Estimation of State Space Models using Particle Filters - applications to Economics and Finance

Hao S Zhou

Department of Economics

University of Leicester

A thesis submitted for the degree of

Doctor of Philosophy

Yet to be decided

This thesis is dedicated to my father and mother: Tie Qiu Zhou and Jiu Ying Zhao. All of my little successes have been the result of their unconditional love and unceasing support. For what they have led me to understand the deep meaning of:

> Life can only be understood backwards, but must be lived forwards.
>
> – Soren Kierkegaard.

Abstract

In recent years, general state space models have been proven to be extremely useful in modelling wide range of economic and financial time series. Subsequently, particle filters, a computational simulation based method along with its related techniques had burst into our spectrum and fill our expectation of estimating general state space models. However, particle methods can be computationally intensive, as well as possibly requiring stringent restrictions on the parameters space to achieve timely convergence. In this thesis, I propose several improvements to particle methods on different aspects. A list of the improvements are: general computational time reduction in particle filters, modified particle smoothing algorithm, more accurate parameter and state variable estimation through the utilizations of Modified Entropy particle filter, and apply novel general state space model estimation method to real economic and financial time series.

Acknowledgements

I am truly grateful for the guidance of my supervisors and members of the Economics Department, University of Leicester throughout this process.
I would like to thank both of my supervisors, Prof Stephen Hall and Prof Stephen Pollock, for their time and willingness to help me when I need the most. I could not have hoped for a better mentor than Prof Stephen Hall. I am in debt with all that he invested in me as his PhD student. I would also like to thank Prof Stephen Pollock for taking me on given a difficult situation I was in. I have become a better researcher and person because of them.
I am also grateful to members of Department of Economics, University of Leicester, for countless useful discussions and knowledgeable advices. Those who I should particularly mentioned are: Dr Subir Bose, Dr Sanjit Dhami, Dr James Rockey, and Dr Martin Foureaux Koppensteiner, for their unwavering faith in me as a student.
I would like to thank my colleagues and supporting staff in the Department of Economics. Their effort and kindness had helped me to survive my graduate years. Especially, I thank Sneha Gaddam, Narges Hajimoladarvish, Miguel Flores, and Johan Rewlak for their support, for the friendships we have built will last more than life long.
I would like to thank the most important people in my life - my parents. It is their collective encouragement and unceasing support throughout has made who I am to-day. I am grateful for the help, and for they have never hesitated to give, year after year. I would also like to thank my girlfriend Crystal Kuo, for being so patient with me during the time while I continuously ramble on non-interesting things to her, and for making me a better person. I would like to thank all my relatives and friends who had supported me over the years.
Finally, I thank God for everything. For what I had done wrong and what I had failed to do, he will also lead me to the right path, and I shall put my trust and faith in him for as long as I am alive.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Real-world economic and financial processes produce observable outcomes that have often been referred as data or information. In nature, such observed information can be discrete (e.g., yearly economic growth rate, monthly inflation rate, etc.) or continuous (e.g., stock prices, exchange rate, etc). The information may be generated from a single source, or corrupted with noise or transformed or indexed from other sources, etc. All these possibilities spark a fascinating underlying problem regarding the characteristics and the behaviour of such real-world information processes. Hidden Markov models or general state space models [1] consist of a Markov chain that is hidden and an observable process that depends on the Markov chain, which have been the most popular choices among other models in modelling observed economic and financial information processes in recent years.

During the 1980s, the Kalman filter of Kalman [1960] has gradually become an extremely popular tool in estimating various economic models that are specified in the form of linear and Gaussian state space model. Such prevalence was largely due to the endeavours of people like Anderson and Moore [1979], Harvey and Phillips [1979], Stock and Watson [1988], Pollock et al. [1993], Hamilton [1994], among others. However, the applicability of Kalman filter has been confined to linear and Gaussian state space model. In recent years, the hidden Markov models or general state space models have been thought of being more realistic and appropriate than the linear and Gaussian state space model in modelling real economic and financial processes. Earlier attempts in estimating general state space models were based on the Kalman filter and related techniques such as: the extended Kalman filter, the unscented Kalman filter, and the grid method. However, all these methods have not

[1] A word of clarification would be that the non-linear and non-Gaussian state space models have been referred to as the general state space in the Statistics literature.

been able to live up to our expectations of accuracy and simplicity. For instance, the extended Kalman filter depends on local linear approximation, which can lead to large approximation error when the function is highly non-linear; the unscented Kalman filter relies on excessive normal approximations, where its performance can be degraded substantially when the state space is non-Gaussian; the grid method requires $O(TG^2)$ computation, where G is the grid and T is the time length, and it requires G to be sufficient large to get a good approximation to the state space. To overcome such problem of estimating general state space models, a computational based simulation technique called the sequential Monte Carlo methods or particle filters have been established and widely implemented in estimating the hidden state variables, as well as unknown parameters within the model.

Nearly 20 years on, since particle filters were firstly established as an alternative to the aforementioned Kalman based methods, they have been able to fulfil our expectation of improving estimation accuracy to certain extend where the previous Kalman based methods have been unsuccessful. Moreover, their applicability in the fields of statistics and engineering was largely the strive of Gordon et al. [1993], Doucet et al. [2001], Crisan and Doucet [2000], Godsill et al. [2001], among others. On the contrary, the particle methods have become a very useful tool in economics and finance through the work of Pitt and Shephard [1999], DeJong and Dave [2006], Fernandez-Villaverde and Rubio-Ramirez [2007], and Pitt et al. [2012]. Envisaging the particle filters literature up to date, the objective of this thesis is to establish and develop more accurate and informative particle filtering and particle smoothing techniques that better serves the analysis purpose in economics and finance. More precisely, the thesis is to suggest a set of improved methods that enable us to handle the estimation of the state variables and population parameters within the general state space models in less computational and more accurate manner. However, before heading towards the main discussions, the following sections introduce and characterize the features of the general state space model, as well as the estimation tasks that will be involved with such type of model.

1.1 Model Specification

Developing estimation methods for learning the hidden state variables and the unknown parameter associated with general state space models have been receiving vast amount attention in the past two decades, for example, Gordon et al. [1993], Durbin and Koopman [2000], Arulampalam et al. [2002], Poyiadjis et al. [2005], among others.

The attraction towards the employment of general state space models in modelling economic and financial time series has been down to their unique characteristics in deriving knowledge from observed information. More specifically, general state space model derives a relationship through linking the hidden variables to observed information, where these hidden variables have been referred as the state variables. The relationship is said to be the key to understand the underlying behaviour of observed processes such as: exchange rate, stock prices, inflation, and many more.

The general state space model has often been referred to as Hidden Markov models in the statistic literature. The mainstream definition of the state space model is to assume that both stochastic processes $\{X_t\}_{t\geq 1}$ and $\{Y_t\}_{t\geq 1}$ are defined on a measurable space $(\Omega, \mathcal{F})$. These stochastic processes depend on parameter $\theta \in \Theta$, for Θ is a subset of $\mathbb{R}^r$, and r denotes the dimension of the parameter space. [2] The process $\{X_t\}_{t\geq 1}$ has been assumed to be hidden or unobserved Markov process with transition density $f_\theta(x'|x)$, that is:

$$X_t|X_{t-1} = x_{t-1} \sim f_\theta(.|x_{t-1}). \tag{1.1}$$

Note that the initial density of the hidden process $\{X_t\}_{t\geq 1}$ is known as $X_1 \sim \mu_1$. Despite that the process $\{X_t\}_{t\geq 1}$ is unknown, it is assumed to be partially observed through the observation process $\{Y_t\}_{t\geq 1}$. Moreover, the conditional density of the observation at time instance t given X_t is:

$$Y_t|X_t = x_t \sim g_\theta(.|x_t). \tag{1.2}$$

A more familiar specification of the general state space model in the stream of the economics literature consists of the following functional forms of the transition equation and the observation equation, such as

$$X_t = \psi_\theta(X_{t-1}, V_t, \theta) \quad \text{and} \quad Y_t = \phi_\theta(X_t, W_t, \theta), \tag{1.3}$$

where $\{W_t\}_{t\geq 1}$ and $\{V_t\}_{t\geq 1}$ are mutually i.i.d. noise sequences. ϕ_θ and ψ_θ are possible non-linear functionals that determine the evolution of the hidden state and observation processes [3].

In the thesis, one of the crucial contributions is the estimation of stochastic volatil-

[2] Loosely speaking, the dimension is simply the number of parameters within the model specification.

[3] Note that the definitions of technical terms such as: stochastic process, Markov process, transition density, along with the subsequent particle filtering, particle smoothing, will be defined and explained in chapter 2

ity (SV) models by using newly developed techniques of particle filtering and particle smoothing. Wherein SV models are prototype state space models, which have been widely applied to model time series such as stock price, market index, and exchange rate. SV models assume that the hidden logarithm of volatility or variance $\{X_t\}_{t\geq 1}$ of the observed time series follows a Markovian process, and has a probabilistic structure $f_\theta(x_t|x_{t-1})$. Moreover, the observed information or data of interest, $\{Y_t\}_{t\geq 1}$ has been assumed to be governed by a probability model $g_\theta(y_t|x_t)$. See Shephard [1996] and Durbin and Koopman [2000] for more detailed discussion. One of the simplest univariate SV model, where certain amount simulation studies in the thesis will be based upon, can be expressed through the following two equations:

$$X_t = \theta_1 X_{t-1} + \theta_2 V_t, \tag{1.4}$$

$$Y_t = \theta_3 \exp\big(X_t/2\big) W_t, \tag{1.5}$$

where V_t and W_t can be assumed to be i.i.d. standard normal with mean 0 and variance 1, and $X_1 \sim \mathcal{N}(\mu_1, \sigma_1^2)$. Equations (1.4) and (1.5) have often been referred as the *state equation* and the *observation equation* respectively. The above transition equation that induces the volatility at time t depends on a linear function of previous time period volatility. Suppose that the exchange rate or the market index is denoted by e_t, then the series of interest (as in the observation equation) is $Y_t = \Delta \log(e_t)$ for $t = 1, \cdots, T$. [3] The stochastic process X_t is indeed the logarithm of variance of the exchange rate or the market index. Modification can be made on the transition (state) equation of the model, which would permit the hidden volatility to take into account more realistic and versatile situations of the hidden volatility, such as non-linear structure and additional information. An example that includes both cases is the following discrete version of the re-parametrized continuous time Cox-Ingersoll-Ross model in Poyiadjis et al. [2011], which takes the following form:

$$X_t = \mu + X_{t-1} + \phi \exp(-X_t) + \exp(-X_t/2) V_t, \tag{1.6}$$

$$Y_t = \beta \exp\big(X_t/2\big) W_t. \tag{1.7}$$

In the transition equation (1.6), the non-linearities have been introduced through the exponential term, and the additional information is μ, which represents the mean reversion value.

In the thesis, I also consider the Phillips curve model in the general state space format. In this model, the inflation rate corresponds the observation process $\{Y_t\}_{t\geq 1}$

[3] Δ denotes the difference operator, e.g. $\Delta \log(e_t) = \log(e_t) - \log(e_{t-1})$.

and the unobserved natural rate unemployment corresponds the hidden state process $\{X_t\}_{t\geq 1}$. In addition, this model can be seen as a very simple dynamic macroeconomic model with fewer parameters.

1.2 Estimation of the Model

The estimation of the general state space involves obtaining the estimate of hidden state sequence $\{X_t\}_{t\geq 1}$ and making inference about the unknown population parameters θ. For general state space models where learning the state sequence is the only task, then such scenario has been referred as the general state space model with known parameters. On the contrary, general state space models with the task of learning both state sequence and unknown parameters has been referred as general state space models with unknown parameters. The conventional assumptions this type of models suggest that the estimation of both state variables and unknown parameters are closely tied to one another. In this section, I elaborate on the difficulties associate with the estimation of the general state space model.

1.2.1 General State Space Model with known Parameters

By state inference, it is referring to the estimation of the hidden state variables or sequence $\{X_t\}_{t\geq 1}$. Hereby I explain the procedure of obtaining the estimate of state variables when the parameters are known in the general state space model. Two very different approaches can be adopted to complete such task, they are: particle filtering and particle smoothing. With particle filtering, it says that provides the set of available observations $\{Y_t\}_{t\geq 1}$ up to time instance t, then the estimates of $\{X_t\}_{t\geq 1}$ can be obtained through the marginal density function $p_\theta(x_t|y_{1:t})$. Where the estimation of the marginal density $p_\theta(x_t|y_{1:t})$ is in fact by its empirical density that obtained through particle samples. On the contrary, particle smoothing is saying that given the full observations set is available, e.g. $\{Y_{1:T}\}$, then the estimate of state variable can be obtained from the marginal smoothing density of $p_\theta(x_t|y_{1:T})$. The estimation of particle smoothing carries out in a backward fashion instead of the forward fashion with the particle filtering. The full extended discussions on particle filtering and particle smoothing have been placed in chapter 2 and 5.

1.2.2 General State Space Model with unknown Parameters

The above discussed state sequence estimation was under the assumption of knowing the value of parameters in the model. However, in many real world time series utilizing the general state space model, such a condition is impossible to fulfil. Therefore, the estimations of general state space models are: obtaining the estimate of state sequence and making inference of unknown parameters. This is a much more complicated problem than the one with merely state inference given the parameters are known. One of the complications lies within the order of estimation. By the order of estimation, it means whether one should firstly estimate the parameters then the state sequence; or the vice-versa; or even at the same time. Despite the various attempts that have been made in the past to handle such difficulty, inadequacies remain either in estimation accuracy and computational timing. Kantas et al. [2011] provided a survey on the existing parameter estimation methods in general state space models with unknown parameters.

Chapter 4 and 5 study two different approaches for both state and parameter estimation, where these two approaches differ from the order of their estimation. In chapter 4, I perform a modification to the existing *Entropy particle filter* of Liverani and Papavasiliou [2006]. Such particle filters estimate firstly the unknown parameters, and then it performs the above state inference based on the estimate of parameters. In which the modified Entropy particle filter follows the order of firstly estimate the parameters then the state sequence. On the contrary, chapter 5 investigates the approach of combining Expectation-Maximization (EM) algorithm with particle techniques for 'off-line' estimation. By off-line, it means that the observations are processed as a batch during the estimation. Such an approach estimates the parameters and the state sequence in an iterative fashion. For example, given the target parameters are $\theta = \{\theta_1, \theta_2, \theta_3\}$ in the SV model of Harvey et al. [1994] and Durbin and Koopman [2000](that defined by equations (1.4) and (1.5)) and $\theta = \{\mu, \phi, \beta\}$ in the Cox-Ingersoll-Ross model (that defined by equations (1.6) and (1.7)), the EM-particle smoothing carries out the estimation in the following fashion: firstly imposes some initial values for the set of parameter, namely θ^0, it then allows to estimate the state sequence $\{X_{t|T}\}_{t\geq 1}$ by particle smoothing. Subsequently, the estimate of $\{X_{t|T}\}_{t\geq 1}$ is utilized to compute a new set of parameter values, θ^1, which allow us to repeat the particle smoothing step. The iteration stops once the convergence is reached.

1.3 Thesis Outline

This thesis is structured as follows: to make the thesis as self-contained as possible, chapter 2 lays the fundamental knowledge of state space model, particle filtering, and particle smoothing. Subsequently, chapter 3 develops a novel degeneracy diagnostics, namely the Shannon information entropy diagnostics. This newly technique can be implemented as an alternative to existing degeneracy detection to all particle filters for as long as resampling procedure is used. In addition, it provides an empirical guide-line over the selection of existing resampling schemes for the application of particle filters. In chapter 4, I present modification on the Entropy particle filter for parameter estimation of the general state space model in Liverani and Papavasiliou [2006]. Within chapter 5, I developed a modified forward filtering and backward sampling smoothing method. In addition, the newly proposed smoothing method has been embedded into the EM algorithm for off-line parameter estimation in general state space models. Chapter 6 ends the thesis with a summary of the results and a prospect of future works.

1.4 Contribution of Thesis

The contribution of chapter 3 is the development of the Shannon information entropy diagnostics. This new diagnostics can be utilized in the same way as other widely applied diagnostics such as the effective sample size and the coefficient variation to determine the necessity of resampling in particle filtering. However, I show that Shannon information entropy diagnostics avoids over-resampling compare to existing diagnostics. The simulation evidences suggest that particle filter utilizing the Shannon information entropy diagnostics provides at least as good precision as the particle filter with the effective sample size diagnostics. However, the resampling percentage of particle filter with the Shannon information entropy diagnostics is lower than that of with the effective sample size diagnostics. The Shannon information entropy diagnostics presented in chapter 3 is widely applicable and conforms to the attendant reductions in computational cost. These two aspects are in-line with the two of the five paradigms that were made by Durham and Geweke [2012] on their work of sequential posterior simulators for Bayesian inference.

The contribution of chapter 4 can be summarised as follows: looking beyond the estimation of state process into the more formidable parameter inference within general state space models, I modified the Entropy particle filter of Liverani and Papavasiliou [2006] by introducing additional refinement procedure. The simulation

evidences suggest that the modified entropy particle filter demonstrates greater improvement over the estimation of both state process and parameter which is by the original entropy particle filter. The modified entropy particle filter is very attractive and particularly useful in parameter estimation for general state space models.

Typically, conventional particle smoothing algorithms will have computational complexity of $O(TN^2)$. Such fact has been one of the major drawbacks that had prevent particle smoothing from being extensively applied in reality. However, the forward filtering and backward sampling smoothing of Godsill et al. [2004] (has often been referred as the GDW smoothing algorithm) has computation complexity of $O(TN)$. In chapter 5, I incorporate modification steps within the GDW smoothing algorithm, where the modification takes into account of the backward smoothing weights prior to the resampling step in the GDW smoothing. In the thesis, for the sake of simplicity, I shall refer the modified algorithm as the modified GDW (MGDW) smoothing algorithm. The MGDW smoothing also has the computational complexity of $O(TN)$. In addition, I show that through simulation studies, the MGDW smoothing performs at least as good as any existing smoothing techniques.

One final contribution of the thesis to the particle filtering literature is the extended work of a novel parameter estimation method for general state space models. Such estimation method brings up two existing, yet quite different techniques together: EM algorithm and particle techniques, to carry out the off-line estimation in the dynamic system. This work extends the novel estimation method by utilizing the Shannon information entropy diagnostics of chapter 3 in particle filtering and employing the aforementioned MGDW smoothing, which forms the EM-MGDW method in the thesis. Such formalization builds a valid estimation techniques that make the EM algorithm and particle techniques to become feasible for real data applications. The method of EM-MGDW smoothing method demonstrates great feasibility in off-line parameter estimation among general state space models. Moreover, I have successfully applied this method to estimate the parameters of the non-linear Phillips curve model and the stochastic volatility models. In the meantime, the implementation of this method will not be effected by the number of unknown parameters.

Chapter 2

Backgrounds

Abstract

Particle filters have been widely applied to filter state sequences and estimate unknown parameters for Hidden Markov models or general state space models. But within economics, the usage of particle filters has only begun recently. The objective of this tutorial is twofold: firstly, to provide a self-contained introduction on the basics of particle filtering and particle smoothing. Then to demonstrate the attractiveness of particle methods in the estimation of general state space models.

2.1 Introduction

This chapter provides a tutorial review of the Kalman filter, Particle filters, and of related matters, which aims to make this thesis as self-contained as possible.

Markov chain is a stochastic process in which the next state depends only on the current state. Markov chains have many applications as statistical modelling tool of real-world observation processes, and it is the single most important background information of the hidden Markov models or general state space models that we will be considering in this thesis.

The Kalman filter has been renowned for its ability to estimate linear and Gaussian state space models, wherein such models are in fact a special type of the general state space model. Most of all, a leading application of the Kalman filter (Kalman [1960]) of the 1960s was in aerospace engineering, when it was used by NASA to guide the Apollo mission on its journey to the moon. Econometricians have been much slower to adopt the Kalman filter than statisticians. During the 80s and 90s, the Kalman filter has become a device for constructing the observed likelihood function of time series being modelled in state space format, which subsequently allows to estimate the population parameters within the dynamic system. A recapitulation of the Kalman filter and of Kalman based filters will provide a comparable basis for the discussion of particle filters. It is the flexibility of particle filters and their ability to cater to non-linear and non-Gaussian models or general state space models that accounts for their importance in the field of economics. The discussion of the Kalman filter and the Kalman smoothing follow very closely to Pollock et al. [1993]. However, I provide further derivations (that were not shown in Pollock et al. [1993]) that need to arrive at the Kalman filter.

The advantage of particle filters, in comparison with other approximation methods, such as the extended Kalman filter, is that, when dealing with the estimation of general state space models, they do not require troublesome local linearisation or functional approximations. Ever since the development of the bootstrap filter by Gordon et al. [1993], generic particle filters have been continuously refined and developed. The developments have come in from two streams. In one stream have been the order of *firstly sampling, then resampling*, and the representatives of such stream is the bootstrap filter Gordon et al. [1993] among others. The other stream has been the development on the order of *firstly resampling then sampling*, where the auxiliary particle filter of Pitt and Shephard [1999] have been the first and foremost filter that follows such order. See Lopes and Tsay [2011] and Creal [2012] for an extensive review of the use of particle filters in econometrics.

Particle smoothing tends to be more computationally challenging than particle filtering. Smoothing consists of estimating the distribution of the state sequence at a particular time given the observations beyond that time. Therefore, given the additional information, one would expect that the trajectory estimates of the state sequence obtained via smoothing will be 'smoother' to those obtained by the particle filter alone. The most widely applied particle smoothing techniques are: the forward filtering and backward smoothing by Kitagawa [1987], the two-filter smoothing formula by Bresler [1986], and the generalized two-filter algorithm by Briers et al. [2010]. The two-filter smoothing combines a forward filter with a backward information filter, where the difference between the two filter smoothing techniques and the generalized two-filter algorithm lies in the construction of the backward information filter.

This chapter is organized as follows: section 2 defines and explains concepts such as: stochastic process, Markov chain, and general state space models. The derivation of the Kalman filter, the construction of likelihood using the prediction error from Kalman filtering, and various smoothing algorithms are discussed in section 3. In section 4, we review the idea of importance sampling and sequential importance sampling. Finally, section 5 introduces the conventional particle smoothing techniques such as: the forward filtering and backward smoothing, the two-filter smoothing formula, and the generalized two-filter algorithm.

2.2 State Space Models

In this section, I introduce the concept of stochastic process and discuss Markov chain and their related features.

2.2.1 Stochastic processes

A familiar example of a stochastic process in continuous time would be the price of a stock recorded continuously in seconds throughout the trading day. On the contrary, an example of a stochastic process in discrete time would be the sequence of temperatures recorded at 9 o'clock every morning at the University of Leicester in Leicester. A formal definition can be given as follows

Definition 2.2.1 (Stochastic process)**.** *A stochastic process X is a family $\{X_t : t \in T\}$ of random variables $X_t : \Omega \to S$. T is called the time index set and S is called the state space.*

In this thesis, I focus on stochastic processes in discrete time, which assumes

$T \subset \mathbb{N}$ or $T \subset \mathbb{Z}$ in the above definition. On the other hand, processes in continuous time is where $T = [0, \infty)$ or $T = \mathbb{R}$.

In the thesis, the term of *random variable* will be repeatedly employed. One can think of it as if there is a set Ω representing the outcomes of a random experiment that can be observed by means of various measurements. These measurements assign numbers to the outcomes, thus the notion of random variable is a function or act to capture such procedure. Readers can refer to Capinski and Kopp [2005] for a systematic treatment on knowledge of measure and probability theory.

Definition 2.2.2 (sample path)**.** *For a given event or outcome $\omega \in \Omega$ the collection $\{X_t(\omega) : t \in T\}$ is called the sample path of X at ω.*

Note that for discrete time of $T \subset \mathbb{N}$, then the sample path is a sequence; for continuous time $T = \mathbb{R}$, the sample path is a function from $\mathbb{R}$ to S. The distintion between processes is not merely restricted on their time index T, but also on state space S. A special case would be that the state space S is defined as a countable set, and then X is then called *discrete state space process* or *discrete process*. Hence a list of type of sample paths that we may refer to in the thesis are: sample path of a discrete process in discrete time, sample path of a discrete process in continuous time, and sample path of a continuous process in continuous time. The above example of the observed stock price of the trading day can be seen as sample path of a continuous process in continuous time; whereas the example of daily temperature can be seen as sample path of a discrete process (for $S \subset \mathbb{T}$) in discrete time.

2.2.2 Markov chains

Markov chain is a special kind of stochastic process. The evolution of the process in the future depends merely on the present and not on where it has been in the past. In result of that the applications of particle methods in the thesis aim to estimate dynamic models of continuous economic and financial information processes in discrete time. We therefore restrict our attention to general state spaces in discrete time. By general state space, it means that the state space $S = \mathbf{R}$.

Definition 2.2.3 (Markov chain)**.** *Let X be a stochastic process in discrete time with general state space S. X is called a Markov chain if X satisfies the Markov property of*

$$P(X_{t+1} \in A | X_0 = x_0, \cdots, X_t = x_t) = P(X_{t+1} \in A | X_t = x_t),$$

for all measurable sets [1] $A \subset S$.

The above definition states that the process depends on the past only through the present. More specifically, given the knowledge to the current state X_t, then the next state X_{t+1} is independent of the past states $X_0, \cdots, X_{t-1}$. Note that if S is countable, e.g. $S \subset \mathbb{N}$, then the preceding definition of general state space Markov chain becomes the definition of discrete version of Markov chain.

This thesis deals merely with the homogeneous Markov chains, which refers to the distributions $P(X_{t+1} \in A | X_t = x_t)$ that are independent of time t. The term of *transition kernel*, which will be frequently mentioned throughout the thesis, describes the probability of the Markov chain moves between states, which is defined as:

$$P(X_{t+1} \in A | X_t = x_t) = \int_A K(x_t, x_{t+1}) dx_{t+1}, \tag{2.1}$$

where the integration is with respect to a suitable dominating measure.[2] The following is an example of such a process.

Example 2.2.2.1 (Random walk)**.** *Suppose the random walk takes the following form*

$$X_{t+1} = X_t + \epsilon_{t+1}, \tag{2.2}$$

where ϵ_{t+1} *is standard normally distributed with mean* 0 *and variance* 1*, and the density function takes the form of* $\phi(z) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{z^2}{2})$*. Let* $X_1 \sim \mathcal{N}(0, 1)$*, then the transition process of* X_t *can be described via*

$$X_{t+1} | X_t = x_t \quad \sim \mathcal{N}(x_t, 1).$$

The following discussion induces that X_t *is a Markov chain,*

$$\begin{aligned} P(X_{t+1} \in A | X_t = x_t, \cdots, X_0 = x_0) &= P\Big(\epsilon_{t+1} \in (A - x_t)\Big) = P(X_{t+1} \in A | X_t = x_t) \\ &= \int_A \phi(x_{t+1} - x_t) dx_{t+1}. \end{aligned}$$

[1] The definition of measurable set is that let sample space Ω be a nonempty set, and $\mathcal{F}$ a σ-algebra over Ω, then the sets in $\mathcal{F}$ are called measurable sets.

[2] Suppose that υ and μ are measures on a measurable space $(\Omega, \mathcal{F})$, we say that υ is absolutely continuous with respect to μ if $\mu(A) = 0$ implies $\upsilon(A) = 0$ for $A \in \mathcal{F}$. We write this as $\upsilon \ll \mu$. Moreover, we say that the measure μ dominates υ when $0 \leqslant \upsilon(\mathcal{F}) \leqslant \mu(\mathcal{F})$.

Thus the transition kernel is

$$K(x_t, x_{t+1}) = \phi(x_{t+1} - x_t) = \frac{1}{\sqrt{2\pi}} \exp\Big\{ - \frac{(x_{t+1} - x_t)^2}{2} \Big\}.$$

In order to obtain the m-step transition kernel, according to equation (2.2), we can write X_{t+m} as follows

$$\begin{aligned} X_{t+m} &= X_{t+m-1} + \epsilon_{t+m} \\ &= X_{t+m-2} + \epsilon_{t+m-1} + \epsilon_{t+m} \\ &= \vdots \\ &= X_t + \epsilon_t + \cdots + \epsilon_{t+m}. \end{aligned}$$

Thus, $X_{t+m}|X_t = x_t \sim \mathcal{N}(x_t, m)$ as ϵ_t are assumed to be i.i.d. Therefore, if we standardize $X_{t+m}|X_t = x_t$ to be of standard normal, the probability distribution will be

$$\begin{aligned} P(X_{t+m} \in A | X_t = x_t) &= P(X_{t+m} - x_t \in A - x_t) \\ &= \int_A \frac{1}{\sqrt{m}} \phi\left(\frac{x_{t+m} - x_t}{\sqrt{m}} \right) dx_{t+m}, \end{aligned}$$

and the m-step transition kernel is therefore

$$K(x_t, x_{t+m}) = \frac{1}{\sqrt{m}} \phi\Big(\frac{x_{t+m} - x_t}{\sqrt{m}} \Big) = \frac{1}{\sqrt{2\pi}} \frac{1}{\sqrt{m}} \exp\Big\{ - \frac{(x_{t+m} - x_t)^2}{2m} \Big\}.$$

To define an ergodic Markov chain (as it will be used in Chapter 4), we need to outline the definitions of irreducibility and recurrence.

Definition 2.2.4 (Irreducibility)**.** *Given a distribution μ on the states S, a Markov chain is said to be μ-irreducible if for all sets A with $\mu(A) > 0$ and for all $x \in S$, there exists a $m \in \mathbb{N}$ such that*

$$P(X_{t+m} \in A | X_t = x) = \int_A K^m(x, y) dy > 0.$$

Irreducible is an important feature when we analyse the limiting behaviour of the Markov chain. The above definition says that the Markov chain will visit any state with a positive probability. In other words, A Markov chain is called irreducible if it only consists of a single class, i.e. all states communicate.

Definition 2.2.5 (Recurrence)**.** *(a) A set $A \subset S$ is said to be recurrent for a Markov*

chain X if for all $x \in A$

$$\mathbb{E}(V_A|X_0 = x) = +\infty,$$

where V_A denotes the number of time of visiting the state in set A.
(b) A Markov chain is said to be recurrent, if

- *The chain is μ-irreducible for some distribution μ.*
- *Every measurable set $A \subset S$ with $\mu(A) > 0$ is recurrent.*

For general state spaces as we are considering here, we need to consider the number of visits to a set of states rather than single state. The above definition emphasizes that a set is recurrent if on average it is visited infinitely often. In other words, there is a non-zero probability of visiting such a set infinity times. Checking recurrence of a Markov chain can be difficult, however a well-known proposition states that if a Markov chain is irreducible and has a unique invariant distribution, then this chain is also recurrent. Finally, an important point that we will be needed is as follows: a Markov chain on a state space S is ergodic, if it is irreducible and positive recurrent.

2.2.3 General State Space Models

The general state space model has been also known as the Hidden Markov models (HMMs), which consists of a Markov chain that is hidden and an observable stochastic process that depends on the Markov chain. More precisely, the model comprises a Markov chain $\{X_t\}_{t\geq 1}$, where the Markov chain is linked to another observable stochastic process $\{Y_t\}_{t\geq 1}$, which says that X_t governs the distribution of the corresponding Y_t. The process Y_t is said conditional on X_t that is independent from its past $\{Y_{t-1}, \cdots, Y_0\}$. The following definition is from chapter 1 of Cappe et al. [2005]:

Definition 2.2.6. *Consider an $\mathcal{X}$-valued discrete-time Markov chain $\{X_t\}_{t\geq 1}$, such that $X_1 \sim \mu(x_1)$ and $X_t|(X_{t-1} = x_{t-1}) \sim f(x_t|x_{t-1})$. The estimation of $\{X_t\}_{t\geq 1}$ rely on observable $\mathcal{Y}$-valued process $\{Y_t\}_{t\geq 1}$. We assume that given $\{X_t\}_{t\geq 1}$, the observations $\{Y_t\}_{t\geq 1}$ are statistically independent and their marginal densities with respect to a dominating measure λ are given by $Y_t|(X_t = x_t) \sim g(y_t|x_t)$. Then such model is called the hidden Markov model.*

Once the reality problems can be boiled down to a framework that can be specified by the general state space, then our aims would be that given some observations, whether we can estimate hidden sequence of states, as well as the unknown population

parameters (if the model is parametrized by both states variable and unknown parameters). Examples of general state space models appeared frequently in economics and finance literature are presented in the following.

Example 2.2.3.1 (Linear and Gaussian State space models Pollock [2003]). *Let* $X_t \in \mathcal{X} = \mathbb{R}^p$, $Y_t \in \mathcal{Y} = \mathbb{R}^q$, $X_1 \sim \mathcal{N}\big(\mathbf{0}, \sigma_v I \sigma_v^{'}(I - F_t F_t^{'})^{-1}\big)$ *and*

$$\begin{aligned} X_t &= F_t X_{t-1} + \sigma_v V_t, \quad \textit{Transition Equation} \\ Y_t &= H_t X_t + \sigma_w W_t, \quad \textit{Observation Equation} \end{aligned} \tag{2.3}$$

where the disturbances V_t *and* W_t *are normally distributed with mean vector* $\mathbf{0}$ *and identity covariance matrix* $\mathbf{I}$ *respectively. Let's omit the time index* t*, and assume matrices* F, H, σ_v *and* σ_w *have appropriate dimensions, then we can obtain that the following densities functions respectively:*

$$\begin{aligned} f(x_t|x_{t-1}) &\sim \mathcal{N}(Fx_{t-1}, \sigma_v I \sigma_v^{'}), \quad \textit{Transition density} \\ g(y_t|x_t) &\sim \mathcal{N}(Hx_t, \sigma_w I \sigma_w^{'}), \quad \textit{Observation density.} \end{aligned}$$

The analytical solution (the Kalman filter) to the above model was given by Kalman [1960] and Kalman and Bucy [1961]. Such model has been widely used for target tracking and time series modelling. An extensive discussion of the linear Gaussian state-space model has been studied by Anderson and Moore [1979]. Finally, the derivation of the Kalman filter and its related are presented in the following section.

Stochastic volatility Model

In the modelling of share price, stock market index, or foreign exchange rate, we use the relative returns or the log-returns to describe the relative change over time of the price process. The variances of these types of return tend to change over time: the large and small values in the sample occur in clusters. More specifically, large changes at certain moments in time tend to be followed by large changes of either sign, and small changes tend to be followed by small changes, such phenomenon has been referred to as *volatility clustering*. The term *volatility* has been interpreted as *variance* in econometrics.

The autoregressive conditional heteroscedasticity (ARCH) model of Engle [1982] and the generalized ARCH of Bollerslev [1986] have been widely used to model the changes in volatility of financial and economic information processes. The stochastic volatility (SV) model is however an alternative approach to the ARCH/GARCH framework. It sets up a model containing an unobserved variance component, the

logarithm of which is specified directly as a linear or non-linear stochastic process. This model has appeared in option pricing and exchange rate modelling for the past two decades. The stochastic volatility model is a prototype general state space model. This thesis considers SV models of continuous process in discrete time. In the following example, we will see a stochastic model in its discrete time form, which has been briefly mentioned in equation (1.4) and (1.5) of the introduction.

Example 2.2.3.2. *The observations* $\{Y_t\}_{t\geq 1}$ *are the log-returns,* $\{X_t\}_{t\geq 1}$ *is the log-volatility, which is assumed to follow a stationary autoregressive of order* 1 *(AR(1)).*

$$\begin{aligned} X_t &= \theta_1 X_{t-1} + \theta_2 V_t, \qquad V_t \sim \mathcal{N}(0,1), \\ Y_t &= \theta_3 \exp(X_t/2) W_t, \qquad W_t \sim \mathcal{N}(0,1), \end{aligned} \tag{2.4}$$

where V_t *and* W_t *are i.i.d. sequences. The parameter* θ_3 *plays the role of the constant scaling factor,* θ_2 *is the volatility of the log-volatility, and* θ_1 *governs the persistence in the volatility. The* X_t *represents the logarithm of variance of the observation process* Y_t*, meanwhile,* Y_t *the process of interest has often been transformed from the operation of* $\Delta \log e_t$*, where* e_t *is the actual observation value such as exchange rate and stock price at time stance* t*, and the notation* Δ *indicates the difference operation. This model can further be transformed into a linear and Gaussian state-space models via log-transformation, for instance, taking logarithms of the squared relative returns, we have*

$$\begin{aligned} X_t &= \theta_1 X_{t-1} + \theta_2 V_t, \\ R_t = \log Y_t^2 &= \log \theta_3^2 + X_t + Z_t, \quad \textit{where} \quad Z_t = \log W_t^2. \end{aligned} \tag{2.5}$$

Note that since W_t *is standard normal,* Z_t *follows the* $\log \chi_1^2$ *distribution and the mean and variance of* Z_t *are known to be* -1.2749 *and* $\pi^2/2 = 4.93$*, respectively, see Harvey et al. [1994]. This model has been study by Durbin and Koopman [2000], who approximated the log-distribution by a finite mixture of Gaussian distribution, which allows the SV model becomes a conditionally linear and Gaussian state space model.*

Regime Switches in Econometrics

The Markov-switching autoregressive models have been used to characterize macroeconomic fluctuations. Such model was developed by Hamilton [1989], it provided a formal statistical representation of the idea of expansion and contraction that constitute two distinct economic phases within the study of business cycle. An improved version of Hamilton's model was discussed in the book by Kim and Nelson [1999],

which is described in the following example.

Example 2.2.3.3 (Linear Markov Switching model)**.** *We have* $\{s_t\}_{t\geq 1}$ *take values from a discrete space, e.g* $s_t \in \mathcal{S} \subset \mathbb{N}$*. For example we can denote economy in an expansion state as* $s_t = 1$ *and in a contraction state as* $s_t = 0$*. We assume* s_t *is a Markov chain such that* $p(s_1 = i) = \mu_1(i)$ *, and* $p(s_t = i|s_{t-1} = j) = f_s(i|j)$ *for* $i, j = \{0, 1\}$*. Suppose that we observe* Y_t *directly but can only make an inference about the value of* s_t *based on the realization of* Y_t*. The inference will be formed in a probability format, which is*

$$\xi_{it} = p(s_t = i|\mathcal{I}_t, X_t; \theta),$$

where θ *denotes the unknown parameters and* $\mathcal{I}_t = \{y_1, \cdots, y_t\}$*, and the hidden state variable* X_t *is* $X_t \in \mathcal{X} = \mathbb{R}^d$*, and the product* σ*-algebra* [3] *is defined as* $\mathcal{Z} = \mathcal{S} \times \mathcal{X}$*. Hence we have* $Z_t = (s_t, X_t) \in \mathcal{Z}$*. Conditional upon* s_t*, we could form the following model when* $X_1|s_1 \sim \mathcal{N}(0, \Sigma_{s_1})$*,*

$$\begin{aligned} X_t &= A_{s_t} X_{t-1} + B_{s_t} V_t, \quad V_t \sim \mathcal{N}(0, I_v) \\ Y_t &= D_{s_t} X_t + G_{s_t} W_t, \quad W_t \sim \mathcal{N}(0, I_w), \end{aligned} \tag{2.6}$$

where V_t *and* W_t *are mutually independent and Gaussian, and* A_{s_t}*,* B_{s_t}*,* G_{s_t}*, and* G_{s_t} *are coefficient matrices with appropriate dimensions. Provided with the above information, we have*

$$\begin{aligned} \mu(z) &= \mu(s, x) = \mu_s(s)\mathcal{N}(0, \Sigma_s), \\ f(z_{t+1}|z_t) &= f\big(s_{t+1}, x_{t+1}|s_t, x_t\big) = f_s(s_{t+1}|s_t)\mathcal{N}(A_{s_t}x_t, \quad B_{s_t}B_{s_t}^{'}), \\ g(y_t|z_t) &= g(y_t|s_t, x_t) = \mathcal{N}(D_{s_t}z_t, \quad G_{s_t}G_{s_t}^{'}). \end{aligned}$$

The inference on the above model can be conduct through the use of the Kalman filter. Further estimation on the desire state of the economy s_t can be computed through the computation of the observed likelihood of Y_t. The detailed formulas of that can be found in Hamilton [1989] and Hamilton [2005].

The state space approach provides significant benefits over traditional time series techniques for problems such as multivariate data and non-linear/non-Gaussian characteristic processes. In order to make inference about a dynamic system in state space model, it requires at least two models: which is the aforementioned transition equation and observation equation. Accordingly, the techniques conduct the inference

[3] A term in measure theory, it simply denotes the collection of all subsets of sample space Ω.

such as seeking filtered estimates of state variable X_t, which will be the discussion in the following sections.

2.3 The Kalman Filter

The linear and Gaussian state space model consists of two equations, which are

$$X_t = F_t X_{t-1} + \epsilon_t, \quad \text{for } \epsilon_t = \sigma_v V_t, \tag{2.7}$$

$$Y_t = H_t X_t + \eta_t, \quad \text{for } \eta_t = \sigma_w W_t, \tag{2.8}$$

where X_t and Y_t are the state vector and the observation vector at time stance t, respectively. The transition disturbance ϵ_t and the observation error η_t are assumed to be mutually uncorrelated random vectors, but normally distributed with zero mean vector and covariance or dispersion matrices

$$V(\eta_t) = \Omega_t \quad \text{and} \quad V(\epsilon_t) = \Phi_t. \tag{2.9}$$

Note that the notation of Ω_t is the covariance at time t, not the sample space Ω. The initial state $X_1 \sim \mathcal{N}(m_1, P_1)$ and the information set (filtration) at time t is defined as the set of observations $\mathcal{I}_t = \{y_1, \cdots, y_t\}$, where y_i are realisation of Y. The insights on the preceding model can be grasped as: the state variable satisfies certain stochastic equation with linear coefficients and Gaussian initial condition; and the observation equation satisfies an evolution of the equation with a linear function. Provided the state disturbance ϵ_t and the observation error η_t are i.i.d normally distributed but mutually independent from each other, then the inference on state variable given the observation up to time t is to learn the distribution of $p(x_t|\mathcal{I}_t)$. Furthermore, for the present problem the Kalman filter (as shown in the following section) derives the mean and covariance of such distribution.

2.3.1 Derivation of Kalman Filter

Before heading straight to the derivations of the Kalman filter, a few necessary definitions are outlined: the state-vector estimates $m_{t|t-1} = E(X_t|\mathcal{I}_{t-1})$ and $m_t = E(X_t|\mathcal{I}_t)$ and their associated covariance matrices $V(X_t - m_{t|t-1}) = P_{t|t-1}$ and $V(X_t - m_t) = P_t$, where $V(X_t)$ denotes covariance of random vector X_t. A summary of equations as-

sociated with the Kalman filter are listed as follows:

$$
\begin{aligned}
m_{t|t-1} &= F_t m_{t-1}, && (2.10)\\
P_{t|t-1} &= F_t P_{t-1} F_t^{'} + \Phi_t, && (2.11)\\
\rho_t &= Y_t - H_t m_{t|t-1}, && (2.12)\\
E_t &= H_t P_{t|t-1} H_t^{'} + \Omega_t, \quad \text{Error dispersion} && (2.13)\\
K_t &= P_{t|t-1} H_t^{'} E_t^{-1}, \quad \text{Kalman gain} && (2.14)\\
m_t &= m_{t|t-1} + K_t \rho_t, && (2.15)\\
P_t &= (I - K_t H_t) P_{t|t-1}. && (2.16)
\end{aligned}
$$

In the above list, equation (2.12) and (2.14) are merely definitions. To derive equation (2.10), we use equation (2.64) in Theorem (2.7.1) in the Appendix A to show that

$$
\begin{aligned}
E(X_t|\mathcal{I}_{t-1}) &= E\big\{E(X_t|X_{t-1})|\mathcal{I}_{t-1}\big\}\\
&= E\{F_t X_{t-1}|\mathcal{I}_{t-1}\}\\
&= F_t m_{t-1}. && (2.17)
\end{aligned}
$$

By equation (2.63) in Theorem (2.7.1) in the Appendix A, equation (2.11) can be obtained as follows

$$
\begin{aligned}
P_{t|t-1} &= V(X_t|\mathcal{I}_{t-1}) = V(F_t X_{t-1} + \epsilon_t|\mathcal{I}_{t-1})\\
&= \Phi_t + F_t P_{t-1} F_t^{'}. && (2.18)
\end{aligned}
$$

The covariance of prediction error, e.g as defined in equation (2.12) can be obtained with the following steps

$$
\begin{aligned}
E_t = V(\rho_t) &= V(Y_t - H_t m_{t|t-1}) = V(H_t X_t + \eta_t - H_t m_{t|t-1})\\
&= V(H_t(X_t - m_{t|t-1}) + \eta_t)\\
&= H_t P_{t|t-1} H_t^{'} + \Omega_t. && (2.19)
\end{aligned}
$$

To obtain equation (2.15), we begin by demonstrating the following expression that

$$\begin{aligned} C(X_t, Y_t|\mathcal{I}_{t-1}) &= E\Big\{\big(X_t - E(X_t|\mathcal{I}_{t-1})\big)\big(Y_t - E(Y_t|\mathcal{I}_{t-1})\big)'\Big\} \\ &= E\Big\{\big(X_t - m_{t|t-1}\big)(H_t X_t + \eta_t - H_t m_{t|t-1}\big)'\Big\} \quad \text{by eq (2.8) and (2.10)} \\ &= E\big\{(X_t - m_{t|t-1})(X_t - m_{t|t-1})' H_t'\big\} \\ &= P_{t|t-1} H_t' \end{aligned} \tag{2.20}$$

Now given equation (2.62) in Theorem (2.7.1) from Appendix A and equation (2.20), we have

$$\begin{aligned} m_t = E(X_t|\mathcal{I}_t) &= E(X_t|\mathcal{I}_{t-1}) + C(X_t, Y_t|\mathcal{I}_{t-1}) V^{-1}(Y_t|\mathcal{I}_{t-1})\{Y_t - E(Y_t|\mathcal{I}_{t-1})\} \\ &= m_{t|t-1} + P_{t|t-1} H_t' E_t^{-1} \rho_t \\ &= m_{t|t-1} + K_t \rho_t. \quad \text{by eq (2.14)} \end{aligned} \tag{2.21}$$

The dispersion of X_t given the information set $\mathcal{I}_t$ can be obtained by using equation (2.63) of Theorem (2.7.1):

$$\begin{aligned} P_t = V(X_t|\mathcal{I}_t) &= V(X_t|\mathcal{I}_{t-1}) - C(X_t, Y_t|\mathcal{I}_{t-1}) V^{-1}(Y_t|\mathcal{I}_{t-1}) C(Y_t, X_t|\mathcal{I}_{t-1}) \\ &= P_{t|t-1} - P_{t|t-1} H_t' E_t^{-1} H_t P_{t|t-1} \\ &= (I - K_t H_t) P_{t|t-1} \end{aligned} \tag{2.22}$$

The prediction errors or innovations $\{\rho_1, \cdots, \rho_t\}$ are mutually uncorrelated random variables, and there is a one-to-one linear relationship between the prediction errors and the observations $\{y_1, \cdots, y_t\}$ that forms the information set $\mathcal{I}_t$. Pollock et al. [1993] and Pollock [2003] have provided clear demonstrations of this point. In addition, the derivation of m_t and P_t are in fact the mean and covariance of the target density function $p(x_t|\mathcal{I}_t)$. In other words, the results we derived provide the optimal solution to the estimation of state variables problem if the highly restrictive linear and Gaussian assumptions hold.

2.3.2 Likelihood Function

In the linear-Gaussian state-space model, the Kalman filter can be used to construct the observed likelihood function. More specifically, given X_t, η_t, and ϵ_t are Gaussian, then the distribution of observations y_t conditional on x_t is Gaussian with mean and

variance that are given by equations (2.12) and (2.13), respectively

$$y_t|x_t, \mathcal{I}_{t-1} \sim \mathcal{N}\big(H_t m_{t|t-1}, \quad H_t P_{t|t-1} H_t^{'} + \Omega_t\big). \tag{2.23}$$

It is saying that

$$\begin{aligned} f_{Y_t|X_t,\mathcal{I}_{t-1}}(y_t|x_t, \mathcal{I}_{t-1}) \quad &= \quad (2\pi)^{-n/2}|H_t P_{t|t-1} H_t^{'} + \Omega_t|^{-1/2} \exp\{-\frac{1}{2}(y_t - H_t m_{t|t-1})^{'} \\ &\quad (H_t P_{t|t-1} H_t^{'} + \Omega_t)^{-1}(y_t - H_t m_{t|t-1})\}, \end{aligned}$$

where $m_{t|t-1}$ and $P_{t|t-1}$ are the mean and covariance of the predictive density $p(x_t|\mathcal{I}_{t-1})$, which has been derived and presented in equation (2.10) and (2.11). Note that t denotes the observation length, which is $t = 1, \cdots, T$, and n is the dimension of the observation vector Y_t. The joint log-likelihood can accordingly be expressed as

$$\sum_{t=1}^{T} \log f_{Y_t|X_t,\mathcal{I}_{t-1}}(y_t|x_t, \mathcal{I}_{t-1}). \tag{2.24}$$

This likelihood can also be derived via the density of the prediction error ρ_t for $t = 1, \cdots, T$. Equation (2.24) can be maximized numerically with respect to the unknown parameter matrices H, F, Ω and Φ. Provide reasonable initial values and the log-likelihood function is convex, then quadratic convergence of the estimates can be achieved with the use of Newton liked numerical methods.

The fact that the Kalman filter allows to form the observation likelihood function conveniently given the framework of the state space model to be linear and Gaussian, and then the subsequent estimation for the unknown parameters become handy. However, cautions should be taken during the estimation. Because in the absence of restrictions on H, F, Ω and Φ, the parameters of the state space representation are unidentified, and the discussion on un-identifications can be found in chapter 11 of Hamilton [1994]. Further discussion on the likelihood estimation in Kalman filter can be found in Pollock [2003].

2.3.3 Kalman Smoothing

This section reviews two of the most widely applied Kalman smoothing algorithms, they are: the classical smoothing algorithm and the de Jong's algorithm. These two algorithms belong to the class of fixed interval smoothing where econometricians have been particularly interested in. Their respective derivations in the proceeding section take the ground of conditional expectation. On the contrary, the Kalman forward-

backward smoothing algorithm has been formed from Bayesian stand-point. Given the Kalman forward-backward algorithm will not be discussed in here, and readers can refer to Pollock et al. [1993] for detailed derivations and discussions. In addition, readers who do not want to be distracted by tedious algebra derivations, can jump right through to the establishment of equation (2.34) and (2.35).

The idea behind fixed-interval smoothing is to revise each of the state estimates conditional upon the full information set $\mathfrak{I}_T$ when it has become available. Then the recursion of smoothing estimation takes the filtering estimates and runs from time T to 1 in a backward fashion. Given the sequence of the prediction errors $\{\rho_1, \cdots, \rho_T\}$ obtained from the Kalman filtering are mutually independent with zero expectations, then by equation (2.62) and (2.63) in Theorem (2.7.1) from the Appendix A and taking notice with their recursiveness, we can arrive at the following two expressions

$$E(X_t|\mathfrak{I}_T) = E(X_t|\mathfrak{I}_m) + \sum_{i=t+1}^{T} C(X_t, \rho_i)V^{-1}(\rho_i)\rho_i, \tag{2.25}$$

$$V(X_t|\mathfrak{I}_T) = V(X_t|\mathfrak{I}_m) + \sum_{i=t+1}^{T} C(X_t, \rho_i)V^{-1}(\rho_i)C(\rho_i, X_t). \tag{2.26}$$

Note that m in the equation (2.25) and (2.26) denotes the time index, which should not be confused with the estimate m_t. The term $V^{-1}(\rho_i)$ in both equations (2.25) and (2.26) is actually the covariance of the prediction errors E_t^{-1}. On the contrary, finding the generic covariance $C(X_t, \rho_i)$ requires more thought. In the following expression, through a recursive formula which allows to represent e_k in terms of $X_t - m_{t|t-1}$ and in terms of the state disturbances and observation errors which occur from time t. The prediction error at time t is therefore

$$\begin{aligned} \rho_t &= Y_t - H_t m_{t|t-1} \\ &= H_t(X_t - m_{t|t-1}) + \eta_t. \end{aligned} \tag{2.27}$$

A new expression of m_t can be obtained by substituting equation (2.12) into equation (2.15), which is

$$m_t = m_{t-1} + K_t(Y_t - H_t m_{t-1})$$

We replace Y_t in this newly expression above by equation (2.8) and lag it by one period, and then substitute it into equation (2.10), we have

$$m_{t|t-1} = \Lambda_t m_{t-1|t-2} + M_t(H_{t-1}X_{t-1} + \eta_{t-1}),$$

where $M_t = F_t K_{t-1}$ and $\Lambda_t = F_t(I - K_{t-1}H_{t-1})$. If we subtract equation (2.7) by the above expression, and then by running the recursion from it, we may deduce that

$$X_t - m_{t|t-1} = \Lambda_{t,q+1}(X_t - m_{t|t-1}) + \sum_{i=m}^{t-1} \Lambda_{m,j+2}(v_{i+1} - M_{i+1}\eta_j). \tag{2.28}$$

For $i \geq t$, follow by equations (2.27) and (2.28), we have

$$\begin{aligned} C(X_t, \rho_i) &= E\{X_t(X_t - m_{t|t-1})' \Lambda_{i,t+1}' H_i'\} \\ &= P_{t|t-1}\Lambda_{i,t+1}' H_i'. \end{aligned} \tag{2.29}$$

Given we know that $F_{t+1}P_t = \Lambda_{t+1}P_{t|t-1}$,[4] for $i > t$, then

$$C(X_t, \rho_i) = P_t F_{t+1}' \Lambda_{i,t+2}' H_i' \quad \text{and} \quad C(X_{t+1}, \rho_i) = P_{t+1|t}\Lambda_{i,t+2}' H_i' \tag{2.30}$$

Note that covariance matrix P is symmetric. Using equation (2.30), it arrives at the following equation

$$C(X_t, \rho_i) = P_t F_{t+1}' P_{t+1|t}^{-1} C(X_{t+1}, \rho_i). \tag{2.31}$$

If we substitute equation (2.31) into equation (2.25) for $m \geq t-1$, and with $V^{-1}(\rho_i) = E_i^{-1}$, then we obtain

$$\begin{aligned} E(X_t|\mathfrak{I}_T) &= E(X_t|\mathfrak{I}_m) + \sum_{i=m+1}^{T} P_{t|t-1}\Lambda_{i,t+1}' H_i' E_i' \rho_i \\ &= E(X_t|\mathfrak{I}_m) + P_{t|t-1}\Lambda_{t,t+1}' q_t. \end{aligned} \tag{2.32}$$

Similarly, equation (2.26) can be re-arranged as

$$V(X_t|\mathfrak{I}_T) = V(X_t|\mathfrak{I}_m) - P_{t|t-1}\Lambda_{m+1,t+1}' Q_t \Lambda_{m+1,t+1} P_{t|t-1}. \tag{2.33}$$

[4] It can be shown through simply algebra manipulations.

Note that the terms of q_t and Q_t are as follows, respectively

$$q_t = \sum_{i=t}^{T} \Lambda_{i,t+1}^{'} H_i^{'} E_i^{-1} \rho_i = H_t^{'} E_t^{-1} \rho_t + \Lambda_{t+1}{}' q_{t+1}, \tag{2.34}$$

$$Q_t = \sum_{i=t}^{T} \Lambda_{i,t+1}^{'} H_i^{'} E_i^{-1} H_i \Lambda_{i,t+1} = H_t^{'} E_t^{-1} H_t + \Lambda_{t+1}{}' Q_{t+1} \Lambda_{t+1}. \tag{2.35}$$

These recursions are set to be initiated with $q_T = H_T^{'} E_T^{-1} \rho_T$ and $Q_T = H_T^{'} E_T^{-1} H_T$. Both terms q_T and Q_T can be computed backwards until time stance reaches 1. Hence all our devoted efforts so far have been set to obtain equation (2.34) and (2.35), and we can now move on to form two of the previous mentioned smoothing algorithms.

Classic smoothing algorithm. If we replace the covariance term $C(.)$ in equation (2.25) with expression in (2.31), and with m set to t, yields

$$E(X_t|\mathfrak{I}_T) = E(X_t|\mathfrak{I}_t) + P_t F_{t+1}^{'} P_{t+1|t}^{-1} \sum_{i=t+1}^{T} C(X_{t+1}, \rho_i) V^{-1}(\rho_i) \rho_i. \tag{2.36}$$

Once more, by equation (2.25), we have,

$$E(X_{t+1}|\mathfrak{I}_T) = E(x_{t+1}|\mathfrak{I}_t) + \sum_{i=t+1}^{T} C(X_{t+1}, \rho_i) V^{-1}(\rho_i) \rho_i, \tag{2.37}$$

so it follows that equation (2.36) can be written as

$$E(X_t|\mathfrak{I}_T) = m_{t|T} = E(x_t|\mathfrak{I}_t) + P_t F_{t+1}^{'} P_{t+1|t}^{-1} \{E(x_{t+1}|\mathfrak{I}_T) - E(x_{t+1}|\mathfrak{I}_t)\}. \tag{2.38}$$

Performing similar procedure as above, the covariance of the smoothed estimate can be expressed as follows

$$V(X_t|\mathfrak{I}_T) = P_{t|T} = P_t - J_t \{P_{t+1|t} - P_{t+1|T}\} J_t^{'}, \tag{2.39}$$

where $J_t = P_t F_{t+1}^{'} P_{t+1|t}^{-1}$. Equations (2.38) and (2.39) represent the fixed interval smoother and dispersion respectively, and such an algorithm have been referred as the *classic smoothing algorithm*.

De Jong's algorithm. If we have $m = t - 1$, and given $\Lambda_{t,t+1} = I$, the above

equations (2.32) and (2.33) can be re-expressed as

$$E(x_t|\mathcal{I}_T) = m_{t|T} = m_{t|t-1} + P_{t|t-1}q_t, \tag{2.40}$$

$$V(x_t|\mathcal{I}_T) = P_{t|T} = P_{t|t-1} - P_{t|t-1}Q_tP_{t|t-1}. \tag{2.41}$$

The terms of q_t and Q_t are given by equations (2.34) and (2.35) respectively, and the new smoothing algorithm has been known as the de Jong's algorithm. Such an algorithm avoids a matrix inversion at each discrete time step, which should be more efficient than the above classical fixed-interval smoother. The above smoothing algorithms should theoretically provide the same result, but slight difference can occur due to the variations of starting value imposed to the starting position of each algorithms. Note that our discussion of Kalman filter has been following the work of Pollock et al. [1993], and similar but more comprehensive discussion can be found in Hamilton [1994] and Anderson and Moore [1979].

2.4 Particle Filters

In more general specifications of state space models, the Kalman filter will be inapplicable to the estimation of their state variables given the conditions of linear and Gaussian are no longer satisfied. Three non-linear filters such as: the extended Kalman filter (EKF), approximated grid-based methods, and particle filters have been developed to overcome the difficulty that had been prosed by non-linear and non-Gaussian state space models. For instance, the EKF utilizes the local linearisation to approximate the non-linearity functions in the model. However, since the EKF always approximates the target density function $p(x_t|\mathcal{I}_t)$ to be Gaussian, the performance will becomes poor if the true density in highly non-Gaussian, e.g. it is bimodal or multi-modal. On the contrary, the approximated grid-based methods take the decomposed cells of the predefined state space. However, sufficiently dense cells are needed to acquire a good approximation of the target density $p(x_t|\mathcal{I}_T)$, which will not be an ideal method for the estimation as the dimensionality of the state space increases.

Particle filters or *Sequential Monte Carlo* methods (SMC) provide the advantages of simplicity and precision over other previously mentioned methods in estimating general state space models. The applicability of the SMC exploits the idea of sequentially approximating intractable target densities by the corresponding empirical measure of these target density function $p(x_t|\mathcal{I}_t)$. In the following, it becomes clear that the empirical measure of target densities are formed through the samples (parti-

cles) that are generated from an importance density. Nevertheless, such method was not fully adapted in practice to estimate general state space models until the work of Gordon et al. [1993], their bootstrap filter was considered as a breakthrough in the area of target tracking and on-line estimation. However, development of the bootstrap filter was largely motivated by the works of *importance sampling* of Muller [1990] and Smith and Gelfand [1992]. The bootstrap filter is a combination of sequential importance sampling and resampling procedures, where generic filters sharing such resemblances have been branded with the name of Sequential Monte Carlo methods.

2.4.1 Derivation of Particle Filter

Part of the estimation task associated with general state space models is to recursively estimating the hidden state X_t at time t given the information $\mathcal{I}_t$ observed at that time. The process has been referred to as *filtering* estimation. One can achieve it by constructing an approximation to the probability density function $p(x_t|\mathcal{I}_t)$ at each time t. We outline the procedures of particle filtering in obtaining such density function recursively in the following.

Prediction Stage. Suppose the observations are available at time $t-1$, and $p(x_{t-1}|\mathcal{I}_{t-1})$ is known to us. Then this density function can be used to estimate the predicting density $p(x_t|\mathcal{I}_{t-1})$ in the following way

$$\begin{aligned} p(x_t|\mathcal{I}_{t-1}) &= \int p(x_t, x_{t-1}|\mathcal{I}_{t-1})dx_{t-1} \\ &= \int p(x_t|x_{t-1}, \mathcal{I}_{t-1})p(x_{t-1}|\mathcal{I}_{t-1})dx_{t-1} \\ &= \int f(x_t|x_{t-1})p(x_{t-1}|\mathcal{I}_{t-1})dx_{t-1}. \quad \text{by def (2.2.6)} \end{aligned} \tag{2.42}$$

From the penultimate expressions to the last line of equation (2.42) is nothing more than the Markov property.

Update State. By the Bay's theorem in the Appendix B and with the new observation y_t becomes available, we have

$$\begin{aligned} p(x_t|\mathcal{I}_t) &= \frac{p(y_t|x_t, \mathcal{I}_{t-1})p(x_t|\mathcal{I}_{t-1})}{p(y_t|\mathcal{I}_{t-1})} \\ &= \frac{g(y_t|x_t)p(x_t|\mathcal{I}_{t-1})}{\int g(y_t|x_t)p(x_t|\mathcal{I}_{t-1})dx_t} \quad \text{by def (2.2.6)} \\ &\propto g(y_t|x_t)p(x_t|\mathcal{I}_{t-1}). \end{aligned} \tag{2.43}$$

In the above equation, $g(y_t|x_t)$ plays the role of likelihood function and $p(x_t|\mathcal{I}_{t-1})$ is in fact the prior density function. Analytical solutions of the above model can only be computed for few special cases, e.g the Kalman filter for the linear and Gaussian state space model.

2.4.2 Sequential Importance Sampling

Sequential Monte Carlo (SMC) performs importance sampling sequentially to approximate the target distribution $p_t(x_t|\mathcal{I}_t)$ for $t = 1, \cdots, T$. More specifically, at time t, the set $\{x_t^i, \omega_t^i\}_{i=1}^N$ is a collection of samples (often referred to as *particles set*) and associated weights, characterizes the target posterior density $p_t(x_t|\mathcal{I}_t)$ as follows:

$$\hat{p}(x_t|\mathcal{I}_t) = \sum_{i=1}^{N} \omega_t^i \delta(x_t - x_t^i) \approx p(x_t|\mathcal{I}_t), \tag{2.44}$$

where $\delta(.)$ is Dirac delta function, which says $\delta(x_t - x_t^i) = 1$ when $x_t - x_t^i$, and $\delta(x_t - x_t^i) = 0$ when otherwise. The eminent tasks would be to provide answers to questions on where are the particles or samples $\{x_t^i\}_{i=1}^N$ coming from and how are the weights formed. A brief answer is that the particles are sampled from the so-called *importance density* $q(x_t|x_{t-1}, \mathcal{I}_t)$, and then weights are formed by the following expression

$$\omega_t^i \propto \omega_{t-1}^i \frac{g(y_t|x_t^i)p(x_t^i|x_{t-1}^i)}{q(x_t^i|x_{t-1}^i, \mathcal{I}_t)}, \tag{2.45}$$

where ω_{t-1}^i is the weight of i^{th} particle at time $t-1$. The above equation (2.45) is indeed the principle of importance sampling. It can be shown that as the number of particles $N \to \infty$, the estimate or empirical measure of the right hand side of equation (2.44) converges to the true target density $p(x_t|\mathcal{I}_t)$.

The sequential importance sampling algorithm (SIS) consists of recursive update of the particles and associated weights as each measurement received sequentially. A brief description of two of the important step are:

- Propagate step: for $i = 1 : N$, sample $x_t^i \sim q(x_t|x_{t-1}^i, \mathcal{I}_t)$.
- Update step: Subsequently, the weights at time $t+1$ can be obtained according to equation (2.45).

One final remark is that the SIS algorithm starts at $t = 1$, where the weights will be assumed to be $1/N$, and $x_1^i \sim q(x_1^i|x_0^i, \mathcal{I}_1)$, for $i = 1, \cdots, N$. Readers can refer

to Crisan and Doucet [2002] for the convergence of particle filters and Arulampalam et al. [2002] for the discussion on selection of importance density function.

2.4.3 Resampling

The applicability of particle filters was held back due to the so called *weight degeneracy* (growing variance) problem. The following example demonstrates the effect of such problem.

Example 2.4.3.1 (Relative variance). *Given the relative variance of importance sampling estimator takes the following form, as it is given by equation (26) of Doucet and Johansen [2008], which is*

$$\frac{\mathbb{V}_{IS}[\hat{Z}_t]}{Z_t^2} = \frac{1}{N}\left(\int \frac{\pi_t^2(x_{1:t})}{q_t(x_{1:t})} dx_{1:t} - 1\right),$$

where N is number of particles. In addition, $\hat{Z}_t$ denotes the importance sampling estimator and Z is the term of normalizing constant, and the target density is

$$\pi_t(x_{1:t}) = \prod_{i=1}^{t} \pi_i(x_i) = \prod_{i=1}^{t} \mathcal{N}(x_i; 0, 1).$$

Suppose the importance density to be

$$q_t(x_{1:t}) = \prod_{i=1}^{t} q_i(x_i) = \prod_{i=1}^{t} \mathcal{N}(x_i; 0, \sigma^2).$$

Considering time is at t, the relative variance is then

$$\begin{aligned}
\frac{\mathbb{V}_{IS}[\hat{Z}_t]}{Z_t^2} &= \frac{1}{N}\left[\int \frac{\left\{(2\pi)^{-t/2}\exp\left(-\sum_i \frac{x_{t,i}^2}{2}\right)\right\}^2}{(2\pi)^{-t/2}(\sigma^2)^{-t/2}\exp\{-\sum_i \frac{x_{t,i}^2}{2\sigma^2}\}} dx_{1:t} - 1\right] \\
&= \frac{1}{N}\left[\int (2\pi)^{-t/2}\exp\left\{-\sum_i x_{t,i}^2 + \sum_i \frac{x_{t,i}^2}{2\sigma^2}\right\}(\sigma^2)^{t/2}\right\} dx_{1:t} - 1\Big] \\
&= \frac{1}{N}\left[\left(\frac{\sigma^2}{2\sigma^2-1}\right)^{t/2}\int (2\pi)^{-t/2}\left(\frac{\sigma^2}{2\sigma^2-1}\right)^{-t/2}\exp\left\{-\frac{2\sigma^2-1}{2\sigma^2}\sum_i x_{t,i}^2\right\} dx_{1:t} - 1\right] \\
&= \frac{1}{N}\left\{\left(\frac{\sigma^2}{2\sigma^2-1}\right)^{t/2} - 1\right\}.
\end{aligned}$$

The above derivation assumed the posterior densities have been updated sequentially up to time t, which explains the reason why there is power of t that indicates the

multiplication of densities.

Now, the last line of the above expression indicates that the relative variance will increase exponentially with time t so long as $\frac{\sigma^2}{2\sigma^2-1} > 1$. For example, if $\frac{\sigma^2}{2\sigma^2-1} = 1.2$, for time $t = 100$, we would have

$$\frac{\mathbb{V}_{IS}[\hat{Z}_t]}{Z_t^2} \approx \frac{1}{N}(8.28 \times 10^7). \tag{2.46}$$

In other words, to make the relative variance 0.001*, one would need to employ* 8.3×10^{10} *particles. Clearly, this is impracticable, and resampling procedure limits such effect.*

According to the above example, the sequential importance sampling is destine to fail as t becomes large. However, in the literature of particle filters, two methods are proposed to limit the effect of weight degeneracy, they are selection of importance distribution and *resampling*. The later has been proven extremely effective in dealing with weight degeneracy in real applications in spite of the side effects might have been associated with its usage. One of those curial side effects is for instance, resampling reduces the diversity of the particles, and hence leads to paths degenerate.

In the following, we explain the idea of resampling procedure in SIS algorithm. Furthermore, we show the convergence of the SIS algorithm remain intact in spite of the introduction of resampling step. In each of the recursive steps of particle filtering, a set of particles $\{x_t^i, \omega_t^i\}_{i=1}^N$ allows us to construct an empirical measure $\hat{p}(x_t|\mathcal{I}_t)$ as defined in equation (2.44). Given the approximation of target distribution as $p(x_t|\mathcal{I}_t)$, suppose one's aim is to estimate $E_p\{(h(x)\}$ for $h(.)$ to be measurable, then, under suitable regularity conditions, Crisan and Doucet [2002] and others have shown that

$$\int_S h(x_t)\hat{p}(x_t|\mathcal{I}_t)dx_t = \sum_{t=1}^{N} \tilde{\omega}_t^i h(x_t^i) \rightarrow \int_S h(x_t)p(x_t|\mathcal{I}_t)dx_t = E_p(h(x)), \tag{2.47}$$

where $\tilde{\omega}_t^i$ is the normalized weight for i^{th} particle, that is

$$\tilde{\omega}_t^i = w_t^i / \sum_{j=1}^{N} w_t^j. \tag{2.48}$$

The convergence property implies that if our intention is to estimate the state variable $X_t = h(X_t)$, for h is an identity function, then the estimates would be the mean of particles $\{x_t^i\}_{i=1}^N$ that are assumed from the approximation density $\hat{p}(x_t|\mathcal{I}_t)$. However, the weight degeneracy problem limits the applicability of SIS algorithm. However,

the resampling algorithm perform the following: given the set $\{x_t^i, \omega_t^i\}_{i=1}^N$, one can draw an index $I = \{I_1, \cdots, I_N\}$ according to the normalized weights $\tilde{\omega}_t^i$, such that the sum of counts $\sum_{j=1}^N I_j = N$ (not necessary equal to N). A new particle set can be formed as $\{\tilde{x}_t^j, \bar{\omega}_t^j = 1/N\}_{j=1}^N$. It has been shown that, as $N \to \infty$, then

$$\frac{1}{N}\sum_{j=1}^{N} h(\tilde{x}_t^j) \to \sum_{t=1}^{N} \tilde{\omega}_t^i h(x_t^i). \tag{2.49}$$

Equation (2.47) and (2.49) together shows that the SIS filter with resampling still converges to the target density function. The intuition of resampling is essentially replicating particles carrying relatively large weight, which effectively produces sufficient amount of particles for the estimation of the target density. Note that particle filtering with resampling carried out at each time step has often been referred as SIS with resampling (SIR), its algorithm has been displayed in the following.

Algorithm 1: Sampling Importance Resampling.

1. At time $t = 1$,
   (a) Sample $x_1^i \sim q(x_1|x_0, \mathcal{I}_1)$.
   (b) Compute the weights $w_1^i = 1/N$.
2. At time $t > 1$
   (a) Sample $x_t^i \sim q(x_t|x_{t-1}^i, \mathcal{I}_t)$.
   (b) Compute the weights ω_t^i via equation (2.45), and normalize it to be
   $$\tilde{\omega}_t^i = \frac{\omega_t^i}{\sum_{i=1}^N \omega_t^i}.$$
   (c) Resample $\{x_t^i, \tilde{\omega}_t^i\}_{i=1}^N$ to obtain $\{\tilde{x}_t^j, \bar{\omega}_t^j = 1/N\}_{j=1}^N$.
   (d) go to $t + 1$, and then cease it at $t = T$.

2.4.4 Generic Particle Filter

Resampling schemes like systematic resampling or stratified resampling are operation that is at least with $O(N)$ computational cost. In addition, as it has been mentioned previously resampling imposes additional 'noise' due to the reduction of particles diversity. Hence one may want to avoid excessive resampling. In order to achieve that, it requires ways of measuring the level of particle degeneracy. In other words,

we execute resampling only when degeneracy is detected. One of the most widely implemented measure of degeneracy is the so called effective sample size (ESS) that introduced by Liu [1996], which is

$$N_{Ess} = \frac{N}{1 + V(w_t^{*,i})},$$

where $w_t^{*,i} = p(x_t^i|\mathcal{I}_t)/q(x_t^i|x_{t-1}^i, \mathcal{I}_t)$ is the "true weight". The small N_{Ess} indicates serious weights degeneracy. Since such weight cannot be computed exactly, an estimate of N_{Ess} has been proposed to be as

$$\hat{N}_{Ess} = \frac{1}{\sum_{i=1}^{N}(\tilde{\omega}_t^i)^2}, \tag{2.50}$$

where $\tilde{\omega}_t^i$ is the normalized weight obtained through equation (2.48). Note that the value of N_{Ess} will be either less or equal to N, which is $N_{Ess} \leq N$. Sampling importance resampling (SIR) is a generic term for particle filters possess sampling, resampling feature, moreover, the resampling step will be performed only when the $\hat{N}_{Ess}$ is less than certain threshold value N_{thre}.

Algorithm 2: Generic particle filter.

1. At time $t = 1$
   (a) Sample $x_1^i \sim q(x_1|x_0, \mathcal{I}_1)$.
   (b) Compute the weights $w_1^i = 1/N$.
2. at time $t > 1$
   (a) Sample $x_t^i \sim q(x_t|x_{t-1}^i, \mathcal{I}_t)$.
   (b) Compute the weights ω_t^i via equation (2.45), and normalize it to be

$$\tilde{\omega}_t^i = \frac{\omega_t^i}{\sum_{i=1}^{N}\omega_t^i}.$$

   (c) Calculate $\hat{N}_{Ess}$ via equation (2.50).
   (d) If $\hat{N}_{Ess} < N_{thre}$,
      i. Resample $\{x_t^i, \tilde{\omega}_t^i\}_{i=1}^N$ to obtain $\{\tilde{x}_t^j, \bar{\omega}_t^j = 1/N\}_{j=1}^N$.
   (e) Go to $t + 1$, and then cease it at $t = T$.

A list of particle filters have been utilized in economics and finance are: the boostrap filter of Gordon et al. [1993], the auxiliary particle filter of Pitt and Shephard [1999],

the Liu-West filter of Liu and West [2001], and the Rao-Balckwellized particle filter of Chen and Liu [2000]. Doucet and Johansen [2008] provide a tutorial on particle filtering and smoothing over their developments in the past two decades. In addition, Lopes and Tsay [2011] and Creal [2012] give extensive review of the use of particle filters in econometrics.

2.5 Smoothing Algorithms

In the discussion over the Kalman filter, we introduced various smoothing algorithms within the spectrum of linear and Gaussian state space models. This section introduces smoothing techniques such as: the forward filtering and backward smoothing, the two-filter formula, and the generalized tow-filter formula that can be used in the cases of non-linear and non-Gaussian state space models. Moreover, among the three smoothing formulas, the generalized two-filter formula, have demonstrated great improvements over the forward filtering backward smoothing technique in terms of precision on state variable estimation, see Briers et al. [2010]. The two-filter smoothing formula was firstly introduced in non-linear Bayesian estimation by Bresler [1986]. Subsequently, the generalized two-filter formula developed by Briers et al. [2010] relies on the introduction of a set of artificial probability distributions to obtain a modified backward time filter. In this section, we review the forward filtering backward smoothing method, the two-filter smoothing formula, and the generalized two-filter smoothing formula in the framework of general state space model.

The forward filtering backward Smoothing

Let $x_{1:T} = \{x_1, \cdots, x_T\}$ and $\mathcal{I}_T = \{y_1, \cdots, y_T\}$, then the joint distribution $p(x_{1:T}|\mathcal{I}_T)$ can be decomposed as

$$\begin{aligned} p(x_{1:T}|\mathcal{I}_T) &= p(x_T|\mathcal{I}_T) \prod_{t=1}^{T-1} p(x_t|x_{t+1}, \mathcal{I}_T) \\ &= p(x_T|\mathcal{I}_T) \prod_{t=1}^{T-1} p(x_t|x_{t+1}, \mathcal{I}_t). \end{aligned} \tag{2.51}$$

The above expression reveals that conditional on x_{t+1} and $\mathcal{I}_t$, x_t is independent of information $\{y_{t+1}, \cdots, y_T\}$. In the meantime, the term $p(x_t|x_{t+1}, \mathcal{I}_t)$ can be expressed as

$$p(x_t|x_{t+1}, \mathcal{I}_t) = \frac{f(x_{t+1}|x_t)p(x_t|\mathcal{I}_t)}{p(x_{t+1}|\mathcal{I}_t)}. \tag{2.52}$$

According to equation (2.52), it says that the density of $p(x_t|x_{t+1}, \mathcal{I}_t)$ can be obtained through available information such as: the prediction density $p(x_{t+1}|\mathcal{I}_t)$, the transition density $f(x_{t+1}|x_t)$, and the posterior density $p(x_t|\mathcal{I}_t)$. Morover, in equation (2.52), where state x_t conditions to the state at future time instance x_{t+1} is where the backward smoothing comes in, which can be explained through the following derivation on the marginal smoothing density,

$$\begin{aligned} p(x_t|\mathcal{I}_T) &= \int p(x_t, x_{t+1}|\mathcal{I}_T)dx_{t+1} \\ &= \int p(x_t|x_{t+1}, \mathcal{I}_T)p(x_{t+1}|\mathcal{I}_T)dx_{t+1} \\ &= \int p(x_t|x_{t+1}, \mathcal{I}_t)p(x_{t+1}|\mathcal{I}_T)dx_{t+1} \\ &= p(x_t|\mathcal{I}_t)\int \frac{f(x_{t+1}|x_t)p(x_{t+1}|\mathcal{I}_T)}{p(x_{t+1}|\mathcal{I}_t)}dx_{t+1}. \quad \text{by eq (2.52)} \end{aligned} \tag{2.53}$$

The above expression states that once we have computed all the prediction and filtering densities $p(x_{t+1}|\mathcal{I}_t)$ and $p(x_t|\mathcal{I}_t)$ respectively for all $t = 0, 1, \cdots, T$, then it is possible to obtain $\{p(x_t|\mathcal{I}_T)\}$ recursively in a backward fashion. The only missing piece in equation (2.53) is to obtain the term $p(x_{t+1}|\mathcal{I}_T)$. However, we can start from $p(x_T|\mathcal{I}_T)$ in order to obtain $p(x_{T-1}|\mathcal{I}_T)$. It also answer the reason that the smoothing algorithm will be executed backwards. This idea was first proposed by Kitagawa [1987]. Alternative view of the estimation of $p(x_t|\mathcal{I}_T)$ is integrating $\{x_{0:t-1}, x_{t+1:T}\}$ out of joint distribution of equation (2.51), where $x_{1:t-1} = \{x_1, \cdots, x_{t-1}\}$ and $x_{t+1:T} = \{x_{t+1}, \cdots, x_T\}$.

The two-filter smoothing formula

The two-filter smoothing formula of Bresler [1986] combines the output of two independent filters: the forward filter given by the particle filtering and the backward information filter calculating $p(y_{t:T}|x_t)$. The computation of $p(y_{t:T}|x_t)$ is given by the following equation,

$$\begin{aligned} p(y_{t:T}|x_t) &= \int p(y_t, y_{t+1:T}, x_{t+1}|x_t)dx_{t+1} \\ &= \int p(y_{t+1:T}|x_{t+1})p(x_{t+1}|x_t)p(y_t|x_t)dx_{t+1} \\ &= \int p(y_{t+1:T}|x_{t+1})f(x_{t+1}|x_t)g(y_t|x_t)dx_{t+1}. \quad \text{by Def (2.2.6)} \end{aligned} \tag{2.54}$$

The functions within the right hand side of the integral are known from the filtering process; therefore, given the prediction and the backward information filter, we can

obtain the smooth density via

$$\begin{aligned} p(x_t|\mathcal{I}_T) &= p(x_t|y_{1:t-1}, y_{t:T}) \\ &= \frac{p(x_t, y_{1:t-1}, y_{t:T})}{p(y_{1:t-1}, y_{t:T})} = \frac{p(x_t|y_{1:t-1})p(y_{t:T}|x_t, y_{1:t-1})}{p(y_{t:T}|y_{1:t-1})} \\ &\propto p(x_t|\mathcal{I}_{t-1})p(y_{t:T}|x_t). \quad \text{as } \mathcal{I}_{t-1} = y_{1:t-1} \end{aligned} \tag{2.55}$$

The backward information filter $p(y_{t:T}|x_t)$ can be obtained according to equation (2.54).

The Generalized two-filter smoothing formula

In the two-filter smoothing formula, the backward information filter is not a probability density in argument x_t and it is even possible that $\int p(y_{t:T}|x_t)dx_t = \infty$. Since sequential Monte Carlo based approximations can only be used to approximate finite measures. To circumvent the problem that we are facing, Briers et al. [2010] introduce a set of artificial probability distributions $\{\gamma_t(x_t)\}$. The modifications are described as follows: the probability distribution $\{\gamma_t(x_t)\}$ for $t = 1, \cdots, T$ are defined such that:

$$p(y_{t:T}|x_t) > 0 \Rightarrow \gamma_t(x_t) > 0,$$

and at $t = T$, we define

$$\tilde{p}(x_T|y_T) = \frac{g(y_T|x_T)\gamma_T(x_T)}{\tilde{p}(y_T)}, \tag{2.56}$$

where

$$\tilde{p}(y_T) = \int g(y_T|x_T)\gamma_T(x_T)dx_T.$$

It is said that the function $\tilde{p}(x_T|y_T)$ in equation (2.56) is by construction a probability measure. In similar fashion, the joint distributions

$$\tilde{p}_t(x_{t:T}|y_{t:T}) = \frac{\gamma_t(x_t)\prod_{k=t+1}^{T} f(x_k|x_{k-1})\prod_{k=t}^{T} g(y_k|x_k)}{\tilde{p}(y_{t:T})}, \tag{2.57}$$

where

$$\tilde{p}(y_{t:T}) = \int \cdots \int \gamma_t(x_t) \prod_{k=t+1}^{T} f(x_k|x_{k-1}) \prod_{k=t}^{T} g(y_k|x_k)dx_{t:T}.$$

Then one can show that the following is true [5]

$$p(y_{t:T}|x_t) = \tilde{p}(y_{t:T})\frac{\tilde{p}(x_t|y_{t:T})}{\gamma_t(x_t)}, \tag{2.58}$$

providing

$$\tilde{p}(x_t|y_{t:T}) = \int \cdots \int \tilde{p}_t(x_{t:T}|y_{t:T})dx_{t+1:T}. \tag{2.59}$$

Up to this moment, one key question that readers may have in mind is how do we select the artificial distribution $\gamma_t(x_t)$ at each time period. The discussion on this aspect can be found in Briers et al. [2010].

Taking equation (2.58), the backward information filter has been re-expressed the marginal density $\tilde{p}(x_t|y_{t:T})$ and the artificial density $\gamma_t(x_t)$. Therefore, by substituting equation (2.58) into equation (2.55), the marginal smoothed distribution of generalized two-filter formula can be expressed as follows

$$p(x_t|\mathcal{I}_T) \propto \frac{p(x_t|\mathcal{I}_{t-1})\tilde{p}_t(x_t|y_{t:T})}{\gamma_t(x_t)}, \tag{2.60}$$

and

$$p(x_1|\mathcal{I}_T) \propto \frac{\mu_1(x_1)\tilde{p}_t(x_1|y_{1:T})}{\gamma_1(x_1)} \tag{2.61}$$

where μ_1 is the initial density of x_1. The steps of computing $\{p_t(x_t|\mathcal{I}_T)\}$ of the generalized two-filter smoother are listed in the following:

- Store the prediction densities $\{p(x_t|\mathcal{I}_{t-1})\}$ from the filtering recursion.
- Compute and store $\{\tilde{p}(x_t|y_{t:T})\}$.
- For all $t = 1, \cdots, T$, obtain $p(x_t|\mathcal{I}_T)$ through the combination of $p(x_t|\mathcal{I}_{t-1})$, $\tilde{p}(x_t|y_{t:T})$, and $\gamma_t(x_t)$, as it has been expressed in equation (2.60).

2.6 Conclusion

In this introduction chapter, we start with definitions on stochastic process and sample path and Markov chain. This prelude categorizes the specific type of economic

[5] The derivation of equation (2.58) has been omitted since it does not contribute to the thesis as while except adding algebra manipulations. However, it can be requested from me or refer to the work of Briers et al. [2010].

and finance processes that this thesis studies. In addition, these definitions serve as building-blocks that allow us to further introduce the concept of ergodicity of Markov chains. Subsequently, Markov chain provides a standing-point to formalize the dynamics structure of state variable that explains the observed economic and financial processes. Building upon on this information, we define and introduce the general state space model, which has proven to be particularly attractive in modelling time series processes in economics and finance.

The rest of the chapter reviews techniques that enable us to estimate general state space models. Those techniques are: the Kalman filter, Kalman smoothing, particle filtering, and particle smoothing. Provided particle filtering and particle smoothing being two of those most appropriate and accurate classes of techniques of all existing ones in dynamic model inference, an overview of particle filtering and its related knowledge are therefore discussed in great length.

2.7 Appendix

Appendix A

Block Matrix Inversion formulas

Definition 2.7.1.

$$(A - BD^{-1}C)^{-1} = A^{-1} + A^{-1}B(D - CA^{-1}B)^{-1}CA^{-1}$$

where A, B, C, and D are matrices with appropriate dimension.

The Calculus of Conditional Expectations

Theorem 2.7.1. *The Calculus of Conditional Expectations Consider the jointly distributed normal random vectors X and Y, which possesses the linear relationship of $E(Y|X) = \alpha + \beta X$. The following conditions are true:*

$$E(Y|X) = E(Y) + C(Y,X)V^{-1}(X)\{X - E(X)\}, \tag{2.62}$$

$$V(Y|X) = V(Y) - C(Y,X)V^{-1}(X)C(X,Y), \tag{2.63}$$

$$E\{E(Y|X)\} = E(Y), \tag{2.64}$$

$$V\{E(Y|X)\} = C(Y,X)V^{-1}(X)C(X,Y), \tag{2.65}$$

$$V(Y) = V(Y|X) + V\{E(Y|X)\}, \tag{2.66}$$

$$C\{Y - E(Y|X), X\} = 0. \tag{2.67}$$

Proof. To begin, for simplicity case, X and Y are assumed to be univariate random variable (although the random vector case can be derived in similar way). Equation of the linear relationship may be multiplying throughout by $f(x)$, and integrates with respect to X, which gives

$$E(Y) = \alpha + \beta E(X), \tag{2.68}$$

whence $\alpha = E(Y) - \beta E(X)$, and substitute it into the equation of linear relationship, which yields

$$E(Y|X) = E(Y) + \beta\{X - E(X)\}. \tag{2.69}$$

Next, let multiply the linear relationship by X and $f(x)$ and then integrated with respect to X to provide the following equation

$$E(XY) = \alpha E(X) + \beta E(X^2). \tag{2.70}$$

Multiplying equation (2.68) by $E(X)$ gives

$$E(X)E(Y) = \alpha E(X) + \beta\big\{E(X)\big\}^2. \tag{2.71}$$

whence, subtract equation (2.71) from equation (2.70), we get

$$\begin{aligned} \beta &= \{E(XY) - E(X)E(Y)\}\{E(X^2) - (E(X))^2\} && (2.72)\\ &= C(x,y)V^{-1}(y). && (2.73) \end{aligned}$$

For (2.62), plugging equation (2.72) into equation (2.69), we have what we need as in equation (2.62).
For (2.63), given equation (2.62), we can re-express it as

$$E(Y|X) - E(Y) = C(Y,X)V^{-1}(X)\{X - E(X)\}.$$

The above expression is multiplied by $\{Y - E(Y)\}$, then take expectation on both sides, after few of the re-arrangements, equation (2.63) will be obtained.

For (2.64), we have

$$\begin{aligned} E\{E(Y|X)\} &= \int_{\mathcal{X}}\int_{\mathcal{Y}} yf(y|x)dyf(x)dx = \int_{\mathcal{Y}} y\int_{\mathcal{X}} f(x,y)dxdy \\ &= \int_{\mathcal{Y}} yf(y)dy = E(Y). \end{aligned}$$

For (2.65), it is nothing more than the *Law of total variance*.
For (2.66), given equations (v) and (ii), we have

$$V(Y|X) = V(Y|X) + V\{E(Y|X)\} - C(Y,X)V^{-1}(X)C(X,Y),$$

then, equation (2.66) will be apparent.
For (2.67), it is

$$\begin{aligned} C\Big\{Y - E(Y|X), X\Big\} &= E\Big\{\big(Y - E(Y|X)\big)X\Big\} - E\Big\{Y - E(Y|X)\Big\}E\{X\} \\ &= E\{YX\} - E\{E(Y|X)X\} = E\{YX\} - E\{YX\} = 0. \end{aligned}$$

□

Appendix B

Conditional Probabilities. There are $P(A|B) = P(A \cap B)/P(B)$ and $P(B|A) = P(B \cap A)/P(A)$. Therefore, the conditional probability can be written as

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A \cap B)}{P(A)}\frac{P(A)}{P(B)} = \frac{P(B|A)P(A)}{P(B)}. \tag{2.74}$$

Moreover, we can deduce that

$$\begin{aligned} P(A \cap B \cap C) &= \frac{P(A \cap B \cap C)}{P(B \cap C)}\frac{P(B \cap C)}{P(C)}P(C) \\ &= P(A|B \cap C)P(B|C)P(C). \end{aligned} \tag{2.75}$$

Bayes' Theorem. The theorem is conveyed by the equation

$$\begin{aligned} P(H_i|E) = \tfrac{P(E \cap H_i)}{P(E)}, \quad \text{wherein} \\ P(E) = \textstyle\sum_i P(E \cap H_i) = \sum_i P(E|H_i)P(H_i). \end{aligned} \tag{2.76}$$

Here, $P(H_i)$ is the prior probability of the hypothesis H_i and $P(H_i|E)$ is its posterior probability in the light of the evidence E.

The Chapman–Kolmogorov Equation. The marginal probability of the event A can be obtained from the joint probability of the events A and B by a process of integration:

$$P(A) = \int_{\mathcal{B}} P(A \cap B) dB.$$

In a Markov chain, the probability of the transition from state B to state A is determined without reference to any the preceding states. That is to say, the next state depends only on the current state and not on the past. If the sequence of events C, B, A is governed by a Markov process, then there is $P(A \cap B \cap C) = P(A|B)P(B|C)P(C)$. Therefore,

$$P(A \cap C) = \int_{\mathcal{B}} P(A|B)P(B|C)P(C) dB, \tag{2.77}$$

and

$$P(A|C) = \frac{P(A \cap C)}{P(C)} = \int_{\mathcal{B}} P(A|B)P(B|C) dB. \tag{2.78}$$

The Strong Law of large number. Suppose that $X_1, X_2, \cdots$ are independent, identically distributed, with $E(X_1) = \mu < \infty$. Let $S_n = X_1 + X_2 + \cdots + X_n$, then

$$\frac{S_n}{n} \Rightarrow \mu \quad \text{almost surely.}$$

Chapter 3

Resampling with Shannon Information Entropy diagnostics and Resampling Schemes Comparison

Abstract

This chapter proposes a novel weight degeneracy diagnostics to particle filtering that is based upon the idea of the Shannon information entropy. This new diagnostics overcomes the potential over-resampling problem associated with the most widely applied effective sample size diagnostics. In addition, according to the evidence of accuracy measure over the inference of hidden state sequence, the new diagnostics exhibits more stable performance than the effective sample size diagnostics. The claims have been demonstrated through the aspects of mathematical derivation and simulation studies. The aspect of simulation studies have been conducted on the comparison between these two weight degeneracy diagnostics of two different types of state space models: linear Gaussian model and non-linear time series model. A separate contribution of this chapter focuses on the usage of finite particle counts, where local linearisation particle filter equipped with the Shannon information entropy diagnostics has been performed in a simulation based investigation on various existing resampling schemes. This investigation provides further supplementary information for theoretical discussion on resampling schemes selection in particle filtering, as well as a synopsis on particle filtering implementation for empirical researchers.

3.1 Introduction

The method of Sequential Monte Carlo or particle filtering is a simulation technique, which gains its prevalence through its renowned ability of estimating non-linear and non-Gaussian state space models. The objective of this chapter will be twofold: first, I propose a novel weight degeneracy diagnostics that derives from the idea of Shannon information entropy. The new diagnostics exhibit superiority over the existing *effective sample size* diagnostics. The specifications of superiority means computational cost will be less expensive whilst sustaining estimation precision, which has been demonstrated through mathematical derivations, and later supported by simulation applications. Second, considering the constraint of finite particles usage in practice, this chapter provides empirical implementation guidance to the selection over various existing resampling schemes in particle filters.

Applications of particle filters in the domain of economics and finance can be retroactive to the mid of 90s, with the work of Kitagawa [1994] and Kitagawa [1996] on economic time series data analysis, and Shephard [1996] on stochastic volatility. The more recent ones are: the formulation and estimation of dynamic equilibrium models utilizing particle filtering of Fernandez-Villaverde and Rubio-Ramirez [2005] and the learning of time series through stochastic volatility modelling by Pitt and Shephard [1999].

The underlying models within the aforementioned applications of particle filtering are belonging to general state space model. Estimating this type of models with previously developed Kalman based filters have emerged to be formidable, and frequently suffered from serious drawback of low estimation precision. Consequently, the attention has lately shifted to utilize the numerically based approach, which has been referred as sequential Monte Carlo methods or particle filters. Despite the fact that particle filtering method provides the solution of suboptimal estimation to the non-linear and non-Gaussian state space model, it has most certainly not prevent particle filters from a tremendous success in various fields. Such prevalence was massive due to the initial contribution of Gordon et al. [1993]. Their breakthrough lies mostly due to the introduction of the multinomial resampling technique into particle filtering. This implementation alleviates the effect of *weight degeneracy*, and it subsequently sustains the stability of sequential updating of particle filtering algorithm in a reasonable time length.

The essence of resampling procedure is to discard particles that carry relatively smaller weights; in the meantime duplicate particles have relatively larger weights. Therefore, this procedure forms a new set of particles at each time instance. Ac-

cordingly, this newly revitalized particle set stabilizes the performance of sequential updating of particle filtering algorithm, providing reasonable approximation for the target (posterior) distribution. In spite of the significant role of resampling plays in particle filtering, its existence has been overshadowed by the introduction of resampling noise and the addition of computation complexity. The latter issue had been tackled by Kong et al. [1994], who devised a weight degeneracy diagnostics through the so called *coefficient variation* formula. This diagnostics detects the randomness across the whole particle weights at given time stance. More specifically, the action of resampling will be proceeded if the diagnostics result falls below a pre-specified threshold, and therefore, the resampling acts as a reinforcement to balance out the particles for their subsequent time instance. Another related but more widely implemented weight degeneracy diagnostics is the *effective sample size* of Liu [1996]. However, both weight degeneracy diagnostics derive from taking the square of particle weight values, which can often result in potential over detection on weight degeneracy. Therefore, the computational cost increases via unnecessary resampling steps. In this chapter, I propose an alternative diagnostics that originated from the idea of Shannon information entropy. This novel diagnostics process the particle weights without modifications, which then determines the necessity of resampling at such time stance. Consequently, I shall demonstrate on the aspect of computational time, the Shannon information entropy diagnostics is more efficient compare to the effective sample size diagnostics and their related diagnostics.

In the light of the aforementioned multinomial resampling technique, other resampling schemes such as: residual resampling of Liu [1996], stratified resampling of Carpenter et al. [1999], and systematic resampling of Kitagawa [1996] was subsequently been developed and implemented in particle filtering. Douc et al. [2005] provide a theoretical comparison for these four resampling schemes in the direction of the conditional variance. An additional contribution of this chapter is that I extend the work Douc et al. [2005], by looking into an important practical issue, which is over the choice of previously listed resampling schemes whilst the particle counts are finite. The investigation conducted through extensive experiment studies, which allows to draw inference that have been derived from their performances in real applications over aspects of estimation accuracy and computational efficiency.

The chapter is organized as follows: section 2 provides insights into a crucial question of why would researchers be interested in utilizing particle filters? Section 3 demonstrates the significant role of resampling procedure and illustrates the various existing resampling schemes in particle filtering. Section 4 proves the newly proposed Shannon information entropy diagnostics that overcomes the potential over-

resampling problem associated with the effective sample size diagnostics. Section 5 conducts various simulation comparisons between the Shannon information entropy diagnostics and the effective sample size in terms of the measure of root mean square error. Moreover, considering the aspects of estimation precision and computational efficiency, I examine four of the most widely implemented resampling schemes in particle filtering literature. Section 6 draws the conclusions and discuss future researches on this topic.

3.2 Extended Kalman Filter vs Particle Filters

This section seeks evidences on why particle filter are invariably superior to the extended Kalman filter for intricate problem such as estimating the non-linear and non-Gaussian state space models. To complete the demonstration, I consider estimating the time series model of Kitagawa [1996]. This time series model has been frequently adopted for simulation studies in the development of various particle filtering and particle smoothing in the literature, for instance, Doucet et al. [2000a] and Doucet and Tadic [2003]. The following example illustrates through the model.

Example 3.2.0.1 (Non-linear time series model)**.** *Let the measurement equation and the transition equation be defined as:*

$$\begin{aligned} y_t &= g(x_t) + w_t = \frac{x_t^2}{20} + w_t, \\ x_t &= f(x_{k-1}) + v_t = \frac{x_{t-1}}{2} + \frac{25x_{t-1}}{1+x_{t-1}^2} + 8\cos(1.2t) + v_t, \end{aligned}$$

where $w_t \sim \mathcal{N}(0, \sigma_w^2)$ *and* $v_t \sim \mathcal{N}(0, \sigma_v^2)$*. Furthermore,* $\sigma_w^2 = 1$*,* $\sigma_v^2 = 10$*, and* w_t *and* v_t *are mutually independent with* $x_1 \sim f(x_1) = \mathcal{N}(0, 10)$*. The derived densities are*

$$\begin{aligned} y_t \sim g(y_t|x_t) &= \mathcal{N}\Big(\frac{x_t^2}{20}, 1\Big), \\ x_t \sim f(x_t|x_{t-1}) &= \mathcal{N}\Big(\frac{x_{t-1}}{2} + \frac{25x_{t-1}}{1+x_{t-1}^2} + 8\cos(1.2t), 10\Big). \end{aligned}$$

The SIR filter or the bootstrap filter has been utilized to estimate the above non-linear time series model. In addition, multinomial resampling has been embedded into the bootstrap filter. In the simulation of the model in Example (3.2.3.1), the true state values have assumed to be known. Furthermore, the *root mean square error* or empirical standard deviation is the measure of discrepancy between the true state values and their estimates. This measure has been undoubtedly the most widely

adapted measure of discrepancies in particle filter literature, which takes the following form

$$RMSE(x_{t|t}) = \frac{1}{R}\sum_{i=1}^{R}\Big(\frac{1}{N}\sum_{j=1}^{N}(x_{t|t}^{j} - x_{t}^{j})^{2}\Big)^{1/2}, \tag{3.1}$$

where

- x_t^j is the true simulated state for the j^{th} particle, with $j = 1, \cdots, N$.
- $x_{t|t}^j$ is the jth estimate at time step t given information set $\mathfrak{I}_t$.
- R is the number of repetitions.

Figure (3.1) reveals that the bootstrap particle filter with multinomial resampling tracks the actual state process rather well. On the contrary, there are apparent discrepancies between the EKF estimates and the true state process.

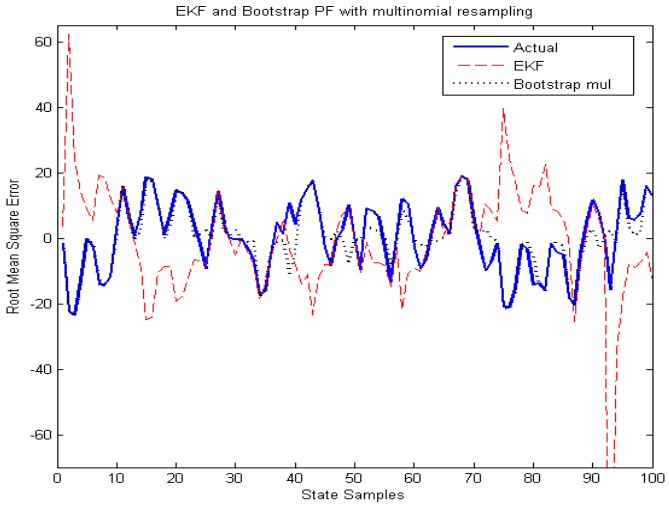

Figure 3.1: Estimation of state variables using the EKF (dashed line) and Bootstrap particle filter with multinomial resampling (dotted line). The actual state variables are represented by the solid line.

Table (3.1) reports the RMSE of EKF and three types of particle filters with systematic resampling (see the following list). Column 2 displays that RMSE of the EKF that are at least 4 times larger than the RMSE of particle filters. Since the EKF does not evolve with particle counts, the differences among the RMSE values

across the entire column 2 (EKF) represent distinctive Monte Carlo error from simulation. In addition, the RMSE of all three particle filters are comparable to each other. It seems as the theory predicts that as the number of particles N becomes large, the RMSE of all particle filters invariably converge to some value. However, so far as the time series model (example 2.2.3.1) concerning, the preceding comparison results between the EKF and particle filters exhibits similar results. Hence for empirical researchers, who have seldom had the experience with particle filters (in estimating non-linear and non-Gaussian state space models), I hope the above sound evidence is self-explanatory regarding to its usefulness. These three particle filters will be re-appearing in our subsequent simulation study, and therefore their respective characteristics are displayed as follows:

Table 3.1: RMSE of the EKF and Particle filters, where the Monte Carlo repetitions R to be 100 runs.

RMSE	EKF	Bootstrap	Linearised	Linearised Ess
N=100	19.3640	5.4767	5.9231	6.0596
N=250	20.2550	5.3179	5.5655	5.5574
N=500	19.6660	5.1981	5.2062	5.3281
N=1000	20.0085	5.1348	5.0333	5.0988
N=2500	20.7216	5.1483	4.9184	4.9785
N=5000	20.5277	5.2243	4.9049	4.9985
N=10000	20.4259	5.1796	4.8646	4.9376

List of Particle filters (LPFs)

1. Kalman filter, the optimal filter for linear and Gaussian state space model, where their target posterior density function can be derived analytically as to be normally distributed.

2. Bootstrap particle filter: the importance density $q_\theta(x_t|x_{t-1}, \mathcal{I}_t)$ is set to be equalling the transition density $f_\theta(x_t|x_{t-1})$ and the resampling will be performed at each time instance.

3. Bootstrap particle filter with Ess: the importance density $q_\theta(x_t|x_{t-1}, \mathcal{I}_t)$ is set to be equalling the prior density $f_\theta(x_t|x_{t-1})$ and the resampling is merely performed when the effective sample size (Ess) fall below the threshold N_{thre}.

4. Linearisation particle filter: the importance density $q_\theta(x_t|x_{t-1}, \mathcal{I}_t)$ is obtained by local Linearisation of observation equation (refer to Doucet et al. [2000a] for more details) and resampling is at each time step.

5. Linearisation particle filter with Ess: same as (4) in this list, but the resampling is merely performed when the effective sample size (Ess) fall below the threshold N_{thre}.

6. Linearisation particle filter with Shannon entropy: same as (4) in this list, but the resampling is merely performed when the Shannon information entropy diagnostics fall below the threshold N_{thre}.

3.3 Resampling

This section focuses on the following tasks: The first one would be to build our understanding on the significance of resampling procedure in particle filtering. In the subsequent task, I introduce and discuss four of the most widely implemented resampling schemes, they are: multinomial resampling, residual resampling, stratified resampling and systematic resampling. The multinomial resampling of Gordon et al. [1993] had emerged to the first and foremost among all other resampling schemes. In fact, the average computational complexity of multinomial resampling is $O(N \log_2(N))$. This implies for a state sample size of T, the computational complexity of particle filtering algorithm such as SIR will be at least $O(TN \log_2(N))$.

3.3.1 Demonstrations

An insightful theoretical example in Doucet and Johansen (2008, p12), which has also been derived and discussed in example (2.4.3.1) in chapter 1 shows that: in the sequential algorithm without resampling, the variance of their estimates grow exponentially with the time step t. Such problem occurs in almost all sequential importance sampling applications. The reason being due to the so called weight degeneracy phenomena. The term of weight degeneracy illuminates a phenomenon within the sequential importance algorithm. That is that as the most of particle weights are nearly zero and only few carry the significant weights, and therefore the pre-determined and great amount of particles loss their capability in approximating the target densities as time step increases. This is one of the reasons why the approach of particle filtering to various fields of research did not take off until the contribution of Gordon et al. [1993].

I demonstrate the importance of resampling through usage of the following simulation example. Simulation will be conducted upon a univariate linear and Gaussian model where all parameters are assumed to be known. Given such dynamic system

model, estimation uses the Kalman filter is known to be an optimal. Nevertheless, the model still serves the purpose of showing how resampling overcomes (as least partially) the problem of weight degeneracy in sequential particle filtering.

Example 3.3.1.1 (Linear and Gaussian Dynamic model)**.** *The following model is the simplest state-space models. It consists of two equations*

$$Y_t = H_t X_t + \sigma_w W_t, \quad \textit{Measurement Equation} \tag{3.2}$$

$$X_t = F_t X_{t-1} + \sigma_v V_t, \quad \textit{Transition Equation} \tag{3.3}$$

where y_t is the observations in the system and x_t is the state variable. The measurement error w_t and the transitional disturbance v_t are pre-assumed to be mutually uncorrelated random variables with the respected mean and variances are

$$W_t \sim \mathcal{N}(0,1) \quad \textit{and} \quad V_t \sim \mathcal{N}(0,1). \tag{3.4}$$

Suppose $\sigma_v = 1$, $\sigma_w = 1$, $H_t = 1$, and the transition parameter $F_t = 1$. Given that information, the transition equation is defined as $AR(1)$ process. The posterior has a closed form, that is known to be normal distribution.

Figure (3.2) shows the distribution of particle weights in the scenarios with and without resampling procedure. The top panel has been produced with sequential importance sampling algorithm without resampling, the weight degeneracy can be easily identified, even with as fewer as 10 time instances. The distribution of particle weights within the bottom panel shows that: for all three separated time instance, all particles possess significant values and with no sign of particular larger weights. In addition, the behaviour of particle weights distribution has been maintained throughout for all three time counts. This type of particle weights distribution implies that we have sufficient amount of particles to obtain a reasonable approximation to the target posterior density $p(x_t|\mathcal{I}_t)$ that guarantees reliable the state estimation.

Figure (3.3) demonstrates that, particle filter with resampling, their standard deviation of 50 sets of particle weights remain stable. On the contrary, particle filter without resampling, as indicated by the thick solid line, shows a clear exponential increment of their weights standard deviation. This implies to a certain extent that resampling has the capability to re-balance the particles, and allows particle filters to alleviate from the effect of weight degeneracy. However, this encouraging evidence revealed by resampling provides no theoretical information regarding to the estimation of state path. To summarise, in the particle filter literature, a well performed

particle filtering algorithm relies on two aspects: first is that the selection of importance density should be close to the optimal density distribution. Second one is that the resampling algorithm should introduce as small variance as possible. The selection of the importance density has direct effects in helping to determine the most suitable amount of particles that is needed for the particle filtering approximation. The detailed discussions on ways of selecting appropriate importance density can be found in Arulampalam et al. [2002] and Pitt and Shephard [1999].

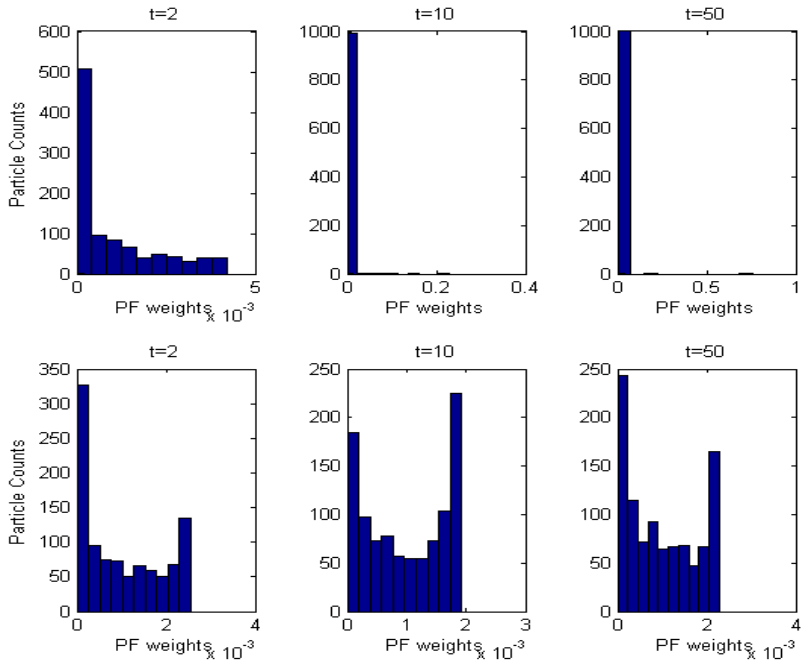

Figure 3.2: Particle weights of particle filtering without resampling (top plot) versus particle weights of particle filtering with resampling (bottom plot). The number of particles N is $1,000$ and time instances t are 2, 10, and 50 respectively.

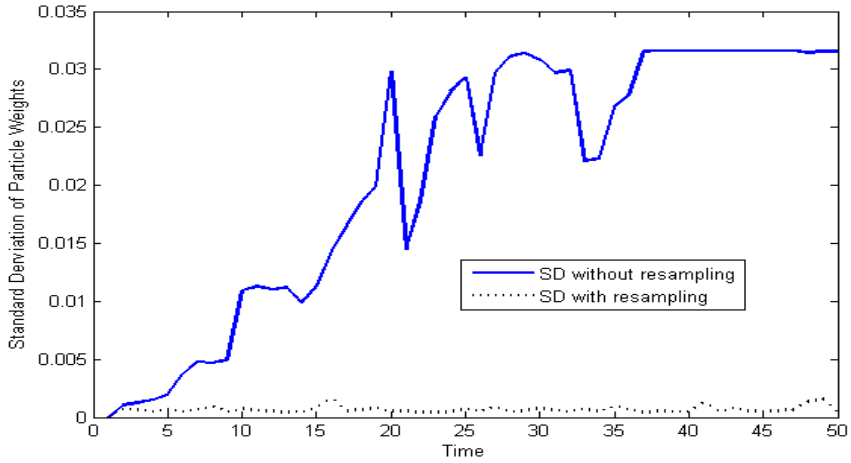

Figure 3.3: The standard deviation of particle weights of particle filtering with and without resampling.

3.3.2 Resampling Schemes

The intuition of a set of particle weights can be thought of as a measure on the likelihood of their respective particles (within the set) being in the target posterior distribution. Smaller weights (in relative sense) indicate their corresponding particles that contribute moderately to the approximation of target posterior distribution. Moreover, with the amount of small particle weights being large; the approximation will be burdened as most of the computational works taken by small weighted particles, which does not actually contribute much to the approximation, as well as the estimation of hidden states. Doucet and Johansen [2008] pointed out that, in sequential importance sampling, the weight degeneracy exists almost always as time instance increases, where majority normalized importance weights in the set close to zero with the exception of few particles carry significant weights.

In particle filtering, the formidable degeneracy issue can be overcome with the implementation of resampling procedure. Resampling performs a probabilistic selection on a given set of standardized weight, which eliminates particles with small weights, whereas those with large particle weights are duplicated. Accordingly, the duplication counts help to form a new set of particles with the set of particle weights to be

reset uniformly equal. Despite the appealing property associated with resampling in particle filter, it should be pointed out that with the addition of resampling step, resampling inflicts some extra 'noise' into the approximation. This is simply due to the duplication process that reduces distinct particles, thus degrading the accuracy of the particle approximation. However, resampling remains to be essential to guarantee bounded conditional variance of the filter in the long-run.

Four of the most widely utilized resampling schemes are: multinomial resampling, residual resampling, stratified resampling, and systematic resampling. Their individual description, as well as their distinctive differences are listed as follows:

Multinomial Resampling : The idea of multinomial resampling can be understood as: given the standardized particle weights $\{\tilde{\omega}^i\}_{i=1}^N$, I intend to generate a N elements random duplication index $I = \{I^1, \cdots, I^N\}$, where the sum of $I's$ are equal to N. To do so, it is suggested by firstly generating a random variable U from uniform distribution of $[0, 1]$, then determining the index I such that whether $u \in \{\sum_{j=1}^{I-1} \omega^j, \sum_{j=1}^{I} \omega^j\}$, where u is the realization of random variable U. More specifically, the index I^i will be recorded once if u_i fall in the interval $\{\sum_{j=1}^{I-1} \omega^j, \sum_{j=1}^{I} \omega^j\}$, such procedure will be carried out N times, where N uniform random variables will be generated. It means that some of the intervals can be picked out multiple times, whereas other may not be picked out at all. Therefore, the index $I = \{I^1, \cdots, I^N\}$ is essentially a duplication vector with the values of I^i determines how many time the i^{th} particle will be duplicated. For instance, suppose I^i can be any integer of $(0, N)$, and $I^s = 3$, which simply indicates the s^th particle will be duplicated 3 times.

Residual Resampling : residual resampling of Liu [1996] has also been referred as *remainder resampling*. This scheme has been proven to give smaller conditional variance than the multinomial resampling, at least theoretically. In this scheme, for $i = 1, \cdots, N$, set

$$N^i = \lfloor N\tilde{\omega}^i \rfloor + \bar{N}^i, \tag{3.5}$$

where the notation $\lfloor . \rfloor$ indicates floor of the numerical value. The set of $\{\bar{N}^1, \cdots, \bar{N}^N\}$ is distributed according to the multinomial distribution $M(N - R, \bar{\omega}^1, \cdots, \bar{\omega}^N)$, where $R = \sum_{i=1}^{N} \lfloor N\omega^i \rfloor$ and

$$\bar{\omega}^i = \frac{N\tilde{\omega}^i - \lfloor N\tilde{\omega}^i \rfloor}{N - R}. \tag{3.6}$$

In other words, the values of $\bar{N}^i$ for $i = \{1, \cdots, N\}$ are obtained using the above procedures as described in above multinomial resampling. The proof of showing the conditional variance of residual resampling is less than the conditional variance of multinomial resampling can be found in Douc et al. [2005].

Stratified Resampling : This resampling scheme partitions the unit interval $(0, 1]$ into N different parts, hence $(0, 1] = (0, 1/N] \cup (1/N, 2/N] \cup \cdots \cup \big((N-1)/N, 1\big]$. Given those small intervals, one can draw a set of samples $\bar{u}^1, \cdots, \bar{u}^N$, e.g. $\bar{u}^i \sim U\Big(\big((i-1)/N, i/N\big]\Big)$ for $i = 1, \cdots, N$. The indices $I(\bar{U}^i)$ can be obtain via the following expression

$$I(\bar{u}_i) = \sum_{i=1}^{N} \mathbf{1}_{\left(\sum_{j=1}^{i-1} \tilde{\omega}^j, \sum_{j=1}^{i} \tilde{\omega}^j\right]}(\bar{u}), \tag{3.7}$$

where $\bar{u}$ is uniformly sampled from those partitioned intervals, and moreover these samples from portioned intervals are independent from each other.

Systematic Resampling : The uniform samples u^i are obtained via $u^i = u_1 + (i-1)/N$, for $i = 2, \cdots, N$, where u_1 is drawn from a uniform $U[0, 1/N]$ distribution. As compare with stratified resampling, systematic resampling is still the case that the unit interval is divided into N sub-intervals $((i-1)/N, i/N]$ and one sample is taken from each of them, which is the same as in stratified resampling. However, given all uniform samples share the same relative position u_1, therefore the samples are no longer independent from each other. As pointed out in Douc et al. [2005], since the samples are not from independent sub-intervals like they are in stratified resampling, it will not be as simple as other resampling simple formulas for the conditional variance of systematic resampling.

If we trace back to the residual resampling, one would notice that instead of using multinomial resampling on given remainder weights $\bar{\omega}^i$ in residual resampling, we could replace it with stratified resampling for their reminder weights. Other scheme such as the coupled methods can form attractive resampling scheme and worth to be exploited on their accuracy, as well as their computational efficiency. Both stratified resampling and systematic resampling possess computational complexity to be approximately over $O(N)$.

3.4 Shannon Information Entropy diagnostics

In the SIR algorithms, resampling has been performed at each time step in the SIR type of filtering algorithms. Such an act demands excessive computational effort. To overcome the large computational complexity associated with SIR filters, Liu [1996] and others suggested the so-called *effective sample size* (ESS) diagnostics. The Ess diagnostics insists that resampling step should only be executed when the value of Ess is smaller than some threshold particle counts, otherwise resampling should be avoided. Such a criterion is defined by equation (51) in Arulampalam et al. [2002]. The *coefficient variation* diagnostics developed by Kong et al. [1994] is another degeneracy diagnostics, and the diagnostics per-Se is closely related with the Ess diagnostics. In this section I demonstrate that the novel Shannon information entropy diagnostics is a more time efficient diagnostics for degeneracy compared to both coefficient variation diagnostics and Ess diagnostics. Note in the following discussion, the subscripts of particle weights $\tilde{\omega}$ such as time t and parameter θ, will be omitted for the sake of simplifying notations.

3.4.1 Effective Sample Size

The previously discussed sequential importance sampling approach is bound to fail in the long run, because of the curses of *weight degeneracy*. One of the important methods that had revived sequential importance sampling or particle filters is through the introduction of resampling procedure. However, the downsides of employing resampling would be the introduction of additional noise, as well as increasing the computation complexity in the model estimations. Therefore, it seems particularly attractive in practice that a dignostics that can be set-up to detect weight degeneracy, and perform resampling only when weight degeneracy is eminent. The most widely applied dignostics is the so-called *effective sample size* $\hat{N}_{Ess}$ (Liu [1996]), which has been defined as

$$\hat{N}_{Ess} = \Big\{ \sum_{i=1}^{N} (\tilde{\omega}^i)^2 \Big\}^{-1}, \tag{3.8}$$

where N is the number of particles, and the standardized i^{th} weight is

$$\tilde{\omega}^i = \frac{\omega^i}{\sum_{j=1}^{N} \omega^j}.$$

The value of N_{Ess} varies between 1 (all weights are insignificant except one) and N (equal weights). Prior to wide applications of the effective sample size dignostics, Kong et al. [1994] had already proposed another dignostics that has been known as the *coefficient variation*, which has been defined through the following expression

$$CV_N = \left\{ \frac{1}{N} \sum_{i=1}^{N} \left(N\tilde{\omega}^i - 1 \right)^2 \right\}^{1/2}. \tag{3.9}$$

The value of CV_N is minimal (equal to 0) when the normalized weights ω^i are all equal to $1/N$, which means that no variation among all weights. On the contrary, the maximum of CV_N equals $(N-1)^{1/2}$, which corresponds to the scenario where the effective sample size N_{Ess} equal to 1. As the matter of fact, the relationship between N_{Ess} and CV_N can be summarized through the following equation

$$\hat{N}_{Ess} = \frac{N}{1 + CV_N^2}. \tag{3.10}$$

The derivation of the above equation can be verified as follow

$$\begin{aligned}
\hat{N}_{Ess} &= \frac{N}{1 + \left[\left\{ \frac{1}{N} \sum_{i=1}^{N} \left(N\tilde{\omega}^i - 1 \right)^2 \right\}^{1/2} \right]^2} \\
&= \frac{N}{1 + \left\{ \frac{1}{N} \sum_{i=1}^{N} \left(\left(N\tilde{\omega}^i \right)^2 + 1 - 2N\tilde{\omega}^i \right) \right\}} \\
&= \frac{N}{1 + \left(N \sum_{i=1}^{N} (\tilde{\omega}^i)^2 + 1 - 2 \right)} \quad \text{by } \textstyle\sum_{i=1}^{N} \tilde{\omega}^i = 1 \\
&= \left\{ \sum_{i=1}^{N} (\tilde{\omega}^i)^2 \right\}^{-1}.
\end{aligned}$$

3.4.2 Shannon Information Entropy

The effective sample size diagnostics has been the most widely applied degeneracy diagnostics till date. However, the usage of square of weights has two potential problems when determining whether or not to resample at their respective time instance. The first is that the large proportion of weights within a set are relatively small, and those weights correspond their respected particles are actually close within the target density. Though resampling is seemingly unnecessary, the square of value of weights in the Ess diagnostics might be saying otherwise. The second is down to the small number recognition of the computer, the square of amount weights can simply be set as zero, and therefore such avoidance can result over detecting weight degeneracy.

Given the potential problems associated with Ess diagnostics and related diagnostics, inspired by the idea entropy idea in Chapter 7 of Cappe et al. [2005], I introduce an alternative weight degeneracy diagnostics based upon the concept of Shannon information entropy. This novel diagnostics takes the value of weights as they originally are in order to detect the existence of weight degeneracy. The motivation of adopting the Shannon information entropy is due to its basic definition. In information theory, Shannon information entropy measures the uncertainty associated with a piece of information (random variable). A mathematical display of this description is, for a random variable X, with N, outcomes $\{x_i : i = 1, \ldots, N\}$, where the Shannon information entropy, a measure of uncertainty denoted by $H(X)$, can be defined as

$$H(X) = -\sum_{i=1}^{N} p(x_i) \log_2 p(x_i),$$

where $p(x_i)$, is the probability mass function of outcome x_i.

The preceding description shares similarity with the standardized particle weights ω^i, where ω^i can be thought as the point mass of the random variable X at the particle (sample) realization x^i. In the meantime, all normalized weights ω^i has value between 0 and 1, with their sum equals to 1. Therefore, the randomness or disorder of standardized weights can be characterized by a similar expression as the one above (Shannon information entropy formula), which is

$$E = -\sum_{i=1}^{N} \omega_i \log_2(\omega_i). \tag{3.11}$$

Equation (3.11) can be understood as if all normalized weights are zero expect for one, then there is a minimum disorder among the particle set, and therefore the Shannon entropy is at its minimal. On the contrary, if all particles carry equal weight, which means all particles are uniformly distributed, this set of particles exhibits great randomness or disorder. This is the situation that the Shannon entropy is at its maximum that equals to $\log_2(N)$. In the light of the above idea, I formed the weight degeneracy diagnostics that can be implemented in any particle filters where the effective sample size diagnostics is applicable. The appealing advantage with the Shannon information entropy is that it does not distort the small weights in their calculations.

Proposition 3.4.1. *Suppose for all* $\omega_i \in (0, 1)$ *and* $\sum_{i=1}^{N} \omega_i = 1$, *with* $i = \{1, \cdots, N\}$,

and for N is large, then

$$2^{-\sum_{i=1}^{N}\omega_i \log_2(\omega_i)} \geq \big(\sum_{i=1}^{N}\omega_i^2\big)^{-1}. \tag{3.12}$$

Proof. To show equation (3.12) is true, is equivalent to showing the following expression is true,

$$2^{\sum_{i=1}^{N}\omega_i \log_2(\omega_i)} \leq \sum_{i=1}^{N}\omega_i^2.$$

Since $\omega_i \in (0,1)$, we can always find a value of $C_i \in (0,1)$ the following is true

$$2^{\omega_i \log_2(\omega_i)} \leq C_i.$$

For example, logarithmic base 2 of 1 is zero, this indicates that $\omega_i = 1$, but this is not true since $\omega_i \in (0,1)$. Given the above expression, we have

$$\begin{aligned} 2^{\omega_1 \log_2(\omega_1)} 2^{\omega_2 \log_2(\omega_2)} \cdots 2^{\omega_N \log_2(\omega_N)} &\leq C_1 C_2 \cdots C_N \\ \Longrightarrow \quad 2^{\sum_{i=1}^{N}\omega_i \log_2(\omega_i)} &\leq \prod_{i=1}^{N} C_i \\ &\leq C^N, \end{aligned}$$

for $C = Max\{C_1, \cdots, C_N\}$ and $C \in (0,1)$. For $N \to \infty$, then $C^N \to 0$, which gives

$$C^N \to 0 < \sum_{i=1}^{N}\omega_i^2 = \omega_1^2 + \cdots + \omega_N^2,$$

for N sufficiently large. Hence we have

$$\sum_{i=1}^{N}\omega_i^2 \geq 2^{\sum_{i=1}^{N}\omega_i \log_2(\omega_i)}.$$

This completes the proof. □

A remark is that the highest information entropy would be for all particles carrying

equal weight of $1/N$, and the following is true

$$\begin{aligned} E = -\sum_{i=1}^{N} \omega_i \log_2(\omega_i) &\leq -\sum_{i}^{N} \frac{1}{N} \log_2(\frac{1}{N}) \\ &= -\log_2(\frac{1}{N}). \end{aligned}$$

The above inequality implies that

$$2^{-\sum_{i=1}^{N} \omega_i \log_2(\omega_i)} \leq 2^{\log_2(N)} = N,$$

where N is the number of particles. Hence for any threshold, N_{thre} is an integer between 0 and N, we have $N_{thre} \leq N$. In other words, the resampling procedure can be avoided since the threshold is less than the dignostics result.

The above proposition shows that when the number of particle counts N is large, the Ess dignostics will detect weight degeneracy more often than the Shannon entropy dignostics. For instance, suppose the threshold $N_{thre} = 0.5N$ and $N = 10000$, then this implies that the Ess dignostics detects degeneracy if $\sum_{i=1}^{N} \omega_i^2 > 0.0002$, but suppose $C = 0.999$, then the Shannon entropy dignostics detects degeneracy if $E^{-1} \geq 0.000045(0.999^{10000})$. According to these numerical comparison, it seems that the Shannon information entropy diagnostics reduces resampling merely when N is large. However, this appears not to be the case in practice. More precisely, the inverse of the Shannon information entropy value is bounded above by $\prod_{i=1}^{N} C_i$, and the value like $C_i = 0.999$ seldom turns up, which implies that the geometric decay accelerates towards zero even with small N. The follow simulation example demonstrates such claim where the Shannon entropy diagnostics performs well with small particle counts.

Demonstration

The set-up of our demonstration is that given the state sample (time) size T to be 200 and the particle counts N to be 2000. The standardized particle weights of performing the local linearisation particle (resample at each step) has been stored in a T by N matrix. Those weights should not exhibit too much degeneracy since the resampling has already been conducted at each time step in the filtering. I set the threshold of N_{thre} to be $0.5 * N$. The Ess dignostics has been performed on the weights (stored in the T by N matrix). In the end, approximately 24% degeneracy was still detected out of 200 time length. On the contrary, the Shannon information entropy dignostics detects merely approximately 5% out of 200 time steps. In the simulation, those time steps have been flagged out by the Shannon information entropy dignostics appear to

be a subset of that of those by the Ess dignostics. In other words, despite all weights in the demonstration are the standardized weights from SIR (resampled at each step) algorithm, a further Ess dignostics on those weights still detects substantial amount degeneracy.

Figure (3.4) shows three different type of sets of particle weights: undetected particle weight set, detected particle weight set by Ess, and detected particle set by both Ess and Shannon entropy dignostics. Clearly if the set is undetected, which implies that all particle weights within such set carry relatively same weight or same value, and all particle weights contribute to filtering estimation. In the mid plot of Figure (3.4), that is flagged out by the Ess dignostics, which does not behave much differently from the undetected particle weight set (top plot), except few stands out. In addition, the standard deviations for top, mid, and bottom plots in Figure (3.4) are 0.00035, 0.00052, and 0.001, respectively. The critical issues with the Ess dignostics is that resampling will be performed in cases that it may not even be required. However, a potential linear computational reduction can be obtained from utilizing the Shannon information entropy dignostics. The following section verifies this claim, as well as making accuracy comparison among these two dignosticss.

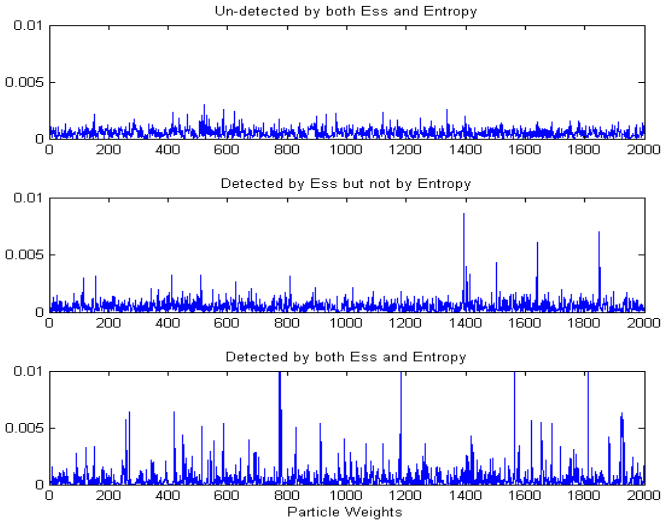

Figure 3.4: The detected and undetected particle set of particle weights.

3.5 Simulations

In this section, the Shannon information entropy diagnostics is embedded into local linearisation particle filter to replace the Ess diagnostics, which forms linearised en-

tropy filter, LPF6. The filter of LPF6 is then compared with the Ess diagnostics coupled with both bootstrap filter and local linearisation particle filter. The second part of this section utilizes LPF6 filter in order to investigate the behaviour of both accuracy and efficiency of aforementioned four of existing resampling schemes. The accuracy comparison derives from the measure of RMSE, which records the deviations between the actual states and its estimates. RMSE has been widely adapted deviation measure in field of particle filtering researches, e.g. Arulampalam et al. [2002]. Alternative ways of measuring deviations are also available, e.g. the mean absolute error in Liverani and Papavasiliou [2006].

The models that will be used for simulation studies have been introduced in section 2 and 3, and they are: the linear and Gaussian model of example (2.3.1.1) and the non-linear time series model of example (2.2.3.1), respectively. In estimation, the importance function is be thought to be the back-bone of each particle filter. For instance, the derived importance density function for linearised entropy filter, LPF6 depends on the linearisation for the linear and Gaussian model and local linearisation for the non-linear time series model, respectively. The detailed discussions and specified derivations of our importance density functions can be found in Doucet et al. [2000a].

3.5.1 Ess diagnostics vs Shannon Information Entropy diagnostics

3.5.1.1 Linear and Gaussian Model

The state time T has been set to be 500 for the simulation on the example of linear and Gaussian model. Table (3.2) compares the results obtained from various particle filters coupled with different degeneracy diagnostics. The comparison has been extended accordingly across 7 different particle counts, as displayed in column 1. Moreover, the RMSE between the linearised entropy filter (LPF6) and linearised Ess filter (LPF5) are comparable, with the former tends to be fractionally better than the latter. Note the Monte Carlo standard deviation is placed in the parentheses. Given the Kalman filter is optimal for linear and Gaussian state space model, the results of Kalman filter (listed in column 2 within Table (3.2)) are predictably smaller than all the particle filters. However, except bootstrap with Ess filter, the rest of filters seem to converge towards the value of Kalman filter as the number of particles increases.

Besides the accuracy comparison, Table (3.3) demonstrates the computational assessment of different degeneracy dignosticss through percentage of resampling mea-

sure. To be more specific, the percentage of resampling is the fraction of number of detected degeneracy over the time length T. The percentages of the linearised entropy filter, LPF6 (displayed in column 4) are smaller than the linearised Ess dignostics, LPF5 across all particle counts. The resampling percentage with LPF6 is invariably up to 20 percent less than LPF5. To summarise, in the linear and Gaussian state space model, our newly proposed Shannon information entropy dignostics (denoted as LPF6) has the capability of producing comparable performance to the existing one, but with a smaller computational cost.

Table 3.2: The RMSE of Kalman filter and Particle filters for linear and Gaussian model, with $N_{thre} = 0.75 * N$ and Monte Carlo repetition R to be 100 runs.

RMSE	Kalman (LPF1)	Bootstrap (LPF2)	Bootstrap Ess (LPF3)	Linearised Ess (LPF5)	Linearised Entropy (LPF6)
N=100	0.7825 (0.0026)	0.8256 (0.0032)	0.8655 (0.0034)	0.8295 (0.0032)	0.8277 (0.0032)
N=250	0.7881 (0.0026)	0.8221 (0.0031)	0.8677 (0.0033)	0.8246 (0.0031)	0.8240 (0.0031)
N=500	0.7880 (0.0025)	0.8172 (0.0030)	0.8604 (0.0032)	0.8192 (0.0031)	0.8183 (0.0031)
N=1000	0.7852 (0.0025)	0.8163 (0.0030)	0.8577 (0.0032)	0.8177 (0.0030)	0.8172 (0.0030)
N=2500	0.7875 (0.0023)	0.8146 (0.0030)	0.8574 (0.0031)	0.8148 (0.0030)	0.8148 (0.0030)
N=5000	0.7834 (0.0025)	0.8139 (0.0031)	0.8575 (0.0032)	0.8142 (0.0031)	0.8141 (0.0031)
N=10000	0.7811 (0.0025)	0.8100 (0.0031)	0.8543 (0.0032)	0.8104 (0.0031)	0.8101 (0.0031)

Table 3.3: Percentage of resampling steps of linear and Gaussian model, ESS vs. SIE.

RMSE	Bootstrap Ess (LPF3)	Linearised Ess (LPF5)	Linearised Entropy (LPF6)
N=100	62.82	50.82	33.62
N=250	63.16	52.64	33.91
N=500	63.06	53.17	33.92
N=1000	63.47	54.19	34.47
N=2500	63.22	54.13	34.25
N=5000	63.22	54.19	34.00
N=10000	63.22	54.42	34.30

3.5.1.2 Non-linear Time Series Model

For the non-linear time series model as illustrated in Example (2.2.3.1), the task of obtaining the optimal filter (as the Kalman filter for linear and Gaussian model) can be a formidable one. The state time T of the non-linear time series is set to be 500, with the particle counts takes to be 7 different amount across from being 100 to $10,000$. In Table (3.4), the results in the column (4) suggest that the linearised particle filter with entropy diagnostics provide comparable precision to particle filters with the Ess diagnostics.

Table (3.5) compares the percentage of resampling obtained from the linearised

entropy filter, LPF6 and linearised Ess filter, LPF5. The values are at least 15 percent less with LPF6 than it is for LPF5. This result is invariably maintained through all particle counts.

Table 3.4: The RMSE of particle filters for non-linear time series model, with $N_{thre} = 0.5 * N$ and Monte Carlo repetitions R to be 100 Runs.

RMSE	Bootstrap (LPF2)	Bootstrap Ess (LPF3)	Linearised Ess (LPF5)	Linearised Entropy (LPF6)
N=100	5.3483 (0.0736)	5.7073 (0.0697)	5.6255 (0.0801)	5.6653 (0.0790)
N=250	5.1414 (0.0727)	5.5720 (0.0689)	5.3287 (0.0766)	5.3490 (0.0789)
N=500	5.0277 (0.0724)	5.4838 (0.0668)	4.9941 (0.0755)	5.0757 (0.0760)
N=1000	4.9725 (0.0734)	5.4550 (0.0682)	5.0916 (0.0775)	5.0058 (0.0760)
N=2500	4.9819 (0.0724)	5.4650 (0.0668)	4.8629 (0.0754)	4.8679 (0.0763)
N=5000	4.9399 (0.0740)	5.4495 (0.0677)	4.8160 (0.0771)	4.8565 (0.0754)
N=10000	4.9251 (0.0743)	5.4305 (0.0689)	4.7460 (0.0780)	4.7598 (0.0762)

Table 3.5: Percentage of resampling steps of non-linear time series model, ESS vs. SIE.

RMSE	Bootstrap Ess (LPF3)	Linearised Ess (LPF5)	Linearised Entropy (LPF6)
N=100	41.74	65.42	47.30
N=250	41.53	71.35	52.49
N=500	41.95	73.89	56.44
N=1000	41.79	75.32	59.71
N=2500	41.05	77.18	63.43
N=5000	42.03	78.06	65.05
N=10000	41.63	79.09	66.17

3.5.2 Resampling Schemes Comparison

In this section, I look into an additional problem, which is on the extension to the resampling schemes comparison by Douc et al. [2005]. For all reality problems involving the implementation of particle filtering, the desire for infinite and extremely large number of particle is infeasible due to the computational constraint of time and cost. Under the frame of finite particles, we hope to provide an empirical guidance over the selection of resampling schemes through simulation experiments derived from two of the prototype state space models. The experiments will be running on the aforementioned four different resampling schemes. More specifically, the performances of those resampling schemes will be diagnostics by embedding them into local linearisation particle filter with the Shannon entropy diagnostics, which is the LPF6 in section 2. Moreover, in order to have a broad picture over the attributes of each individual scheme, I proceed as follows: first, I set the particle counts to vary whilst the state sample size to be fixed. This would allow us to perceive how each resampling scheme

would behave when the number of particle counts change. Secondly, I allow the state sample size to vary while set the particle counts to be fixed.

In practice, despite particles have to be finite, the question of what is the optimal number of particles in a specific problem remains as a debatable topic in the literature. Doucet and Johansen [2008] and Arulampalam et al. [2002] believe that number of particles to be employed should be determined largely by the selection of importance density distribution. Such an answer seems far from being satisfying, possibly with the thought of the rule of thumb may indeed be infeasible to obtain. However, we leave this debate as it is given the concern in the very chapter is to observe the performance of resampling schemes.

3.5.2.1 Linear and Gaussian Model

For the linear and Gaussian model, I examine the precision of each resampling scheme. Their results are displayed in Figure (3.5), which contains the RMSE of all four resampling schemes in particle filter LPF6 at various particle counts whilst set the state size to be fixed ($T = 500$). All four resampling schemes produce higher RMSE with small particle counts, and then the value of RMSE stabilizes as the selected particle counts become larger. Moreover, the overall behaviour of each resampling schemes demonstrated in Figure (3.5) are very similar at different particle counts.

The same behaviour is exhibited in the Figure (3.6), where the state sample sizes are set to vary whilst the number of particles is fixed ($N = 2000$) . Moreover, in Figure (3.6), there appears to be a sharp drop of RMSE at state sample size of 50, then it bounces back as the state size dimensions increase. However, the RMSE soon remains stable as the state size dimension reaching 1000. Moreover, through the increment on the state size, all RMSE of the resampling schemes are no different from each other. I suspect such an 'up-and-down' phenomena could be potentially due to the fact of additional 'noise' introduced by resampling, which exhibit rather strongly when there is only few states in the filtering estimation. Such conjecture will require further investigation. Finally, the specifications of these two simulation studies are: the Monte Carlo repetitions is set to be $R = 50$ and the threshold $N_{thre} = 0.75 * N$, where N is the number of particle counts.

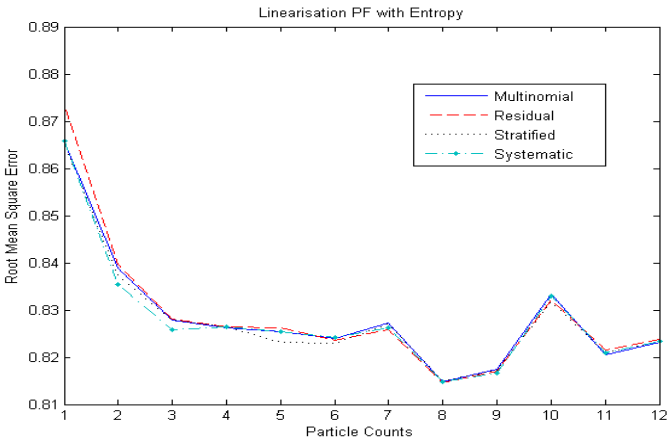

Figure 3.5: The RMSE comparison of resampling schemes using linear and Gaussian model. 12 different particle counts are considered, which has been listed in the horizontal axis, that are $(25, 50, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000)$.

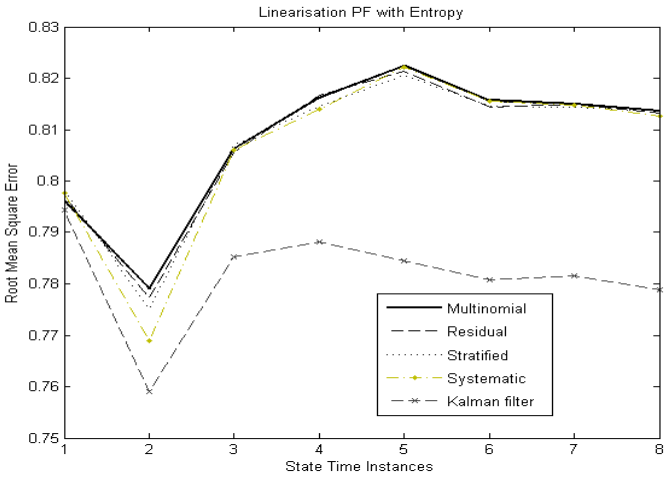

Figure 3.6: The RMSE comparison of resampling schemes using linear and Gaussian model. 8 different state sample sizes are considered, which has been listed in the horizontal axis, that are $(25, 50, 100, 250, 500, 1000, 2500, 5000)$.

3.5.2.2 Non-linear Time Series Model

For the non-linear time series model, with the Monte Carlo repetitions $R = 50$ and threshold at $0.75N$, the behaviour of those four resampling schemes behaves similarly

in both cases of particle counts and sample state are changing, as they are displayed in Figure (3.7) and (3.8), respectively. In Figure (3.7), the RMSE of each resampling scheme remain closely throughout. On the contrary, for case with the changing of state sample size , the behaviour of the resampling schemes among themselves are slightly more unsettled as compare their behaviour for case in Figure (3.7). However, the RMSE of all resampling schemes remain comparable.

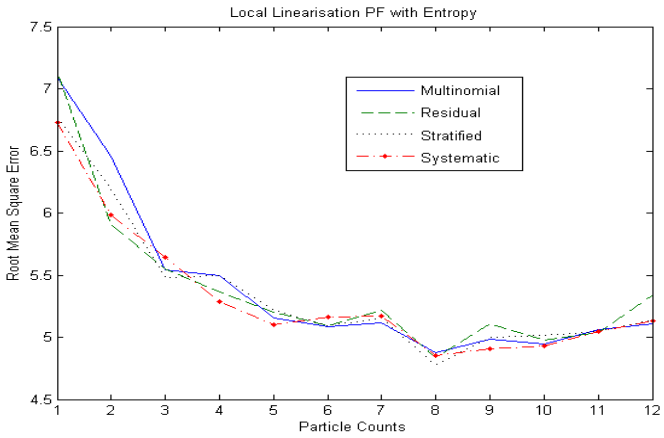

Figure 3.7: The RMSE comparison of resampling schemes using non-linear time series model. 12 different particle counts are considered, which has been listed in the horizontal axis, that are $(25, 50, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000)$.

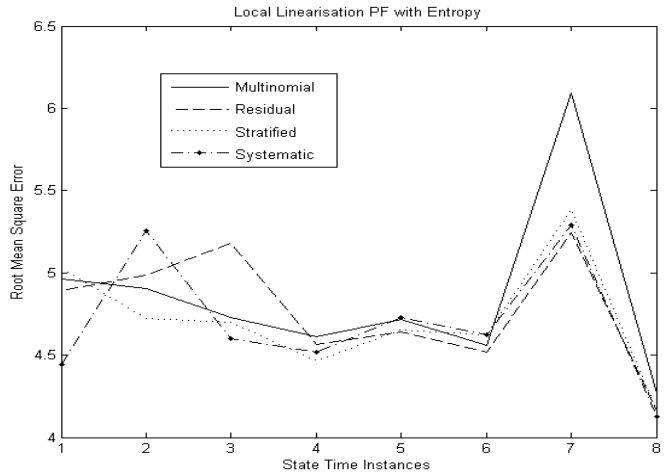

Figure 3.8: The RMSE comparison of resampling schemes using non-linear time series model. 8 different state sample sizes are considered, which has been listed in the horizontal axis, that are $(25, 50, 100, 250, 500, 1000, 2500, 5000)$.

A brief summary leads to perceive that: providing the evidence of the simulation studies of both of the linear and Gaussian model and the non-linear time series model, with the constraint of finite amount particles, all four resampling schemes have almost identical RMSE values. No trace has been found over the concerns of multinomial resampling produces larger estimation deviations over residual and stratified resampling. Building upon on the resampling schemes comparison using particle filter with the Shannon information entropy diagnostics, another investigation is to conduct similar comparison for particle filter with the Ess diagnostics. Hence it would provide additional information regarding to the comparison between the Shannon information entropy diagnostics and Ess diagnostics.

3.5.3 Computational Efficiency

The previous simulation results (considering the scenarios of the changing of both particle counts and state sample sizes) have demonstrated that all four resampling schemes produce comparable estimation accuracy. I further the resampling schemes comparison by examining their computational efficiency, in which the efficiency will be measured in terms of CPU time. Such comparison will be purely for the purpose of providing a way for empirical researchers in their selection of the most suitable resampling schemes for their respective problems. To do so, I use the linear and Gaussian model as the basis model, and then apply the local linearisation particle

filter with Shannon information entropy diagnostics to different resampling schemes. Within this set-up, two distinct state sample sizes 100 and 500 and three various particle counts 100, 500 and 1000 have been considered simultaneously. Similar examinations can be applied to other state space models such as the non-linear time series model.

Table (3.6) reveals the computational efficiency of the four different resampling schemes discussed in the local linearisation particle filters. For example, given state time instances to be 500 and number of particle counts to be 1000, the average computational time to complete local linearisation filter with systematic resampling is approximately 0.3344 seconds, whereas the time of that local linearisation filter with multinomial resampling is almost 143 folds more than that of systematic resampling. Such difference (multinomial vs. systematic) will be even greater when the number of particles increases. Table (3.6) further demonstrates that both systematic resampling and stratified resampling in our simulation requires less computational time than the others. Moreover, as far as finite particle counts are used, the multinomial resampling of original idea of Gordon et al. [1993] has a serious drawback in practice over its computational time efficiency. Despite the results of preceded computational examinations may vary among problems, it however can still be adapt as a trail exercise in determining the most suitable resampling scheme before embarking on comprehensive work.

Table 3.6: CPU time of each resampling scheme implemented within the local Linearisation filter, time is measured in seconds (that is averaged over 50 Monte Carlo runs) .

	Number of particles	resampling	100	500	1000
State	100	multinomial	0.1141	2.4144	9.4359
		residual	0.0537	1.0700	4.1803
		stratified	0.0184	0.0681	0.1409
		systematic	0.0138	0.0362	0.0672
State	500	multinomial	0.5537	12.2994	47.9787
		residual	0.2791	5.4800	21.2959
		stratified	0.0931	0.3488	0.7131
		systematic	0.0619	0.1747	0.3344

3.6 Conclusion

The conclusion of this chapter is two fold: firstly, in sequential importance sample and resampling, the Shannon information entropy diagnostics offers a linear computational reduction over the effective sample size diagnostics. Moreover, this reduction

introduce no effect to the precision of the estimation of utilizing sequential importance sample and resampling. The amount computational efficiency gain with the Shannon information entropy diagnostics will depend on the underlying model that I aim to estimate. Secondly, for finite particle counts, existing resampling schemes provide almost identical estimation in terms of root mean squared error. However, according to the derived evidences from our simulation studies, stratified resampling and systematic resampling possess computational time efficiency over multinomial resampling.

Regarding to future research, the Shannon information entropy diagnostics can certainly be applied to all sequential importance sample and resampling particle filters such as: auxiliary particle filter, regularized particle filter, and so on. Combination of different resampling schemes will certainly worth further investigation, where their attributes can be compared in similar fashion as in this chapter. Finally, considering the diagnostics decision of resampling, further study should be dedicated in knowing the possible existence of a greatest lower bound for any given set of particle weights.

3.7 Appendix

All the simulations and calculation of CPU times are coded with Matlab, then they are performed on Matlab 7.10.0 (R2010 a) on a Desktop computer with Intel(R) Core(TM) 2 Duo, CPU of 3.00GHZ, and RAM of 3.00GB.

Chapter 4

Modified Entropy Particle Filters

Abstract

The Entropy particle filter of Liverani and Papavasiliou [2006] is an attractive tool to estimate both state variable and unknown parameter in the general state space model. In order to reduce the variation of convergence of unknown parameter and obtain more accurate estimates for state variable, this chapter proposes a set of modifications to the original Entropy particle filter of Liverani and Papvasiliou. Such modification is essentially a process of refinement for the prior parameter distribution. The modified Entropy particle filter has the same uniform convergence property as the original Entropy particle filter. Simulation experiments demonstrate that the modified Entropy particle filter outperforms its predecessor in terms of state variable estimation; in addition, it also exhibits superior statistical stability in estimating the unknown parameters.

4.1 Introduction

Estimation of static parameters in non-linear and non-Gaussian state space model or general state space model has been a long-standing problem in spite of the various attempts that have been made in the past two decades.

General state space models, also known as Hidden Markov models (HMM) in the statistics literature, they have the capability to handle a remarkably wide range of economic time series processes. The advantage of the general state space model is that it can model the behaviour of various processes separately and then put the substructures together to form a concrete system for the processes. Earlier works on general state space models are Kitagawa [1987], Gordon et al. [1993], Liu and West

[2001], and among many others.

For general state space models with unknown static parameters, the estimation task becomes much more complicated than merely state variables estimation whilst the parameters are known. An approach that involves the artificial evolution of parameters in Liu and West [2001] assumes the parameters behave as Markov chains that evolve over time. Other approaches like Doucet and Tadic [2003], estimate the parameters through the aid of computing the derivatives of the particle filter. Kantas et al. [2011] provided an extensive summary over different parameter estimation approaches for general state space models.

Liverani and Papavasiliou [2006] proposed the so called *Entropy based particle filter* for estimating the hidden state variable when the static parameters are unknown in the general state space models. Notwithstanding the claim that the work of Liverani and Papavasiliou [2006] focuses on the estimation of the state variable with the intention of avoiding resampling the unknown static parameters, the Entropy particle filter is capable of estimating the parameters of the general state space models. The simulation results of Liverani and Papavasiliou [2006] indicate that the Entropy particle filter is an extremely promising technique to the estimation of general state space model. More specifically, the Entropy particle filter possesses the ability to produce deviation errors of the state variable that are comparable to those provided by conventional particle filtering techniques. In addition, their experiments showed that providing that the initialized parameters are generated from a reasonable prior distribution, the estimates of unknown parameters converge to the true parameter values.

In order to reduce the convergence variation of the parameters, I propose a set of modification on the Entropy particle filter algorithm that is called the modified Entropy particle filter (MEPF). The modification is essentially re-computing weights in the Entropy particle filter that refines the prior distributions of unknown parameters. The MEPF not only stabilizes the parameter estimation, but also provides more precise inference on state variable. Such improvements have been demonstrated in simulation experiments, and the modification procedure can be universally applied to Entropy particle filter in its general applications.

This chapter proceeds as follows: section 2 discusses the state space models that we are interested in and outline the problems we are dealing with. In addition, I describe the idea of the Entropy particle filter, and outline its implementation procedures. The modified Entropy particle filter and its algorithm are explained and discussed in section 3. Section 4 performs several experiments on the application of both MEPF and EPF to linear and Gaussian state space model. Finally, section 5

draws the conclusions that are derived from our simulation experiments. Moreover it points out potential problems that are associated with the application of the Entropy particle filter.

4.2 Particle Filters

4.2.1 Setting

This chapter extends the Entropy particle filter of Liverani and Papavasiliou [2006] to reinforce the estimation capabilities on general state space models. Hence, for the sake of notation consistency, the following setting will be in-line with the authors, and therefore such setting will be slightly different from the one appeared in the introduction and Chapter 2 of the thesis. The system takes the following form:

$$\begin{cases} X_{t+1} \sim K_{\theta_t}(X_t, .) \\ \theta_{t+1} = \theta_t. \end{cases} \tag{4.1}$$

where θ_t has been referred as a non-dynamic component. This non-dynamic component corresponds to the fixed but unknown parameter, and the parameter θ_t is assumed to takes values in a compact subset of a Euclidean space $\Theta \subseteq \mathbf{R}^r$, where r can be understood as the dimensions of the space. [1] Assuming that the initial distribution of X_1 is known, where $X_1 \sim \mu_1$, and the process of $\{X_t\}_{t\geq 1}$ is the so called transition process that has been partially observed through the following observation equation

$$Y_t = h(X_t) + W_t. \tag{4.2}$$

Equation (4.2) is the observation equation, where the observation process $\{Y_t\}_{t\geq 0}$ depends on X_t through function $h(.)$. It also is assumed that we know the distribution of the stochastic process $\{W_t\}_{t\geq 0}$.

The notation of $K_{\theta_t}(X_t, .)$ in equation (4.1) defines the transition density. The subscript of θ_t indicates that for the transition process or the Markov process to be ergodic, [2] one would require additional information on the non-dynamic component, e.g. knowing the fixed value of θ.

[1] One can think of the inducing compactness follows natural from the use of metric space $\mathbf{R}^r$, which it is known that any compact subset of a metric space $\Theta \subseteq \mathbf{R}^r$ is closed and bounded. In addition, given it is $\mathbf{R}^r$, therefore closeness and boundedness induce compactness.

[2] Definition of Ergodicity has been given in page 16 of Chapter 2 in the thesis.

The strong motivation for studying the above system (defined through equation (4.1) and (4.2)) is that such system can be employed to model many real world information processes, e.g. to the monthly inflation data (as economic theory claims) is associated with a unobserved process that is called natural rate unemployment; the missile movement tracking system, and the underlying volatility of exchange rate . The goal can be summarized as: estimate the hidden state sequence X_t and possibly parameter θ, provided all the information up to time instance t. The observed information can be formalized into an information set such that $\mathfrak{I}_t = \{y_0, y_1, \cdots, y_t\}$. The applicability of particle filters become an extremely useful tool once we consider to estimate the dynamic system that is in a non-linear and (or) non-Gaussian format.

4.2.2 Estimation

4.2.2.1 States Estimation

Consider a model where θ is assumed to be known, the focus will be entirely on filtering the state variable X_t. The filtering can be understood as the process of obtaining the target probability distribution function $p_\theta(x_{t+1}|\mathfrak{I}_{t+1})$ via

$$p_\theta(x_{t+1}|\mathfrak{I}_{t+1}) \propto g_\theta(y_{t+1}|x_{t+1})p_\theta(x_{t+1}|\mathfrak{I}_t). \tag{4.3}$$

The derivation of equation (4.3) is the result of a recursive two stages process: prediction and update, which has been shown in detail in Chapter 2 of the thesis, and also explained in Gordon et al. [1993] and Arulampalam et al. [2002]. A set of comments will be made accordingly on each of the individual terms in equation (4.3) in spite of the omission of derivation. $p_\theta(x_{t+1}|\mathfrak{I}_t)$ is the so called predictive distribution of x_{t+1}, which can be obtained by integrating out of x_t from the joint density $p(x_{t+1}, x_t|\mathfrak{I}_t)$. [3] The term $g_\theta(y_{t+1}|x_{t+1})$ is the likelihood function.

In the state variables estimation with fixed parameter θ, the particle filtering produces the following sets at time stance t

$$\{x_t^i, \theta^i : i = 1, \cdots, N\} \quad \theta^i = \theta^j = c, \text{ for } i = j$$

and their corresponding standardized weights such as

$$\{\tilde{\omega}_t^i : i = 1, \cdots, N\},$$

[3] Strictly speaking, we need to invoke the Lebesgue measure over this integration.

where c denotes some known real value. Then the sets $\{x_t^i, \theta^i\}_{i=1}^N$ and $\{\tilde{\omega}_t^i\}_{i=1}^N$ approximate the target distribution $p_\theta(x_t|\mathcal{I}_t)$ at time stance t. A snapshot of all the reason is that constructing analytical target probability density function is feasible for limited cases such as the Kalman filter for linear and Gaussian models. Once the system is non-linear, approximation for the target density function $p_\theta(x_{t+1}|\mathcal{I}_{t+1})$ can be made through the application of particle filters. An extensive review of different type of particle filters can be found in Arulampalam et al. [2002].

4.2.2.2 States and Parameters Estimation

In reality, situations tend to be more complicated. The inclusion of non-dynamic component or unknown fixed parameter θ, as described in equation (4.1) has been proven to be useful from modelling perspective. However, working with the involvement of this additional unknown parameter has turned out to be more complex than merely handling the estimation of state variables along. The following discussion outlines the estimation framework with unknown parameter.

The state equation (4.1) in the previous setting can be represented in following functional function (as in the introduction of the thesis)

$$X_{t+1} = \psi_\theta(X_t, V_t, \theta). \tag{4.4}$$

The addition of θ on the right hand side indicates that we are dealing with both state and parameter estimation; V_t denotes an independent and identically distributed stochastic process. Alternatively, the unknown parameter can be inserted into the aforementioned predictive density function and likelihood function which become

$$f_\theta(x_t|x_{t-1}, \theta) \quad \text{and} \quad g_\theta(y_t|x_t, \theta). \tag{4.5}$$

This means that the probability density functions are known if they are conditional on the parameter θ. Subsequently, the objective becomes constructing the joint posterior distribution $p(x_t, \theta|\mathcal{I}_t)$, which by the Bayes' theorem, can be expressed as

$$\begin{aligned} p_\theta(x_{t+1}, \theta|\mathcal{I}_{t+1}) &= \frac{p_\theta(y_{t+1}|x_{t+1}, \theta)p(x_{t+1}|\theta, \mathcal{I}_t)p_\theta(\theta|\mathcal{I}_t)p_\theta(\mathcal{I}_t)}{p_\theta(y_{t+1}|\mathcal{I}_t)p(\mathcal{I}_t)} \\ &\propto p_\theta(y_{t+1}|x_{t+1}, \theta)f_\theta(x_{t+1}|\theta, \mathcal{I}_t)p_\theta(\theta|\mathcal{I}_t). \end{aligned} \tag{4.6}$$

The above equation indicates that the density function $p_\theta(\theta|\mathcal{I}_t)$ will be an important piece of ingredient in the filtering process. However, this probability density function is unknown to us. The first two probability density functions on the right hand side

are given by equation (4.5). Therefore, the situation that lies ahead is that we have the usual problems of the state estimation outlined previously and the additional issue of the estimation of $p_\theta(\theta|\mathcal{I}_t)$. Notice that the term $p_\theta(\theta|\mathcal{I}_t)$ drops out if θ is known, and then equation (4.6) simplifies to become the state estimation of equation (4.3) .

4.2.3 Entropy Particle Filter

This section reviews the Entropy particle filter and related knowledge. The following description derives from the introduction on particle filter in Liverani and Papavasiliou [2006], which differs from our previous discussion of particle filters only from the notational aspect.

The standing-point of Liverani and Papavasiliou [2006] aims to avoid resampling over the parameters and construct a particle filter that will converge uniformly in time. To do so, they take an approach of weighted average of particle filters whilst treating parameters θ as constant, and also able to sample θ from a 'reasonable' prior distribution u. More specifically, suppose a set of independent samples $\{\theta_j\}_{j=1}^M$ is generated from distribution u, it then would result a total of M particle filters that correspond to M generated θ values. The particle filter of unknown parameters is therefore defined as:

$$\widetilde{\Phi}_t^{M,N}(\mu_1 \otimes u) = \sum_{j=1}^{M} \omega_t(\theta_j)\Phi_n^N(\mu_1 \otimes \delta_{\theta_j}), \tag{4.7}$$

where μ_1 is the previous defined initial distribution of state variable X_0 and u is the prior distribution of θ. The term of $\mu_1 \otimes u$ denotes the particle filter is constructed based on the outer products of initial particle set (from a probability measure of μ_1) and the parameter samples (from a probability measure of u). The left hand side of the equation denotes the particle filter for unknown static parameters. The term $\Phi_n^N(\mu \otimes \delta_{\theta_j})$ on the right hand side represents the particle filter where the unknown static parameter has been fixed to θ_j. In addition, the term $\omega_t(\theta_j)$ can be computed in a way that it in fact approximates the likelihood of parameters θ being given as θ_j:

$$\omega_t(\theta_j) \approx p(\theta_j|\mathcal{I}_t), \tag{4.8}$$

where t indicates time instance. Equation (4.7) is indeed another representation of equation (4.6). However, the new representation conveys extra information regarding

the possible approach that deals with the estimation problems in general state space models. This possible approach takes into account the fact that the correct initial distribution δ_a of parameter θ is unknown, where a is the truth of θ. It then disentangles this problem by initializing parameters θ from a good prior u, and associate it with particle filter, which will eventually converge to the optimal filter as if it has been correctly initialized according to δ_a.

At this point, computing the weighted parameter functions $\omega_t(\theta_j)$ that correspond to the particle filter needs immediate attention. This is actually where the idea of Entropy comes in and plays the crucial part in approximating the likelihood of parameters $p_\theta(\theta|\mathfrak{I}_t)$. More specifically, it starts by assuming that there is a one-to-one correspondence between each θ and the limiting distribution of the observation process v_θ. Furthermore, it assumes that for each θ the observation process satisfies the large deviation principle,[4] it therefore allows to derive the weights of initialized parameters at time t as

$$\begin{aligned} \omega_t(\theta_i) &= \frac{e^{\sum_{t=1}^{T} \log\left\{T v_{\theta_i}(y_t)\right\}}}{\sum_{j=1}^{M} e^{\sum_{t=1}^{T} \log\left\{T v_{\theta_i}(y_t)\right\}}} \\ &= \frac{1}{\sum_{j=1}^{M} e^{\sum_{t=1}^{T} \log\left\{\frac{v_{\theta_j}(y_t)}{v_{\theta_i}(y_t)}\right\}}}, \end{aligned} \tag{4.9}$$

where $i, j = 1, \cdots, M$. Note that the derivations of Entropy measure of Liverani and Papavasiliou [2006] can be referred to in the Appendix of this paper. If we plug the above expression into either equation (4.6) or (4.7), we achieve our approximation for the term of parameter weights density $p_\theta(\theta|\mathfrak{I}_t)$. Moreover, the value of those limiting distributions can be obtained provided we know the value of θ_j and observations y_t, and therefore the whole process does not involve the particle filters. From the implementation stand-point, the second line of equation (4.9) is better in computation since it avoids the troublesome sum of large number of small values.

A point that has been made in Liverani and Papavasiliou [2006] was that, as T becomes large, the weight or mass of parameters will be concentrated or close to one $\theta \in \{\theta_1, \cdots, \theta_M\}$, which would be the one that minimizes the entropy distance or distribution deviation between v_θ and v_a. Moreover, it would lead to the truth of a,

[4]The mathematical definition of large deviation theory is beyond the scope of this thesis. However, the following explanation may be offer some insights. In some sense, the large deviation principle is an analogue of weak convergence of probability measures, but one which takes account of how well the rare events behave. Hence it allows to discover the behaviour of each sampled parameters given the observations towards the true value.

provided we could generate infinite number of sample parameters from the prior of u and T is large.

4.2.4 The Algorithm

The following algorithm implements the Entropy particle filter delineated above. Note that since the parameters θ have been assumed to be static, therefore an index of time will be omitted, and disclosed as θ.

Algorithm 1: Entropy particle filter.

1. **Initialization**. At $t = 1$, we generate M independent random samples from the prior distribution u. These M samples are said to be the set of initialization parameters $\{\theta_j : j = 1, \cdots, M\}$. For each sample θ_j at time instance t, we form the corresponding weight $\omega_t(\theta_j)$ and the number of particles N_t^j of the particle filter. Nevertheless, the initial weights are assigned to be

$$w_1(\theta_j) = \frac{1}{M}, \quad j = 1, \cdots, M,$$

and the number of particles that correspond to each particle filter at given sample θ_j is

$$N_1^j = N \quad j = 1, \cdots, M.$$

Hence, according to the representation of equation (4.7), the particle filter with unknown static parameters at time $t = 1$ takes the following form

$$\widetilde{\Phi}_t^{M,N}(\mu_1 \otimes u) = \frac{1}{M} \sum_{j=1}^{M} \Phi_t^N(\mu_1 \otimes \delta_{\theta_j}), \tag{4.10}$$

where $\Phi_t^N(\mu_1 \otimes \delta_{\theta_j})$ is the particle filter at time $t = 1$ corresponding to the initialized parameter θ_j.

2. **Weighted parameters computation**. The initialization does not involve the calculation of parameters weight such as $\omega_t(\theta_j)$. For $t > 0$, the weighted parameters are calculated according to equation (4.9). The number of particles for each parameter θ_j are rounded up to the nearest integers via

$$N_t^j = \lceil \omega_t(\theta_j) \times (N \times M) \rceil, \tag{4.11}$$

Where $\lceil \cdot \rceil$ denotes the *ceiling* of some value. One would expect the total number

of particles that are used at time t will be

$$N \times M \leqslant \sum_{j=1}^{M} N_t^j \leqslant (N+1)M. \tag{4.12}$$

3. **Evolution** The number of particles N_t^j and the weighted parameter $\omega_t(\theta_j)$ need to be computed from time instance of $t = 1$, and ceased it when time reaches at T. It would eventually provide us with a $M \times T$ matrix that contains number of particles at each time instance for all initialized parameters. Moreover, this part of the task has to be completed before the running of particle filer. For example, the number of particles that correspond to each sample parameters θ_j are N at the initialization, however the number of particles at time T might be greater or less than the initial amount N that depends of the weight of sample parameters. A remark would be: the total number of particles that are utilized by particle filter at time t is $\sum_j^M N_t^j$. Hence this should not be confused with the previous notation where N denotes the number of particles as in Chapter 2.

4. **Particle filter**. To obtain the estimates of the states, we use the following particle filter

$$\widetilde{\Phi}_t^{M,N}(\mu_1 \otimes u) = \sum_{j=1}^{M} \omega_t(\theta_j) \Phi_t^{N_t^j}(\mu_1 \otimes \delta_{\theta_j}). \tag{4.13}$$

   The difference between equation (4.10) and (4.13) is the number of particles that are implemented at each time instance t given sample parameter θ_j. More precisely, in equation (4.13), the number of particles utilized in particle filter have been determined by the weighted parameter functions, whereas the particle numbers for equation (4.13) was pre-set to be N.

4.3 Modified Entropy Particle Filter

In the first stage of calculating the weighted function of parameters, the estimates of the static parameters at each time instance can be obtained by taking the sum of weighted function of parameters times their respected initialized sample parameters.

For example, the estimates of the parameters at time t would be

$$\hat{\theta}_t = \sum_{j=1}^{M} \theta_j * \tilde{\omega}_t(\theta_j), \tag{4.14}$$

where $\tilde{\omega}_t(\theta_j)$ denotes the standardized weight function of parameters at time t given the sample parameters to be θ_j. Such estimates converge to the true parameters provided the state time t is sufficiently large and the initialized sample parameters are generated from a reasonable prior distribution u. The simulation example in the following section demonstrates that the sum of weighted parameters indeed converges to the true value.

Despite the subtle notion of acquiring the estimate of the static parameter with the Entropy particle filter, a more precise calculation of the estimates may be possible if one has more information about the initial prior distribution. Following this stream of thought, I propose a set of modifications to the Entropy particle filter, where it refines the prior distribution for the generation of initialized parameters. The validity of our modification relies on the convergence of the estimates to the true parameters. Such fact has been ensured by the employment of limiting distribution as discussed in Liverani and Papavasiliou [2006]. In addition, the uniform convergence of the weighted average of particle filter has been shown in Papavasiliou [2005].

To achieve the refinement for the prior u, I propose to utilize the set of T estimates acquired from the first stage of weighted function parameter calculation. However, we are merely interested in those estimates that are near the region of the true parameters value. Provided the estimates are converging in time, the establishment of a **break point** will allow to determine the subset of estimates that are needed for the purpose of refinement on the prior density u. By break point, it means for instance, suppose the state time starts from 0 is T, the break point can be any time instance within 0 and T. In the following particle filter algorithm, I adapt the *Golden ratio section* method to determine such a breaking point. The breaking point picks out the estimates values from the Golden section point to time instance T. For example, suppose that we have $T = 2,000$ time states, then the breaking-point would be $\lfloor 2000 \times (1/1.68) \rfloor$. Hence we work with the estimates from $1,236^{th}$ to $2,000^{th}$. Admittedly, other endogenous or systematic ways of determining the break point are possible. This new set of estimate values would allow us to form a more precise prior distribution u_1, and then this new prior u_1 can be used to generate a new set of sample parameters that corresponds the particle filter. By repeating this refinement procedure, we would expect to obtain a more accurate estimation over the static

parameter, as well as the state variables.

The algorithm of modified Entropy particle filter (MEPF) differs from the original Entropy particle filter (EPF) of Liverani and Papavasiliou [2006] in the step of weight computation (step 2 and 3 in Algorithm 1). The modifications are:

Algorithm 2: Modification Procedures.

1. At the completion of step 2 and 3 in the Entropy particle filter algorithm (Algorithm 1) of (4.2.4), given the number of time states to be T and the number of initialized sample parameters θ_1 to be M, then the equation (4.14) will be computed at every time instance for $t = 1, \cdots, T$. Therefore it forms a set of estimates $\hat{\theta} = \{\hat{\theta}_1, \cdots, \hat{\theta}_T\}$.

2. Compute the breaking point via

$$b = \lfloor T * g \rfloor,$$

   where *ad-hoc* $g = 0.618$ is the golden ratio point. Providing the break point, we will be able to form a subset of $\hat{\theta}$ that is $\hat{\theta}_{b:T} = \{\hat{\theta}_b, \hat{\theta}_{b+1}, \cdots, \hat{\theta}_T\}$. Subsequently, we can compute the mean and standard deviation of the set of estimates $\hat{\theta}_{b:T}$ to be $\bar{\theta}^m$ and σ, respectively.

3. Compute the root of mean square variation (RMSV) of $\hat{\theta}_{b:T}$ via

$$RMSV_{\hat{\theta}} = \sqrt{\frac{\sum_{t=b}^{T}(\hat{\theta}_t - \bar{\theta}^m)^2}{(T-b)}}. \tag{4.15}$$

   Note that the above equation (4.15) measures the deviation of among the set of estimates for the parameter θ, which differs from the definition of RMSE for the state variable estimation as it was given by equation (3.1) in chapter 3.

4. (a) if $RMSV_{\hat{\theta}} < \epsilon$, where ϵ denotes a tolerance bound, then go to step 4 in Algorithm 1.

   (b) If $RMSV_{\hat{\theta}} \geqslant \epsilon$, the following procedures will be executed:

      i. Form a new prior distribution $u_1(.)$ derived from information such as: the set of estimates $\hat{\theta}_{b:T}$, $\bar{\theta}^m$, and σ.

      ii. Simulate a new set of M initialized sample parameters from $u_1(.)$.

      iii. Go back to step 1 above and repeat the steps until 4.a is satisfied.

The algorithm 2 of (4.3) can be embedded into the Entropy particle filter algorithm of (4.2.4), which forms the modified Entropy particle filter. Note that setting

the value for the tolerance bound ϵ will be pre-specified in the algorithm. It would certainly be more interesting if the tolerance bound can be determined endogenously within the algorithm.

4.4 Experiment

In this section, I consider the linear and Gaussian state space model that was proposed in Liverani and Papavasiliou [2006],

$$X_t = \theta X_{t-1} + V_t, \quad \theta \in (0,1) \tag{4.16}$$
$$Y_t = X_t + W_t, \quad t = 1, \cdots, T \tag{4.17}$$

where V_t and W_t are standard normally distribute and independent random processes, and parameter θ is assumed to be static but unknown. Y_t is observable and we aim to learn the parameter θ and state variable X_t. The limiting distribution of Y_t can be derived as

$$Y_t \sim \mathcal{N}(E(X_t), V(X_t) + 1).$$

Given $t \to \infty$, we have

$$E(x_t) = 0,$$
$$V(x_t) = \frac{1}{1-\theta^2}.$$

Hence the limiting distribution $v_\theta(Y_t)$ is

$$v_{\theta_j}(Y_t) = \frac{1}{\sqrt{2\pi V_{\theta_j}(y_t)}} \exp\Big\{ -\frac{1}{2}\frac{y_t^2}{V_{\theta_j}(y_t)} \Big\},$$

where

$$V_{\theta_j}(y_t) = \frac{2-\theta^2}{1-\theta^2}.$$

Utilizing the Algorithm 2 in place of the 2nd and 3rd steps of the EPF algorithm, forms the MEPF algorithm. For the purpose of comparison, I implement both MEPF and EPF to the above state space model. Notice that the sampled initialized parameters are assumed from uniform distribution $u(0,1)$ and the tolerance bound $\epsilon = 0.001$, hence the refined distribution within MEPF will also be uniform distribution. The interval for the refined uniform distribution (Algorithm 2 step 4.b) is as follows

- 1. According to the Algorithm 2 point 4.b, given the present model, the uniform interval $u(I_{min}, I_{max})$ will be constructed via

$$I_{max} = \bar{\theta}^m + 4 * \sigma \quad and \quad I_{min} = \bar{\theta}^m - 4 * \sigma,$$

  where $\hat{\theta}^m$ and σ are the mean and standard deviation of the refinement set of selected estimates, which have been discussed in section 3. Notice that the use of $4 * \sigma$ is due to the intention of avoiding potential over and under estimation of the parameter.

- 2.a if $I_{max} < 1$ and $I_{min} > 0$, then $u(I_{min}, I_{max})$ will be the refined prior distribution.

- 2.b if $I_{max} > 1$ and/or $I_{min} < 0$, then $u(min\{\hat{\theta}_{b:T}\}, max\{\hat{\theta}_{b:T}\})$ will be the refined prior distribution.

Figure (4.1) demonstrates the estimates of parameter θ given by both the Entropy particle filter (EPF) and the modified Entropy particle filter (MEPF). In this simulation study, the number of initialized sample parameters $M = 100$ and the number of particles that correspond to each initialized parameter is set to be $N = 100$. In addition, the time states is $T = 2,000$ and the true parameter θ in the system of equation (4.16) is 0.7. Top plot in Figure (4.1) is produced by the EPF with initialized parameter prior distribution setting to be uniform $u(0, 1)$. Despite that the top plot indicates the convergence of the estimates towards the true parameter, the time that it takes for the estimates to settle down occupies at least 20% of total time state. On the contrary, the bottom in Figure (4.1) produced by the MEPF shows the estimates converge immediately to the truth. The reason of such behaviour is that the obtained estimates of MEPF are based on repeated refinements of the estimates of EPF. The variation of the convergence of the estimates associated with the MEPF is clearly much smaller compared to that of the EPF.

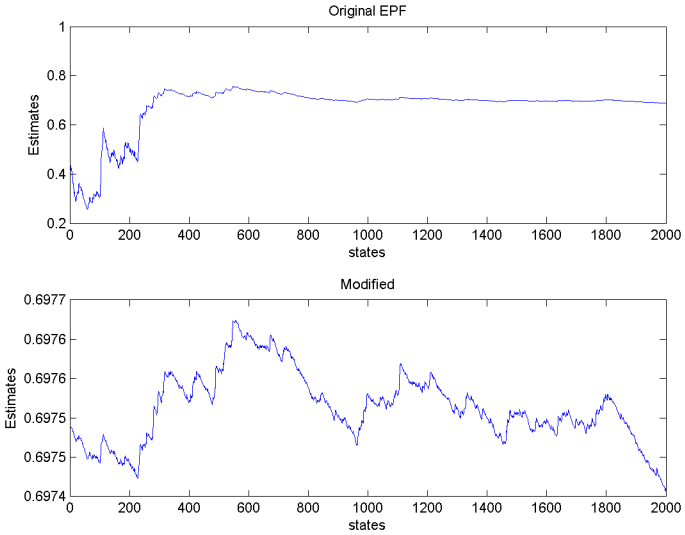

Figure 4.1: The estimates of the parameter obtained through the weighted function of parameter $\omega_t(\theta_j)$ at each time instance.

The Changing of Time State

The absolute deviation or L_1 error will be utilized for the following comparison study between the EPF and the MEPF, which is consistent with the use of deviation measure in Liverani and Papavasiliou [2006]. Note that other measure of deviations such as mean squared error (MSE) that can also be adapted for similar comparison. Provided that the system defined in equation (4.16) is linear and Gaussian, the Kalman filter will be an optimal filter, and therefore will be used as the benchmark for the comparison. The simulation experiment that produces Figure (4.2), the time instances or the length of observations are set to vary from 50 to $10,000$ over 8 different values, and the initialized sample parameters $M = 100$ and number of particles $N = 100$. The dotted line represents the L_1-error of MEPF tends to be lower than the solid line (L_1-error of EPF), except at the size of observations at 50. Furthermore, the L_1-error decreased by significant amount as the size of observations increases, then it stabilizes with no further room of error reductions, e.g. at $T = 50$, the L_1-error of MEPF and EPF are 0.126 and 0.125 respectively, then they are reduced to 0.080 and 0.086 respectively at time states $T = 10,000$. Furthermore, the MEPF improves from the EPF for the state variable estimation by approximately 8 per cent.

Plot b in Figure (4.2) provides the comparison of the root mean squared error (RMSV, as defined in Algorithm 2) of the estimates for the parameter that are given by the EPF and the MEPF, respectively. Overall, the RMSV of the estimates of pa-

rameter with the MEPF tends to be much more stable and smaller than the RMSV produced by EPF.

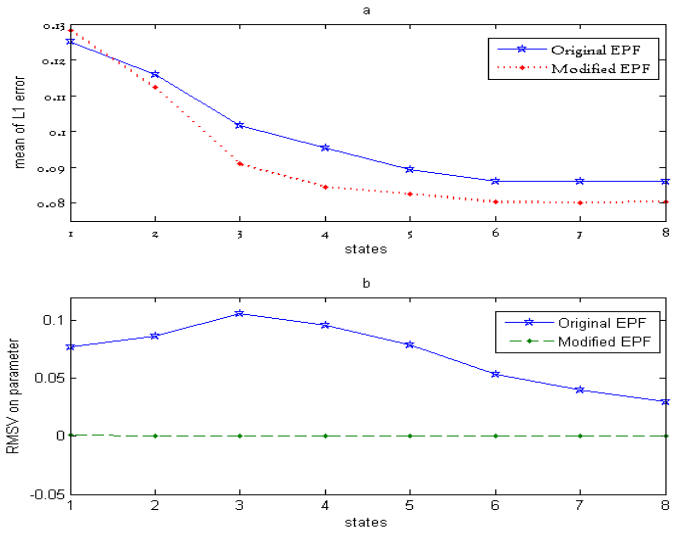

Figure 4.2: The comparison between original EPF and modified EPF, where 8 different observation lengths are examined, as indicated by the horizontal axis with (50, 100, 250, 500, 1000, 2000, 5000, 10000), and MC repetition R equals 50.

The Changing of the Number of Initialized Parameters

The following simulation investigates the performances of both the EPF and the MEPF whilst the number of initialized sample parameters are set to vary from as less as 20 to as much as $1,000$. Once more, the observations $T = 1,000$, number of particles $N = 100$, and the prior distribution for the initialized parameter is uniformly distributed between 0 and 1. Figure (4.3) displays the results that are produced by the methods of EPF and MEPF, where the L_1-error of the state estimates is in plot a and the RMSV of the estimates of parameter is in plot b. These evident results reveal that the MEPF outperforms the EPF at each different number of initialized parameters. In addition, for state variable estimation, the MEPF is capable of producing comparable results that is given by the conventional *bootstrap particle filter* for known parameter in Liverani [2006].

To summarize, the performance of both of the EPF and MEPF depends on two crucial conditions: the number of observations and the prior distribution of initialized parameter. More specifically, the number of observations have to be large for acquiring good approximation with the Entropy measure, which has been shown in the above simulation study of changing time state. On the other hand, the prior distri-

bution where the initialized parameters are generated have significant influence over the accuracy of both state variable and parameter estimation. For example, a simulation example carried out was that, if the prior uniformly distribution changes its interval from (0, 1) to (-1, 1), with the setting of $M = 100$, $N = 100$, and $T = 2,000$, the L_1 error of both the EPF and the MEPF will be around 0.28 compared to the 0.17 given by the bootstrap particle filter for unknown parameter state space model. Hence, one should be aware of these two points while using Entropy particle filters for estimating dynamic model with unknown static parameters.

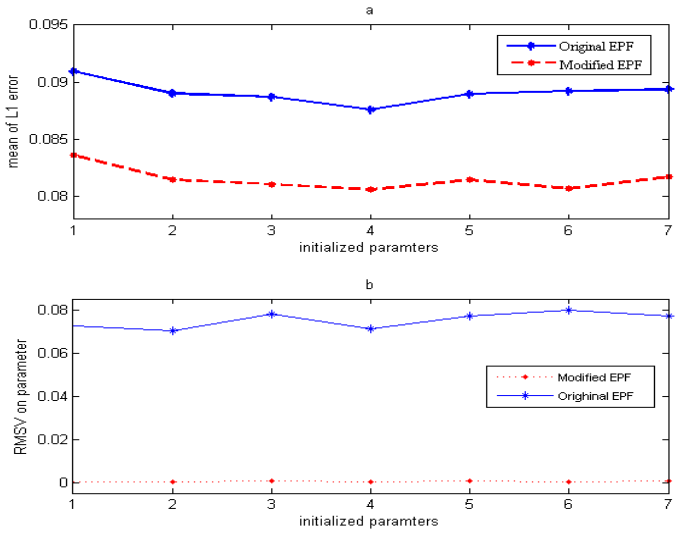

Figure 4.3: The comparison between original EPF and modified EPF. where 7 different sets of initialized sample parameters are examined, as indicated by the horizontal axis with $(20, 50, 100, 250, 500, 750, 1000)$, and MC repetition R equals 100.

4.5 Discussion

The modified Entropy particle filter is an extension of the Entropy particle filter of Liverani and Papavasiliou [2006]. Results derived from the above experiments demonstrate that, given specified error measure, the improvements of MEPF over EPF are not only to the state estimates, but also to the stability of the estimates of the unknown parameter.

Similar comparison procedures can certainly be conducted to state space models that are non-linear and non-Gaussian with multi unknown parameters. This would certainly provide a more complete picture over the performance of both of the Entropy particle filter and the modified Entropy particle filter. Moreover, application

to real economic and financial models will be the next task. However, in the implementation of Entropy particle filter, difficulties may arise due to the non-existence of the close form for the limiting distribution of the observation sequence. Therefore, approximation methods would be required in deriving the limiting distribution even before the Entropy particle filter can be used.

In the direction of future researches, we are making a healthy progress in developing another method - 'truncated particle filter' to overcome the entropy reliance of both MEPF and EPF to the large size of the observations. I hope this newly developed particle filter can give comparable errors to the conventional particle filter with unknown parameter at any size of observations. Another interesting application of entropy particle filters would be to non-ergodic Markov chains state space model. By non-ergodic, it means that the initial value of the Markov chain is assumed unknown. In similar application, Liverani and Papavasiliou [2006] concluded that given the transition process to be non-ergodic, after certain amount of time states, the particle filter converge and behave as like the one in an ergodic case. To avoid the *ad-hoc* method of Golden ratio section, systematic way of determining the breaking point of the estimates sequence for obtaining the refined prior distribution can be exploited, however, it could most likely burden the computational time of the method of MEPF. Finally, a crucial problem associated with Entropy particle filters will be the increasing of the dimension of parameter space. More specifically, as the number of parameters increase, the computation of the weight of the parameter functions become $O(M^d)$ (which becomes computationally cumbersome), where d is the number of parameter and M is the number of initialization of each parameter.

4.6 Appendix

The following derivation of the weighted function of parameters is derived from the entropy measure, wherein it aims to provide the background information for those who are interested to know what exactly the role of entropy in the Entropy particle filters. This introductory material follows closely the work by Liverani and Papavasiliou [2006].

By the large deviation principle, as the number of observations tend to infinity, one would know the limiting distribution v_θ and consequently learn the parameter θ. The likelihood of θ_j should be approximately proportional to the distance of v_{θ_j} from v_a,

where a is the true value θ in the sense that

$$\frac{1}{T}\sum_{t=1}^{T}\delta_{Y_t} \xrightarrow{w} v_a, \quad T \to \infty,$$

where w denotes weak convergence. Given the above observations, we have the following approximation:

$$p(\mathcal{I}_T|\theta) \approx p\big(L_T(y) \in \mathcal{B}(v_a, \epsilon)|\theta\big),$$

where $L_T(y) = \frac{1}{T}\sum_{t=1}^{T}\delta_{Y_t}$, and $\mathcal{B}(v_a, \epsilon)$ is the ball of radius around the distribution v_α with respect to the Levy-Prohorov metric that metrizes the weak convergence of measures. Hence we have following probability density function approximated by the entropy

$$p\big(L_T(y) \in \mathcal{B}(v_a, \epsilon)|\theta\big) \approx e^{-TI_\theta},$$

where I_θ is the appropriate rate function or entropy function. For independent, identically distributed $(i.i.d)$ random variables, the entropy function is the entropy distance between the distributions v_θ and v_a, which is

$$I_\theta \approx \int_{\mathcal{R}^q} \log\Big\{\frac{dv_\alpha}{dv_\alpha}(y)\Big\}v_a(dy).$$

Since the knowledge of the limiting distribution v_α is missing, we replace it by the large deviation, that is $\frac{1}{T}\sum_{t=1}^{T}\delta_{y_t}$, which converges to v_α. Then the entropy function can be computed via a discrete formula as

$$I_T(\theta) := -\frac{1}{T}\sum_{t=1}^{T}\log\{Tv_\theta(Y_t)\}. \tag{4.6.1}$$

Given the above equation (4.6.1), the probability density of observation conditional on the parameters can be expressed as

$$p(\mathcal{I}_T|\theta) \approx e^{\sum_{t=1}^{T}\log\{Tv_\theta(Y_t)\}}. \tag{4.6.2}$$

Now equation (4.8) can be represented as

$$\omega_t(\theta_j) \approx p(\theta_j|\mathcal{I}_t) = \frac{p(\mathcal{I}_t|\theta_j)}{p(\mathcal{I}_t)}. \tag{4.6.3}$$

Substitute equation (4.6.2) into (4.6.3), the approximation of distribution $p(\mathcal{I}_t|\theta_j)$ takes the form of equation (4.9).

Chapter 5

Modified Backward Sampling Smoothing with EM algorithm - Applications to Economics and Finance

Abstract

This chapter demonstrates the attractiveness of particle smoothing in state variable inference and unknown parameters estimation for state space models. I propose a backward weight modification to the $O(TN)$ forward filtering backward sampling (FFBSa) algorithm of Godsill et al. [2004], namely GDW smoothing. The modified GDW (MGDW) smoothing algorithm takes into account the backward information while performing resampling at each time instance, and the MGDW smoothing remains computation complexity of $O(TN)$. A novel approach consists of the MGDW smoothing algorithm and the EM algorithm has been carried out for off-line parameter estimation in general state space models. This novel method has been applied and tested through a series of simulated studies and real economic data applications. In which, I observe that, in terms of parameter estimation for general state space models, this newly proposed method produces comparable results to the existing ones.

5.1 Introduction

In recent years, the focus in modelling time series processes in economics and finance, has shifted from previous linear and Gaussian type of system to less restrictive and perhaps more realistic non-linear and non-Gaussian state space models. The latter models have often been referred as the general state space model or the hidden Markov model. Despite that the attractiveness and usefulness of the general state space model have been shown in modelling observation process, the optimal state variables estimation in such given setting will clearly be infeasible with the conventional Kalman filter. In various approaches to overcome the aforementioned problem, a simulation based technique that has been branded as sequential Monte Carlo (SMC) methods or particle filters have gained massive popularity in many scientific domains, including economics and finance. For example, Fernandez-Villaverde and Rubio-Ramirez [2007] demonstrate and justify the usefulness of particle filtering in facilitating likelihood-based inference in a dynamic business cycle model. The reasons of prevalence of particle filters in recent time have been largely due to their simplicity and accuracy compared to those Kalman based filters such as: the Extended Kalman filter, the unscented Kalman filter, and the grid method. For example, in estimating macroeconomic dynamic models, Fernandez-Villaverde and Rubio-Ramirez [2005] and Fernandez-Villaverde and Rubio-Ramirez [2007] show the benefits of using particle filtering to the Kalman filter. A brief description on particle filtering would be that it is a simulation technique that recursively propagates and updates a set of weighted particles (samples), which forms a empirical measure that approximates the target posterior probability distributions of the state variable. A list of references on the development and the application of particle filtering over the past two decades are Gordon et al. [1993], Liu and Chen [1998], Doucet et al. [2000a], Durbin and Koopman [2000], Doucet et al. [2001], Doucet et al. [2000b], and so on.

On the contrary, particle smoothing did not receive as much attention as particle filtering, at least during the first half of the past two decades. An important reason for that was due to the extensive computational complexity associated with particle smoothing methods. For instance smoothing algorithms such as: the forward filtering and backward smoothing Kitagawa [1987], two-filter smoothing Bresler [1986], and the generalized two-filter smoothing Briers et al. [2010] have computational effort of $O(TN^2)$. Moreover, with these smoothing methods, obtaining of marginal smoothing density has less of interest, as investigations of historical states generally focus on trajectories and hence requires the knowledge of the collection of states together. An ample progress on smoothing was due to Godsill et al. [2004] with the forward filter-

ing and backward sampling (FFBSa) smoothing technique, which has computational complexity of $O(TN)$, and it allows the random generation of entire historical trajectories from the forward joint density. Considering the role that particle smoothing can play in parameter estimation for the dynamic model, this chapter not only investigates the FFBSa liked smoothing algorithm with order $O(TN)$, but also proposes and tests a newly modified FFBSa algorithm.

Over the years, great amount of researches have been devoted to both theory and application of the particle filtering and the particle smoothing in estimating the hidden state variables for general state space models. On the contrary, the attention on the estimation of the unknown parameters within general state space models has only started in recent years. Initial attempts on estimating unknown but fixed parameters were relying on augmenting the state whilst includes the unknown parameters, which transforms the problem to be a complete filtering problem. The representatives of such approach are: Liu and West [2001], Storvik [2002] and Fearnhead [2002]. The work of Liu and West [2001] proposes to introduce artificial dynamics to the fixed parameters, whereas Fearnhead [2002] developed MCMC rejuvenation steps that takes into account the state sufficient statistics from the posterior distribution. The limitation associated with the aforementioned methods have been well documented in the literature in, for instance, the work of Doucet and Tadic [2003] and Kantas et al. [2011].

The most recent development on parameter estimation for general state space models can be broadly divided into two classes: the maximum likelihood approach and Bayesian approach. Moreover, the maximum likelihood approach has further been categorized into two subclasses: the first one derives from the approximation of the derivatives of the particle filter that had been proposed by Doucet and Tadic [2003] and Poyiadjis et al. [2005]. An alternative class is the combination of particle smoothing (PS) and EM algorithm of Briers et al. [2010] . The PS-EM approach avoids the approximations of particle filter derivatives, wherein it has often been considered to be less troublesome and more stable in really applications. Bayesian approach has been heavily based on the use of Markov Chain Monte Carlo (MCMC) method, where the MCMC is embedded into sequential filtering. References on such idea are Andrieu et al. [2010] and Kantas et al. [2011]. The pros and cons of the aforementioned methods in those two classes to the problem of on-line (estimation is performed recursively given the observations are processed sequentially) and off-line (estimation is performed iteratively given the observations are processed as a batch) parameter estimation in general state space models have been summarised in Kantas et al. [2011].

In line with Briers et al. [2010], as well inspired by Kim [2005], the present chapter constructs the modified PS-EM approach to estimate two economic models that have been formed into the general state space format. The modifications are two folds: firstly the $O(TN^2)$ forward filtering and backward smoothing or generalized two-filter smoothing of Briers et al. [2010] have been replaced with $O(TN)$ type of forward filtering and backward sampling such as Godsill et al. [2004]. Subsequently, building upon Godsill et al. [2004], I proposed a more stable version of forward filtering and backward sampling method (MGDW smoothing), which takes into account both forward and backward particle weights during the backward sampling process. The experiment studies have shown that the MGDW smoothing performs at least as good as the smoothing proposed by Godsill et al. [2004]. The modified PS-EM approach is essentially a combination of the MGDW smoothing with EM estimation, which will be referred as the EM-MGDW method. Overall, our simulation experiment results provide comparable results to other parameters estimation approaches that had mentioned above, but computationally less expensive. Moreover, our real application results such as the estimation of the Phillips curves and the Cox-Ingersoll-Ross volatility model demonstrate that the technique of the combination of MGDW smoothing and EM algorithm or the EM-MGDW method can be widely applied in economic and financial modelling. Other methods of off-line parameter learning are: Doucet and Tadic [2003] use maximum likelihood that based on computing the derivatives of particle filter and Andrieu et al. [2010] and Pitt et al. [2012] develop the particle filter Markov Chain Monte Carlo sampler.

The remaining of the chapter is organized as follows: Section 2 presents the statistical model of interest, as well as reviewing the idea of particle filtering. Building upon the knowledge of particle filtering, section 3 studies few existing smoothing techniques. In addition, we investigate the performance of the newly proposed forward filtering and backward sampling method (MGDW smoothing) over the existing ones. Section 4 reviews the role of the EM algorithm in parameter estimation. In Section 5 we examine the performance of newly proposed smoothing algorithms combined with the EM method through simulation study, as well as compare it to their counterpart parameter estimation techniques. Section 6 looks through two real data applications in economics. Finally, section 7 summarises the results and provides some concluding remarks.

5.2 Comparison of Particle Smoothing Algorithms

An alternative approach to state variable inference is particle smoothing. Particle smoothing tends to be computationally more challenging than particle filtering. However, smoothing estimates the distribution of the state sequences at a particular time given the observations are beyond this given time. Therefore, with the additional information provided, one would expect that the trajectory estimates obtained via smoothing will be 'smoother' than those obtained through particle filter.

In the earlier part of particle smoothing literature, the most well-known and applied particle smoothing techniques are: the forward-backward smoothing by Kitagawa [1987], the two-filter smoothing formula by Bresler [1986], and the generalized two-filter algorithm by Briers et al. [2010]. In Kantas et al. [2011], the forward-backward smoothing has been referred as a class of methods that includes the forward filtering backward smoothing (FFBSm) of Doucet et al. [2000b] and the forward filtering backward sampling (FFBSa) of Godsill et al. [2004]. The two-filter smoothing consists of forward filter and backward information filter. The difference between the two filter smoothing and the generalized two-filter smoothing lies in their construction of the backward information filter. The generalized two-filter smoothing introduces a *artificial prior distribution* that allows it to cater to the possibility of the integration of backward information filter over the support of state variable that might not be finite in the two-filter algorithm. The detailed discussions on the above smoothing algorithms can be found in Briers et al. [2010] and Chapter 2 of the thesis.

Both the FFBSm algorithm and the (generalized) two-filter algorithm require to store the information of particle filtering distribution $p(x_t|\mathcal{I}_t)$ for the purpose of facilitating smoothing calculation. The result of it leads to the memory requirement to be of $O(TN)$. Moreover, these smoothing techniques have computational complexity of $O(TN^2)$. In this chapter, my aim is to reduce the computation complexity of the method of combining EM approach with conventional particle smoothing for the parameter estimation in general state space models. Therefore, we replace those two aforementioned conventional smoothing algorithms with the FFBSa $O(TN)$ computational complexity algorithm, such as: the Godsill-Doucet-West (GDW) smoothing in Godsill et al. [2004] and the modified GDW (MGDW) smoothing. The MGDW smoothing is an alteration of the algorithm Godsill et al. [2004] that takes into account both forward and backward weights. Such modification produces well balanced sampling backward weights for the resampling in smoothing at each time instance. Before moving onto the discussion of those $O(TN)$ algorithms, we provide a brief overview of the forward-backward concept.

5.2.1 Forward-backward Concept

In contrast with forward recursive particle filtering, particle smoothing is performed recursively backward in time according to the forward-backward smoothing formula, which is

$$\begin{aligned} p_\theta(x_t|\mathcal{I}_T) &= \int p_\theta(x_t, x_{t+1}|\mathcal{I}_T)dx_{t+1} \\ &= \int p_\theta(x_{t+1}|\mathcal{I}_T)p_\theta(x_t|x_{t+1}, \mathcal{I}_T)dx_{t+1} \\ &= \int p_\theta(x_{t+1}|\mathcal{I}_T)p_\theta(x_t|x_{t+1}, \mathcal{I}_t)dx_{t+1} \\ &= p_\theta(x_t|\mathcal{I}_t)\int \frac{p_\theta(x_{t+1}|\mathcal{I}_T)f_\theta(x_{t+1}|x_t)}{p_\theta(x_{t+1}|\mathcal{I}_t)}dx_{t+1}. \end{aligned} \tag{5.2.1}$$

The preceding expression reveals that the marginal smoothing posterior density $p_\theta(x_t|\mathcal{I}_T)$ can be obtained through a forward filtering recursion of the filter density $p_\theta(x_t|\mathcal{I}_t)$, the predictive density $p_\theta(x_{t+1}|\mathcal{I}_t)$, and a previous backward recursion smoothed density $p_\theta(x_{t+1}|\mathcal{I}_T)$. More specifically, unlike particle filtering, the smoothing algorithm starts from the newest time instance T, then runs backward and ceases at time $t = 1$. This type of smoothing has been referred to as the forward filter backward smoothing (FFBSm) in the literature.

5.2.2 FFBSa-GDW Algorithm

Similar to FFBSm and generalized two-filter smoothing, the GDW smoothing also requires that the previously discussed particle filtering has been performed. In other words, the information of the set of particles and weights set $\{x_t^i, \tilde{\omega}_{\theta,t}^i\}_{i=1}^N$ should be stored. The GDW smoothing serves the traditional idea of smoothing, which is to obtain sample realizations from the entire smoothing density, and therefore approximate the state variable in a backward fashion. This method can be described as follows: where the joint posterior density can be factorized as

$$p_\theta(x_{1:T}|\mathcal{I}_T) = p_\theta(x_T|\mathcal{I}_T)\prod_{t=1}^{T} p_\theta(x_t|x_{t+1:T}.\mathcal{I}_T), \tag{5.2.2}$$

By the Markov assumption on the transition equation, we have

$$\begin{aligned} p_\theta(x_t|x_{t+1:T}, \mathcal{I}_T) &= p_\theta(x_t|x_{t+1}, \mathcal{I}_t) \\ &= \frac{p_\theta(x_t|\mathcal{I}_t) f_\theta(x_{t+1}|x_t)}{p_\theta(x_{t+1}|\mathcal{I}_t)} \\ &\propto p_\theta(x_t|\mathcal{I}_t) f_\theta(x_{t+1}|x_t). \end{aligned} \tag{5.2.3}$$

The first term $p_\theta(x_t|\mathcal{I}_t)$ in the above equation has been revealed as the marginal posterior density that can be approximated by particle filtering, and the second term is identified as the transition density. Immediately, we would notice that the smooth density function in equation (5.2.3) can be approximated by the following expression:

$$p_\theta(x_t|x_{t+1:T}, \mathcal{I}_T) \approx \sum_{i=1}^{N} \omega^i_{\theta,t|t+1} \delta(x_t - x^i_t), \tag{5.2.4}$$

where $\delta(.)$ is the Dirac delta measure, and the smoothing weights are defined by

$$\omega^i_{\theta,t|t+1} = \frac{\tilde{\omega}^i_{\theta,t} f_\theta(x_{t+1}|x^i_t)}{\sum_{j=1}^{N} \tilde{\omega}^j_{\theta,t} f_\theta(x_{t+1}|x^j_t)}, \tag{5.2.5}$$

where the normalized weights $\tilde{\omega}^i_{\theta,t}$ have been stored from the filtering. The idea boils down to taking the particle sets that approximate the marginal posterior $p_\theta(x_t|\mathcal{I}_t)$ for $t = 1, \cdots, T$, then re-weighting the particles via equation (5.2.5), and this would allow us to approximate the smoothing density. The following algorithm is from Godsill et al. [2004].

Algorithm 1: FFBSa-GDW smoothing.

Given the pair of particles and its weights $\{x^i_t, \tilde{\omega}^i_{\theta,t}\}$ for $i = 1, \cdots, N$ and $t = 1, \cdots, T$, have been computed through particle filtering.

- 1. For $t = T$, Re-sample $\bar{x}^j_T = x^i_T$ with probability $\tilde{\omega}^i_{\theta,T}$.
- 2. For $t = T - 1$ to 1,
  - a. Compute the smoothing weight $\omega^i_{\theta,t|t+1} \propto \tilde{\omega}^i_{\theta,t} f_\theta(x_{t+1}|x^i_t)$ for each $i = 1, \cdots, N$.
  - b. Re-sample $\bar{x}^j_t = x^i_t$ with probability $\omega^i_{\theta,t|t+1}$.
  - c. Go back to step a, and stop when $t = 1$.

5.2.3 FFBSa-MGDW Algorithm

The modified GDW smoothing algorithm shares the resemblances with the GDW smoothing. As the matter of fact, MGDW algorithm is a modification of the GDW smoothing algorithm above. The modification has been done in the position of step 2.b in Algorithm 2 of (5.2.2). More specifically, within GDW algorithm, the backward sampling weight $\omega_{\theta,t|t+1}$ has been purely governed by the forward filtering density $f_\theta(x_{t+1}|x_t^i)$ and forward weight $\tilde{\omega}_{\theta,t}^i$. In the MGDW algorithm, I propose the backward sampling weight to be a linear combination of the smoothing weight $\omega_{\theta,t|t+1}^i$ (step 2.a in Algorithm 2 of (5.2.2)) and the smoothing weight from time instance $t+1$ of $\omega_{\theta,t+1|t+2}^i$. In this manner, the backward sampling weight will therefore be determined not only by the forward filtering process, but also the backward sampling information, which is:

$$\hat{\omega}_{\theta,t|t+1}^i = \lambda\omega_{\theta,t+1|t+2}^i + (1-\lambda)\omega_{\theta,t|t+1}^i, \tag{5.2.6}$$

where $\lambda \in \{0, 1\}$. In the following practical implementation of the MGDW algorithm, the value of λ has been set to be 0.5. Another way of choice of λ may be: directly sample from a uniform distribution of interval $(0, 1)$. However, other endogenous way of determining the value of λ would be preferred in practical applications. Based upon on the extra information given by the backward weights, the MGDW algorithm should perform at least as good as the GDW smoothing empirically, especially for the state process has frequent large change of values. This point requires further investigation. Nevertheless, equation (5.2.6) should provide a more balanced sampling weights, and therefore it should provide more stable estimation for the state variable.

Algorithm 2: FFBSa-MGDW smoothing

Given the pair of particles and weights $\{x_t^i, \tilde{\omega}_{\theta,t}^i\}$ for $i = 1, \cdots, N$ and $t = 1, \cdots, T$, has been obtain from particle filtering.

- 1. For $t = T$, set $\{\bar{x}_T^j = \tilde{x}_T^j, \tilde{\omega}_{\theta,T|T}^j = \tilde{\omega}_{\theta,T}^j = 1/N\}$ for $j = 1, \cdots, N$.
- 2. For $t = T-1$ to 1,
  - a. Compute the smoothing weight $\omega_{\theta,t|t+1}^i \propto \tilde{\omega}_{\theta,t}^i f_\theta(x_{t+1}|x_t^i)$ for each $i = 1, \cdots, N$.
  - b. obtain $\hat{\omega}_{\theta,t|t+1}^i$ through equation (5.2.6).
  - c. Re-sample $\bar{x}_t^j = x_t^i$ with probability $\hat{\omega}_{\theta,t|t+1}^i$.
  - d. Go back to step a, and stop when $t = 1$.

Providing the normalized sampling weights in MGDW smoothing remains in the unit interval and its sum equals to 1, therefore the sampling based upon on those weights does not effect the approximation of the target distribution. Hence convergence of the MGDW smoothing algorithm will follow exactly the GDW smoothing, which has been given in Godsill et al. [2004]. Furthermore, the computational complexity of MGDW smoothing remains to be of $O(TN)$. The performance of MGDW smoothing will be examined through a simple stochastic volatility model in the proceeding section.

5.2.4 GDW vs. MGDW

A proposed model for the smoothing algorithms comparison is a stochastic volatility model in Durbin and Koopman [2000], which can be written as follows

$$\begin{aligned} X_t &= \theta_1 X_{t-1} + \theta_2 V_t, \quad X_1 \sim \left(0, \tfrac{\theta_2^2}{1-\theta_1^2}\right) \\ Y_t &= \theta_3 \exp(X_t/2) W_t, \end{aligned} \tag{5.2.7}$$

where both V_t and W_t are i.i.d normally distributed with mean 0 and variance 1. In this model, θ_1 is the so called persistence parameter, and the observed process Y_t depends on the hidden volatility (state variable) X_t. Hence, taking the simulated process $\{X_t\}_{t=1}^T$ as the benchmark, our present task is to discover the performance of MGDW smoothing derive from the inference of the hidden volatility X_t.

The following Root mean square error (RMSE) has been used as the accuracy assessment for comparisons such as: between particle filtering and smoothing algorithm (bootstrap filter vs. MGDW and GDW smoothing), and among smoothing algorithms (GDW vs. MGDW).

$$RMSE(x_{t|t}) = \frac{1}{R}\sum_{i=1}^{R}\left(\frac{1}{N}\sum_{j=1}^{N}(x_{t|t}^j - x_t^j)^2\right)^{1/2},$$

where

- x_t^j is the true simulated state for the jth particle at time t, with $j = 1, \cdots, N$.
- $x_{t|t}^j$ is the jth estimate at time step t given observation set $\mathfrak{I}_t$.
- R is the number of repetitions.

Note that the purpose of this example is to access the performance of the MGDW smoothing in estimating the state variable. The repetition R is set to be 100, and the true underlying parameter values are: $\theta_1 = 0.95$, $\theta_2^2 = 1$, and $\theta_3 = 0.4221$.

In figure (5.1), particle counts on the horizontal display means different particle numbers, which are $[50, 100, 250, 500, 750, 1000, 2500, 5000]$. The RMSE of both GDW smoothing and MGDW smoothing are less than the bootstrap filtering. In addition, the performance of MGDW smoothing is at least as good as GDW smoothing. Similar results were found (not listed) for the study of linear and Gaussian state space models.

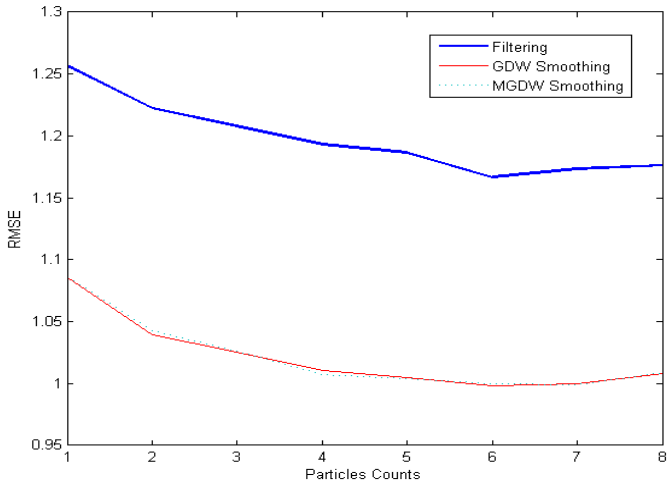

Figure 5.1: Particle filtering and particle smoothing comparison based on simple volatility model.

5.3 Off-line Parameter Estimation - EM Algorithm

To be able to infer the hidden state variables is merely part of the jobs in estimating the general state space model. A more important and interesting part of the job would be to estimate the unknown but fixed population parameters that parameterizes the state space model. This problem has received huge amount of attention over the past 15 years. Existing parameter estimation methods for state space models have been classified as: Bayesian or Maximum Likelihood (ML) and whether they are implement off-line (batch) or on-line (recursively). In the approach of Bayesian inference, the unknown parameters will be considered random, with a suitable prior density attached to it for its inference. In the approach of ML, the inference of unknown parameters is the maximizing argument in the observed likelihood function.

In an off-line framework, the inference on the parameters is carried out by iterating over a fixed set of observations $y_{1:t} = \{y_1, \cdots, y_t\}$ or $\mathcal{I}_t$. On the contrary, in an on-line framework, the inference is done by updating the parameters estimation sequentially as new observations $\{y_t\}_{t>1}$ are available. This paper takes particular interest in off-

line parameter estimation using maximum likelihood approach, more specifically, it is the method of EM algorithm with the usage of particle smoothing. Provided the EM technique possesses features of numerical stability and computationally cheaper (for a large dimension parameter space) over the gradient ascent in off-line estimation. This chapter extended the EM technique by replacing conventional $O(TN^2)$ smoothing algorithms such as: FFBSm and generalized two-filter, by our newly proposed $O(TN)$ MGDW smoothing.

5.3.1 Likelihood Function

In general, given the realisations (observed data) $\mathcal{I}_T = y_{1:T} = \{y_1, \cdots, y_T\}$ of random vector $Y = \{Y_{1:T}\}$ with the parametric density $p_{Y|\theta}(Y)$, the goal would be to find the maximum likelihood estimate (MLE) of θ:

$$\hat{\theta}_{MLE} \equiv \underset{\theta \in \Theta}{\operatorname{argmax}} \; p_\theta(Y|\theta) = \underset{\theta \in \Theta}{\operatorname{argmax}} \; L_\theta(\theta|Y), \tag{5.3.1}$$

which is the equivalent as maximizing the log-likelihood of $y_{0:t}$:

$$\hat{\theta}_{MLE} \equiv \underset{\theta \in \Theta}{\operatorname{argmax}} \; \log p_\theta(Y|\theta) = \underset{\theta \in \Theta}{\operatorname{argmax}} \; \mathcal{L}_\theta(\theta|Y). \tag{5.3.2}$$

Note that there is a change of notation, such as we use Y instead of the information set $\mathcal{I}$ in the above likelihood formulation.

However, in state space models, as the state variables X_t is unobserved, it will be difficult to solve either with equation (5.3.1) or (5.3.2). Therefore, we employ EM algorithm, which fills the complete data $Z = \{X, Y\}$, then maximizes θ over the expected log-likelihood of Z. Hence our previous maximization of observed likelihood function has been transformed to be maximizing the joint likelihood function of $Z = \{X, Y\}$, which is

$$\hat{\theta} \equiv \underset{\theta}{\operatorname{argmax}} \; \mathcal{L}_\theta(\theta|X, Y), \quad \text{where } \mathcal{L}_\theta(\theta|X, Y) \equiv \log p_\theta(X, Y|\theta). \tag{5.3.3}$$

The EM algorithm provides a tool to solve the above missing data problem as it is proposed in equation (5.3.3). Note that the applicability of EM algorithm has also been used for learning Gaussian mixture models. In order to obtain the expected log-likelihood of Z, we make the use of particle smoothing matrix, which has been computed from the estimation of state variables. This is the reason being of our particular interest in particle smoothing that has previous discussed in the chapter. The goal would be to apply our newly proposed MGDW smoothing in obtaining the

most suitable smoothing values that can be used to aid the parameter estimation through EM algorithm.

5.3.2 The EM Algorithm

The following description of EM algorithm includes the conventional *expectation* and *maximization* steps that will be used in estimating general state space models:

S1 : Let m denote the m^{th} iteration. Suppose we have the initial guess or initial estimates θ^m. Given the observed data y and estimates θ^m, we can formulate the conditional probability distribution of $p_\theta(x, y|y, \theta^m)$ for the complete data $z = (x, y)$.

S2 : **Expectation** - Our subsequent concern would be that in order to allow the estimation of parameter θ through maximisation, we would have to make further adjustment over the new conditional probability distribution $p(z|y, \theta^m)$. Such an adjustment is through the approximation of expected joint log-likelihood, which has often been referred as the *Q-function*

$$\begin{aligned}
\mathcal{L}(\theta|z) &\approx E_{Z|y,\theta^m}[\log p_\theta(Z|\theta)] = \int_{\mathcal{Z}} \log p_\theta(z|\theta) p_\theta(z|y, \theta^m) dz \qquad (5.3.4)\\
&= \int_{\mathcal{Z}} \log p_\theta(x, y|\theta) p_\theta(x, y|y, \theta^m) dz \\
&= \int_{\mathcal{X}} \log p_\theta(x, y|\theta) p_\theta(x|y, \theta^m) dx \\
&= E_{X|y,\theta^m}[\log p_\theta(y, X|\theta)] = Q(\theta, \theta^m).
\end{aligned}$$

Note that the integral over the set X is the closure of the set $\{x|p_\theta(x|y, \theta^m) > 0\}$, and the above Q-function is a function of θ. One important remark over the expectation step is that we transform the problem of estimation the maximum likelihood estimates θ_{MLE} from observation density $p_\theta(y|\theta)$ to be estimating θ with joint density $p_\theta(x, y|\theta)$. Since the missing part of the information X in Z is unknown to us, the complete likelihood function $\mathcal{L}_\theta(\theta|z)$ is set to be approximated through its expectation, where the expectation of the log-likelihood can be obtained through finite Monte Carlo simulations.

S3 : **Maximization** - The $(m + 1)^{th}$ guess of θ will be obtained by maximizing the following Q-functions

$$\theta^{m+1} = \underset{\theta \in \Theta}{\operatorname{argmax}} \; Q(\theta, \theta^m). \qquad (5.3.5)$$

S4 : Two types of Stopping criterion can be employed, the first one is that Step S2 and S3 will be iterated until the estimates value stops changing: $\|\theta^{m+1} - \theta^m\| < \epsilon_a$ for some $\epsilon_a > 0$. Alternative one is that iterates until the log-likelihood $\mathcal{L}_\theta(\theta|z)$ stop changing $\|\mathcal{L}_\theta(\theta^{m+1}|z) - \mathcal{L}_\theta(\theta^m|z)\| < \epsilon_b$ for some $\epsilon_b > 0$.

In real applications using EM algorithm, with the most appropriate determination of initial estimate θ^0 is unavailable, then it is common to start EM from multiple random initial guesses, and choose the one with the largest likelihood as the final guess for θ.

5.3.3 The Objective Function

One of the remarkable features of the EM algorithm is that the maximization of its objective function $Q(\theta, \theta^m)$ guarantees an increase of the joint probability density function, hence the joint likelihood function. Such a feature is summarized in the following monotonicity property of the EM algorithm.

Proposition 5.3.1. *The sequence of $\theta^1, \theta^2, \theta^3, \cdots$, of EM estimates satisfy*

$$p_\theta(y|\theta^1) \leq p_\theta(y|\theta^2) \leq p_\theta(y|\theta^3) \leq \cdots .$$

The proof of the above proposition is given in the Appendix A. The monotonicity of the EM algorithm demonstrates that the EM iterates will not get worse in terms of their observed likelihood value. But this property alone does not guarantee the convergence of the sequence of estimates $\{\theta^m\}$ to the MLE of the observed likelihood of $p_\theta(y|\theta)$. Moreover, the convergence of the sequence $\{\theta^m\}$ rely on the characteristics of the log-likelihood $\mathcal{L}_\theta(\theta|x, y)$ and the Q-function $Q(\theta, \theta^m)$, and the initial point θ^0. Quadratic convergence of EM algorithm can be achieved with additional work of the inverse of Hessian matrix of the likelihood $\mathcal{L}_\theta(\theta|x, y)$; superlinear convergence can be achieved with the calculation of gradient of likelihood $\mathcal{L}_\theta(\theta|x, y)$ such as BFGS update. Note that the definitions of different convergences are given in the Appendix A.

To be able to draw inference on the unknown parameters is an important goal in modelling data sets with general state space models. Given the total observations or the end time step to be $t = T$, then joint density function that includes the parameters would be

$$p_\theta(\mathbf{x}, \mathbf{y}|\theta) = \mu_\theta(x_1)\Big\{\prod_{i=2}^{T} f_\theta(x_i|x_{i-1}, \theta)\Big\}\Big\{\prod_{i=1}^{T} g_\theta(y_i|x_i, \theta)\Big\} \tag{5.3.6}$$

where $\mathbf{x} = \{x_1, \cdots, x_T\}$ and $\mathbf{y} = \{y_1, \cdots, y_T\}$. The essential parameters estimation can be obtained through iterative EM algorithm on the likelihood of equation (5.3.3). The objective function $Q(\theta, \theta^k)$ can be formed by taking logarithms and conditional expectations over the joint density function of equation (5.3.6), which is

$$\begin{aligned} Q(\theta, \theta^k) &= \int \mathcal{L}(\theta|\mathbf{x}, \mathbf{y}) p_\theta(\mathbf{x}|\mathbf{y}, \theta^k) d\mathbf{x} = \int \log p_\theta(\mathbf{x}, \mathbf{y}|\theta) p_\theta(\mathbf{x}|\mathbf{y}, \theta^k) d\mathbf{x} \\ &= \int \log \Big[\mu_\theta(x_1) \Big\{ \prod_{i=2}^{T} f_\theta(x_i|x_{i-1}, \theta) \Big\} \Big\{ \prod_{i=1}^{T} g_\theta(y_i|x_i, \theta) \Big\} \Big] p_\theta(\mathbf{x}|\mathbf{y}, \theta) d\mathbf{x} \quad \text{by eq(5.3.6)} \\ &= \int \log \mu_\theta(x_1) p_\theta(\mathbf{x}|\mathbf{y}, \theta^k) dx_1 + \sum_{t=2}^{T} \int \log f_\theta(x_t|x_{t-1}, \theta) p_\theta(x_t|\mathbf{y}, \theta^k) dx_{t-1} \\ &+ \sum_{t=1}^{T} \int \log g_\theta(y_t|x_t, \theta) p_\theta(x_t|\mathbf{y}, \theta^k) dx_t. \end{aligned} \tag{5.3.7}$$

A first brief glance of the above equation sparks an additional reason as to why we are interested in particle smoothing. This identifies the role of particle smoothing in the purpose of parameter estimation using EM algorithm. Suppose the above smoothing densities can be approximated using the smoothing procedure that described by equation (5.2.4), then the set of smoothing particles $\{x_{t|T}^{(i)}\}$ for $i = 1, \cdots, N$ can be utilized as the generated samples for us to approximate the above objective function $Q(\theta, \theta_k)$, that is

$$\begin{aligned} \hat{Q}(\theta, \theta_k) &= \sum_{i=1}^{N} \omega_{1|T}^{(i)} \log p_\theta(x_{1|T}^i) + \sum_{t=2}^{T} \sum_{i=1}^{N} \omega_{t|T}^i \log f_\theta(x_{t|N}^{(i)}|x_{t-1}) \\ &+ \sum_{t=1}^{T} \sum_{i=1}^{N} \omega_{t|T}^i \log g_\theta(y_t|x_{t|N}^{(i)}). \end{aligned} \tag{5.3.8}$$

Given the density functions $f(.)$ and $g(.)$ are known, and the above objective function is simply a function of parameter θ. Then M-step can be apply to obtain the $k+1$ estimate, θ^{k+1}. Such estimate θ^{k+1} will be adopted during the next round of particle filtering and smoothing, sub-sequentially it would allow to obtain the $(k+2)^{th}$ iteration estimate θ^{k+2}. These procedures will be repeated until the convergence is reached.

5.3.4 The Standard Deviation of Parameter Estimates

Since we have known that the observed likelihood function $f_\theta(Y|\theta)$ is hard to work with, which was the reason that we were using EM method in the first place. More

specifically, which is to use the joint likelihood function $f_\theta(X, Y|\theta)$. Such diversion has led to a much complicated route of deriving the variance and covariance of the estimates. We aim to explain how does this complication arises, as well as provide the new formula for the covariance of the estimates in this very section.

Conventionally, the variance and covariance matrix is the inverse of the observed information matrix, which is

$$-\frac{\partial^2 \log p_\theta(Y|\theta)}{\partial\theta\partial\theta'}, \tag{5.3.9}$$

where y is the realisation of random vector of Y and θ is the unknown but fixed parameter vector. The problem is that we do not know $p_\theta(Y|\theta)$ since the state variable is hidden in the model. But we do have $p_\theta(Y|\theta) = \int p_\theta(X, Y|\theta)dX$, where the above observed information matrix can be re-expressed as

$$\begin{aligned} -\frac{\partial^2 \log p_\theta(Y|\theta)}{\partial\theta\partial\theta'} \quad = \quad & -E\Big[\frac{\partial^2}{\partial\theta\partial\theta'}\log p_\theta(X, Y|\theta)\Big|\mathcal{I}_T\Big] \\ & -E\Big[\Big\{\frac{\partial}{\partial\theta}\log p_\theta(X, Y|\theta)\Big\}\Big\{\frac{\partial}{\partial\theta'}\log p_\theta(X, Y|\theta)\Big\}\Big|\mathcal{I}_T\Big] \\ & +E\Big[\frac{\partial}{\partial\theta}\log p_\theta(X, Y|\theta)\Big|\mathcal{I}_T\Big]E\Big[\frac{\partial}{\partial\theta'}\log p_\theta(X, Y|\theta)\Big|\mathcal{I}_T\Big]. \end{aligned} \tag{5.3.10}$$

The procedure enables us to derive the above expression is: firstly plug $p_\theta(Y|\theta) = \int p_\theta(X, Y|\theta)dX$ into the above observed information matrix of equation (5.3.9), then through a series of 'chain-rule' and expectation formula applications, we would have the new observed information matrix that will be implementable in estimation. For detailed derivations and explanations of the above equation (5.3.10), I would encourage reader refer to Kim [2005]. Note that the joint log-likelihood function $\log p_\theta(X, Y|\theta)$ has been defined above.

The potential practical difficulty with obtaining the observed information matrix from using the above equation (5.3.10) is that, the value of second term on the right hand side of the equation (expectation of the square of the gradients) can frequently be larger than the combination of first term and third term (square of the expectation of the gradients). This is due to the fact that for a random variable X, we have $\big(E(X)\big)^2 < E(X^2)$. [1] Such problem can result in the inverse of the observed information matrix to be non-positive definite. We adopt the trimming idea of Kim [2005] to deal with this problem in our estimation. Note that in the following model

[1] In the context of probability theory, it is generally stated in the following form: if X is a random variable and h is a convex function, then $h\big(E(X)\big) \leq E\big(h(X)\big)$. Since X^2 is a convex function, the proof of our case would follow the proof of the Jensen's inequality.

estimations, the terms where the observed information matrix involved have been derived and listed in the Appendix B and C.

For the models are nested, they can be compared using a likelihood-ratio test by computing their respected likelihood values. On the contrary, for models are not nested, the conventional Akaike's information criterion (AIC) or Bayesian information criterion (BIC) can be used for the model comparison. The model comparison task has been omitted during the studies to the Phillips model and the Stochastic volatility model.

5.3.5 The Algorithm

The EM and particle smoothing method of estimating state space that has been discussed so far can be summarised in the following algorithm.

Algorithm 3: EM - MGDW smoothing algorithm

Provided the model takes the form that has been defined as it is in equations (**??**) and (**??**), and a set of initial parameter estimate θ^0. Let k denote the k^{th} estimates and set $k = 0$, we perform:

- 1. Given the initial parameter value θ^k for $k = 0$, perform particle filtering algorithm (Algorithm 1 of (2.4.4)) and store both particle values and particle weights $\{x^i_{t|t}, \tilde{\omega}^i_{t|t}\}$ for i denotes the i^{th} particle and t denotes the t^{th} time instance.

- 2. Provided the values obtained through filtering, we perform the MGDW smoothing (Algorithm 2 of (5.2.3)) to obtain the smoothing particle values and smoothing particle weights $\{x^i_{t|T}, \omega^i_{t|T}\}$.

- 3. Using the given parameters values θ^k and smoothing values, we maximizes equation (5.3.8), and obtains the $(k+1)^{th}$ estimates.

- 4. Takes the $(k+1)^{th}$ set of values and go back to step 1. Iterated till the observed likelihood deviation between k and $k+1$ is smaller than some tolerance ϵ_b.

- 5. Compute the variance and covariance matrix of the last iteration of the set of estimates.

The cons with EM algorithm are: its convergence can be slow (linear convergence). In addition, since the EM method is locally optimal, it thus sensitive to initialization and might be trapped in a local maximum. However, If the joint likelihood of models

can be expressed in exponential form, then the M-step can be done analytically.

Finally, the selection of the particle size N depends on the presence of problem. As far as my knowledge goes, there is actually no rule of thumb on this issue. In this chapter, we start with a smaller N, increasing its value gradually and observe the room of improvement in terms of filtering and parameter estimations. Having done that, we admit this may not be the most optimal way of choosing the most ideal particle size.

5.4 Test and Comparison

In this section, we intend to learn the performance of the method of EM algorithm combined with MGDW smoothing in parameters estimation for general state space models. We hope to achieve it through its applications onto studies using both simulated data and real data.

5.4.1 Simulation

The simulation study employs the discrete version of Cox-Ingersoll-Ross (CIR) stochastic volatility model that have been used in Poyiadjis et al. [2011]. The model is defined as follows:

$$\begin{aligned} X_{t+1} &= \mu + X_t + \phi \exp(-X_t) + \exp(-X_n/2)V_{t+1}, \\ Y_t &= \beta \exp(X_t/2)W_t, \end{aligned} \tag{5.4.1}$$

where V_{t+1} and W_t are i.i.d normal with mean 0 and variance 1. The model parameters are $\theta = (\mu, \phi, \beta)$, where the parameter μ denotes the speed of mean reversion[2] , β is the volatility term of the square root volatility diffusion, and ϕ represents the persistence of the volatility. The transition equation of CIR model incorporates a non-linear dynamic term, and the stochastic disturbance term has been assumed to depend on the volatility at previous time period. Inspired by Shephard [1996], in the following implementation, we transform the above observation equation in equation (5.4.1) to form the more familiar linear structure. More specifically, it is by squaring the observation equation and taking the logarithm of it, which gives:

$$R_t = \log(Y_t^2) = \alpha + X_t + \tilde{W}_t, \tag{5.4.2}$$

[2]In general terms, the essence of the concept of mean reversion is that a stochastic process such as a stock price, its high and low values are temporary, and the value tends to be average value overtime. In other words, the deviation from the average value is expected to revert to the average.

where

$$\begin{aligned}\alpha &= \log(\beta^2) + E(\log(W_t^2)) \\ \tilde{W}_t &= \log(W_t^2) - E(\log(W_t^2)) \sim \log(\chi_1^2) - E(\log(\chi_1^2)),\end{aligned}$$

and $\tilde{W}_t$ is the centred Chi-square distribution in Shephard [1996], with $E(\log(\chi_1^2)) = -1.2749$. Moreover, Harvey et al. [1994] pointed out that the mean and variance of $\log(W_t^2)$ are to be -1.2749 and $\pi^2/2 = 4.9348$, where W_t is standard normal random variable. The first and second derivatives of the joint likelihood function of the observations $p_\theta(X, R|\theta^m)$ have been given in the Appendix B under the section of the Cox-Ingersoll-Ross model, which would allow us to compute the observation standard deviations. Note that let $\mathfrak{I}_T = \{r_1, \cdots, r_T\}$, with r_i is the realization of R_i, and the $Q(\theta, \theta^m)$ function is

$$Q(\theta, \theta^m) = E\big\{ -2 \log p_\theta(X, R|\theta^m)|\mathfrak{I}_T\big\}, \tag{5.4.3}$$

where

$$\begin{aligned}p_\theta(X, R|\theta^m) \propto & \prod_{t=2}^{T} -\frac{1}{2\pi}\big\{\exp(-X_{t-1})\big\}^{-\frac{1}{2}} \\ & \exp\Big\{ -0.5 \exp(X_{t-1})\big(X_t - \mu - X_{t-1} - \phi \exp(-X_{t-1})\big)^2\Big\} \\ & \prod_{t=1}^{T} C_1 \exp\Big\{\frac{1}{2}\big(\exp(R_t - X_t - \alpha - 1.2749) - (R_t - x_t - \alpha - 1.2749)\big)\Big\},\end{aligned}$$

where C_1 is a constant, and therefore the logarithm of function $p(.)$ is

$$\begin{aligned}\log p_\theta(X, R|\theta^m) \propto & -0.5 \sum_{t=2}^{T} \exp(X_{t-1})\big\{X_t - \mu - X_{t-1} - \phi \exp(-X_{t-1})\big\}^2 \\ & -0.5 \sum_{t=1}^{T} \big\{ \exp(R_t - X_t - \alpha - 1.2749) - (R_t - x_t - \alpha - 1.2749)\big\}.\end{aligned}$$

The $(m+1)^{th}$ estimates are:

$$\begin{aligned}
\mu^{m+1} &= \frac{\sum_{t=2}^{T} E\{\exp(X_{t-1})(X_t - X_{t-1} - \phi^m \exp(-X_{t-1}))|\mathcal{I}_T\}}{\sum_{t=2}^{T} E\{\exp(X_{t-1})|\mathcal{I}_T\}} \\
\phi^{m+1} &= \frac{\sum_{t=2}^{T} E\{(X_t - \mu^{m+1} - X_{t-1})|\mathcal{I}_T\}}{\sum_{t=2}^{T} E\{\exp(-X_{t-1})|\mathcal{I}_T\}} \\
\alpha^{m+1} &= \log\Big[\frac{1}{T}\sum_{t=1}^{T} E\{\exp(r_t - X_t - 1.2479)|\mathcal{I}_T\}\Big].
\end{aligned}$$

Note that the initial state is discarded in the Q function. The true parameter values are set to be $\mu = -0.1$, $\phi = 0.5$, and $\alpha = -4.9$ with the use of number of particles $N = 2000$, and length of the simulated observation $T = 500$. We use the EM and MGDW smoothing method iteratively, and the 200^{th} estimates are $-0.09949(0.0269)$, $0.55(0.2151)$, and $-4.956(0.0067)$ for parameter $\theta = \{\mu, \phi, \alpha\}$, respectively. Figure (5.2) demonstrates the behaviour of each estimates during 200 iterations.

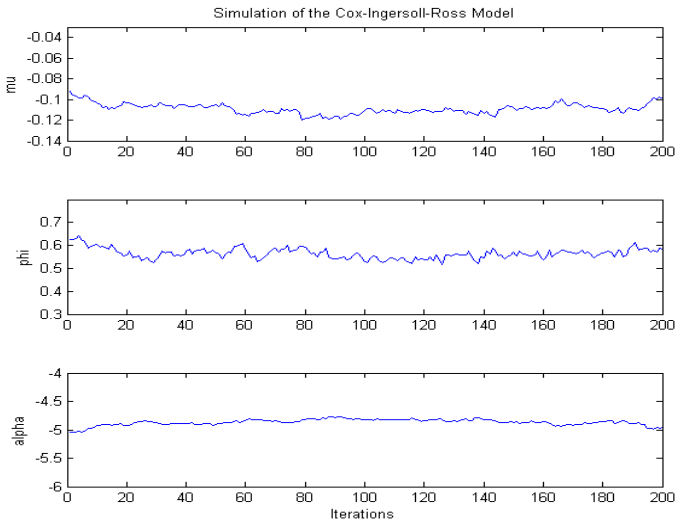

Figure 5.2: Parameter estimation using the EM-MGDW method for the Cox-Ingersoll-Ross model with simulated data.

5.4.2 Comparison

In the following section, we investigate the performance of the EM-MGDW method in real data application. In order to draw more concrete comparison to the existing parameter estimation methods for general state space models, the investigation will be deriving from the stochastic volatility model of Durbin and Koopman [2000], where

the model takes the form of equation (5.2.7), that is

$$\begin{aligned} X_t &= \theta_1 X_{t-1} + \theta_2 V_t, \quad X_1 \sim \left(0, \tfrac{\theta_2^2}{1-\theta_1^2}\right) \\ Y_t &= \theta_3 \exp(X_t/2) W_t. \end{aligned}$$

In the observation equation above, the series of interest is $Y_t = \Delta \log(e_t)$, where e_t denotes the actual exchange rate and Δ is the difference operator for discrete sequence. Once more, for implementation simplicity, we re-write the observation equation in the form of equation (5.4.2), which is

$$R_t = \log(Y_t^2) = \alpha + X_t + \tilde{W}_t. \tag{5.4.4}$$

The objective Q function and their estimates can be straightforwardly derived as we have done so for the Cox-Ingersoll-Ross model. For the detailed derivation of the above stochastic volatility model, reader can refer to Briers et al. [2010].

The parameters of interest would be $\theta = \{\theta_1, \theta_2, \alpha\}$ for the above model. Our estimation employs the same data of Durbin and Koopman [2000], that is: the pound-dollar daily exchange rates from October 1st, 1981 to June 28th, 1985. Our results are drawn to compare with different estimation techniques that were developed in Durbin and Koopman [2000] and Doucet and Tadic [2003]. In the estimation, we use the number of particles $N = 2000$ and the iteration number $k = 200$. In Figure (5.3), the volatility persistence value indicates the volatility follows mostly a unit root process, and the bottom plot of the relative likelihood displays the behaviour of the likelihood function. The 200^{th} estimates and its standard deviations are given in Table (5.1), and the estimate values are comparable to two of the existing methods. Finally, the estimates $\hat{\alpha} = -2.1720$ can be converted back and obtain the estimate of parameter θ_3 in the original observation equation. Note that the value in the parentheses indicates standard deviations of the estimates.

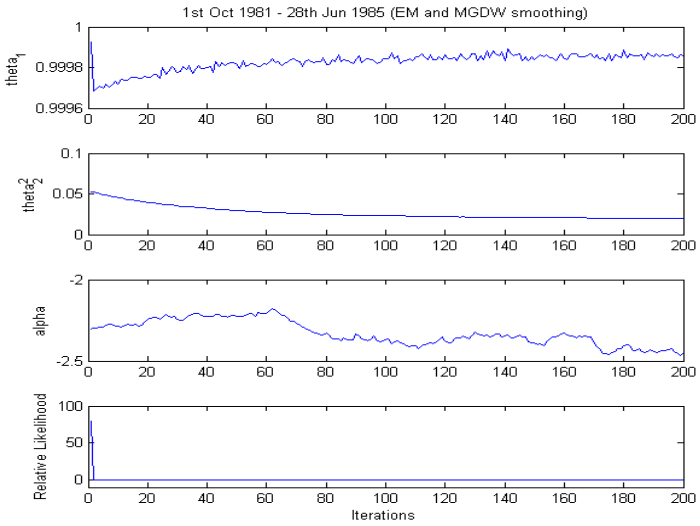

Figure 5.3: Parameter estimation using the EM-MGDW method for the stochastic volatility model using Durbin and Koopman's exchange rate data.

Table 5.1: The comparison between the EM-MGDW smoothing and existing estimation methods.

Ref.	θ_1	θ_2^2	α
Durbin and Koopman	0.973	0.0299	-2.1863
Doucet and Tadic	0.968	0.0353	-2.1737
EM and MGDW smoothing			
MGDW	0.9998(0.0000)	0.0259(0.0001)	-2.172(0.0038)

5.5 Applications

5.5.1 Phillips Curves - US Inflation and Unemployment

The Phillips curve is an economic theory that describes the inverse relationship between the rate of unemployment and the rate of inflation in an economy. The new classic version of the short run Phillips curve can be derived from the aggregated supply function, then via the use of the Okun's law, [3] which is

$$\Pi_t = \Pi_t^e + \beta(U_t - U_t^n) + W_t, \tag{5.5.1}$$

[3]The Okun's law describes a relationship between output and unemployment. However, this law was primarily empirically observed rather than a result derived from theory.

where W_t denotes the unexpected exogenous shocks to the supply, and Π_t and Π_t^e are the inflation and expected inflation respectively. In addition, U_t represents unemployment and U_t^n is the natural rate of unemployment. Another interesting econometric problem that apart form the estimation of parameter β would be making inference regarding the underlying behaviour of U_t^n. In order to be able to use the Kalman filter, economic researchers have been make the assumption that the natural rate of unemployment follows a linear Markov process, which can be expressed as:

$$U_t^n = \gamma U_{t-1}^n + V_t. \tag{5.5.2}$$

Equation (5.5.1) and (5.5.2) form a linear and Gaussian state space model. However, providing us with the EM-MGDW estimation method, we can now estimate a less restrictive Phillips curve model. This motivates us to re-construct the transition equation of the Phillips model by including an addition of nonlinear term and the modified supply shock. In Fernandez-Villaverde and Rubio-Ramirez [2007], who solve the real business cycle model, and then use it for the purpose of investigating the usefulness of particle filtering in estimating dynamic macroeconomic models. I take a different route to these authors in this chapter, which imposes the Phillips curve model to take a specific form, where it is similar to the those observation equations and transition equations of the model given in Fernandez-Villaverde and Rubio-Ramirez [2007], but with merely three parameters involved. However, the EM-MGDW method is capable of incorporating more parameters. The newly proposed model is defined as

$$\begin{aligned} U_t^n &= \gamma U_{t-1}^n + \phi \frac{1}{1+\exp(-U_{t-1}^n)} + \exp\left\{ - \frac{1}{|U_{t-1} - U_{t-1}^n|} \right\} V_t \\ \Pi_t &= \Pi_t^e + \beta(U_t - U_t^n) + W_t, \end{aligned}$$

where W_t and V_t are assumed to be i.i.d normal with mean 0 and variance 1. The parameters of interest are $\theta = \{\gamma, \phi, \beta\}$, where β indicates the relationship between unemployment and inflation. For the transition equation, both γ and ϕ are persistence parameters, with ϕ describes the effects of the nonlinear dynamics. Monthly U.S unemployment and inflation data over from the period from Feb, 1970 to Sep 2011, total of $T = 500$ observations will be used for the estimation of the above Phillips model. The plots of the U.S inflation and unemployment are given in Figure (5.4).

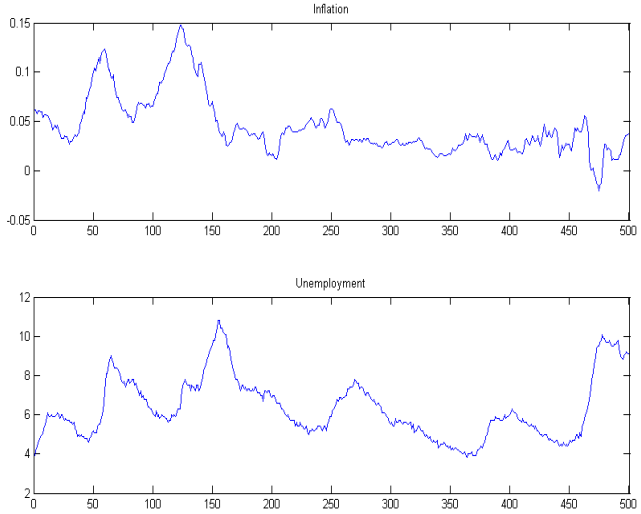

Figure 5.4: The monthly U.S inflation rate and unemployment rate from Feb 1970 - Sep 2011.

The joint likelihood function $f_\theta(U^n, \Pi|\theta^m)$ consists of the following densities:

$$\begin{aligned} f_\theta(U_1^n|\theta) &\sim \mathcal{N}\Big(0, 1\Big), \\ f_\theta(U_1^n|\theta, U_{t-1}^n) &\sim \mathcal{N}\Big(\gamma U_{t-1}^n + \phi\frac{1}{1+\exp(-U_{t-1}^n)}, \exp(-\frac{2}{|U_t - U_{t-1}^n|})\Big), \\ f_\theta(\Pi_t|U_t^n, \theta) &\sim \mathcal{N}\Big(\Pi_t^e + \beta(U_t - U_t^n), 1\Big). \end{aligned}$$

Similar as it were for the estimation of the Cox-Ingersoll-Ross model previously, we discard the first state in the estimation, the $Q(\theta, \theta^m)$ function for the Phillips model will be expressed as follow. Note that the standard deviations of estimates will be given in the Appendix C under the section of Phillips curve.

$$Q(\theta, \theta^m) = E\big[-2\log f_\theta(U^n, \Pi|\theta^m)|\mathfrak{I}_T\big], \tag{5.5.3}$$

where

$$\begin{aligned} \log f_\theta(U^n, \Pi|\theta^m) \quad \propto \quad & -0.5\sum_{t=1}^{T}\big\{\Pi_t - \Pi_t^e - \beta^m(U_t - U_t^n)\big\}^2 \\ & -0.5\sum_{t=2}^{T} B_t\big\{U_t^n - \gamma^m U_{t-1}^n - \phi^m(1+\exp\big(-U_{t-1}^n\big))^{-1}\big\}^2. \end{aligned}$$

and

$$B_t = \exp\left\{\frac{2}{|U_t - U_{t-1}^n|}\right\}$$

The $(m+1)^{th}$ estimates are:

$$\begin{aligned}
\gamma^{m+1} &= \frac{\sum_{t=2}^{T} E\Big[B_t U_{t-1}^n \big\{U_t^n - \phi^m(1+\exp(-U_{t-1}^n))^{-1}\big\}|\mathcal{I}_T\Big]}{\sum_{t=2}^{T} E\Big[B_t\big\{U_{t-1}^n\big\}^2|\mathcal{I}_T\Big]} \\
\phi^{m+1} &= \frac{\sum_{t=2}^{2} E\Big[B_t\big(1+exp(-U_{t-1}^n)\big)^{-1}\big(U_t^n - \gamma^m U_{t-1}^n\big)|\mathcal{I}_T\Big]}{\sum_{t=2}^{T} E\Big[B_t\big(1+exp(-U_{t-1}^n)\big)^{-2}|\mathcal{I}_T\Big]} \\
\beta^{m+1} &= \frac{\sum_{t=1}^{T} E\Big[(U_t - U_t^n)(\Pi_t - \Pi_t^e)|\mathcal{I}_T\Big]}{\sum_{t=1}^{T} E\Big[(U_t - U_t^n)^2|\mathcal{I}_T\Big]}
\end{aligned}$$

In the estimation, the number of particles $N = 2000$ and the iteration number $k = 200$. In Figure (5.5) shows the convergence through iterations of estimates for each parameter. The 200^{th} estimates and their standard deviations are $\hat{\gamma} = 0.9315(0.0176)$, $\hat{\phi} = 0.4204(0.1089)$, and $\hat{\beta} = -0.0331(0.0228)$.

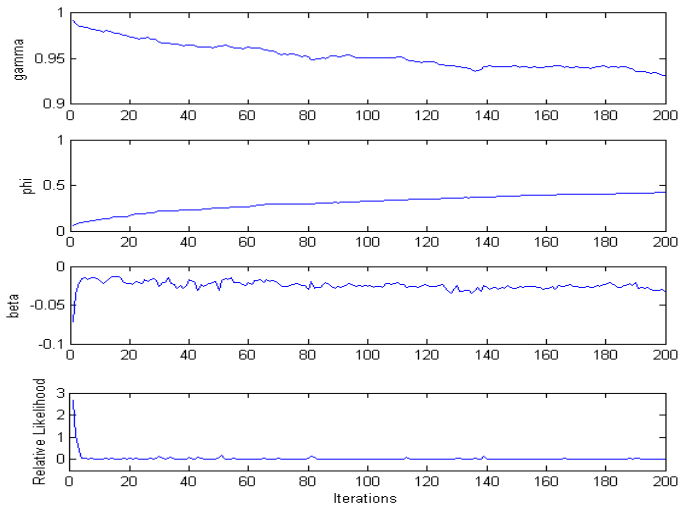

Figure 5.5: Parameter estimation using the EM-MGDW method for the Phillips curve model.

The negative value of estimate $\hat{\beta}$ indicates that there exists a negative relationship between inflation and unemployment based upon on the study of the U.S data. Provided with the estimates of the parameters in the above Phillips curve model, we

can estimate the hidden natural rate unemployment rate once more either by particle filtering or by particle smoothing (MGDW smoothing). With the number of particles are set to be 10000, where the simulated natural rates of unemployment are plotted in Figure (5.6), where dotted line indicates estimates by filtering and dash line indicates estimates by MGDW smoothing. The simulated natural rates of unemployment seem reasonable and they do vary overtime.

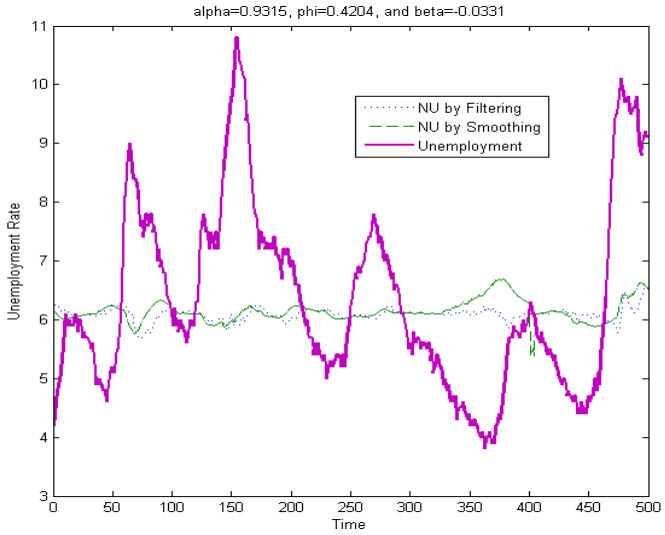

Figure 5.6: The Simulated U.S natural rate of unemployment from Feb 1970 to Sep 2011 using particle filtering and particle smoothing (MGDW).

5.5.2 Stochastic Volatility - UK vs. US Exchange Rate

This section investigates how good is the EM-MGDW smoothing method in estimating the previously outlined and discussed Cox-Ingersoll-Ross model when real data is used. The model has been previously defined that was presented in equations (5.4.1) and (5.4.2), the state equation and transformed observation equation are

$$X_{t+1} = \mu + X_t + \phi \exp(-X_t) + \exp(-X_n/2)V_{t+1}, \tag{5.5.4}$$

$$R_t = \log(Y_t^2) = \alpha + X_t + \tilde{W}_t, \tag{5.5.5}$$

where

$$Y_t = \log e_t - \log e_{t-1},$$

$$\alpha = \log(\beta^2) + E(\log(W_t^2)),$$

$$\tilde{W}_t = \log(W_t^2) - E(\log(W_t^2)) \sim \log(\chi_1^2) - E(\log(\chi_1^2)).$$

In addition, the exchange rate of interest $Y_t = \Delta \log(e_t)$, where e_t is the real exchange rate, and X_t denotes the logarithm of volatility or variance. The unknown population parameters in the preceding model are $\theta = \{\mu, \phi, \alpha\}$. Then the CIR model is utilized to model the UK - US daily exchange rate from the period of 2nd Apr, 2007 to 25th Apr, 2012, with total number of observations to be 1323. The logarithm of the exchange rate, $\log(e_t)$ over the specified period has been displayed in panel (a) of figure (5.7); the transformed exchange of interest, R_t is plotted in panel (b) of figure (5.7).

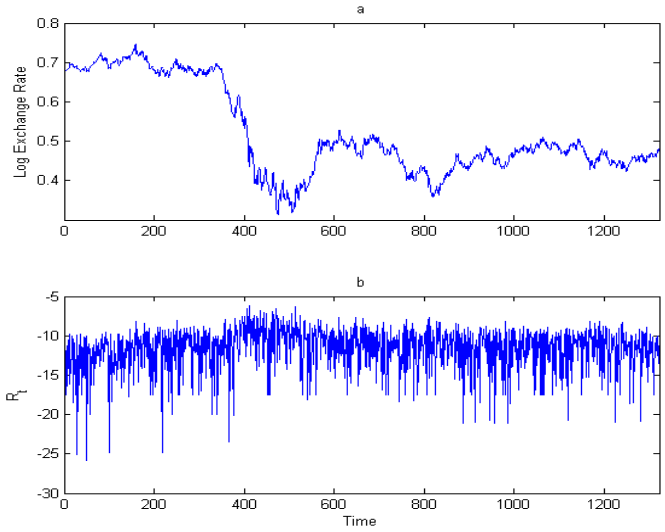

Figure 5.7: UK vs. US exchange rate. Plot a is the logarithmic UK and US exchange rate; plot b is the transformed UK and US exchange rate.

The derivations of the Q-function and the estimates of the population parameters for the CIR model have been previously obtained, which are given in the section of (5.5.1). The estimation of parameters $\theta = \{\mu, \phi, \alpha\}$ that are derived have been plotted in Figure (5.8), where the settings are: particles is $N = 2000$ and number of iterations is $k = 150$. In addition, the estimates of the parameter at the $150th$ iteration are $\hat{\mu} = -1.0489(0.0169)$, $\hat{\phi} = 1.7703(0.0455)$, and $\hat{\alpha} = -12.3320(0.0009)$ respectively. Note that the values within the parentheses are standard deviation of the estimates.

Provided with the estimates, our subsequent aim would be to obtain the hidden volatility of the UK and US exchange rate. To do so, we estimate the hidden volatility through both particle filtering (with entropy test) and MGDW smoothing, where they are displayed in Figure (5.9). Overall, the simulated volatility either by particle filter or particle smoothing behaves in a similar fashion. Note that the volatility

estimation uses 5000 number of particles as compared to the 2000 particles in the parameter estimation.

Providing the available simulated volatility values ($\hat{X}$, that obtained through particle filtering and particle smoothing) and the estimated parameters value (θ), we plug all those values into the Cox-Ingersoll-Ross model. Subsequently, it allows us to obtain a 'fitted' transformed exchange rate $\hat{R}$, which have been given in Figure (5.10). In the top plot, the fitted transformed exchange rate is based on the filtered volatility; whereas the fitted value in the bottom plot is from the volatility obtained from MGDW smoothing. It seems that both the fitted values provide a base in terms of representing the overall structure of the transformed exchange rate. Note that such fitted value has been calculated through the usage of equation (5.4.2).

Finally, the proposed Phillips curve model and the Cox-Ingersoll-Ross model along with the EM-MGDW estimation method can certainly be adapted for other data sets. For instance, given the Phillips model, we may be able to learn the Phillips relationship for the U.K; or the CIR model may allow us to see how the underlying volatility of Yuan vs. Dollar does for the past 5 years.

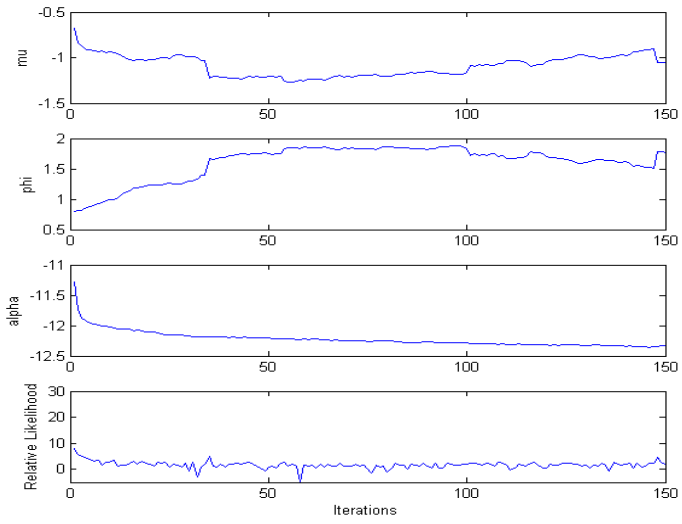

Figure 5.8: UK vs. US exchange rate, parameter estimation using the EM-MGDW method.

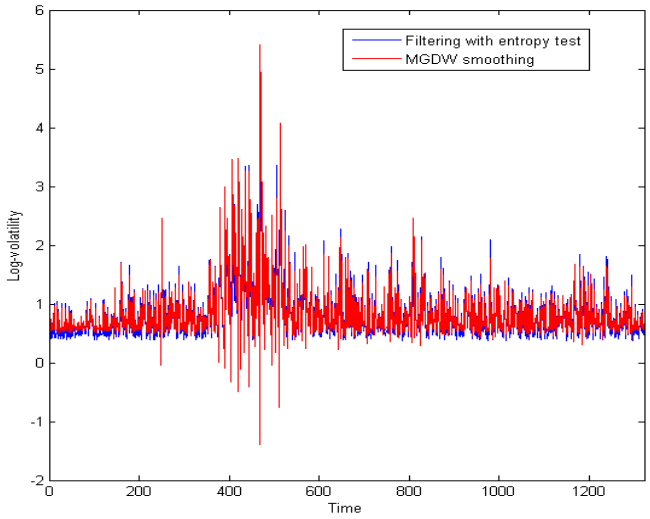

Figure 5.9: Comparison between the simulated Log-volatility of Particle filtering and MGDW Particle smoothing.

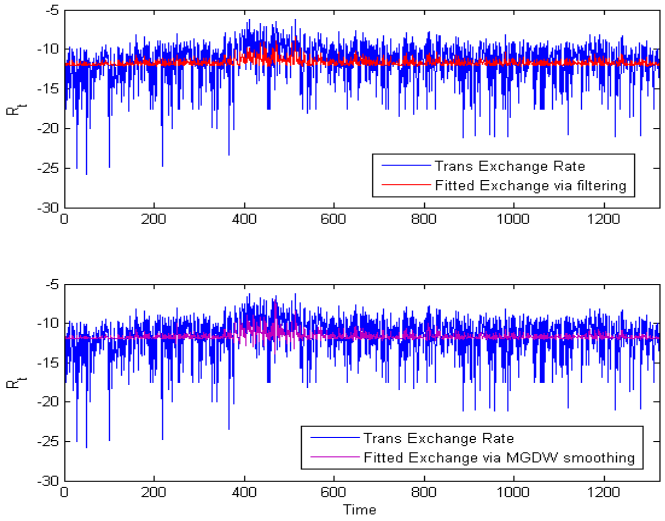

Figure 5.10: The fitted of UK vs. US exchange rate using the simulated log-volatility obtained from particle filtering and particle smoothing (MGDW).

5.6 Conclusion

In this chapter, we demonstrated empirically that the newly proposed MGDW smoothing performs at least as good as the forward filtering backward sampling of the GDW smoothing. The use of MGDW smoothing reduces the computational effort substan-

tially as it is compared to existing smoothing techniques. This advantage has been utilized in the approach of EM algorithm, which provides the EM-MGDW smoothing in off-line parameter estimation for general state space models. Through both simulation and real data applications, this novel method has proven to be reliable in estimating unknown parameters. However, it has been pointed out that this method tends to be sensitive to the selection of the initial values. The EM-MGDW method caters the task of learning large unknown parameters, and also allows to compute the likelihood for the purpose of model comparison.

We are in search of theoretic up-ground for the MGDW smoothing over the GDW smoothing. Further researches lie to extend the EM-MGDW smoothing to multivariate state space models, which will undoubtedly be more challenging as we will be facing the problem of curse of dimensionality. However, it will be interesting to see how far we could progress on this aspect. Finally, we would certainly like to know more about the idea of forecasting based upon the general state space model, and how would the roles be for both particle filtering and particle smoothing within it. An important application of the EM-MGDW method would be for macroeconomic dynamic models, e.g. Fernandez-Villaverde et al. [2006] and Fernandez-Villaverde and Rubio-Ramirez [2007].

5.7 Appendix

Appendix A: Definitions and Proofs

Linear Convergence : Linear convergence means that there exists $M > 0$ and $0 < C < 1$ such that $\|\theta^{m+1} - \theta^*\| \leq C\|\theta^m - \theta^*\|$ for all $m \geq M$, where θ^* is the optimal value of θ.

Quadrtic Convergence : Quadrtic Convergence means that there exists $M > 0$ and $0 < C < 1$ such that $\|\theta^{m+1} - \theta^*\| \leq C\|\theta^m - \theta^*\|^2$ for all $m \geq M$, where θ^* is the optimal value of θ.

Superlinear Convergence : Superlinear Convergence means $\|\theta^{m+1} - \theta^*\|/C\|\theta^m - \theta^*\| \to 0$ as $m \to \infty$.

Lemma 5.7.1. *Suppose we have* $\mathcal{L}_\theta(\theta|y) = \log p_\theta(y|\theta)$ *and* $h_{\theta_0}(\theta) = Q(\theta, \theta_0) + g(\theta_0)$, *then the following two statements are valid:*

i.

$$\mathcal{L}_\theta(\theta^0|y) \geq Q(\theta, \theta^0) + g(\theta^0), \quad \forall \theta, \theta^0,$$

ii.

$$\mathcal{L}_\theta(\theta|y) = Q(\theta,\theta) + g(\theta), \quad \forall\theta.$$

Proof. Within this proof, we will use the concept of information entropy to describe the discrepancy between two density functions. Readers need no prior knowledge of information entropy to understand the proof of the above lemma. For part (i), By definition $\theta^{t+1} \in \underset{\theta}{\text{argmax}} \quad Q(\theta,\theta^m)$, where $Q(\theta,\theta^m) = \mathbb{E}_{X|y,\theta^m}\big\{\log p_\theta(y,X|\theta)\big\}$.

$$\begin{aligned}
\mathcal{L}_\theta(\theta|y) &= \log p_\theta(y|\theta) = \sum_{\mathfrak{X}} q(x)\log p_\theta(y|\theta) \\
&= \sum_{\mathfrak{X}} q(x)\log\Big[p_\theta(y|\theta)p_\theta(x|y,\theta)\frac{q(x)}{p_\theta(x|y,\theta)}\frac{1}{q(x)}\Big] \\
&= \sum_{\mathfrak{X}} q(x)\log p_\theta(x,y|\theta) + \sum_{\mathfrak{X}} q(x)\log\frac{q(x)}{p_\theta(x|y,\theta)} - \sum_{\mathfrak{X}} q(x)\log q(x),
\end{aligned}$$

where

$$D\big\{q||p_\theta(.|y)\big\} = \sum_{\mathfrak{X}} q(x)\log\frac{q(x)}{p_\theta(x|y,\theta)}$$

has often been referred to as the relative entropy, and

$$H(q) = \sum_{\mathfrak{X}} q(x)\log q(x)$$

is called as the entropy of q. Moreover, since we are able to choose the density function of q, such that $q(x) = p_\theta(x|y,\theta^0)$, then providing with the 0^{th} estimates, the above expression can be re-expressed as

$$\begin{aligned}
\mathcal{L}_\theta(\theta^0|y) &= \sum_{\mathfrak{X}} p_\theta(x|y,\theta^0)\log p_\theta(x,y|\theta) + D\big\{p_\theta(x|y,\theta^0)||p_\theta(x|y,\theta)\big\} \quad (5.7.1) \\
&\quad -H\big\{p_\theta(x|y,\theta^0)\big\} \\
&\geq \mathbb{E}_{\theta_0}\{\log p_\theta(x,y|\theta)\} + g(\theta^0) \\
&= Q(\theta,\theta^0) + g(\theta^0).
\end{aligned}$$

The reasons for the above equation (5.7.1) to stand are: the relative entropy will always be non-negative and the definition of function $g()$, and the definition of Q function such as:

$$D\big\{p_\theta(x|y,\theta^0)||p_\theta(x|y,\theta)\big\} \geq 0,$$

and

$$g(\theta^0) = H\big\{p_\theta(x|y,\theta^0)\big\}.$$

Hence we have proved part (i).
For part (ii), we know that if θ_0 is set to be θ, then the relative entropy will be equal to zero, which is:

$$D\big\{p_\theta(x|y,\theta)||p_\theta(x|y,\theta)\big\} = 0.$$

therefore we have

$$\mathcal{L}_\theta(\theta|y) = Q(\theta,\theta) + g(\theta).$$

□

Provided with the establishment of the above lemma, the proof of Proposition (5.3.1) becomes straightforward.

Proof.

$$\begin{aligned}
\log p_\theta(y|\theta^k) = \mathcal{L}_\theta(theta^k|y) &= Q(\theta^k,\theta^k) + g(\theta^k) \quad \text{by (ii) of Lemma A.1} \\
&\leq \max_\theta Q(\theta,\theta^k) + g(\theta^k) \\
&= Q(\theta,\theta^{k+1}) + g(\theta^k) \quad \text{by definition} \\
&\leq \mathcal{L}_\theta(y|\theta^{k+1}), \quad \text{by (i) of Lemma A.}
\end{aligned}$$

hence, $\mathcal{L}_\theta(\theta^k|y) \leq \mathcal{L}_\theta(\theta^{k+1}|y)$. □

Appendix B. The Cox-Ingersoll-Ross model

The variance and covariance of the joint log-likelihood function $\log p_\theta(X, R|\theta^m)$ is

$$\begin{aligned}
\frac{\partial \log p_\theta(X,R|\theta^m)}{\partial \mu} &= \sum_{t=2}^{T} \exp(x_{t-1})\big\{x_t - \mu^m - x_{t-1} \\
&\quad -\phi^m \exp(-x_{t-1})\big\} \qquad (5.7.2)\\
\frac{\partial \log p_\theta(X,R|\theta^m)}{\partial \phi} &= \sum_{t=2}^{T} \big\{x_t - \mu^m - x_{t-1} - \phi^m \exp(-x_{t-1})\big\} \qquad (5.7.3)\\
\frac{\partial \log p_\theta(X,R|\theta^m)}{\partial \alpha} &= 0.5\sum_{t=1}^{T}\{\exp(r_t - x_t - \alpha^m - 1.2479) - 1\}, \qquad (5.7.4)\\
\frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \mu^2} &= -\sum_{t=2}^{T} \exp(x_{t-1}), \qquad (5.7.5)\\
\frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \mu \partial \phi} &= -(T-1), \qquad (5.7.6)\\
\frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \mu \partial \phi} &= \frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \phi \partial \mu}, \qquad (5.7.7)\\
\frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \phi^2} &= -\sum_{t=2}^{T} \exp(-x_{t-1}), \qquad (5.7.8)\\
\frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \alpha^2} &= -0.5\sum_{t=1}^{T}\{\exp(r_t - x_t - \alpha^m - 1.2479)\}, \qquad (5.7.9)\\
\frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \mu \partial \alpha} &= 0, \qquad (5.7.10)\\
\frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \phi \partial \alpha} &= 0. \qquad (5.7.11)
\end{aligned}$$

Therefore the gradient is:

$$\frac{\partial \log p_\theta(X,R|\theta^m)}{\partial \theta'} = \left\{ \begin{array}{ccc} \frac{\partial \log p_\theta(X,R|\theta^m)}{\partial \mu} & \frac{\partial \log p_\theta(X,R|\theta^m)}{\partial \phi} & \frac{\partial \log p_\theta(X,R|\theta^m)}{\partial \alpha} \end{array} \right\}.$$

and the second derivative is:

$$\frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \theta \partial \theta'} = \left\{ \begin{array}{ccc} \frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \mu^2} & \frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \phi \partial \mu} & \frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \alpha \partial \mu} \\ & \frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \phi^2} & \frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \alpha \partial \phi} \\ & & \frac{\partial^2 \log p_\theta(X,R|\theta^m)}{\partial \alpha^2} \end{array} \right\}.$$

Appendix C. The Phillips curve

The variance and covariance of the joint log-likelihood function $\log p_\theta(U^n, \Pi|\theta^m)$ is

$$\frac{\partial \log p_\theta(U^n,\Pi|\theta^m)}{\partial\gamma} = \sum_{t=2}^{T} B_t U_{t-1}^n \big\{U_{t-1}^n - \gamma^m U_{t-1}^n - \phi^m(1+\exp(-U_{t-1}^n))^{-1}\big\}, \quad (5.7.12)$$

$$\frac{\partial \log p_\theta(U^n,\Pi|\theta^m)}{\partial\phi} = \sum_{t=2}^{T} B_t(1+\exp(-U_{t-1}^n))^{-1}\big\{U_{t-1}^n - \gamma^m U_{t-1}^n - \phi^m(1+\exp(-U_{t-1}^n))^{-1}\big\}, \quad (5.7.13)$$

$$\frac{\partial \log p_\theta(U^n,\Pi|\theta^m)}{\partial\beta} = \sum_{t=1}^{T}(U_t - U_t^n)\big\{\Pi_t - \Pi_t^e - \beta^m(U_t - U_t^n)\big\}, \quad (5.7.14)$$

$$\frac{\partial^2 \log p_\theta(U^n,\Pi|\theta^m)}{\partial\gamma^2} = -\sum_{t=2}^{T} B_t(U_{t-1}^n)^2, \quad (5.7.15)$$

$$\frac{\partial^2 \log p_\theta(U^n,\Pi|\theta^m)}{\partial\gamma\partial\phi} = -\sum_{t=2}^{T} B_t U_{t-1}^n(1+\exp(-U_{t-1}^n)^{-1}, \quad (5.7.16)$$

$$\frac{\partial^2 \log p_\theta(U^n,\Pi|\theta^m)}{\partial\gamma\partial\phi} = \frac{\partial^2 \log p_\theta(U^n,\Pi|\theta^m)}{\partial\phi\partial\alpha}, \quad (5.7.17)$$

$$\frac{\partial^2 \log p_\theta(U^n,\Pi|\theta^m)}{\partial\phi^2} = -\sum_{t=2}^{T} B_t\big(1+\exp(-U_{t-1}^n)\big)^{-2}, \quad (5.7.18)$$

$$\frac{\partial^2 \log p_\theta(U^n,\Pi|\theta^m)}{\partial\beta^2} = -\sum_{t=1}^{T}(U_t - U_t^n)^2, \quad (5.7.19)$$

$$\frac{\partial^2 \log p_\theta(U^n,\Pi|\theta^m)}{\partial\gamma\partial\beta} = 0, \quad (5.7.20)$$

$$\frac{\partial^2 \log p_\theta(U^n,\Pi|\theta^m)}{\partial\phi\partial\beta} = 0. \quad (5.7.21)$$

The gradient and the second derivative of the joint log-likelihood function can be expressed in the same manner as in Appendix B.

Appendix D. Computing and Data Sources

All the calculations reported in this paper were condcuted using my own Matlab code on Matlab 2010a on a Intel(R) Core(TM)2 Duo CPU E8400 @3.00GHZ, and 2.96GB of RAM. All data series used for application in this paper are taken from the UK's Datastream. For the daily exchange rate data, Datastream does not record the exchange rates at weekends, which gives approximately 261 observations per year.

Chapter 6

Conclusions and Future Work

6.1 Summary and Contributions

This thesis makes several contributions to the literature of both particle filtering and particle smoothing. The main contribution of Chapter 3 was the establishment of the Shannon information entropy diagnostics. The proofed proposition and various demonstrations have shown that the new test provides computational reduction to the widely implemented efficient sample size test in particle filtering. In addition, the empirical evidences reveal that particle filters with Shannon information entropy diagnostics maintain or even improve the estimation accuracy compared to other particle filters without degeneracy diagnostics or particle filters with the effective sample size diagnostics.

The contribution of Chapter 4 has been the development of the modified Entropy particle filter. In which the modified filter seems to provide substantial improvements compared to the Entropy particle filter, which was due to Liverani and Papavasiliou [2006]. The improvements that derived from the simulation studies can be categorized on two aspects: the state variable estimation accuracy and the stability of the unknown parameter.

In particle smoothing, the forward filtering backward sampling algorithm has massive computational advantage over both the forward filtering backward smoothing algorithm and the generalized two-filter smoothing algorithm. Chapter 5 forms a new forward filtering backward sampling algorithm (HPZ smoothing), which is through the modification on the basis of the forward filtering backward sampling algorithm of Godsill et al. [2004]. In which the HPZ smoothing shows that it performs as least as good as the other smoothing algorithm in terms of state variable estimation. Furthermore, Chapter 5 makes several extensions on the combination

of EM algorithm and particle techniques for off-line parameter estimation of general state space models. More specifically, the extensions were made on the particle techniques of the combination estimation method. The particle techniques are figuratively pointing to the classes of particle filtering and particle smoothing. A new combined EM and particle techniques were formed by equipping using the Shannon information entropy diagnostics to particle filtering and HPZ smoothing to particle smoothing. This novel and newly formalized estimation method has been given the name of EM-HPZ method in the thesis. The novel parameter estimation method has been implemented in both simulated processes and real time series processes, where their respective performances have shown very promising results as comparing to existing estimation methods for general state space models.

6.2 Future Work

In the work of the Shannon information entropy diagnostics, I made no further attempt in exploring and determining the existence of a diagnostics formula that gives the greatest lower bound for each particle set at each time stance. My initial thought of the question was that each different diagnostics formulas could be essentially different ways of measuring variations of the set of particle weights. It may be incredibly difficult to find a measuring formula that fulfils both expectations of the greatest computational cost reduction and the most accurate estimation at once. However, additional efforts to the problem should provide great return.

In the work of the modified Entropy particle filter, both the modified Entropy particle filter and the Entropy particle filter face great computational difficulty whilst the number of unknown parameters increases. In addition, Entropy particle filters in general require the number of observations to be large in order to achieve precise state variable and parameter estimations. However, large observations may not always be possible in reality situations. These two outlined problems have been studied and with their potential solutions are due to be revealed in the near future.

A natural extension of the work of Chapter 5 on the combined EM algorithm and particle techniques would be to estimate multivariate state space models. Though embarking upon the investigation of the preceding problem will encounter formidable challenges. Two of the notable challenges are: firstly how to handle the derivation and implementation difficulties involved with the inference that are caused by the increasing dimensions. Secondly, how would one cater the performance of particle techniques as it degrades due to the state and parameter dimensions increase?

Though it may be the case that fewer of the challenges are foreseeable and others are mostly unforeseeable, it will be interesting to discover the potential ideas that would allow us to go further.

Appendix

A list of Notations

- X_t: random variable/the state variable at time stance t.
- $\{X_t\}_{t=1}^{T}$: the sequence of random variable/the state variable process.
- Y_t: random variable/the Observation variable at time stance t.
- y_t: the realization or observation of Y_t.
- $E(X)$: the expectation of random variable X.
- $\mathfrak{I}_t$: the collection of observation $\{y_1, \cdots, y_t\}$.
- V_t: the disturbance of transition equation.
- W_t: the disturbance of observation equation.
- $K(.,.)$: the transition kernel function.
- θ: the population parameter vector.
- Θ: the population parameter space.
- $p(.)$: probability density/mass function.
- $p_\theta(x_t|\mathfrak{I}_{t-1})$: the predictive density function.
- $p_\theta(x_t|\mathfrak{I}_t)$: the filter density function.
- $p_\theta(x_t|\mathfrak{I}_T)$: the smoothing density function.
- $f_\theta(x_t|x_{t-1})$: the transition density function.
- $g_\theta(y_t|x_t)$: the observation density function.

- x_t^i: the i^{th} particle at time stance t.
- ω_t^i: the i^{th} particle weight at time stance t.
- $\tilde{\omega}_t^i$: the normalized i^{th} particle weight at time stance t.
- $\Phi_t^{M,N}(\mu \otimes u)$: the notation used in Liverani and Papavasiliou [2006], which is the particle filter with the initial state density μ and prior density of the parameter u.
- a: the true distribution of population parameter θ.
- $\omega_t(\theta_i)$: the weight of the i^{th} parameter sample at time stance t in Entropy particle filter.
- $\lceil . \rceil$: the ceiling of a real number.
- $\lfloor . \rfloor$: the floor of a real number.
- $\psi_\theta(X_{t-1}, V_t, \theta)$: the functional form for transition equation.
- $\phi_\theta(X_{t-1}, W_t, \theta)$: the functional form for observation equation.
- E: denote the entropy value.
- $Q(.,.)$: the objective function in EM algorithm.
- μ_1: the initial distribution of state variable X_1.

Data

- Π_t: the inflation rate at time stance t.
- Π_t^e: the expected inflation at time stance t.
- U_t: the unemployment rate at time stance t.
- U_t^n: the natural unemployment rate at time stance t.
- R_t: the logarithmic of square of the transformed exchange rate (the exchange rate of interest) for time stance t.

References

Anderson, B. and J. Moore (1979). *Optimal Filtering*. Prentice Hall. 1, 16, 26

Andrieu, C., A. Doucet, and R. Holenstein (2010). Particle markov chain monte carlo methods. *Journal of Royal Statistical Society. B 72*, 269–342. 89, 90

Arulampalam, M., S. Maskell, N. Gordon, and T. Clapp (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Trans on Signal Proc, 50*(2), 174–188. 2, 29, 49, 53, 59, 62, 71, 72

Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics 31*(3), 307–327. 16

Bresler, Y. (1986). Two-filter formula for discrete-time non-linear bayesian smoothing. *International Journal of Control 43*(2), 629–641. 11, 33, 34, 88, 91

Briers, M., A. Doucet, and S. Maskell (2010). Smoothing algorithms for state-space models. *Ann Inst Stat Math 62*, 61–89. 11, 33, 35, 36, 88, 89, 90, 91, 106

Capinski, M. and E. Kopp (2005). *Measure, Integral and Probability*. Springer. 12

Cappe, O., E. Moulines, and T. Ryden (2005). *Inference in Hidden Markov Models*. Springer. 15, 55

Carpenter, J., P. Clifford, and P. Fearnhead (1999). An improved particle filter for non-linear problems. *IEE proceedings - Radar, Sonar and Navigation 146*, 2–7. 43

Chen, R. and J. Liu (2000). Mixture kalman filter. *Journal of the Royal Statistical Society, Series B 62*, 493–508. 33

Creal, D. (2012). A survey of sequential monte carlo methods for economics and finance. *Econometric Reviews 31*, 245–296. 10, 33

Crisan, D. and A. Doucet (2000). Convergence of sequential monte carlo methods. Technical report, Cambridge University, Engineering Dept. Technical Report CUED/F-INFENG/TR.381. 2

Crisan, D. and A. Doucet (2002). A survey of convergence results on particle filtering methods for parctitioners. *IEEE Transactions on Signal Processing 50*(3), 736–746. 29, 30

DeJong, D. and C. Dave (2006). *Structural Marcoeconometrics*. Princeton Univeristy Press, Princeton and Oxford. 2

Douc, R., O. Cappe, and E. Moulines (2005). Comparison of resampling schemes for particle filtering. In *International Symposium on Image and Signal Processing and Analysis*. 43, 52, 61

Doucet, A., N. de Freitas, and N. gordon (2001). *Sequential Monte Carlo in Practice*. Springer. 2, 88

Doucet, A., S. Godsill, and C. Andrieu (2000a). On sequential monte carlo sampling methods for bayesian filtering. *Statistics and Computting 10*, 197–208. 44, 46, 59, 88

Doucet, A., S. Godsill, and C. Andrieu (2000b). On sequential monte carlo sampling methods for bayesian filtering. *Statistics and Computing 10*, 197–208. 88, 91

Doucet, A. and A. Johansen (2008, Dec). A tutorial on partticle filtering and smoothing: Fifteen years later. 29, 33, 50, 62

Doucet, A. and V. Tadic (2003). Parameter estimation in general state-space models using particle methods. *Ann. Inst. Statist. Math. 55*(2), 409–422. 44, 69, 89, 90, 106

Durbin, J. and S. J. Koopman (2000). Time series analysis of non-gaussian observations based on state space models from both classical and bayesian perspectives. *Journal Of The Royal Statistical Society Series B 62*(1), 3–56. 2, 4, 6, 17, 88, 95, 105, 106

Durham, G. and J. Geweke (2012, June). Adaptive sequential posterior simulators for massively parallel computing environments. 7

Engle, R. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica 50*(4), 987–1007. 16

Fearnhead, P. (2002). Mcmc, sufficient statistics and particle filter. *Journal of Computational Graphical Statistics 11*, 848–862. 89

Fernandez-Villaverde, J. and J. Rubio-Ramirez (2005). Estimating dynamic equilibrium economies: Linear versus nonlinear likelihood. *Journal of Applied Econometrics 20*, 891–910. 42, 88

Fernandez-Villaverde, J. and J. Rubio-Ramirez (2007). Estimating marcoeconomic models: A likelihood approach. *Review of Economic Studies 74*, 1059–1087. 2, 88, 108, 115

Fernandez-Villaverde, J., J. Rubio-Ramirez, and M. Santos (2006). Convergence properties of the likelihood of computed dynamic models. *Econometrica*. 115

Godsill, S., A. Doucet, and M. West (2001). Maximum a posteriori sequence estimation using monte carlo particle filters. *Ann Inst and Statist Math*, 1–15. 2

Godsill, S., A. Doucet, and M. West (2004). Monte carlo smoothing for nonlinear time series. *Journal of the American Statistical Association 99*, 156–168. 8, 87, 88, 90, 91, 93, 95, 120

Gordon, N., S. Salmond, and A. Smith (1993). Novel approach to nonlinear/non-gaussian bayesian state estimation. *IEE Proceeding-F 140*, 107–113. 2, 10, 27, 32, 42, 47, 66, 68, 71, 88

Hamilton (1994). *Time Series Analysis*. Princeton University Press. 1, 22, 26

Hamilton, J. (1989). A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica 57*, 357384. 17, 18

Hamilton, J. (2005, May). Regime-switiching models. 18

Harvey, A. and G. Phillips (1979). Maximum likelihood estimation of regression models with autoregressive - moving average disturbances. *Biometrika 66*(1), 49–58. 1

Harvey, A., E. Ruiz, and N. Shephard (1994). Multivariate stochastic variance models. *Review of Economic Studies 61*, 247–264. 6, 17, 104

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering 82*, 35–45. 1, 10, 16

Kalman, R. and R. Bucy (1961). New results in linear filtering and prediction theory. *Journal of Basic Engineering 83*, 95–107. 16

Kantas, N., A. Doucet, S. Singh, and J. Maciejowski (2011, Nov). An overview of sequential monte carlo methods for parameter estimation in general state-space models. 6, 69, 89

Kantas, N., A. Doucet, S. Singh, J. Maciejowski, and N. Chopin (2011, Nov). On particle methods for parameter estimation in general state-space models. 89, 91

Kim, C. and C. Nelson (1999). *State-Space Models with Regime Switching: Classical and Gibbs-Sampling Approaches with Applications*. MIT Press. 17

Kim, J. (2005). *Parameter Estimation in Stochastic Volatility Models with Missing Data Using Particle Methods and the EM Algorithm*. Ph. D. thesis, Department of Statistics, University of Pittsburgh. 90, 101

Kitagawa, G. (1987). Non-gaussian state-space modeling of nonstationary time series. *Journal of the American Statistical Association 82*, 1032–1063. 11, 34, 68, 88, 91

Kitagawa, G. (1994). The two-filter formula for smoothing and an implementation of the gaussian-sum smoother. *Ann. Inst. Statist. Math 46*(4), 605–623. 42

Kitagawa, G. (1996). Monte carlo filter and smoother for non-gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics 5*(1), 1–25. 42, 43, 44

Kong, A., J. Liu, and W. Wong (1994). Sequential imputations and bayesian missing data problems. *Journal of the American Statistical Association 89*, 278–288. 43, 53, 54

Liu, J. (1996). Metropolized independent sampling with comparison to rejection sampling and importance sampling. *Statistics and Computing 6*, 113–119. 32, 43, 51, 53

Liu, J. and R. Chen (1998). Sequential monte carlo methods for dynamical systems. *Journal of American Statistical Association 93*, 1032–1044. 88

Liu, J. and M. West (2001). *Sequential Monte Carlo in Practice*, Chapter Combined parameter and state estimation in simulation-based filtering, pp. 197–223. Springer. 33, 68, 69, 89

Liverani, S. (2006). Particle filters and adaptive estimation. Master's thesis, University of Warwick, Statistics. 82

Liverani, S. and A. Papavasiliou (2006). Entropy based adaptive particle filter. In *Nonlinear Statistical Signal Processing Workshop, 2006 IEEE*, pp. 87 –90. 6, 7, 59, 68, 69, 70, 73, 74, 77, 78, 79, 81, 83, 84, 120, 124

Lopes, H. and R. Tsay (2011). Particle filters and bayesian inference in financial econometrics. *Journal of Forecasting 30*, 168–209. 10, 33

Muller, P. (1990). Monte carlo integration in general dynamic models. *Working paper version, later published in Contemporary Mathematics*. 27

Papavasiliou, A. (2005). A uniformly convergent adaptive particle filter. *Journal of Applied Probability 42*, 1053–1068. 77

Pitt, M. and N. Shephard (1999). Filtering via simulation: Auxiliary particle filters. *Journal of the American Statistical Association 94*, 590–599. 2, 10, 32, 42, 49

Pitt, M., R. Silva, P. Giordani, and R. Kohn (2012). On some properties of markov chain monte carlo simulation methods based on the particle filter. *Journal of Econometrics*. 2, 90

Pollock, D. (2003, April). Recursive estimation in econometrics. 16, 21, 22

Pollock, D., H. Merkus, and A. de Vos (1993). A synopsis of the smoothing formulae associated with the kalman filter. *Computational Economics 6*, 177–200. 1, 10, 21, 23, 26

Poyiadjis, G., A. Doucet, and S. Singh (2005). Maximum likelihood parameter estimation in general state-space models using particle methods. 2, 89

Poyiadjis, G., A. Doucet, and S. Singh (2011). Particle approximations of the score and observed information matrix in state space models with application to parameter estimation. *Biometrika 98*(1), 65–80. 4, 103

Shephard, N. (1996). *In Econometrics, finance, and other fields*, Chapter Statistical aspects of ARCH and stochastic volatility. Chapman and Hall, London. 4, 42, 103, 104

Smith, A. and A. Gelfand (1992). Bayesian statistics without tears: A sampling-resampling perspective. *The American Statistician 46*, 84–88. 27

Stock, J. and M. Watson (1988). A probability moel of the coincident economic indicators. Technical report, NBER Working Paper Series. 1

Storvik, G. (2002). Particle filters for state-space models with the presence of unknown static parameters. *IEEE Transactions on Signal Processing 50*(2), 281–289. 89