delta t \approx \bar{P}_{t_n} H(t_n)^{\mathrm{T}} R(t)^{-1}\delta t, \tag{B.1.33}$$

we take the limit as $\delta t \to 0$, which leads to

$$d\hat{x} = F(t)\hat{x}dt + PH(t)^{\mathrm{T}}R(t)^{-1}(dy - H(t)\hat{x}dt)dt. \tag{B.1.34}$$

For the error covariance matrix, we begin with eq. (B.1.18). substituting P_{t_n} with its expression in eq. (B.1.16) and then using the discrete form of eq. (B.1.21), we obtain

$$\begin{aligned}\bar{P}_{n+1} &= \Pi(t_n)\bar{P}_{t_n}\Pi(t_n)^{\mathrm{T}} - \Pi(t_n)\bar{P}_{t_n}H(t_n)^{\mathrm{T}}[H(t_n)\bar{P}_k H(t_n)^{\mathrm{T}} \\ &\quad + R(t_n)/\delta t]^{-1}H(t_n)\bar{P}_{t_n}\Pi(t_n)^{\mathrm{T}} + G(t_n) \\ &= [I + F(t_n)\delta t]\bar{P}_{t_n}[I + F(t_n)\delta t]^{\mathrm{T}} \\ &\quad - [I + F(t_n)\delta t]\delta t K_{t_n}H(t_n)\bar{P}_{t_n}[I + F(t_n)\delta t]^{\mathrm{T}} + G(t_n).\end{aligned} \tag{B.1.35}$$

Expanding and neglecting $O(\delta t^2)$ terms yields

$$\bar{P}_{n+1} = \bar{P}_{t_n} + F(t_n)\delta t\bar{P}_{t_n} + \bar{P}_{t_n}F(t_n)^{\mathrm{T}}\delta t - \delta t K_{t_n}H(t_n)\bar{P}_{t_n} + G(t_n). \tag{B.1.36}$$

Next, we use eqs. (B.1.25) and (B.1.33) and get

$$\delta\bar{P}_{t_n} = F(t_n)\delta t\bar{P}_{t_n} + \bar{P}_{t_n}F(t_n)^{\mathrm{T}}\delta t - K_{t_n}H(t_n)\bar{P}_{t_n}\delta t + G(t)\delta t. \tag{B.1.37}$$

Finally, we take the limits as $\delta t \to 0$ to obtain

$$dP = F(t)dtP + PF(t)^{\mathrm{T}}dt - KH(t)Pdt + G(t)dt. \tag{B.1.38}$$

Equations (B.1.34) and (B.1.38) are the equations for the Kalman-Bucy filter.