MAXIMUM LIKELIHOOD ESTIMATION UNDER EFFICIENT IMPORTANCE SAMPLING FOR NON-LINEAR STATE SPACE MODELS

By

SHITENG YANG

Bachelor of Science in Statistics
Hangzhou Dianzi University
Hangzhou, China
2013

Master of Science in Applied Statistics
Rochester Institute of Technology
Rochester, NY
2016

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
December, 2021

MAXIMUM LIKELIHOOD ESTIMATION UNDER EFFICIENT IMPORTANCE SAMPLING FOR NON-LINEAR STATE SPACE MODELS

Dissertation Approved:

Dr. Ye Liang

Dissertation Advisor

Dr. Lan Zhu

Dr. Pratyaydipta Rudra

Dr. Wenyi Shen

ACKNOWLEDGMENTS

I would like to acknowledge and give my warmest thanks to my advisor Professor Ye Liang for the continuous support of my PhD study and related research, for his patience, motivation,and immense knowledge. His guidance and advice carried me through all the stages of writing my dissertation. The door to his office was always open whenever I ran into a trouble spot or had a question about my research or writing. He consistently allowed this dissertation to be my own work, but steered me in the right direction whenever he thought I needed it.

I would also like to thank my committee members: Professor Lan Zhu, Professor Pratyaydipta Rudra and Professor Wenyi Shen, for letting my defense be an enjoyable moment, and for their insight comments and suggestions which incent me to widen my research from various perspectives, thanks to you.

Finally, I must express my very profound gratitude to my parents for providing me with unfailing support and continuous encouragement throughout my years of PhD study and through the process of researching and writing this dissertation. This accomplishment would not have been possible without them.

Acknowledgments reflect the views of the author and are not endorsed by committee members or Oklahoma State University.

Name: SHITENG YANG

Date of Degree: DECEMBER, 2021

Title of Study: MAXIMUM LIKELIHOOD ESTIMATION UNDER EFFICIENT IMPORTANCE SAMPLING FOR NON-LINEAR STATE SPACE MODELS

Major Field: STATISTICS

Abstract: The interest of this dissertation lays on the Likelihood Evaluation and Maximum Likelihood (ML) Parameter Estimation on the Non-linear State Space Model in which the analytical solution is not available. An algorithm known as Efficient Importance Sampling (EIS) is adopted for the continuous approximation of likelihood function and we proposed a method to further improve its performance by accomplishing a more precise calculation on the weight functions. With respect to the ML parameter estimation, we proposed a Monte Carlo EM algorithm based on EIS procedure and Constant-Weight principle to achieve lower computational complexity and better performance on parameter estimation in comparison with algorithms based on Particle Filters. Moreover, by paying a small price on the estimation performance, we further developed a technique known as Fast-Sampling for our proposed EIS-based EM algorithm to realize more computational efficiency gain. Finally, we illustrate these developed algorithm and technique in applications to the Dynamic Stochastic General Equilibrium modeling which is a very popular methodology designed for Macroeconomics analysis.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

LITERATURE REVIEW

1.1 Nonlinear State Space Models

The State space model (SSM), also referred to as the Hidden Markov Model (HMM) [33] or the Latent Process Model (LPM), is a well-known statistical model for sequential data analysis [20]. It describes the statistical probabilistic dependence between the hidden variable and the observable data. Numerous applications of SSM have already been found in the fields of economics, engineering, statistics, environmental sciences and neuroscience to solve complicated dynamical system problems[3].

This model consists of two stochastic processes. The first process, denoted as $\{s_t\}_{t\geq 0}$, is called hidden Markov process where s_t are the latent variables with initial density $\mu(s_0)$ and transition density $f(s_t|s_{t-1})$ satisfying Markov property, that is,

$$s_0 \sim \mu(s_0) \tag{1.1.1}$$

$$s_t|(s_{0:t-1}) = s_t|s_{t-1} \sim f(s_t|s_{t-1}) \tag{1.1.2}$$

The second process, denoted as $\{y_t\}_{t\geq 0}$, is referred to as measurement process where y_t are the observations with a density conditional on s_t and satisfying the following property,

$$y_t|(s_{0:t}, y_{0:t-1}) = y_t|s_t \sim g(y_t|s_t) \tag{1.1.3}$$

The simplest form of SSM is the dynamic linear model [39] which is just the state space model with linearity and Gaussian property, that is,

$$y_t = F_t s_t + v_t, v_t \sim N(0, V_t) \tag{1.1.4}$$

$$s_t = G_t s_{t-1} + w_t, w_t \sim N(0, W_t) \tag{1.1.5}$$

The target of interest is of likelihood function $f(y_{0:t})$ evaluation and state inference, i.e., filtering density $f(s_t|y_{0:t})$ and moment $E(h(s_t)|y_{0:t})$ evaluation. This target can be directly achieved in dynamic linear model by applying Kalman Filter Procedure [38]. However, when state space model is non-linear or non-Gaussian, analytic solution or expression is not available. In this context, various numerical procedures have already been proposed over the last few decades to perform the approximation of likelihood and filtering density. We focus on two well known methods, the PF and EIS in this dissertation.

1.2 Particle Filters

Particle filters (PF) are firstly introduced in 1993[14] and have already become the most commonly used numerical procedures to perform state inference and likelihood approximation for non-linear non-Gaussian state space model [10]. Essentially speaking, PF methods are the procedures consisting of two steps. The first step is the sampling. In this step, an appropriate sampler will be selected and further be used to draw particles and calculate corresponding weights. The second step is the approximation. In this step, previously obtained particles and weights will be used to perform approximation to filtering density $f(s_t|y_{0:t})$ and likelihood $f(y_{0:t})$.

Numerous PF algorithms have already been proposed over last few decades. To have a better understanding about the advantage or weakness of these algorithms, a brief review will be given in the following sections.

1.2.1 Sequential Importance Sampling (SIS)

The Sequential Importance Sampling (SIS) [15, 16] is the earliest developed algorithm of Particle Filters. The sampler in SIS, denoted as $q_t(s_{1:t})$, has the sequential structure, that is,

$$\begin{aligned} q_t(s_{0:t}) &= q_t(s_t|s_{0:t-1})q_{t-1}(s_{0:t-1}) \\ &= q_0(s_0)\prod_{k=1}^{t} q_k(s_k|s_{0:k-1}) \end{aligned} \tag{1.2.1}$$

By the theory of Importance sampling and Bayesian inference, we can further have,

$$f(s_{0:t}|y_{0:t}) = \frac{\frac{f(s_{0:t},y_{0:t})}{q_t(s_{0:t})}q_t(s_{0:t})}{f(y_{0:t})} = \frac{\frac{f(s_{0:t},y_{0:t})}{q_t(s_{0:t})}q_t(s_t|s_{0:t-1})q_{t-1}(s_{0:t-1})}{f(y_{0:t})} \tag{1.2.2}$$

$$w_t = \frac{f(s_{0:t}, y_{0:t})}{q_t(s_{0:t})} \tag{1.2.3}$$

$$W_t^i = \frac{w_t^i}{\sum_{i=1}^N w_t^i} \tag{1.2.4}$$

Here, w_t^i and W_t^i are referred to as the unnormalized and normalized weight respectively.

Under the context of non-linear non-Gaussian state space model, the analytical expression of filtering density $f(s_{0:t}|y_{0:t})$ is not tractable. Instead, an appropriate sampler $q_t(s_{0:t})$ with sequential structure will be selected. By Bayesian inference shown in equation1.2.2, particles $\{s_{0:t}^i\}_{i=1}^N$ will be draw from this sampler and corresponding weights $\{w_t^i\}_{i=1}^N$ will be calculated. As a result, the filtering density $f(s_{0:t}|y_{0:t})$ will be approximated in the form of mixtures of Dirac measures, that is,

$$\hat{f}(s_{0:t}|y_{0:t}) = \sum_{i=1}^{N} W_t^i \times \delta_{s_{0:t}^i}(s_{0:t}) \tag{1.2.5}$$

Here, $\delta_{s^i_{0:t}}(s_{0:t}) = 1$ when $s_{0:t} = s^i_{0:t}$, otherwise, it is equal to zero.

Next, let $\alpha_t = \frac{f(y_t|s_t)f(s_t|s_{t-1})}{q_t(s_t|s_{0:t-1})}$, equation1.2.3 can be further simplified as,

$$\begin{aligned} w_t = \frac{f(s_{0:t}, y_{0:t})}{q_t(s_{0:t})} &= \frac{f(s_{0:t}, y_{0:t})}{f(s_{0:t-1}, y_{0:t-1})q_t(s_t|s_{0:t-1})} \times \frac{f(s_{0:t-1}, y_{0:t-1})}{q_{t-1}(s_{0:t-1})} \\ &= \frac{f(y_t|s_t)f(s_t|s_{t-1})f(s_{0:t-1}, y_{0:t-1})}{f(s_{0:t-1}, y_{0:t-1})q_t(s_t|s_{0:t-1})} \times w_{t-1} \\ &= \frac{f(y_t|s_t)f(s_t|s_{t-1})}{q_t(s_t|s_{0:t-1})} \times w_{t-1} \\ &= \alpha_t \times w_{t-1} \end{aligned} \tag{1.2.6}$$

Here, α_t is referred to as incremental importance weight. The detailed SIS procedure is as follows,

---

Sequential Importance Sampling

- At time t=0
  - Draw particles $\{s^i_0\}^N_{i=1}$ from $q_0(s_0)$
  - Compute the unnormalized weight $w^i_0 = f(y_0|s^i_0)f(s^i_0)/q_0(s^i_0)$ and normalized weight $W^i_0 \propto w^i_0$
- At time t$\geq$ 1, suppose we inherit $\{(w^i_{t-1}, s^i_{0:t-1})\}^N_{i=1}$
  - Draw particles $\{s^i_t\}^N_{i=1}$ from $q_t(s_t|s^i_{0:t-1})$
  - Compute incremental important weight $\alpha^i_t = f(y_t|s^i_t)f(s^i_t|s^i_{t-1})/q_t(s^i_t|s^i_{0:t-1})$
  - Compute unnormalized weights $w^i_t = \alpha^i_t \times w^i_{t-1}$ and normalized weights $W^i_t \propto w^i_t$

By equation1.2.2, the likelihood value can be evaluated by

$$f(y_{0:t}) = \int \frac{f(s_{0:t}, y_{0:t})}{q_t(s_{0:t})} q_t(s_{0:t}) ds_{0:t} \tag{1.2.7}$$

$$\hat{f}(y_{0:t}) = \frac{1}{N}\sum_{i=1}^{N} \frac{f(s^i_{0:t}, y_{0:t})}{q_t(s^i_{0:t})} = \frac{1}{N}\sum_{i=1}^{N} w^i_t \tag{1.2.8}$$

And the moments can be evaluated by

$$E(h(s_t)|y_{0:t}) = \int h(s_t) \times f(s_t|y_{0:t}) ds_t \tag{1.2.9}$$

$$\hat{E}(h(s_t)|y_{0:t}) = \sum_{i=1}^{N} W^i_t \times h(s^i_t) \tag{1.2.10}$$

While Sequential Importance Sampling is intuitive and very easy to implement, there always arise one challenge problem known as weight degeneracy, i.e., as time goes on, the variance of the normalized weight will keep increasing [21, 2], as a result, in the extreme case, there is only one particle taking 1 as the value of normalized weight and all the others taking 0. Increasing the sample size is one way to remedy this problem, but this is at the cost of efficiency. By far, the most commonly recognized solution is to introduce resampling step in SIS. This resampling-based SIS is referred to as Sampling Importance Resampling (SIR)[14, 27].

1.2.2 Sampling Importance Resampling (SIR)

The Sampling Importance Resampling (SIR) algorithm is developed with the motivation of solving weight degeneracy problem. The basic idea of SIR originates from the fact that performing resampling can continuously initialize the weights at each time step. As a result, the normalized weights after resampling will all equal to $\frac{1}{N}$.

The framework of SIR algorithm is almost same with SIS with the exception of the inclusion of resampling procedure[35]. This resampling procedure is implemented to the discrete distribution of the approximation of filtering density $\hat{f}(s_{0:t}|y_{0:t}) = \sum_{i=1}^{N} W_t^i \times \delta_{s_{0:t}^i}(s_{0:t})$. The most intuitive resampling scheme is known as Multinomial Resampling, that is,

$$\{N_t^i\}_{i=1}^N \sim Multinomial\left(N, \{W_t^i\}_{i=1}^N\right) \tag{1.2.11}$$

An important advantage of resampling is reflected by the fact that, after resampling, with a very high probability, the particle with low weight will be removed while the particle with high weight will be duplicated, as a result, the computational power will be concentrated on the region of high probability mass. This may be the reason why SIR is the most commonly used approach against weight degeneracy problem.

Instead of multinomial resampling, the most widely praised resampling scheme is known as Systematic Resampling[19]. In most situation, it outperforms other resampling schemes. The detailed implementation is as follows,

---

Systematic Resampling

- Sample $U_1 \sim \mathbf{U}(0, \frac{1}{N})$ and calculate $U_i = U_1 + \frac{i-1}{N}$ for i=2,3,...,N
- Find the Cardinality of the set $\{U_j : \sum_{k=1}^{i-1} W_t^k \leq U_j \leq \sum_{k=1}^{i} W_t^k\}$ and denote it as N_t^i

---

Due to the resampling procedure, the sampler $q_t(s_{0:t})$ in SIR is a little different from the one in SIS, that is,

$$\begin{aligned} q_t(s_{0:t}) &= q_t(s_t|s_{0:t-1})q_{t-1}(s_{0:t-1}) \\ &= q_t(s_t|s_{0:t-1})f(s_{0:t-1}|y_{0:t-1}) \end{aligned} \tag{1.2.12}$$

Here, we can see that $q_{t-1}(s_{0:t-1})$ is replaced by the smoothing density $f(s_{0:t-1}|y_{0:t-1})$ at time

t-1. This replacement is due to the fact that,

$$\begin{aligned}
&\textbf{At time t-1: } \left\{W_{t-1}^i, s_{0:t-1}^i\right\}_{i=1}^N \xrightarrow{resample} \left\{\frac{1}{N}, \bar{s}_{0:t-1}^i\right\}_{i=1}^N \sim f(s_{0:t-1}|y_{0:t-1}) \\
&\textbf{At time t: } \left\{\frac{1}{N}, \bar{s}_{0:t-1}^i, s_t^i\right\}_{i=1}^N = \left\{\frac{1}{N}, s_{0:t}^i\right\}_{i=1}^N \sim q_t(s_t|s_{0:t-1}) f(s_{0:t-1}|y_{0:t-1})
\end{aligned} \tag{1.2.13}$$

Moreover, the Bayesian inference of filtering density in SIR is given by

$$\begin{aligned}
f(s_{0:t}|y_{0:t}) &= \frac{\frac{f(s_{0:t},y_{0:t})}{q_t(s_{0:t})} q_t(s_{0:t})}{f(y_{0:t})} \\
&= \frac{\frac{f(s_{0:t},y_{0:t})}{q_t(s_{0:t})} q_t(s_t|s_{0:t-1}) f(s_{0:t-1}|y_{0:t-1})}{f(y_{0:t})} \\
&= \frac{\frac{f(y_t|s_t) f(s_t|s_{t-1}) f(s_{0:t-1}|y_{0:t-1}) f(y_{0:t-1})}{q_t(s_t|s_{0:t-1}) f(s_{0:t-1}|y_{0:t-1})} q_t(s_t|s_{0:t-1}) f(s_{0:t-1}|y_{0:t-1})}{f(y_t|y_{0:t-1}) f(y_{0:t-1})} \\
&= \frac{\frac{f(y_t|s_t) f(s_t|s_{t-1})}{q_t(s_t|s_{0:t-1})} q_t(s_t|s_{0:t-1}) f(s_{0:t-1}|y_{0:t-1})}{f(y_t|y_{0:t-1})}
\end{aligned} \tag{1.2.14}$$

And, the unnormalized weight is,

$$w_t = \frac{f(y_t|s_t) f(s_t|s_{t-1})}{q_t(s_t|s_{0:t-1})} \tag{1.2.15}$$

Here, we can see that, the unnormalized weight in SIR is just equal to the incremental importance weight in SIS. And, instead of evaluating likelihood value for the whole time steps, in SIR, at each time step, we only evaluate conditional likelihood $f(y_t|y_{0:t-1})$ which is given by,

$$f(y_t|y_{0:t-1}) = \int \frac{f(y_t|s_t) f(s_t|s_{t-1})}{q_t(s_t|s_{0:t-1})} q_t(s_t|s_{0:t-1}) f(s_{0:t-1}|y_{0:t-1}) ds_{0:t} \tag{1.2.16}$$

$$\hat{f}(y_t|y_{0:t-1}) = \frac{1}{N} \sum_{i=1}^N \frac{f(y_t|s_t^i) f(s_t^i|\bar{s}_{t-1}^i)}{q_t(s_t^i|\bar{s}_{0:t-1}^i)} = \frac{1}{N} \sum_{i=1}^N w_t^i \tag{1.2.17}$$

Overall, the Sampling Importance Resampling algorithm is summarized as follows:

Sampling Importance Resampling

- At time t=0
  - Draw particles $\{s_0^i\}_{i=1}^N$ from $q_0(s_0)$
  - Compute the unnormalized weight $w_0^i = f(y_0|s_0^i) f(s_0^i)/q_0(s_0^i)$ and normalized weight $W_0^i \propto w_0^i$
  - Resample $\{W_0^i, s_0^i\}_{i=1}^N$ to obtain N equally-weighted particles $\{\frac{1}{N}, \bar{s}_0^i\}_{i=1}^N$

- At time t$\geq$ 1, suppose we inherit $\{\frac{1}{N}, \bar{s}^i_{0:t-1}\}^N_{i=1}$
  - Draw particles $\{s^i_t\}^N_{i=1}$ from $q_t(s_t|\bar{s}^i_{0:t-1})$ and set $s^i_{0:t} = (\bar{s}_{0:t-1}, s^i_t)$
  - Compute weight $w^i_t = f(y_t|s^i_t)f(s^i_t|\bar{s}^i_{t-1})/q_t(s^i_t|\bar{s}^i_{0:t-1})$ and $W^i_t \propto w^i_t$
  - Resample $\{W^i_t, s^i_{0:t}\}^N_{i=1}$ to obtain N equally-weighted particles $\{\frac{1}{N}, \bar{s}^i_{0:t}\}^N_{i=1}$

The most popular and basis SIR algorithm is known as Bootstrap Particle Filter (BPF) [14] in which the sampler consists of transition density $f(s_t|s_{t-1})$, that is,

$$q_t(s_{0:t}) = \mu(s_0) \times \prod_{k=1}^{t} f(s_k|s_{k-1}) \tag{1.2.18}$$

Clearly, as we can see, two kinds of weights are generated during each time step of SIR algorithm: weight before resampling and weight after resampling. Therefore, there are two ways to approximate the filtering density, that is,

$$\hat{f}(s_{0:t}|y_{0:t}) = \sum_{i=1}^{N} W^i_t \times \delta_{s^i_{0:t}}(s_{0:t}) \tag{1.2.19}$$

$$\bar{f}(s_{0:t}|y_{0:t}) = \frac{1}{N}\sum_{i=1}^{N} \delta_{\bar{s}^i_{0:t}}(s_{0:t}) \tag{1.2.20}$$

However, for the moments evaluation, the approximation 1.2.19 outperforms 1.2.20.

Although the performance of SIR is pretty good against weight degeneracy, another challenge problem, known as sample impoverishment, emerges as the consequence of the continuous implementation of resampling. As time goes on, particles with relatively high weights will be duplicated for many times while the ones with relatively low weights will be removed during the resampling procedure. This will end up with the loss of diversity for the particles in earlier time [32].

One way to partially solve this problem replies on the adaptive resampling, that is, the implementation of resampling is only allowed when pre-specified rule is met. The most commonly used rule is referred to as Effective Sample Size (ESS) criterion [24], which is given by,

$$ESS = \left(\sum_{i=1}^{N}(W^i_t)^2\right)^{-1} \tag{1.2.21}$$

The value range of ESS is between 1 and N and the resampling is allowed only when it is below a threshold N^* which typically equal to $N/2$. The detailed implementation of adaptive resampling is as follows,

---

Sampling Importance Adaptive Resampling

- At time t=0
  - Draw particles $\{s^i_0\}^N_{i=1}$ from $q_0(s_0)$

- Compute the weights $w_0^i = f(y_0|s_0^i)f(s_0^i)/q_0(s_0^i)$ and $W_0^i \propto w_0^i$
- Compute ESS and if ESS $< N/2$, then resample $\{W_0^i, s_0^i\}_{i=1}^N$ to obtain N equally-weighted particles $\{\frac{1}{N}, \bar{s}_0^i\}_{i=1}^N$ and set $\{\frac{1}{N}, \bar{s}_0^i\}_{i=1}^N \to \{\bar{W}_0^i, \bar{s}_0^i\}_{i=1}^N$;
  Otherwise, $\{W_0^i, s_0^i\}_{i=1}^N \to \{\bar{W}_0^i, \bar{s}_0^i\}_{i=1}^N$

- At time t$\geq$ 1, suppose we inherit $\{\bar{W}_{t-1}^i, \bar{s}_{0:t-1}^i\}_{i=1}^N$
  - Draw particles $\{s_t^i\}_{i=1}^N$ from $q_t(s_t|\bar{s}_{0:t-1}^i)$ and set $s_{0:t}^i = (\bar{s}_{0:t-1}, s_t^i)$
  - Compute weight $\alpha_t^i = f(y_t|s_t^i)f(s_t^i|\bar{s}_{t-1}^i)/q_t(s_t^i|\bar{s}_{0:t-1}^i)$ and $W_t^i \propto \alpha_t^i \times \bar{W}_{t-1}^i$
  - Compute ESS and if ESS $< N/2$, then resample $\{W_t^i, s_{0:t}^i\}_{i=1}^N$ to obtain N equally-weighted particles $\{\frac{1}{N}, \bar{s}_{0:t}^i\}_{i=1}^N$ and set $\{\frac{1}{N}, \bar{s}_{0:t}^i\}_{i=1}^N \to \{\bar{W}_t^i, \bar{s}_{0:t}^i\}$,
    otherwise, $\{W_t^i, s_{0:t}^i\}_{i=1}^N \to \{\bar{W}_t^i, \bar{s}_{0:t}^i\}$

However, even with Sampling Importance Adaptive Resampling, the impoverishment problem is still inevitable when time goes to infinity. A widely accepted reason for this challenge is that the sampling process at time t is only based on the particles $\{s_{0:t-1}^i\}_{i=1}^N$ from previous time and the information conveyed by observation y_t is missing during the process. To overcome this problem, particle filters with adaption have already been developed and the most well-known ones are Fully Adapted Particle Filter (FAPF) [1] and Auxiliary Particle Filter (APF) [32]. The adaption in this context is referred to as the inclusion of y_t information.

1.2.3 Fully Adapted Particle Filter (FAPF)

Fully Adapted Particle Filter, also known as Conditionally Optimal Particle Filter, has the same framework with Sampling Importance Resampling with the exception of the particularity of sampler which is based on the availability of the analytical expression of the following factorization:

$$f(s_t|s_{t-1})f(y_t|s_t) = f(s_t|s_{t-1}, y_t)f(y_t|s_{t-1}) \tag{1.2.22}$$

Here, we can see that the production of transition and measurement density can be factorized into the production of fully adapted sampler $f(s_t|s_{t-1}, y_t)$ and the predictive density $f(y_t|s_{t-1})$. At each time step, the sampling process is always implemented based on a sampler taking into account y_t information. To be specific, the sampler in FAPF is given by,

$$\begin{aligned} q_t(s_{0:t}) &= q_t(s_t|s_{0:t-1})q_{t-1}(s_{0:t-1}) \\ &= q_t(s_t|s_{0:t-1})f(s_{0:t-1}|y_{0:t-1}) \\ &= f(s_t|s_{t-1}, y_t)f(s_{0:t-1}|y_{0:t-1}) \end{aligned} \tag{1.2.23}$$

Here, $q_{t-1}(s_{0:t-1})$ is replaced by $f(s_{0:t-1}|y_{0:t-1})$ for the reason of resampling.

And By equation 1.2.14, the Bayesian inference of filtering density in FAPF is,

$$\begin{aligned} f(s_{0:t}|y_{0:t}) &= \frac{\frac{f(y_t|s_t)f(s_t|s_{t-1})}{q_t(s_t|s_{0:t-1})} q_t(s_t|s_{0:t-1}) f(s_{0:t-1}|y_{0:t-1})}{f(y_t|y_{0:t-1})} \\ &= \frac{\frac{f(y_t|s_t)f(s_t|s_{t-1})}{f(s_t|s_{t-1},y_t)} f(s_t|s_{t-1},y_t) f(s_{0:t-1}|y_{0:t-1})}{f(y_t|y_{0:t-1})} \\ &= \frac{[f(y_t|s_{t-1})] f(s_t|s_{t-1},y_t) f(s_{0:t-1}|y_{0:t-1})}{f(y_t|y_{0:t-1})} \end{aligned} \tag{1.2.24}$$

Therefore, the weights are given by

$$\begin{aligned} w_t^i &= f(y_t|s_{t-1}^i) \\ W_t^i &= w_t^i / \sum_{i=1}^N w_t^i \end{aligned} \tag{1.2.25}$$

Here, we can see that the normalized weight W_t^i in FAPF does not rely on s_t^i but only on s_{t-1}^i. So, in the context of the inheritance of particles $\{s_{t-1}^i\}_{i=1}^N$ from previous time period, the conditional variance $Var(W_t^i|s_{t-1}^i)$ is just equal to zero which achieves the objective of weight variance minimization optimally and conditionally. This is referred to as conditional optimality which explains why FAPF is also known as Conditionally Optimal Particle Filter[1, 41].

The conditional likelihood $f(y_t|y_{0:t-1})$ and its approximation in FAPF is given by,

$$\begin{aligned} f(y_t|y_{y_{0:t-1}}) &= \int f(y_t|s_t) f(s_t|s_{t-1}) f(s_{t-1}|y_{0:t}) ds_t ds_{t-1} \\ &= \int f(y_t|s_{t-1}) f(s_t|s_{t-1},y_t) f(s_{t-1}|y_{0:t}) ds_t ds_{t-1} \\ &= \int f(y_t|s_{t-1}) f(s_{t-1}|y_{0:t}) ds_{t-1} \int f(s_t|s_{t-1},y_t) ds_t \\ &= \int f(y_t|s_{t-1}) f(s_{t-1}|y_{0:t}) ds_{t-1} \end{aligned} \tag{1.2.26}$$

$$\hat{f}(y_t|y_{y_{0:t-1}}) = \frac{1}{N} \sum_{i=1}^N f(y_t|s_{t-1}^i) \tag{1.2.27}$$

Here, we can see that, given the fixed support of $\{s_{t-1}^i\}_{i=1}^N$, the variance of the estimator of conditional likelihood is just equal to zero which implies the conditional optimality. However, in most situation, the factorization in equation 1.2.22 is not tractable. Thus, we have to resort to other ways to achieve the adaption. And the most popular one is referred to as Auxiliary Particle Filter (APF)[32].

1.2.4 Auxiliary Particle Filters

Auxiliary Particle Filter Algorithm was first proposed by Pitt and Shephard in 1999 [32]. It has the same implementation procedure with the SIR algorithm except the design of sampler in which the observation information at current time step is incorporated. Moreover, it is the method for the replacement of Fully Adapted Particle Filter (FAPF) in which the factorization shown in 1.2.22 is not available.

To be specific, the APF algorithm is actually based on an auxiliary variable which is derived by,

$$\begin{aligned} f(s_t|y_{0:t}) &= \frac{f(y_t|s_t)\int f(s_t|s_{t-1})f(s_{t-1}|y_{0:t-1})ds_{t-1}}{f(y_t|y_{0:t-1})} \\ &\approx \frac{f(y_t|s_t)\sum_{k=1}^{N} W_{t-1}^k f(s_t|s_{t-1}^k)}{f(y_t|y_{0:t-1})} \end{aligned} \tag{1.2.28}$$

$$\begin{aligned} f(s_t,k|y_{0:t}) &= \frac{f(y_t|s_t)f(s_t|s_{t-1}^k)W_{t-1}^k}{f(y_t|y_{0:t-1})} \\ &= \frac{\frac{f(y_t|s_t)f(s_t|s_{t-1}^k)W_{t-1}^k}{g(s_t,k|y_t)}g(s_t,k|y_t)}{f(y_t|y_{0:t-1})}, \forall k = 1,2,..,N \end{aligned} \tag{1.2.29}$$

Here, $f(s_t,k|y_{0:t})$ is referred to as auxiliary filtering density. Our objective is to find an appropriate auxiliary sampler $g(s_t,k|y_t)$ taking into account y_t information and use this sampler to draw particles, calculate weights and perform resampling. As a result, after discarding the auxiliary variable k, N equally-weighted particles $\{\frac{1}{N}, s_t^i\}_{i=1}^N$ distributed approximately to filtering density $f(s_t|y_{0:t})$ will be obtained.

For the selection of auxiliary sampler, in [32], a generic form is proposed and given by,

$$g(s_t,k|y_t) \propto f(y_t|\mu_t^k)f(s_t|s_{t-1}^k)W_{t-1}^k \tag{1.2.30}$$

where μ_t can be the mean, the mode, a randomly draw or some other likely values of the transition density $f(s_t|s_{t-1}^k)$. To simulate from this auxiliary sampler, the first step is to sample from $g(k|y_t)$ which is given by,

$$\begin{aligned} g(k|y_t) &\propto \int f(y_t|\mu_t^k)f(s_t|s_{t-1}^k)W_{t-1}^k ds_t \\ &= f(y_t|\mu_t^k)W_{t-1}^k, \forall k = 1,2,3,...,N \end{aligned} \tag{1.2.31}$$

$$\lambda_k = \frac{g(k|y_t)}{\sum_{k=1}^{N} g(k|y_t)} \tag{1.2.32}$$

Here, λ_k is known as first-stage weight and the first step sampling procedure is implemented by drawing particles from this probability mass $\{\lambda_k, k\}_{k=1}^N$. The next step is to perform simulation from $g(s_t|k,y_t)$ which is given by,

$$g(s_t|k,y_t) = \frac{g(s_t,k|y_k)}{g(k|y_t)} \propto \frac{f(y_t|\mu_t^k)f(s_t|s_{t-1}^k)W_{t-1}^k}{f(y_t|\mu_t^k)W_{t-1}^k} = f(s_t|s_{t-1}^k) \tag{1.2.33}$$

As a result, random draws $\{s_t^i, k^i\}_{i=1}^M$ where $N < M$, are obtained from $g(s_t, k|y_t)$. The final step is to perform resampling. By equation 1.2.29, the resampling weight, also known as second stage weight, denoted as π_i is given by:

$$\begin{aligned} u_i &= \frac{f(y_t|s_t^i)f(s_t^i|s_{t-1}^{k^i})W_{t-1}^{k^i}}{g(s_t^i, k^i|y_t)} = \frac{f(y_t|s_t^i)}{f(y_t|\mu_t^{k^i})} \\ \pi_i &= u_i / \sum_{i=1}^N u_i \end{aligned} \tag{1.2.34}$$

The resampling procedure is implemented based on the probability mass $\{\pi_i, s_t^i\}_{i=1}^M$ and the resulting particles $\{\frac{1}{N}, \bar{s}_t^i\}_{i=1}^N$ will be passed into next time period.

The conditional likelihood $f(y_t|y_{0:t-1})$ is given by:

$$\begin{aligned} f(y_t|y_{0:t-1}) &= \int f(y_t|s_t)f(s_t|s_{t-1}^k)W_{t-1}^k ds_t dk \\ &= \int \frac{f(y_t|s_t)f(s_t|s_{t-1}^k)W_{t-1}^k}{g(s_t, k|y_t)} g(s_t, k|y_t) ds_t dk \\ &= \int \frac{f(y_t|s_t)}{f(y_t|\mu_t^k)} g(s_t, k|y_t) ds_t dk \end{aligned} \tag{1.2.35}$$

$$\hat{f}(y_t|y_{0:t-1}) = \frac{1}{N}\sum_{i=1}^N \frac{f(y_t|s_t^i)}{f(y_t|\mu_t^{k^i})} = \frac{1}{N}\sum_{i=1}^N u_i \tag{1.2.36}$$

As we can see, the variance of the likelihood estimator is based on the ratio $\frac{f(y_t|s_t^i)}{f(y_t|\mu_t^{k^i})}$. Generally speaking, we tend to make the selection of the auxiliary sampler such that the variance of this ratio can be as small as possible and the ultimate gold is to achieve the variance minimization. However, in the whole family of Particle Filters, there is no such algorithm that can achieve this ultimate goal. Moreover, due to the random sampling, the likelihood values approximated by Particle Filters are not continuous leading to the convergence problem on Maximum Likelihood Estimation (MLE). To solve these problems, the Efficient Importance Sampling (EIS) procedures are developed in which both variance minimization and continuous approximation of likelihood estimate can be achieved.

1.3 Efficient Importantce Sampling (EIS)

While Particle Filter algorithms are very simple and easy to implement, its way to approximate filtering density $f(s_t|y_{0:t})$ greatly restricts its performance. In Particle Filters, the approximation of $f(s_t|y_{0:t})$ relies on particles that can no longer be changed since its generation. This limits the performance of PF algorithm to be only conditional optimality, that is, the optimization relies on the fixed support. However, our ultimate goal is to pursue unconditional optimality and this challenge has already been conquered by Efficient Importance Sampling methodology.

1.3.1 The Original EIS

The EIS algorithm is firstly proposed by Richard and Zhang [34] in 2007. The key component of this algorithm consists of the search for a sampler such that the variance of likelihood estimate can be minimized and this optimization process can be implemented by applying the least square method in the modelling of linear regression. The accomplishment on variance minimization makes EIS becomes highly efficient method to numerically evaluate the associated integrals. Compared with Particle Filters, EIS can achieve the maximal performance on likelihood approximation with least particle draws.

Essentially speaking, EIS is originally designed for the numerical evaluation of integral which does not have analytical solution. Suppose we have an integral of the form:

$$g(\theta) = \int f(x;\theta)dx \tag{1.3.1}$$

Here, x can be a vector leading to multiple integral evaluation problem and θ denotes the fixed parameters. To numerically evaluate this integral, we need to find an appropriate sampler, denoted as $m(x|a)$, and the above equation can be rewritten as,

$$g(\theta) = \int \frac{f(x;\theta)}{m(x|a)} m(x|a)dx \tag{1.3.2}$$

$$w(x;\theta,a) = \frac{f(x;\theta)}{m(x|a)} \tag{1.3.3}$$

Here, $w(x;\theta,a)$ denotes unnormalized weights and the estimator of $g(\theta)$ is given by,

$$\hat{g}(\theta) = \frac{1}{N}\sum_{i=1}^{N} w(x_i;\theta,a) \tag{1.3.4}$$

The task of EIS procedure is firstly to select an appropriate family of distribution $M = \{m(x|a) : a \in \mathbb{A}\}$, then to find an optimal value a so that the variance of the weight can be minimized.

In general, the class of distribution for the EIS sampler $m(x|a)$ is selected from the exponential family. This is due to the fact that a natural parametrization of this sampler density can be performed resulting in a linear log kernel which makes it possible to directly apply least square method to achieve the optimization. The natural parametrization and log kernel of the exponential sampler $m(x|a)$ is given by

$$m(x|a) = \frac{k(x;a)}{\chi(a)} = \frac{b(x)\cdot exp(a^{'}t(x))}{\chi(a)} \tag{1.3.5}$$

$$lnk(x;a) = ln(b(x)) + a^{'}\cdot t(x) \tag{1.3.6}$$

Here, $k(x;a)$ is the density kernel and $\chi(a)$ is the integrating constant which satisfies

$$\chi(a) = \int k(x;a)dx \tag{1.3.7}$$

Usually, we assume that this integrating constant is known analytically.

In equation 1.3.6, the term $ln(b(x))$ is the intercept part and $a^{'} \cdot t(x)$ represents regression parameters and explanatory variables. To be specific, the EIS baseline algorithm can be implemented as follows,

---

EIS baseline Algorithm

- At initial step k=0
  - Find an appropriate initial sampler $q_0(x)$
  - Draw particles $\{x_i^0\}_{i=1}^S$ from this initial sampler
- At step k=1
  - With linear regression least square method, solve the following minimization problem,

$$(\hat{a}_1, \hat{c}_1) = \underset{a \in A, c \in \mathbb{R}}{ArgMin} \frac{1}{S} \sum_{i=1}^{S} \left[lnf(x_i^0; \theta) - c - lnk(x_i^0; a) \right]^2 \tag{1.3.8}$$

    Here, $lnf(x_i^0; \theta)$ denotes the dependent variable, c is the intercept and $lnk(x_i^0; a)$ can be factorized into linear regression equation wherein optimal parameters $\hat{a}_1$ will be found.
  - Draw particles $\{x_i^1\}_{i=1}^S$ from step k=1 EIS sampler $m(x|\hat{a}_1)$ and pass them into next step.
- Repeat the process in step k=1 until the convergence of $\{\hat{a}_k\}_{k=1}$ and denote the convergent parameter as $\hat{a}$

For the problem of initial sampler selection, in general, there exists a factorization of the integrand $f(x; \theta)$ in equation 1.3.1 with the form

$$f(x; \theta) = g(x; \theta) \cdot p(x|\theta) \tag{1.3.9}$$

where the resulting density function $p(x|\theta)$ is often regarded as the initial sampler $q_0(x)$ in EIS baseline algorithm, known as local sampler. Moreover, it should be noted that the convergence observed is based on the sequence of least square estimates $\{\hat{a}_k\}_{k=1}$ and the failure to converge can be remedied by either re-selecting initial sampler or extending the family of EIS sampler.

While the EIS baseline algorithm is very efficient, it can only be used to numerically evaluate low-dimensional integrals. This limitation makes it impossible in evaluating likelihood value in state space model where the integrals are usually in very high-dimensional form. One way to overcome this challenge is based on the feasibility where the integrand can be decomposed into the product of a sequence of conditional densities. This decomposition can achieve problem transformation from high-dimensional integral evaluation to a sequence of low-dimensional optimization. This methodology is referred to as Sequential EIS.

1.3.2 Sequential EIS with Backward-iteration

The Sequential EIS of Backward-iteration is developed by Richard and Zhang [34] in 2007. It is an extension of EIS baseline algorithm designed for the high-dimensional problem of likelihood evaluation on State Space Model. The feasibility of this algorithm relies on the following decomposition of likelihood integrand,

$$\begin{aligned} f(y_{0:t}) &= \int f(y_{0:t}, s_{0:t}) ds_{0:t} \\ &= \int f(s_0) \prod_{i=1}^{t} f(s_i|s_{i-1}) \prod_{i=0}^{t} f(y_i|s_i) ds_{0:t} \\ &= \int \prod_{i=0}^{t} \varphi_i(s_i, s_{i-1}, y_i) ds_{0:t} \end{aligned} \tag{1.3.10}$$

where $\varphi_i(s_i, s_{i-1}, y_i) = f(s_i|s_{i-1})f(y_i|s_i), \forall i = 1, 2, .., t$ and $\varphi_0(s_0, y_0) = f(s_0)f(y_0|s_0)$. The sampler in this Sequential EIS algorithm is denoted as $m(s_{0:t}|a_{0:t})$ and it has the following sequential structure

$$m(s_{0:t}|a_{0:t}) = m_0(s_0|a_0) \prod_{i=1}^{t} m_i(s_i|s_{i-1}, a_i) \tag{1.3.11}$$

$$m_i(s_i|s_{i-1}, a_i) = \frac{k_i(s_i, s_{i-1}; a_i)}{\chi_i(s_{i-1}; a_i)} \tag{1.3.12}$$

where $k_i(s_i, s_{i-1}; a_i)$ is the kernel density and $\chi_i(s_{i-1}; a_i)$ denotes the integrating constant. With this structure, equation 1.3.10 can be further rewritten as,

$$\begin{aligned} f(y_{0:t}) &= \int \frac{f(y_{0:t}, s_{0:t})}{m(s_{0:t}|a_{0:t})} m(s_{0:t}|a_{0:t}) ds_{0:t} \\ &= \int \prod_{i=0}^{t} \frac{\varphi_i(s_i, s_{i-1}, y_i)}{m_i(s_i|s_{i-1}, a_i)} m_i(s_i|s_{i-1}, a_i) ds_{0:t} \\ &= \chi_0(a_0) \cdot \int \prod_{i=1}^{t} \left[\frac{\varphi_i(s_i, s_{i-1}, y_i) \cdot \chi_{i+1}(s_i; a_{i+1})}{k_i(s_i, s_{i-1}; a_i)} \right] m_i(s_i|s_{i-1}, a_i) ds_{0:t} \end{aligned} \tag{1.3.13}$$

$$R_i(s_i, s_{i-1}, y_i; a_{i+1}, a_i) = \frac{\varphi_i(s_i, s_{i-1}, y_i) \cdot \chi_{i+1}(s_i; a_{i+1})}{k_i(s_i, s_{i-1}; a_i)} \tag{1.3.14}$$

where $\chi_t(s_t; a_{t+1}) \equiv 1$. It is clearly unfeasible to directly perform joint optimization for this high-dimensional integral. Instead, a backward iterative sequence of individual EIS optimizations is constructed in which the task is to find the optimal value a_i such that the variance of $R_i(s_i, s_{i-1}, y_i; a_{i+1}, a_i)$ can be minimized. Thus, the highly complicated joint optimization problem can be decomposed into a sequence of simple individual EIS procedure in which the target integral is,

$$\int R_i(s_i, s_{i-1}, y_i; a_{i+1}, a_i) \cdot m_i(s_i|s_{i-1}, a_i) ds_i \tag{1.3.15}$$

It should be noted that the individual EIS optimization starts from the last time step and the optimal parameter estimated by least square method will be passed into previous time step to take participate in the calculation of dependent variable in EIS linear regression. This backward iterative procedure will continue until reaching the most beginning time period. On the other hand, instead of the ratio $\frac{\varphi_i(s_i,s_{i-1},y_i)}{m_i(s_i|s_{i-1},a_i)}$, the variance minimization is performed for $R_i(s_i, s_{i-1}, y_i; a_{i+1}, a_i)$. This is due to the fact that the dependent variable in EIS linear regression is $ln\left[\varphi_i(s_i, s_{i-1}, y_i) \cdot \chi_i(s_{i-1}; a_i)\right]$ which relies on the unknown parameters a_i if we employ the ratio $\frac{\varphi_i(s_i,s_{i-1},y_i)}{m_i(s_i|s_{i-1},a_i)}$ for the variance minimization.

To be specific, this Backward Iterative Sequential EIS can be implemented in the following detailed procedure,

---

Sequential EIS of Backward-iteration

- At initial step k=0
  - Select an appropriate initial sampler $q_0(s_{0:t})$
  - Draw particles $\{s^j_{0:t}\}_{j=1}^S$ from this initial sampler
- At step k=1, for i in t:0 (start from the last time period)
  - With linear regression least square method, solve the following minimization problem,

$$(\hat{a}_i^k, \hat{c}_i^k) = \underset{a\in A, c\in\mathbb{R}}{ArgMin} \sum_{j=1}^{S} \left\{ln\left[\varphi_i(s_i^j, s_{i-1}^j, y_i) \cdot \chi_{i+1}(s_i^j; \hat{a}_{i+1}^k)\right] - c - lnk_i(s_i^j, s_{i-1}^j; a)\right\}^2 \tag{1.3.16}$$

    Here, $ln\left[\varphi_i(s_i^j, s_{i-1}^j, y_i) \cdot \chi_{i+1}(s_i^j; \hat{a}_{i+1}^k)\right]$ denotes the dependent variable, c is the intercept and $lnk_i(s_i^j, s_{i-1}^j; a)$ can be factorized into linear regression equation wherein optimal parameter $\hat{a}_i^k$ will be estimated and passed into backward time period i-1 for the calculation of $\chi_i(s_{i-1}^j; \hat{a}_i^k)$. As a result, the step k tentative EIS sampler $m(s_{0:t}|\hat{a}_{0:t}^k)$ will be acquired.
  - Draw particles $\{s^j_{0:t}\}_{j=1}^S$ from $m(s_{0:t}|\hat{a}_{0:t}^k)$ and pass them into next step.
- Repeat above process until the convergence of $\{\hat{a}_{0:t}^k\}_{k=1}$

By the equation 1.3.13, the likelihood evaluation by Backward iterative Sequential EIS is given by,

$$\begin{aligned}\hat{f}(y_{0:t}) &= \frac{1}{N}\sum_{i=1}^{N}\frac{f(y_{0:t}, s_{0:t}^i)}{m(s_{0:t}^i|\hat{a}_{0:t})} \\ &= \frac{1}{N}\sum_{i=1}^{N}\left\{\prod_{j=0}^{t}\frac{\varphi_j(s_j^i, s_{j-1}^i, y_j)}{m_j(s_j^i|s_{j-1}^i, \hat{a}_j)}\right\}\end{aligned} \tag{1.3.17}$$

One thing we need to be aware of is that the convergence is based on the parameters $\{\hat{a}_{0:t}^k\}$ for the whole time periods. This may lead to the convergence problem when we have a very long time span. To remedy this problem, some researchers have proposed another version of Sequential EIS in which the individual optimizations are completely independent with each other.

1.3.3 Sequential EIS with Continuous Approximation

The Sequential EIS with continuous approximation of filtering density is proposed by David and Roman [5] in 2013. In comparison with the Backward-iterative EIS introduced in previous section, this EIS procedure constructs a sequence of independent joint samplers for each individual optimization by employing the continuous approximations for the filtering densities. As a result, the EIS implementation at each time step is completely independent with each other and this independence can help us bypass the convergence challenge in Backward-iterative EIS. On the other hand, the introduction of continuous approximations to the filtering densities also makes it possible to perform the evaluation of the filtering moments given by 1.2.9 without relying on fixed particles which are employed by Particle Filter algorithms, leading to a more efficient moments approximation.

Instead of directly evaluating the likelihood for the whole time span, in this EIS procedure, the joint likelihood can be factorized into a product of a sequence of conditional densities, that is,

$$f(y_{0:t}) = \prod_{i=0}^{t} f(y_i|y_{0:i-1}) \tag{1.3.18}$$

where $f(y_i|y_{0:i-1}) = f(y_0)$ for $i = 0$. And the evaluation of conditional likelihood on the period i can be performed by the following double integral,

$$\begin{aligned} &f(y_i|y_{0:i-1}) \\ &\qquad = \int\int \frac{\varphi_i(s_i, s_{i-1})}{g_i(s_i, s_{i-1}|a_i)} g_i(s_i, s_{i-1})|a_i) ds_{i-1} ds_i \end{aligned} \tag{1.3.19}$$

$$\varphi_i(s_i, s_{i-1}) = f(y_i|s_i) f(s_i|s_{i-1}) \hat{f}(s_{i-1}|y_{0:i-1}) \tag{1.3.20}$$

$$w_i(s_i, s_{i-1}) = \frac{\varphi_i(s_i, s_{i-1})}{g_i(s_i, s_{i-1}|a_i)} \tag{1.3.21}$$

where the continuous approximation of filtering density $f(s_{i-1}|y_{0:i-1})$ will be given so that the EIS procedure can be applied to this double integral for each time step. As a result, a joint sampler $g_i(s_i, s_{i-1}|a_i)$ from exponential family will be constructed and the task is to find the optimal parameter a_i such that the variance of weight $w_i(s_i, s_{i-1})$ can be minimized. The resulting approximation of conditional likelihood is given by,

$$\hat{f}(y_i|y_{0:i-1}) = \frac{1}{N} \sum_{j=1}^{N} w_i(s_i^j, s_{i-1}^j) \tag{1.3.22}$$

where $\{s_i^j, s_{i-1}^j\}_{j=1}^N$ denotes N i.i.d. draws from $g_i(s_i, s_{i-1})|a_i)$. The continuous approxima-

tion of filtering density at time period i can be derived by,

$$\begin{aligned} f(s_i|y_{0:i}) &= \frac{\int f(y_i|s_i)f(s_i|s_{i-1})\hat{f}(s_{i-1}|y_{0:i-1})ds_{i-1}}{f(y_i|y_{0:i-1})} \\ &= \frac{\int w_i(s_i,s_{i-1})g_i(s_i,s_{i-1}|a_i)ds_{i-1}}{f(y_i|y_{0:i-1})} \\ &= g_i(s_i)\frac{\int w_i(s_i,s_{i-1})g_i(s_{i-1}|s_i)ds_{i-1}}{f(y_i|y_{0:i-1})} \end{aligned} \tag{1.3.23}$$

where $g_i(s_i, s_{i-1}|a_i) = g_i(s_i)g_i(s_{i-1}|s_i)$ and $f(y_i|y_{0:i-1})$ can be approximated by 1.3.22. Moreover, the integral in the numerator of equation 1.3.23 can be evaluated by,

$$\bar{w}_i(s_i) = \frac{1}{N}\sum_{j=1}^{N} w_i(s_i, s_{i-1}^j(s_i)) \tag{1.3.24}$$

where $\{s_{i-1}^j(s_i)\}_{j=1}^N$ denotes i.i.d draws from $g(s_{i-1}|s_i)$. As a result, $f(s_i|y_{0:i})$ can be approximated by,

$$\hat{f}(s_i|y_{0:i}) = g_i(s_i)\tilde{w}_i(s_i) \tag{1.3.25}$$

$$\tilde{w}_i(s_i) = \frac{\bar{w}_i(s_i)}{\hat{f}(y_i|y_{0:i-1})} = \frac{\sum_{j=1}^N w_i(s_i, s_{i-1}^j(s_i))}{\sum_{j=1}^N w_i(s_i^j, s_{i-1}^j)} \tag{1.3.26}$$

The success of this EIS procedure mainly owes to the feasibility of the continuous approximation to the filtering density and this makes it possible to decompose a joint EIS problem into a sequence of completely independent individual optimizations. The detailed implementation is demonstrated as follows,

---

Sequential EIS with Continuous Approximation of Filtering Density

- At time period i, suppose we inherit a continuous approximation of filtering density $\hat{f}(s_{i-1}|y_{0:i-1})$
- At initial step L=0
  - Select an appropriate initial sampler $q_0(s_i, s_{i-1})$
  - Draw particles $\{s_i^j, s_{i-1}^j\}_{j=1}^S$ from this initial sampler
- At step L=1
  - With linear regression least square method, solve the following minimization problem,

$$(\hat{a}_i^L, \hat{c}_i^L) = \underset{a\in A, c\in\mathbb{R}}{ArgMin} \sum_{j=1}^{S} \left\{ ln\left[\varphi_i(s_i^j, s_{i-1}^j)\right] - c - lnk_i(s_i^j, s_{i-1}^j; a)\right\}^2 \tag{1.3.27}$$

Here, $ln\left[\varphi_i(s_i^j, s_{i-1}^j)\right]$ denotes the dependent variable, c is the intercept and $lnk_i(s_i^j, s_{i-1}^j; a)$ can be factorized into a linear regression equation wherein optimal parameter $\hat{a}_i^L$ will be estimated. As a result, a tentative EIS sampler $g_i(s_i, s_{i-1}|\hat{a}_i^L)$ at step L on time period i will be acquired.

  - Draw particles $\{s_i^j, s_{i-1}^j\}_{j=1}^S$ from $g_i(s_i, s_{i-1}|\hat{a}_i^L)$ and pass them into next step.

- Repeat above process until the convergence of $\{\hat{a}_i^L\}_{L=1}$ and we denote the convergent value as $\hat{a}_i$. Therefore, the EIS sampler $g_i(s_i, s_{i-1}|\hat{a}_i)$ at time period i is acquired.

- Follow the derivation process from 1.3.23 to 1.3.26, a continuous approximation of the filtering density $\hat{f}(s_i|y_{0:i})$ will be acquired and further passed into next time period.

Here, $k_i(s_i, s_{i-1}; a)$ is the density kernel of $g_i(s_i, s_{i-1}|a)$ satisfying,

$$g_i(s_i, s_{i-1}|a) = \frac{k_i(s_i, s_{i-1}; a)}{\chi(a)} \tag{1.3.28}$$

where $\chi(a)$ is the integrating constant which can be unknown. One point we need to be aware of is that the convergence relies on parameters $\{\hat{a}_i^L\}_{L=1}$ for each individual time period instead of the whole time span. This property help us bypass the convergence challenge met in Backward-iterative EIS and the continuous approximations of the filtering densities played a vital role in constructing these independent EIS procedures. For the selection of initial sampler $q_0(s_i, s_{i-1})$, the most commonly way is to make draws from the local approximation $f(s_i|s_{i-1})\hat{f}(s_{i-1}|y_{0:i-1})$ shown in 1.3.19.

CHAPTER II

LIKELIHOOD EVALUATION FOR STATE SPACE MODELS

2.1 Introduction

State Space Model (SSM), also known as Hidden Markov Model (HMM) or Latent Process Model (LPM), is a very famous statistical Markov model for sequential data analysis and can be used to depict the evolution of observable data relying on some hidden unobservable factors, also referred to as state. Lots of applications of State Space Model have already been found in the fields such as, statistics, engineering, biology, pattern recognition and reinforcement learning, see [40, 33, 30, 23]. Formally, a State Space Model consists of two discrete-time stochastic processes: measurement process $\{y_t\}_{t\geq 0}$ and hidden state process $\{s_t\}_{t\geq 0}$. The measurement process delivers the observable information depending on state process and its measurement density satisfies the following property,

$$g_\theta(y_t|s_t) = g_\theta(y_t|s_{0:t}, y_{0:t-1}) \tag{2.1.1}$$

The hidden state process is an unobservable transition process with an initial density $\mu(s_0)$ and its transition density is satisfying the Markov property, that is

$$f_\theta(s_t|s_{t-1}) = f_\theta(s_t|s_{0:t-1}) \tag{2.1.2}$$

where θ is the parameter vector assumed to be known in this chapter for the purpose of likelihood evaluation.

Our interests of inference in State Space Model lay on the likelihood evaluation and parameter estimation. When the model is linear and Gaussian distributed, the analytical solution of likelihood function can be accessible by directly applying Kalman Filer algorthm [38, 4]. However, under the non-linear or non-Gaussian context, we have to rely on some numerical procedures to perform the likelihood approximation.

Particle Filter (PF) Algorithms are the most commonly used procedures to numerically evaluate the likelihood function for Non-linear State Space Model [31]. Its popularity is based on the flexibility and simplicity. The earliest developed PF algorithm is known as the Sequential Importance Sampling (SIS) [9] which is intuitive and very easy to implement. However, SIS suffers from the weight degeneracy problem which refers to such a fact that there will exist only few particles with significant weights as the time goes on[2]. To remedy this problem, researchers further proposed Sampling Importance Resampling (SIR) algorithm in which a resampling step is embedded into the SIS procedure [25]. An important advantage of resampling is that it allows us to have a high probability to remove particles of low weight values. As a result, we can focus our computational efforts on regions of high

probability mass. However, the overuse of resampling leads to another challenge, known as sample impoverishment: particles with relatively high weights end up being duplicated too many times during the implementation of resampling, resulting in a loss of diversity among these samples. Some researchers impute this challenge to the absence of information conveyed by observable data y_t from current time period during the construction of the sampler. To conquer this challenge, particle filters with adaption have been proposed and the most popular one is known as the Auxiliary Particle Filter (APF) in which the sampler construction process relies on the incorporation of y_t information [32, 17].

Although these PF algorithms are very simple and flexible, their performance in likelihood approximation are greatly restricted by the quality of sampler. Generally, we tend to select a sampler such that the variance of weight can be controlled as small as possible and the ultimate goal is to achieve the variance minimization. However, in real application, there is no such procedure in Particle Filters that can realize this ultimate goal. If the sampler selected is not appropriate, we may have to draw more particles to reach a specific performance. Thus, Particle Filter is not an efficient algorithm as we expected. On the other hand, due to the resampling step, the continuity of likelihood evaluation can not be achieved by simply using Common Random Number technique leading to the infeasibility of the convergence in Maximum Likelihood Estimation (MLE). Although researchers have recently proposed some extensions of Particle Filters for continuous likelihood evaluation [26], without achieving variance minimization, the MLE implemented by Particle Filters is doomed to be inefficient.

To conquer these challenges, a numerical procedure known as Efficient Importance Sampling (EIS) has been proposed. This EIS methodology is originally developed by Richard and Zhang [34] in 2007. Its baseline algorithm is designed for the numerical evaluation on the integrals without tractable analytical solutions. The variance minimization procedure is introduced and its implementation relies on the Least Square method of linear regression. The key thought of EIS on numerically evaluating likelihood function for the Non-linear State Space Model is based on such a fact that the infeasible EIS implementation on high-dimensional integrals can be decomposed into a sequence of simple individual optimization problems. However, the EIS procedure developed by Richard and Zhang suffers from the convergence problem which can be imputed to the dependence of these individual EIS samplers. To remedy this problem, an extension of EIS is proposed by David and Roman [5] in 2013. The cornerstone of this EIS procedure is based on the introduction of continuous approximations to the filtering densities leading to the independence of these individual EIS implementations. On the other hand, the continuity (smoothness) of EIS likelihood estimation can be achieved by using the technique known as Common Random Number (CRN) [8] where all random particles in EIS procedure are obtained by transformation of a single draw from a canonical distribution which does not depend on any parameters, i.e. the transformation of standard normal distribution.

While the EIS method proposed by David and Roman does not suffer from the convergence problem, another challenge arises from such a fact that the computational complexity of these continuously approximated filtering densities is $\mathcal{O}(N^t)$ leading to an infeasible calculation when t takes very large value. One way to partially remedy this problem is known as Constant Weight (CW) approximation proposed in [5] by David and Roman. However, this method may introduce an non-negligible bias into corresponding likelihood estimation for the State Space Model with extremely complicated structure. To further reduce this

bias and meanwhile make the computation still feasible, in this chapter, we proposed an improved method to make a more precise calculation on these continuously approximated filtering densities with the computational complexity be reduced to only $\mathcal{O}(N)$. A simulation study is given and the result showed that our proposed computational method outperforms the Constant-Weight approximation in likelihood approximation.

2.2 Numerical Evaluation of Likelihood for State Space Models

2.2.1 Particle Filters and EIS

As we know, the likelihood evaluation is the most important prerequisite for parameter estimation by MLE. Therefore, in this chapter, we will focus our concentration on this prerequisite and explore the efficiency of different numerical methods. Specifically speaking, in State Space Model, the likelihood function can be evaluated by the following joint integral,

$$f(y_{0:T}) = \int ... \int f(y_{0:T}, s_{0:T}) ds_{0:T} \tag{2.2.1}$$

When the State Space Model is linear and Gaussian distributed, this integral can be directly solved by applying Kalman Filter algorithm. However, in most real applications, the models we met are usually Non-linear or Non-Gaussian distributed and the analytical solution for above joint integral is not tractable. Therefore, we have to resort to the numerical methods for the likelihood approximation. The most commonly used methods are known as Particle Filters (PF) and Efficient Importance Sampling (EIS). Instead of directly evaluating the high-dimensional integral in 2.2.1, both of these two methods factorize this joint likelihood into a product of conditional likelihood densities, that is,

$$f(y_{0:T}) = \prod_{t=0}^{T} f(y_t|y_{0:t-1}) \tag{2.2.2}$$

For the PF algorithms, this conditional likelihood can be evaluated by,

$$f(y_t|y_{0:t-1}) = \int \frac{f(y_t|s_t)f(s_t|s_{t-1})}{q_t(s_t|s_{0:t-1})} q_t(s_t|s_{0:t-1}) f(s_{0:t-1}|y_{0:t-1}) ds_{0:t} \tag{2.2.3}$$

and can be estimated by,

$$\hat{f}(y_t|y_{0:t-1}) = \frac{1}{N}\sum_{i=1}^{N} \frac{f(y_t|s_t^i)f(s_t^i|\bar{s}_{t-1}^i)}{q_t(s_t^i|\bar{s}_{0:t-1}^i)} = \frac{1}{N}\sum_{i=1}^{N} \pi_t^i \tag{2.2.4}$$

where $\{\bar{s}_{0:t-1}^i\}_{i=1}^N$ is the i.i.d draws obtained from $f(s_{0:t-1}|y_{0:t-1})$ by using resampling and $\{s_t^i\}_{i=1}^N$ is the i.i.d draws from $q_t(s_t|s_{0:t-1})$. For the EIS algorithm, the conditional likelihood can be evaluated by,

$$f(y_t|y_{0:t-1}) = \int\int \frac{f(y_t|s_t)f(s_t|s_{t-1})f(s_{t-1}|y_{0:t-1})}{g_t(s_t, s_{t-1})} g_t(s_t, s_{t-1}) ds t ds_{t-1} \tag{2.2.5}$$

and can be estimated by,

$$\hat{f}(y_t|y_{0:t-1}) = \frac{1}{N}\sum_{i=1}^{N}\frac{f(y_t|s_t^i)f(s_t^i|s_{t-1}^i)f(s_{t-1}^i|y_{0:t-1}^i)}{g_t(s_t^i, s_{t-1}^i)} = \frac{1}{N}\sum_{i=1}^{N} w_t^i \tag{2.2.6}$$

where $\{s_t^i, s_{t-1}^i\}_{i=1}^N$ is the i.i.d draws from $g_t(s_t, s_{t-1})$. In comparison with Particle Filters, the superiority of EIS lies in the extra step of variance minimization on the weight w_t. The goal of EIS principle is to find a sampler $g(s_{0:T})$ such that the variance of w_t can be minimized leading to a likelihood estimate with minimized variance. This is referred to as global optimization and we can achieve maximal performance on likelihood approximation with minimal sample draws.

The most widely recognized method of Particle Filters is known as Bootstrap Particle Filter which will be further used for our simulation study. Its popularity is mainly attributed to the simplicity of implementation. To be specific, the transition density $f(s_t|s_{t-1})$ is selected as the sampler $q_t(s_t|s_{0:t-1})$ and we can see the weight function π_t is just equal to the measurement density $f(y_t|s_t)$. The detailed implementation process is as follows,

---

Bootstrap Particle Filter

- At time t=0
  - Draw particles $\{s_0^i\}_{i=1}^N$ from $\mu(s_0)$
  - Compute the unnormalized weight $\pi_0^i = g(y_0|s_0^i)$ and normalized weight $W_0^i \propto \pi_0^i$
  - Resample $\{W_0^i, s_0^i\}_{i=1}^N$ to obtain N equally-weighted particles $\{\frac{1}{N}, \bar{s}_0^i\}_{i=1}^N$
- At time t$\geq$ 1, suppose we inherit $\{\frac{1}{N}, \bar{s}_{t-1}^i\}_{i=1}^N$
  - Draw particles $\{s_t^i\}_{i=1}^N$ from $f(s_t|\bar{s}_{t-1}^i)$
  - Compute weight $\pi_t^i = f(y_t|s_t^i)$ and $W_t^i \propto \pi_t^i$
  - Resample $\{W_t^i, s_t^i\}_{i=1}^N$ to obtain N equally-weighted particles $\{\frac{1}{N}, \bar{s}_t^i\}_{i=1}^N$

There exists a total of two versions of EIS. The first version is proposed by Richard and Zhang in 2007. Its implementation of variance minimization process starts from the last time period end in the most beginning time point, known as Backward-iteration. However, when we have a very large time length T, this EIS may suffer from the convergence problem during the optimization process. Therefore, we only focus on the second version of EIS proposed by David and Roman in 2013 in which the convergence problem is solved by implementing the EIS procedure independently for each time period. This independence essentially relies upon the continuous approximations of filtering densities which can transform this infeasible high-dimensional global optimization problem into a sequence of individual EIS implementations.

2.2.2 Continuous Approximation of Filtering Density

The joint likelihood function $f(y_{0:T})$ in State Space Model can be factorized into the following product of a sequence of conditional densities,

$$f(y_{0:T}) = \prod_{t=0}^{T} f(y_t|y_{0:t-1}) \tag{2.2.7}$$

And the conditional likelihood $f(y_t|y_{0:t-1})$ can be evaluated by,

$$\begin{aligned} f(y_t|y_{0:t-1}) &= \int\int f(y_t|s_t)f(s_t|s_{t-1})\hat{f}(s_{t-1}|y_{0:t-1})ds_{t-1}ds_t \\ &= \int\int \frac{f(y_t|s_t)f(s_t|s_{t-1})\hat{f}(s_{t-1}|y_{0:t-1})}{g_t(s_t, s_{t-1}|a_t)} g_t(s_t, s_{t-1}|a_t)ds_{t-1}ds_t \end{aligned} \tag{2.2.8}$$

$$w_t(s_t, s_{t-1}) = \frac{f(y_t|s_t)f(s_t|s_{t-1})\hat{f}(s_{t-1}|y_{0:t-1})}{g_t(s_t, s_{t-1}|a_t)} \tag{2.2.9}$$

where $\hat{f}(s_{t-1}|y_{0:t-1})$ is the continuous approximation of filtering density inherited from the last time period. After applying the EIS procedure, an optimal sampler $g_t(s_t, s_{t-1}|\hat{a}_t)$ will be found such that the variance of weight function $w_t(s_t, s_{t-1})$ can be minimized and this conditional likelihood will be estimated by,

$$\hat{f}(y_t|y_{0:t-1}) = \frac{1}{N}\sum_{i=1}^{N} w_t(s_t^i, s_{t-1}^i) = \frac{1}{N}\sum_{i=1}^{N} \frac{f(y_t|s_t^i)f(s_t^i|s_{t-1}^i)\hat{f}(s_{t-1}^i|y_{0:t-1})}{g_t(s_t^i, s_{t-1}^i|\hat{a}_t)} \tag{2.2.10}$$

Next, we need to find a way to continuously approximate the filtering density in current time period and pass it into next time period to start a new iteration. By the Bayesian inference, the filtering density at current time point can be expressed as,

$$\begin{aligned} f(s_t|y_{0:t}) &= \frac{\int f(y_t|s_t)f(s_t|s_{t-1})\hat{f}(s_{t-1}|y_{0:t-1})ds_{t-1}}{f(y_t|y_{0:t-1})} \\ &= \frac{\int w_t(s_t, s_{t-1})g_t(s_t, s_{t-1}|\hat{a}_t)ds_{t-1}}{f(y_t|y_{0:t-1})} \\ &= g_t(s_t)\frac{\int w_t(s_t, s_{t-1})g_t(s_{t-1}|s_t)ds_{t-1}}{f(y_t|y_{0:t-1})} \\ &= g_t(s_t)\frac{H(s_t)}{f(y_t|y_{0:t-1})} \end{aligned} \tag{2.2.11}$$

$$H(s_t) = \int w_t(s_t, s_{t-1})g_t(s_{t-1}|s_t)ds_{t-1} \tag{2.2.12}$$

where $g_t(s_t, s_{t-1}|\hat{a}_t) = g_t(s_t)g_t(s_{t-1}|s_t)$. The denominator is the conditional likelihood which can be just approximated by 2.2.10 and the numerator $H(s_t)$ can be evaluated by,

$$\hat{H}(s_t) = \frac{1}{N}\sum_{i=1}^{N} w_t(s_t, s_{t-1}^i(s_t)) = \frac{1}{N}\sum_{i=1}^{N} \frac{f(y_t|s_t)f(s_t|s_{t-1}^i(s_t))\hat{f}(s_{t-1}^i(s_t)|y_{0:t-1})}{g_t(s_t, s_{t-1}^i(s_t)|a_t)} \tag{2.2.13}$$

where $\{s_{t-1}^i(s_t)\}_{i=1}^N$ denotes i.i.d draws from $g(s_{t-1}|s_t)$. As a result, $f(s_t|y_{0:t})$ can be continuously approximated by,

$$\hat{f}(s_t|y_{0:t}) = g_t(s_t)\frac{\hat{H}(s_t)}{\hat{f}(y_t|y_{0:t-1})} = g_t(s_t)\frac{\sum_{i=1}^N w_t(s_t, s_{t-1}^i(s_t))}{\sum_{i=1}^N w_t(s_t^i, s_{t-1}^i)} \tag{2.2.14}$$

As we can see, the computational complexity of above equation is $\mathcal{O}(N^t)$ and it is impossible to perform the calculation when t takes very large value. To remedy this calculation problem, David and Roman further proposed an intuitive method, known as Constant-Weight approximation.

2.2.3 Constant-Weight Approximation

The challenge on the calculation of $\hat{f}(s_t|y_{0:t})$ can be imputed to the dependence of these continuously approximated filtering densities $\{\hat{f}(s_i|y_{0:i})\}_{i=0}^t$. Specifically speaking, to compute $\hat{f}(s_t|y_{0:t})$, we have to perform calculation on $\hat{f}(s_{t-1}|y_{0:t-1})$ which relies on the evaluation of $\hat{f}(s_{t-2}|y_{0:t-2})$. As a result, given a value of s_t, we have to perform the calculation for all the filtering densities from previous time periods leading the computational complexity to be $\mathcal{O}(N^t)$ which is infeasible.

The method proposed by David and Roman, Constant-Weight approximation, is based on such standpoint that the weight generated by EIS variance minimization process has a very small fluctuation and can be just viewed as a constant function of state variables. By applying this method, we can see that $\hat{H}(s_t) = \hat{f}(y_t|y_{0:t-1})$ leading to

$$\hat{f}(s_t|y_{0:t}) = g_t(s_t) \tag{2.2.15}$$

However, it ignores the fact that the implementation of this EIS optimization is based on the least square method leading to a great restriction such that the family of sampler can only be chosen from the exponential class. Therefore, even if the variance has already been minimized, the fluctuation of weight still can not be ignored when exponential family is not a good choice for the sampler. Moreover, this method may introduce an non-negligible bias to corresponding likelihood estimation in the State Space Model with extremely complicated structure and we need to find a way to further reduce this bias.

2.2.4 Proposed Computational Method

Here, we proposed an improved method to make a more precise calculation than Constant-Weight method. This improved method is still based on Constant-weight approximation, but instead of simply viewing weight as a constant function, we apply the CW method

on $\hat{H}(s_t)$ to break the dependence of these continuous approximations of filtering densities $\{\hat{f}(s_i|y_{0:i})\}_{i=0}^{t}$.

To be specific, with Constant-Weight principle, we can replace $\hat{f}(s_{t-1}^i(s_t)|y_{0:t-1})$ in $\hat{H}(s_t)$ by $g_{t-1}(s_{t-1}^i(s_t))$ to construct a new weight function, denoted as $\bar{w}_t(s_t, s_{t-1})$, leading to an approximation for $\hat{H}(s_t)$, denoted as $\bar{H}(s_t)$, that is

$$\bar{H}(s_t) = \frac{1}{N}\sum_{i=1}^{N} \bar{w}_t(s_t, s_{t-1}^i(s_t)) = \frac{1}{N}\sum_{i=1}^{N} \frac{f(y_t|s_t)f(s_t|s_{t-1}^i(s_t))g_{t-1}(s_{t-1}^i(s_t))}{g_t(s_t, s_{t-1}^i(s_t)|\hat{a}_t)} \tag{2.2.16}$$

Thus, starting from $t = 1$, the continuous approximation of filtering density can be calculated by,

$$\hat{f}(s_t|y_{0:t}) = g_t(s_t)\frac{\bar{H}(s_t)}{\hat{f}(y_t|y_{0:t-1})} = g_t(s_t)\frac{\sum_{i=1}^{N}\bar{w}_t(s_t, s_{t-1}^i(s_t))}{\sum_{i=1}^{N} w_t(s_t^i, s_{t-1}^i)} \tag{2.2.17}$$

$$w_t(s_t^i, s_{t-1}^i) = \frac{f(y_t|s_t^i)f(s_t^i|s_{t-1}^i)\hat{f}(s_{t-1}^i|y_{0:t-1})}{g_t(s_t^i, s_{t-1}^i|a_t)} \tag{2.2.18}$$

$$\begin{aligned}\hat{f}(s_{t-1}^i|y_{0:t-1}) &= g_{t-1}(s_{t-1})\frac{\bar{H}(s_{t-1})}{\hat{f}(y_{t-1}|y_{0:t-2})} \\ &= \frac{g_{t-1}(s_{t-1})}{\hat{f}(y_{t-1}|y_{0:t-2})}\frac{1}{N}\sum_{i=1}^{N}\frac{f(y_{t-1}|s_{t-1})f(s_{t-1}|s_{t-2}^i(s_{t-1}))g_{t-2}(s_{t-2}^i(s_{t-1}))}{g_{t-1}(s_{t-1}, s_{t-2}^i(s_{t-1})|\hat{a}_{t-1})}\end{aligned} \tag{2.2.19}$$

By applying this improved method, we can see that the weight function $w_t(s_t, s_{t-1})$ only relies on $\hat{f}(s_{t-1}|y_{0:t-1})$ leading to a feasible calculation and the computational complexity of $\hat{f}(s_t|y_{0:t})$ is reduced to $\mathcal{O}(N)$. Theoretically speaking, in comparison with the Constant-Weight approximation, this method should generate a likelihood estimate with smaller Mean-Square Error (MSE). The detailed procedure is shown as follows,

---

Proposed Computational Method for Continuous Approximation of Filtering Density

- At initial time period t=0
  - The likelihood function $f(y_0)$ can be evaluated by

$$\begin{aligned}f(y_0) &= \int f(y_0|s_0)f(s_0)ds_0 \\ &= \int \frac{f(y_0|s_0)f(s_0)}{g_0(s_0|a_0)}g_0(s_0|a_0)ds_0 \\ &= \int w_0(s_0)g_0(s_0|a_0)ds_0\end{aligned} \tag{2.2.20}$$

$$w_0(s_0) = \frac{f(y_0|s_0)f(s_0)}{g_0(s_0|a_0)} \tag{2.2.21}$$

  - After applying the EIS procedure, we will find an optimal value denoted as $\hat{a}_0$

such that the variance of $w_0(s_0)$ can be minimized and $f(y_0)$ can be estimated by

$$\hat{f}(y_0) = \frac{1}{N}\sum_{i=1}^{N} w_0(s_0^i) \tag{2.2.22}$$

where $\{s_0^i\}_{i=1}^N$ are the draws from $g_0(s_0|\hat{a}_0)$

– The filtering density $f(s_0|y_0)$ can be derived by

$$f(s_0|y_0) = \frac{f(y_0|s_0)f(s_0)}{f(y_0)} \tag{2.2.23}$$

and can be estimated by

$$\hat{f}(s_0|y_0) = \frac{f(y_0|s_0)f(s_0)}{\hat{f}(y_0)} = \frac{f(y_0|s_0)f(s_0)}{\sum_{i=1}^{N} w_0(s_0^i)}N \tag{2.2.24}$$

which is a continuous approximation with the computational complexity be just $\mathcal{O}(1)$ and will be passed into next time period.

- Start from time period t=1, the following process will be iterated

– The likelihood function $f(y_t|y_{0:t-1})$ can be evaluated by

$$\begin{aligned} f(y_t|y_{0:t-1}) &= \int\int f(y_t|s_t)f(s_t|s_{t-1})\hat{f}(s_{t-1}|y_{0:t-1})ds_0 ds_1 \\ &= \int\int \frac{f(y_t|s_t)f(s_t|s_{t-1})\hat{f}(s_{t-1}|y_{0:t-1})}{g_t(s_t, s_{t-1}|a_t)} g_t(s_t, s_{t-1}|a_t)ds_{t-1}ds_t \\ &= \int\int w_t(s_t, s_{t-1})g_t(s_t, s_{t-1}|a_t)ds_{t-1}ds_t \end{aligned} \tag{2.2.25}$$

$$w_t(s_t, s_{t-1}) = \frac{f(y_t|s_t)f(s_t|s_{t-1})\hat{f}(s_{t-1}|y_{0:t-1})}{g_t(s_t, s_{t-1}|a_t)} \tag{2.2.26}$$

– After applying the EIS procedure, we will find an optimal value denoted as $\hat{a}_t$ such that the variance of $w_t(s_t, s_{t-1})$ can be minimized and $f(y_t|y_{0:t-1})$ can be estimated by

$$\hat{f}(y_t|y_{0:t-1}) = \frac{1}{N}\sum_{i=1}^{N} w_t(s_t^i, s_{t-1}^i) \tag{2.2.27}$$

where $\{s_t^i, s_{t-1}^i\}_{i=1}^N$ are the draws from $g_t(s_t, s_{t-1}|\hat{a}_t)$

– The filtering density $f(s_t|y_{0:t})$ can be derived by 2.2.11

$$\begin{aligned} f(s_t|y_{0:t}) &= \frac{\int f(y_t|s_t)f(s_t|s_{t-1})\hat{f}(s_{t-1}|y_{0:t-1})ds_{t-1}}{f(y_t|y_{t-1})} \\ &= g_t(s_t)\frac{\int w_t(s_t, s_{t-1})g_t(s_{t-1}|s_t)ds_{t-1}}{f(y_t|y_{t-1})} \end{aligned} \tag{2.2.28}$$

and can be estimated by 2.2.17,

$$\begin{aligned}\hat{f}(s_t|y_{0:t}) &= g_t(s_t)\frac{\bar{H}(s_t)}{\hat{f}(y_t|y_{0:t-1})} \\ &= \frac{g_t(s_t)}{\hat{f}(y_t|y_{0:t-1})}\frac{1}{N}\sum_{i=1}^{N}\frac{f(y_t|s_t)f(s_t|s_{t-1}^i(s_t))g_{t-1}(s_{t-1}^i(s_t))}{g_t(s_t, s_{t-1}^i(s_t)|\hat{a}_t)}\end{aligned} \tag{2.2.29}$$

which is a continuous approximation with computational complexity be $\mathcal{O}(N)$ and will be passed into next time period for new iteration.

One important thing we need to be aware of is that the likelihood estimate by EIS is not unbiased even though minimum variance has been achieved. If we take the expectation on the EIS likelihood estimate, we can have the following derivations,

$$\begin{aligned}E\left(\hat{f}_{EIS}(y_t|y_{0:t-1})\right) &= E\left(\frac{f(y_t|s_t)f(s_t|s_{t-1})\hat{f}(s_{t-1}|y_{0:t-1})}{g_t(s_t, s_{t-1}|a_t)}\right) \\ &= \int\int \frac{f(y_t|s_t)f(s_t|s_{t-1})\hat{f}(s_{t-1}|y_{0:t-1})}{g_t(s_t, s_{t-1}|a_t)} g_t(s_t, s_{t-1}|a_t)ds_t ds_{t-1} \\ &= \int\int f(y_t|s_t)f(s_t|s_{t-1})\hat{f}(s_{t-1}|y_{0:t-1})ds_t ds_{t-1} \\ &\neq \int\int f(y_t|s_t)f(s_t|s_{t-1})f(s_{t-1}|y_{0:t-1})ds_t ds_{t-1} = f(y_t|y_{0:t-1})\end{aligned} \tag{2.2.30}$$

On the other hand, the likelihood estimate by Particle Filters is unbiased which can be derived by,

$$\begin{aligned}E\left(\hat{f}_{PF}(y_t|y_{0:t-1})\right) &= E\left(\frac{1}{N}\sum_{i=1}^{N}\frac{f(y_t|s_t^i)f(s_t^i|\bar{s}_{t-1}^i)}{q_t(s_t^i|\bar{s}_{0:t-1}^i)}\right) \\ &= E\left(\frac{f(y_t|s_t)f(s_t|s_{t-1})}{q_t(s_t|s_{0:t-1})}\right) \\ &= \int \frac{f(y_t|s_t)f(s_t|s_{t-1})}{q_t(s_t|s_{0:t-1})} q_t(s_t|s_{0:t-1})f(s_{0:t-1}|y_{0:t-1})ds_{0:t} \\ &= \int f(y_t|s_t)f(s_t|s_{t-1})f(s_{0:t-1}|y_{0:t-1})ds_{0:t} \\ &= f(y_t|y_{0:t-1})\end{aligned} \tag{2.2.31}$$

It seems that the likelihood estimate of Particle Filters is better than the one of EIS by simply considering unbiasedness. However, if we take the variability into account, the EIS demonstrates its tremendous superiority against Particle Filters. By the variance minimization in EIS procedure, we can achieve maximal efficiency gains with a very small price be paid. On the other hand, since we can never have the access to the true value of $f(s_{t-1}|y_{0:t-1})$ under the case of Non-linear State Space Model, the EIS likelihood estimate is doomed to be biased.

So, the only thing we can do to achieve good performance on likelihood approximation is to reduce this bias as small as possible. Obviously, compared with the Constant-Weight approximation, our proposed computational method can make the continuously approximated filtering density shown in 2.2.14 closer to $f(s_{t-1}|y_{0:t-1})$ leading to the bias reduction.

2.3 Simulation Study

In this section, a simulation study is given to make a comparison between Particle Filter and EIS on their efficiency of likelihood approximation. For PF algorithm, we adopt the bootstrap Particle Filter mainly for the reason of simplicity and popularity. For EIS procedure, both Constant-Weight approximation and our proposed computational method are implemented and their performance are measured by using Mean Square Error (MSE) which is given by,

$$\begin{aligned} MSE &= Var\left(\hat{f}(y_{0:T})\right) + Bias^2\left(\hat{f}(y_{0:T})\right) \\ &= Var\left(\hat{f}(y_{0:T})\right) + \left[E\left(\hat{f}(y_{0:T})\right) - f(y_{0:T})\right]^2 \end{aligned} \tag{2.3.1}$$

where the variance and mean can be obtained by applying Monte Carlo approximation with replication time be set to 100. Obviously, it is necessary to have the analytical solution of likelihood function for the evaluation of MSE. Therefore, the State Space Model for this simulation study must be Linear and Gaussian distributed so that we can have the access to the true likelihood value. To be specific, the State Space Model for this simulation study is given by,

$$\begin{aligned} y_t &= 2s_t + \epsilon_t,\ \epsilon_t \sim N(0,1) \\ f(y_t|s_t) &= N(y_t; 2s_t, 1) \\ s_t &= \alpha s_{t-1} + \sigma\eta_t,\ \eta_t \sim N(0,1) \\ f(s_t|s_{t-1}) &= N(s_t; \alpha s_{t-1}, \sigma^2) \end{aligned} \tag{2.3.2}$$

where $\alpha = \frac{1}{2}$ and $\sigma = 1$. The initial density is given by,

$$f(s_0) = N\left(s_0; 0, \frac{\sigma^2}{1-\alpha^2}\right) \tag{2.3.3}$$

Note that this choice of initial density ensures that $f(s_t) = f(s_0)$ for all t so that the computational process can be simplified for the Particle Filter algorithm. Moreover, we can see that this State Space Model is Linear Gaussian distributed which implies that its true likelihood value can be directly evaluated by applying Kalman Filter algorithm. Therefore, with the true likelihood value on hand, we will implement three different methods for the likelihood approximation and compare their performance by using Mean Square Error (MSE). The first algorithm is known as Bootstrap Particle Filter (BPF) which is a very simple and also most commonly used method of Sampling Importance Resampling. The transition density $f(s_t|s_{t-1})$ serves as the sampler and the unnormalized weight values are computed by measurement density $f(y_t|s_t)$. The second algorithm is the EIS procedure in which the filtering density is continuously approximated by Constant-Weight method, abbreviated to

Table 1: MC means, Bias and Standard Deviations of log-likelihood estimates

	Sample Size	True Likelihood	MC Mean	Absolute Bias	NSE
BPF	1 million	-2239.421	-2239.588	0.1670	0.554
CWEIS	100	-	-2239.421	1.948×10^{-7}	5.612×10^{-6}
PCEIS	100	-	-2239.421	1.462×10^{-7}	1.359×10^{-6}

Table 2: Mean Square Error and CPU time

	Sample Size	MSE	CPU Time
BPF	1 million	0.3348	648 min
CWEIS	100	3.15×10^{-11}	2.59 min
PCEIS	100	0.187×10^{-11}	6.17 min

CWEIS. The last algorithm is still the EIS procedure, but instead of applying Constant-Weight method, we continuously approximate the filtering density by using our proposed computational method, abbreviated to PCEIS.

Specifically speaking, a sequence of observations $\{y_t\}_{t=0}^{T}$ with $T = 1000$ are generated by simulating this Linear Gaussian State Space Model and its true log-likelihood value can be directly solved by using Kalman Filter algorithm. Figure 1 shows a short section of the simulated data for both observations and state variables. For BPF, the sample size is set to N = 1,000,000 (1 Million) and 100 i.i.d. log-likelihood estimates are generated by using 100 different seeds. For CWEIS and PCEIS, the sample sizes for likelihood approximation shown in 1.3.22 and optimization process shown in 1.3.27 are both set to $N = S = 100$. For the replication, we generated 100 i.i.d CWEIS and PCEIS log-likelihood estimates by using 100 different sets of Common Random Numbers. The Monte Carlo (MC) means, numerical standard errors (NSEs), Absolute Biases, Mean Square Errors (MSEs) and CPU time of these log-likelihood estimates by using BPF, CWEIS and PCEIS are reported in Table 1 and 2. Moreover, Figure 2 also shows the variability of these log-likelihood estimates for a more intuitive comparison on the efficiency between BPF and EIS.

Note first from Table 1 that the bias generated by either CWEIS or PCEIS is extremely small so that the influence caused by the biased likelihood estimation of EIS algorithm can be ignored to some extent. Therefore, we may claim that this EIS procedure can provide us with a nearly unbiased likelihood estimator. On the other hand, in comparison with CWEIS, the PCEIS generates a smaller bias and this also justifies the statement that our proposed computational method on the continuous approximation of filtering density demonstrates a better performance on likelihood estimation.

Generally speaking, the more samples taken for likelihood approximation, the smaller variance of the estimator. However, the efficiency of BPF on variance reduction by increasing sample size greatly far behind the one of EIS. This can be mainly attributed to the fact that variance minimization is achieved in EIS procedure and from table 1, the comparison on Numerical Standard Errors (NSEs) between BPF and EIS sufficiently demonstrates the great success of variance minimization. Even with one million particles, the BPF is still unable to compete with the EIS in which only 100 samples are taken for the likelihood

approximation. Moreover, the NSE of PCEIS is about 4 times smaller than the one of CWEIS which implies that our proposed computational method for the continuous approximation of filtering density facilitates a better performance on variance minimization.

The comparisons on Mean Square Errors (MSEs) given in Table 2 between BPF, CWEIS and PCEIS further justify the superiority of EIS procedure on the efficiency of likelihood approximation over BPF. Moreover, we can also notice that the MSE of PCEIS is approximately 17 times smaller than the one of CWEIS. Therefore, the computational method we proposed can exploit the potentiality of EIS algorithm in more depth and realize more efficiency gains than Constant-Weight EIS. Additional information provided in Table 2 are the CPU time for 100 replications (calculated on 3.6 GHz desktop using R) and BPF shows its hopelessly extreme inefficiency on likelihood approximation in comparison with EIS procedure.

2.4 Conclusion and Future Work

In this chapter, we proposed an improved computational method which can achieve a more accurate calculation performance on the continuous approximation of filtering density. In comparison with the Constant-Weight approach developed by David and Roman, our proposed method can realize a deeper exploitation on the potentiality of EIS procedure which have already been demonstrated in detail from our simulation study.

On the other hand, to make a meaningful comparison, it is necessary to have the access to the true likelihood value and this requirement imposes the restriction of linearity and normality on our simulated model selection. As a result of model simplicity, in our simulation study, we can see that the performance difference between these two methods measured by MSE is so tiny that our modification seems to be negligible. However, one important thing we should be aware of is that, in most real applications, the models we deal with usually have extremely complicated structure and the length of its time series is always very large. As a result, the EIS likelihood estimate generated by simply applying Constant-Weight approach in such kind of model tends to have an non-negligible bias. Therefore, under this scenario, bias reduction appears to be particularly important and our proposed method will demonstrates its necessity of existence.

In the next chapter, we will discuss the topic about Maximum Likelihood Parameter Estimation in State Space Model. Due to the infeasibility of having access to the closed-form solution of MLE for Non-linear State Space Model, we have to resort to algorithm of either Gradient Ascent (GA) or Expectation-Maximization (EM). Attributed to the common random number technique and variance minimization process, the likelihood estimate generated by EIS does not suffer from the problem of discontinuity and its fluctuation can be negligible. As a result, finite difference method can be reliably applied for the evaluation of score vector leading to the popularity of Gradient Ascent. However, there does not exist too much research works with respect to the EM algorithm. Therefore, in next chapter, we will propose a new procedure to achieve the EM algorithm by introducing the EIS optimization process.

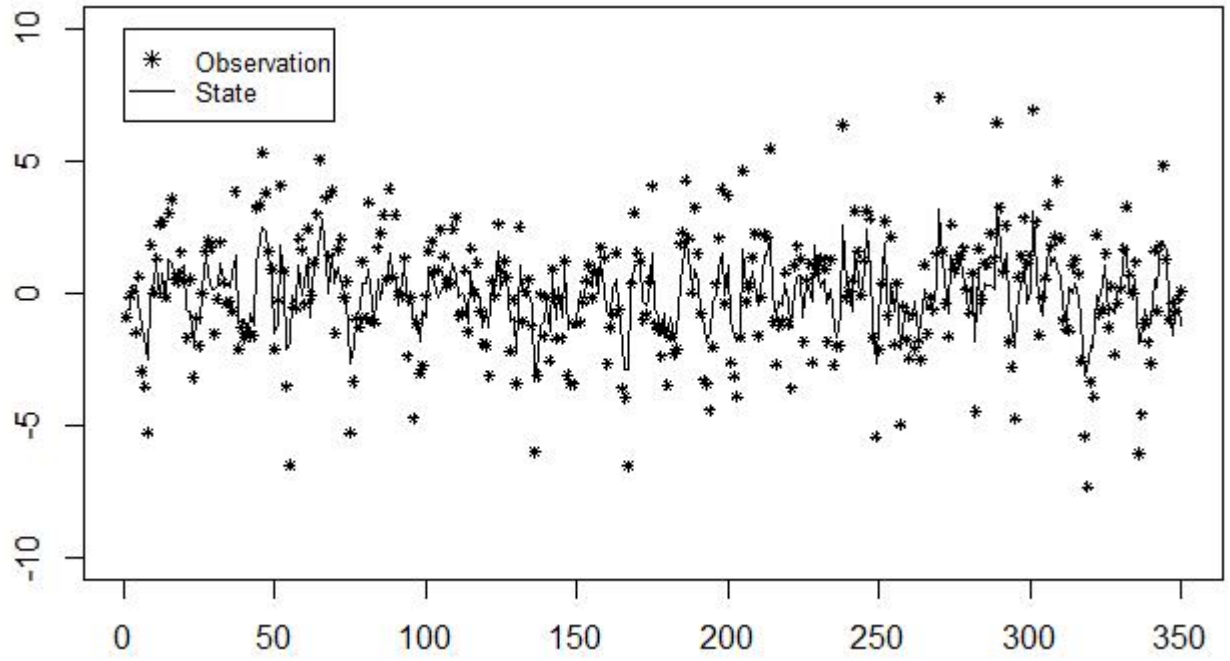

Figure 1: The variation of simulated Data

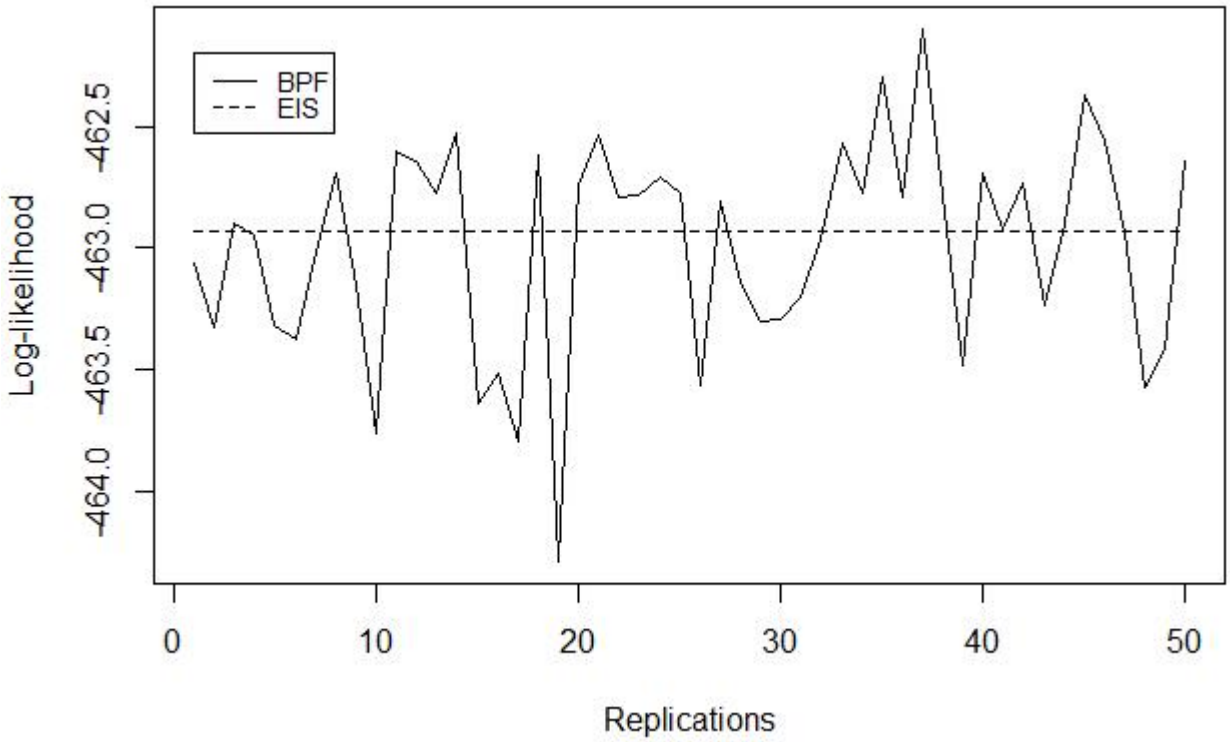

Figure 2: Comparison between BPF and EIS on Likelihood Approximation

CHAPTER III

MAXIMUM LIKELIHOOD PARAMETER ESTIMATION

3.1 Introduction

In this chapter, we will talk about the topic about the parameter estimation in State Space Model with Maximum Likelihood procedure which aims to solve the following maximization problem,

$$\begin{aligned} \hat{\theta}_{MLE} &= \underset{\theta}{ArgMax}\mathcal{L}_T(\theta) \\ \mathcal{L}_T(\theta) &= log\left\{f_\theta(y_{0:T})\right\} \end{aligned} \tag{3.1.1}$$

where $\mathcal{L}(\theta)$ is the log-likelihood function. When the State Space Model we dealing with is Linear and Gaussian distributed, the analytical expression of log-likelihood function is available and the closed-form solution of MLE can be just obtained by standard procedures [29]. However, when the model we met is Non-linear, the analytical expression of log-likelihood will be infeasible and it is impossible to have access to the closed-form solution of MLE. Therefore, we have to resort to the numerical methods to help us find the MLE for Non-linear State Space Model. So far, the most widely used method that can achieve this task is known as the Gradient Ascent algorithm in which the log-likelihood function is approximated by EIS procedure introduced before and the score vector is evaluated by applying finite difference method [12]. The detailed implementation is described as follows,

---

Gradient Ascent Algorithm of MLE for Non-linear State Space Model

- At iteration k, suppose we inherited θ_{k-1} from previous iteration
- Perform the EIS procedure to generate a sequence of optimal samplers $\{g_t(s_t, s_{t-1})\}_{t=0}^T$ for each time period, by making i.i.d draws from these samplers, the likelihood can be evaluated by

$$\begin{aligned} \hat{f}_{\theta_{k-1}}(y_t|y_{0:t-1}) &= \frac{1}{N}\sum_{i=1}^{N}\frac{f(y_t|s_t^i)f(s_t^i|s_{t-1}^i)\hat{f}(s_{t-1}^i|y_{0:t-1})}{g_t(s_t^i, s_{t-1}^i)} \\ \hat{f}_{\theta_{k-1}}(y_{0:t}) &= \prod_{t=0}^{T}\hat{f}(y_t|y_{0:t-1}) \end{aligned} \tag{3.1.2}$$

- By applying the Finite Difference Method, the score vector can be approximated by

$$\bigtriangledown_\theta \mathcal{L}_T(\theta)|_{\theta=\theta_{k-1}} = \frac{\mathcal{L}_T(\theta+h) - \mathcal{L}_T(\theta)}{h} \tag{3.1.3}$$

- By selecting an appropriate step size, the parameter can be updated as,

$$\theta_k = \theta_{k-1} + \gamma_k \bigtriangledown_\theta \mathcal{L}_T(\theta)|_{\theta=\theta_{k-1}} \tag{3.1.4}$$

- Repeat above steps until reaching the convergence of θ_k

One important thing we need to be aware of is the smoothness or continuity of the estimate of likelihood function. In general, given fixed parameters and observations, the EIS procedure by using random draws from these optimal samplers will generate a likelihood estimator with minimized variance. When this minimized variance is negligible, then we can say that this estimate of likelihood function is nearly smooth and the Maximum Likelihood Estimation (MLE) can be performed without any convergence issue under non-strict rule. However, in most scenarios, the variation of likelihood estimate can not be ignored and we may suffer from the convergence problem if the MLE is performed by using this likelihood estimate. One way to remedy this problem is known as Common Random Number (CRN) technique where all random draws in EIS procedure are obtained by transformation of a single set $\{u_i\}_{i=1}^N$ from a canonical distribution which does not depend on any parameters, i.e. the transformation of standard normal distribution.

Benefit from the variance minimization step in EIS procedure and the Common Random Number technique, the score vector in Gradient Ascent algorithm can be reliably evaluated by applying Finite Difference Method. However, this algorithm can be numerically unstable as it requires us to be very careful with the selection of the step sizes for each component of the score vector. On the other hand, the Expectation-Maximization (EM) algorithm is another very popular alternative procedure which can achieve the maximization task without log-likelihood approximation, score vector calculation and the step size selection. These properties make EM algorithm more attractive and robust than Gradient Ascent. However, by far, there does not exist too much research works about performing the EM algorithm for likelihood maximization on Non-linear State Space Model. Therefore, we proposed a new procedure, referred to as EIS Monte Carlo EM algorithm, in which the EIS principle is introduced into the E-step for the Monte Carlo approximation of expectation function leading to the feasibility of analytical solution of expectation maximization in M-step.

3.2 EM algorithm for Maximum Likelihood Estimation

3.2.1 Expectation-Maximization algorithm

The EM algorithm is proposed by Arthur Dempster, Nan Laird and Donald Rubin in 1977 [7]. It is an iterative method designed for the models involving hidden or missing variables to achieve the ML parameter estimation without evaluating the likelihood function. The EM iteration alternates between implementing an Expectation (E) step, in which a Q function

is built up for the evaluation of the expectation of the log-likelihood function involving both observations and hidden variables, and a Maximization (M) step, in which optimal parameters will be found such that this Q function can be maximized. Specifically, the detailed implementation of EM algorithm on Non-linear State Space Model is shown as follows,

---

EM algorithm on State Space Model

- At iteration k, suppose we inherited θ_{k-1} from previous iteration
- E-Step:

$$\begin{aligned} Q(\theta_{k-1}, \theta) &= E_{s_{0:T}|y_{0:T}} \left\{ log \left[f_\theta(s_{0:T}, y_{0:T}) \right] \right\} \\ &= \int log \left[f_\theta(s_{0:T}, y_{0:T}) \right] \cdot f_{\theta_{k-1}}(s_{0:T}|y_{0:T}) ds_{0:T} \end{aligned} \tag{3.2.1}$$

$$f_\theta(s_{0:T}, y_{0:T}) = f(s_0) \prod_{t=1}^{T} f(s_t|s_{t-1}) \prod_{t=0}^{T} f(y_t|s_t) \tag{3.2.2}$$

- M-Step

$$\theta_k = \underset{\theta}{ArgMax} Q(\theta_{k-1}, \theta) \tag{3.2.3}$$

- Repeat until reaching the convergence of θ_k

3.2.2 Monte Carlo EM algorithm with Forward-Backward Smoothing

Generally speaking, the analytical expression of $Q(\theta_{k-1}, \theta)$ is not available for the Non-linear State Space Model. Thus, we have to rely on the numerical method and the most intuitive way is to perform Monte Carlo approximation in which we make i.i.d draws from $f_{\theta_{k-1}}(s_{0:T}|y_{0:T})$ leading to,

$$\hat{Q}(\theta_{k-1}, \theta) = \frac{1}{N} \sum_{i=1}^{N} log \left[f_\theta(s^i_{0:T}, y_{0:T}) \right] \tag{3.2.4}$$

This is referred to as Monte Carlo EM algorithm. Unfortunately, this approximation is not feasible due to the fact that the analytical expression of this smoothing density $f_{\theta_{k-1}}(s_{0:T}|y_{0:T})$ is intractable under the case of Non-linear State Space Model. Therefore, it is impossible to directly make i.i.d draws from $f_{\theta_{k-1}}(s_{0:T}|y_{0:T})$. One way to solve this problem is known as the Forward-Backward Smoothing proposed by Godsill, Doucet and West in 2004 [13]. This method relies on the Particle Filter algorithms and its principle is based on the following key decomposition,

$$f_{\theta_{k-1}}(s_{0:T}|y_{0:T}) = f_{\theta_{k-1}}(s_T|y_{0:T}) \prod_{t=0}^{T-1} f_{\theta_{k-1}}(s_t|y_{0:t}, s_{t+1}) \tag{3.2.5}$$

where $f_{\theta_{k-1}}(s_t|y_{0:t}, s_{t+1})$ is referred to as backward Markov transition density. By Bayesian principle, it can be written by

$$f_{\theta_{k-1}}(s_t|y_{0:t}, s_{t+1}) = \frac{f_{\theta_{k-1}}(s_{t+1}|s_t) f_{\theta_{k-1}}(s_t|y_{0:t})}{f_{\theta_{k-1}}(s_{t+1}|y_{0:t})} \tag{3.2.6}$$

By implementing the Particle Filter algorithm from time $t = 0$ to T, a sequence of approximated filtering densities $\{\hat{f}_{\theta_{k-1}}(s_t|y_{0:t})\}_{t=0}^{T}$ will be obtained and is given by the following form of mixture of Dirac measure,

$$\hat{f}_{\theta_{k-1}}(s_t|y_{0:t}) = \sum_{i=1}^{N} W_t^i \delta_{s_t^i}(s_t) \tag{3.2.7}$$

By substituting $\hat{f}_{\theta_{k-1}}(s_t|y_{0:t})$ for $f_{\theta_{k-1}}(s_t|y_{0:t})$ in 3.2.6, this backward Markov transition density $f_{\theta_{k-1}}(s_t|y_{0:t}, s_{t+1})$ can be approximated by,

$$\begin{aligned}
\hat{f}_{\theta_{k-1}}(s_t|y_{0:t}, s_{t+1}) &= \frac{\sum_{i=1}^{N} W_t^i f_{\theta_{k-1}}(s_{t+1}|s_t^i)\delta_{s_t^i}(s_t)}{\hat{f}_{\theta_{k-1}}(s_{t+1}|y_{0:t})} \\
f_{\theta_{k-1}}(s_{t+1}|y_{0:t}) &= \int f_{\theta_{k-1}}(s_{t+1}|s_t) f_{\theta_{k-1}}(s_t|y_{0:t}) ds_t \\
\hat{f}_{\theta_{k-1}}(s_{t+1}|y_{0:t}) &= \sum_{i=1}^{N} W_t^i f_{\theta_{k-1}}(s_{t+1}|s_t^i)
\end{aligned} \tag{3.2.8}$$

To make the i.i.d draws from this smoothing density $f_{\theta_{k-1}}(s_{0:T}|y_{0:T})$, we just need to sample firstly from the approximated filtering density $\hat{f}_{\theta_{k-1}}(s_T|y_{0:T})$ at the last time period, then make draws from $\{\hat{f}_{\theta_{k-1}}(s_t|y_{0:t}, s_{t+1})\}_{t=0}^{T-1}$ backward iteratively and the Q function can be just approximated by equation 3.2.4. This sampling method is also referred to as Forward Filtering Backward Sampling (FFBSa) and it needs $\mathcal{O}(NT+1)$ operations to generate a single path $s_{0:T}$, as sampling from $\{\hat{f}_{\theta_{k-1}}(s_t|y_{0:t}, s_{t+1})\}_{t=0}^{T-1}$ for each time period need to calculate $f_{\theta_{k-1}}(s_{t+1}|s_t^i)$ for N times leading to $\mathcal{O}(N)$ operations. Therefore, it requires a total of $\mathcal{O}(N^2T + N)$ operations to generate a full path i.i.d draws $\{s_{0:T}^i\}_{i=1}^{N}$. As a result, the implementation of this Monte Carlo EM algorithm will be extremely inefficient and time-consuming if we take very large value for N.

These drawbacks can be mainly imputed to the use of mixture of Dirac measures for the approximation of filtering densities. Therefore, to remedy these problems, we need to find a way to achieve the Monte Carlo EM algorithm without the use of mixture of Dirac measures. Based on the fact that the EIS principle introduced before can generates estimate with minimized variance and the filtering density is approximated continuously, in next section, we will propose a EIS-based approach to achieve the Monte Carlo EM algorithm with the abandonment of mixture of Dirac measure leading to a great efficiency gain.

3.2.3 Monte Carlo EM algorithm using EIS

Similar to the FFBSa sampling method of smoothing density $f_{\theta_{k-1}}(s_{0:T}|y_{0:T})$ introduced before, our proposed method also consists of Forward and Backward procedures except that we perform EIS algorithm instead of Particle Filter during the implementation of Forward procedure. Specifically speaking, the joint likelihood function in State Space Model can be factorized into a product of a sequence of conditional likelihood densities, that is,

$$f_{\theta_{k-1}}(y_{0:T}) = \prod_{t=0}^{T} f_{\theta_{k-1}}(y_t|y_{0:t-1}) \tag{3.2.9}$$

and each conditional likelihood density can be written by,

$$\begin{aligned} f_{\theta_{k-1}}(y_t|y_{0:t-1}) &= \int\int f(y_t|s_t)f(s_t|s_{t-1})f(s_{t-1}|y_{0:t-1})ds_t d_{t-1} \\ &\approx \int\int \frac{f(y_t|s_t)f(s_t|s_{t-1})\hat{f}(s_{t-1}|y_{0:t-1})}{g_t(s_t, s_{t-1})} g_t(s_t, s_{t-1})ds_t d_{t-1} \end{aligned} \tag{3.2.10}$$

where $\hat{f}(s_{t-1}|y_{0:t-1})$ is the continuous approximation of filtering density generated from last time period. By applying EIS algorithm, an optimal sampler $g_t(s_t, s_{t-1})$ will be generated such that the variance of $\frac{f(y_t|s_t)f(s_t|s_{t-1})\hat{f}(s_{t-1}|y_{0:t-1})}{g_t(s_t,s_{t-1})}$ can be minimized leading to the likelihood estimate with smallest variation. Moreover, this optimal sampler can also be written by,

$$g_t(s_t, s_{t-1}) = g_t(s_t)g_t(s_{t-1}|s_t) \tag{3.2.11}$$

where the marginal density $g_t(s_t)$ will be stored and further passed into the Backward smoothing procedure. The continuous approximation of filtering density $f(s_t|y_{0:t})$ can be obtained either by Constant-Weight approximation or our proposed computational method introduced in Chapter 2.

After the completion of the Forward EIS procedure, a sequence of marginal densities $\{g_t(s_t)\}_{t=0}^{T}$ will be obtained and we can proceed to the Backward smoothing procedure by rewriting the smoothing density $f(s_{0:T}|y_{0:T})$ shown in 3.2.5 and 3.2.6 as

$$f(s_{0:T}|y_{0:T}) = g_T(s_T) \prod_{t=0}^{T-1} f(s_t|y_{0:t}, s_{t+1}) \tag{3.2.12}$$

$$f(s_t|y_{0:t}, s_{t+1}) = \frac{f(s_{t+1}|s_t)g_t(s_t)}{f(s_{t+1}|y_{0:t})} \tag{3.2.13}$$

where the filtering densities $\{f(s_t|y_{0:t})\}_{t=0}^{T}$ are replaced by $\{g_t(s_t)\}_{t=0}^{T}$ based on the Constant-Weight approximation which claims that,

$$f(s_t|y_{0:t}) \approx g_t(s_t) \tag{3.2.14}$$

To make the i.i.d draws $\{s_{0:T}^i\}_{i=1}^{N}$ from the smoothing density $f(s_{0:T}|y_{0:T})$, we just need to firstly sample s_T^i from $g_T(s_T)$, then for $t = T-1, T-2, ..., 0$, taking draws s_t^i from

$f(s_t|y_{0:t}, s_{t+1})$. However, sampling directly from $f(s_t|y_{0:t}, s_{t+1})$ is not feasible since its analytical expression does not exist. To remedy this problem, one intuitive way is to perform the Importance sampling method in which $g_t(s_t)$ serves as the proposal sampler and $f(s_t|y_{0:t}, s_{t+1})$ can be approximated by the following mixture of Dirac measures,

$$\begin{aligned} \hat{f}(s_t|y_{0:t}, s_{t+1}) &= \sum_{i=1}^{N} W_t^i \delta_{s_t^i}(s_t) \\ W_t^i &= \frac{w_t^i}{\sum_{i=1}^{N} w_t^i} \\ w_t^i &= f(s_{t+1}|s_t^i) \end{aligned} \tag{3.2.15}$$

where $\{s_t^i\}_{i=1}^N$ is the i.i.d draws from $g_t(s_t)$. However, as we can see, to generate a single path of $f(s_{0:T}|y_{0:T})$, we have to perform N times calculation on $\{W_t^i\}_{i=1}^N$ at each time period for $t = T-1, T-2, ..., 0$ leading to $\mathcal{O}(NT+1)$ operations which is the same with the Forward Filtering Backward Smoothing. Moreover, this intuitive method also involves the use of mixture of Dirac measures which is the root of inefficiency problem.

Since the EIS procedure introduced before can achieve the variance minimization on weight function, we developed a Monte Carlo EM algorithm for ML parameter estimation on Non-linear State Space Model based on Constant-Weight principle proposed by David and Roman [5] which claims that the weight function of minimized variance can be just viewed as a constant. Specifically speaking, the equation 3.2.13 can be rewritten as,

$$f(s_t|y_{0:t}, s_{t+1}) = \frac{\frac{f(s_{t+1}|s_t)g_t(s_t)}{q_t(s_t)} q_t(s_t)}{f(s_{t+1}|y_{0:t})} \tag{3.2.16}$$

Our objective is to find an optimal sampler $q_t(s_t)$ such that the variance of weight function $\frac{f(s_{t+1}|s_t)g_t(s_t)}{q_t(s_t)}$ can be made as small as possible and the ultimate goal is to achieve the variance minimization. Since the closed-form of $f(s_{t+1}|s_t)g_t(s_t)$ is available, this ultimate goal can be accomplished by just following the standard EIS procedure introduced before. Moreover, as we can see, the normalized constant $f(s_{t+1}|y_{0:t})$ in the denominator of above equation can be evaluated by

$$f(s_{t+1}|y_{0:t}) = \int \frac{f(s_{t+1}|s_t)g_t(s_t)}{q_t(s_t)} q_t(s_t) ds_t \tag{3.2.17}$$

and its Monte Carlo approximation can be performed by just making i.i.d draws from $q_t(s_t)$ leading to,

$$\hat{f}(s_{t+1}|y_{0:t}) = \frac{1}{N} \sum_{i=1}^{N} \frac{f(s_{t+1}|s_t^i)g_t(s_t^i)}{q_t(s_t^i)} \tag{3.2.18}$$

By the Constant-Weight principle, we can view this weight function as a constant and the

backward Markov transition density $f(s_t|y_{0:t}, s_{t+1})$ can be estimated by,

$$\begin{aligned}\hat{f}(s_t|y_{0:t}, s_{t+1}) &= \frac{\frac{f(s_{t+1}|s_t)g_t(s_t)}{q_t(s_t)}q_t(s_t)}{\hat{f}(s_{t+1}|y_{0:t})} \\ &= q_t(s_t)\end{aligned} \tag{3.2.19}$$

Therefore, to make a single draw from $f(s_t|y_{0:t}, s_{t+1})$, we just need to take one sample from $q_t(s_t)$ and the operation of making a single path of $f(s_{0:T}|y_{0:T})$ is reduced to $\mathcal{O}(ET+1)$ in which $E << N$ is the computational complexity of EIS procedure and it usually takes less than 10 operations to reach the convergence.

After the generation of i.i.d draws from $f(s_{0:T}|y_{0:T})$, denoted as $\{s^i_{0:T}\}_{i=1}^N$, the Monte Carlo approximation of the expectation function in E-step is given by,

$$\begin{aligned}\hat{Q}(\theta) &= \frac{1}{N}\sum_{i=1}^{N} log\left[f_\theta(s^i_{0:T}, y_{0:T})\right] \\ &= \frac{1}{N}\sum_{i=1}^{N} log\left\{f(s^i_0)\prod_{t=1}^{T} f(s^i_t|s^i_{t-1})\prod_{t=0}^{T} f(y_t|s^i_t)\right\}\end{aligned} \tag{3.2.20}$$

And this MC approximation will be passed into M-step to find an optimal θ such that the maximization can be achieved for $\hat{Q}(\theta)$. When the analytical expressions of transition density $f(s_t|s_{t-1})$ and measurement density $f(y_t|s_t)$ are available, this maximization problem can be directly solved by standard derivation method leading to a closed-form solution which will be passed into next iteration until the convergence be reached. The detailed implementation process is shown as follows,

---

Monte Carlo EM algorithm of EIS principle

- At iteration k, suppose we inherited θ_{k-1} from previous iteration
- E-Step:
  - Start from $t = 0, 1, 2, ..., T$, implement EIS procedure to the following integral,

$$\int\int \frac{f(y_t|s_t)f(s_t|s_{t-1})\hat{f}(s_{t-1}|y_{0:t-1})}{g_t(s_t, s_{t-1})}g_t(s_t, s_{t-1})ds_t d_{t-1} \tag{3.2.21}$$

    in which a sequence of optimal samplers $\{g_t(s_t, s_{t-1})\}_{t=0}^T$ are generated and the marginal densities $\{g_t(s_t)\}_{t=0}^T$ will be stored and passed into next step.
  - For i=1,2,...,N, repeat the following process
    - Take one sample from $g_T(s_T)$, denoted as s^i_T
    - Start from $t = T-1, ..., 2, 1, 0$, implement EIS procedure to the following integral,

$$\int \frac{f(s^i_{t+1}|s_t)g_t(s_t)}{q_t(s_t)}q_t(s_t)ds_t \tag{3.2.22}$$

in which an optimal sampler $q_t(s_t)$ will be generated and a single draw will be taken from it, denoted as s_t^i

– After the generation of i.i.d draws $\{s_{0:T}^i\}_{i=1}^N$ from $f(s_{0:T}|y_{0:T})$, the Q-function can be approximated by,

$$\hat{Q}(\theta_{k-1},\theta) = \frac{1}{N}\sum_{i=1}^{N} log\left\{ f(s_0^i)\prod_{t=1}^{T} f(s_t^i|s_{t-1}^i)\prod_{t=0}^{T} f(y_t|s_t^i)\right\} \tag{3.2.23}$$

- M-Step

$$\theta_k = \underset{\theta}{ArgMax}\hat{Q}(\theta_{k-1},\theta) \tag{3.2.24}$$

- Repeat until reaching the convergence

As we can see, it requires a total of $\mathcal{O}(NET+N)$ operations to generate a full path i.i.d draws from $f(s_{0:T}|y_{0:T})$. Although it takes less operations than FFBSa-based EM algorithm, its performance is barely satisfactory. So, in next section, we will propose an improved EIS-based EM algorithm which can achieves the i.i.d draws from $f(s_{0:T}|y_{0:T})$ with very low computational complexity.

3.2.4 A Fast-Sampling Technique

From the detailed implementation process of EIS-based Monte Carlo EM algorithm shown in the previous section, the high computational complexity of the generation of i.i.d draws from $f(s_{0:T}|y_{0:T})$ is mainly caused by the overuse of EIS procedure. As we can see, it requires to implement the EIS algorithm for a total of NT times to generate a full path of i.i.d draws. To remedy this problem, we developed a procedure, referred to as Fast-Sampling technique which can further reduces the operation time from $\mathcal{O}(NET+N)$ to $\mathcal{O}(PET+1)$ where $P<N$.

Specifically speaking, the detailed implementation process of EIS procedure for the integral shown in 3.2.22 is shown as follows,

---

Efficient Importance Sampling of Backward Sampling

- At time period t, suppose we inherited $\{s_{t+1}^i\}_{i=1}^N$ from the previous time point
- Select an appropriate exponential family of distribution $\{q_t(s_t|a) : a \in A\}$
- For $i = 1, 2, ...N$,
  - At step L=0, the initial particles $\{s_{t,j}^0\}_{j=1}^S$ can be obtained by making draws from $g_t(s_t)$
  - At step L=1, solve the following minimization problem by least square method,

$$(\hat{a}_{t,i}^1, \hat{c}_{t,i}^1) = \underset{a\in A, c\in\mathbb{R}}{ArgMin}\sum_{j=1}^{S}\left\{ ln\left[f(s_{t+1}^i|s_{t,j}^0)g_t(s_{t,j}^0)\right] - c - lnk_t(s_{t,j}^0|a)\right\} \tag{3.2.25}$$

- Draw particles $\{s_{t,j}^1\}_{j=1}^S$ from $q_t(s_t|\hat{a}_{t,i}^1)$ and pass it into next step
- repeat the above process until reaching the convergence of $\{\hat{a}_{t,i}^L\}$ and the final optimal value is denoted as $\hat{a}_{t,i}$

- After the generation of optimal samplers $q_t(s_t|\hat{a}_{t,i})$ for $i = 1, 2, ..., N$, we take one single draw s_t^i from each of these samplers and the resulting particles $\{s_t^i\}_{i=1}^N$ will be passed into next time period t-1 for new iteration.

Here, $k_t(s_t|a)$ is the kernel density of $q_t(s_t|a)$. By the property of exponential family distribution, this kernel density can be factorized into,

$$lnk_t(s_t|a) = a^{'} \cdot t(s_t) \tag{3.2.26}$$

where a is the vector of regression parameters and $t(s_t)$ denotes the vector of explanatory variables. On the other hand, the term $ln\left[f(s_{t+1}^i|s_t)g_t(s_t)\right]$ serves as the dependent variable. One important thing we should be aware of is that the only difference of these EIS procedures for $i = 1, 2, ..., N$ lays on the term $f(s_{t+1}^i|s_t)$. If we have a sequence of particles $\{s_{t+1}^{r_i}\}_{i=1}^R \subset \{s_{t+1}^i\}_{i=1}^N$ and their values do not differ too much, we can expect that the optimal values $\{\hat{a}_{r_i}\}_{i=1}^R$ generated by their respective EIS procedures will be approximated equal with each other. Based on this fact, we can just take their mean value for EIS implementation and the generated optimal value can be shared by these particles of similar values. So, instead of implementing EIS procedure for R times, we just need to take the mean value and perform one single EIS which achieves great efficiency gain. Based on this idea, we developed the Fast-Sampling technique for the EIS-based EM algorithm proposed in previous section to efficiently generate i.i.d draws $\{s_{0:T}^i\}_{i=1}^N$ from smoothing density $f(s_{0:T}|y_{0:T})$ and the detailed implementation process is shown as follows,

---

Fast-Sampling technique of EIS-based EM algorithm

- At the time period t, suppose we inherited a sequence of particles $\{s_{t+1}^i\}_{i=1}^N$
- Sort $\{s_{t+1}^i\}_{i=1}^N$ ascendingly and denote the resulting sorted sequence as $\{s_{t+1}^{(i)}\}_{i=1}^N$
- Partition $\{s_{t+1}^{(i)}\}_{i=1}^N$ into $P = \frac{N}{R}$ sections and each section has R particles of similar values.
- Take the mean value for each section and denote these mean values as $\{\mu_r\}_{r=1}^P$
- For $r = 1, 2, ...P$,
  - Implement the EIS procedure to the following integral,

$$\int \frac{f(\mu_r|s_t)g_t(s_t)}{q_t(s_t)} q_t(s_t)ds_t \tag{3.2.27}$$

    in which an optimal sampler will be generated and we take R draws from it, denoted as $\{s_{t,r}^j\}_{j=1}^R$

- Return these generated particles $\left\{\{s_{t,r}^j\}_{j=1}^R\right\}_{r=1}^P$ to their original location before sorting leading to the sequence $\{s_t\}_{t=1}^N$ which will be passed into next time period $t-1$

Here, R is referred to as the section size and P is known as partition length. When $R=1$, this EIS-based EM algorithm is equivalent with the one without Fast-Sampling technique.

In the next section, a simulation study of ML parameter estimation on an Non-linear State Space Model will be given to demonstrate the superiority of our proposed method against Gradient Ascent and FFBSa methods.

3.3 Simulation Study

3.3.1 Stochastic Volatility Model

In this simulation study, we will explore the Maximum Likelihood Parameter Estimation on Stochastic Volatility (SV) Model which is a very popular economic model used in the field of mathematical finance to evaluate derivative securities, such as options. To be specific, the SV model for our simulation study is given by,

$$\begin{aligned} y_t &= \beta exp\left(\frac{s_t}{2}\right) W_t, W_t \sim N(0,1) \\ s_t &= \alpha s_{t-1} + \sigma V_t, V_t \sim N(0,1) \\ s_0 &\sim N(0,1) \end{aligned} \tag{3.3.1}$$

Clearly, this is an Non-linear State Space model which implies the fact that the analytical expression of its likelihood function is not available leading to the infeasibility of closed-form solution of MLE by standard derivation method. Therefore, we have to rely on numerical method to achieve the likelihood maximization and the most intuitive one is Gradient Ascent.

To implement the ML parameter estimation by Gradient Ascent algorithm, the first thing we need to consider about is the approximation of score vector. Generally speaking, this task can be achieved by just using Finite Difference Method which requires the evaluation of likelihood function. However, as we said before, the Non-linearity of this SV model makes it unavailable to apply any computational methods such as Kalman Filter algorithm to find the true value of likelihood function. Therefore, we have to rely on the numerical methods and the most commonly used one is the Particle Filter. Unfortunately, due to the resampling step involved in Particle Filters, it is not available to apply Common Random Number technique to have the access to a continuous likelihood approximation, and without continuity, the convergence problem will arises. Moreover, as shown in previous simulation study, Particle Filter shows hopeless inefficiency on likelihood approximation when we compare it with EIS procedure. So in this simulation study, for likelihood evaluation in Gradient Ascent, instead of implementing Particle Filter, we follow the EIS procedure to generate likelihood approximations with the property of continuity. Another advantage of performing EIS lays on the fact that the variance of likelihood estimate is minimized. As a result, the score vector can be reliably approximated by the Finite Difference Method.

Except of Gradient Ascent, the EM algorithm is another alternative which can also be used for ML parameter estimation. As introduced before, the challenge of implementing this

algorithm lays on the Monte Carlo approximation of expectation function in E-step which requires to make i.i.d draws from the smoothing density $f(s_{0:T}|y_{0:T})$. The original method to conquer this challenge is proposed by Godsill, Doucet and West in 2004 and referred to as Forward Filter and Backward Sampling (FFBSa). However, this method also suffers from the inefficiency problem. To remedy this, we proposed an EIS-based Monte Carlo EM algorithm with Fast-Sampling technique and its superiority will be demonstrated by comparison with Gradient Ascent and FFBSa.

3.3.2 Simulation Setting and Results

Specifically speaking, a sequence of observations $\{y_t\}_{t=0}^{T}$ of T=300 are generated by simulating this Stochastic Volatility Model with parameters be set as: $\alpha = 0.91, \sigma = 1, \beta = 0.5$ and this generation process will be repeated for 50 replication times. So, a total of 50 observation sequences, denoted as $\{y_{0:T}^{i}\}_{i=1}^{50}$, are generated and one of these observation sequences is plotted in Figure 3. A total of three ML parameter estimation methods are performed. The first method is based on the Gradient Ascent algorithm in which the likelihood function is evaluated by just following EIS procedure and this method is abbreviated to GA-EIS-MLE. The second method is based on EM algorithm in which the Monte Carlo approximation of expectation function in E step is performed by Forward Filtering Backward Sampling (FFBSa) method and this method is referred to as EM-FFBSa-MLE. The last method is also based on the EM algorithm, instead of implementing FFBSa, in this method, the Monte Carlo approximation of Q-function in E-step is achieved by performing our proposed EIS-based EM algorithm abbreviated to EIS-EM-MLE with Fast-Sampling technique in which the section sizes are set to $R = 1, 4, 10$.

In GA-EIS-MLE, the sample size of the likelihood approximation in 3.1.2 and EIS variance minimization process in 1.3.27 are both set to $N = S = 200$. The increment value in Finite Difference Method is selected as $h = 0.01$ and the step size of Gradient Ascent is determined by $\gamma = 0.001$ for all the components of score vector. In the Expectation step of the Monte Carlo EM algorithm, the sample size for the $\hat{Q}(\theta_{k-1}, \theta)$ in 3.2.4 is set to 100. Moreover, in the maximization step of EM algorithm, the closed-form solution is available and given by,

$$\alpha = \frac{\sum_{i=1}^{N}\sum_{t=1}^{T} s_t^i s_{t-1}^i}{\sum_{i=1}^{N}\sum_{t=1}^{T}\left(s_{t-1}^i\right)^2}$$

$$\beta = \sqrt{\frac{1}{T+1}\frac{1}{N}\sum_{i=1}^{N}\sum_{t=0}^{T}\frac{y_t^2}{exp(s_t^i)}} \tag{3.3.2}$$

$$\sigma = \sqrt{\frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}(s_t^i - \alpha s_{t-1}^i)^2}$$

The Monte Carlo (MC) means, Absolute Biases, Numerical Standard Errors (NSEs) and

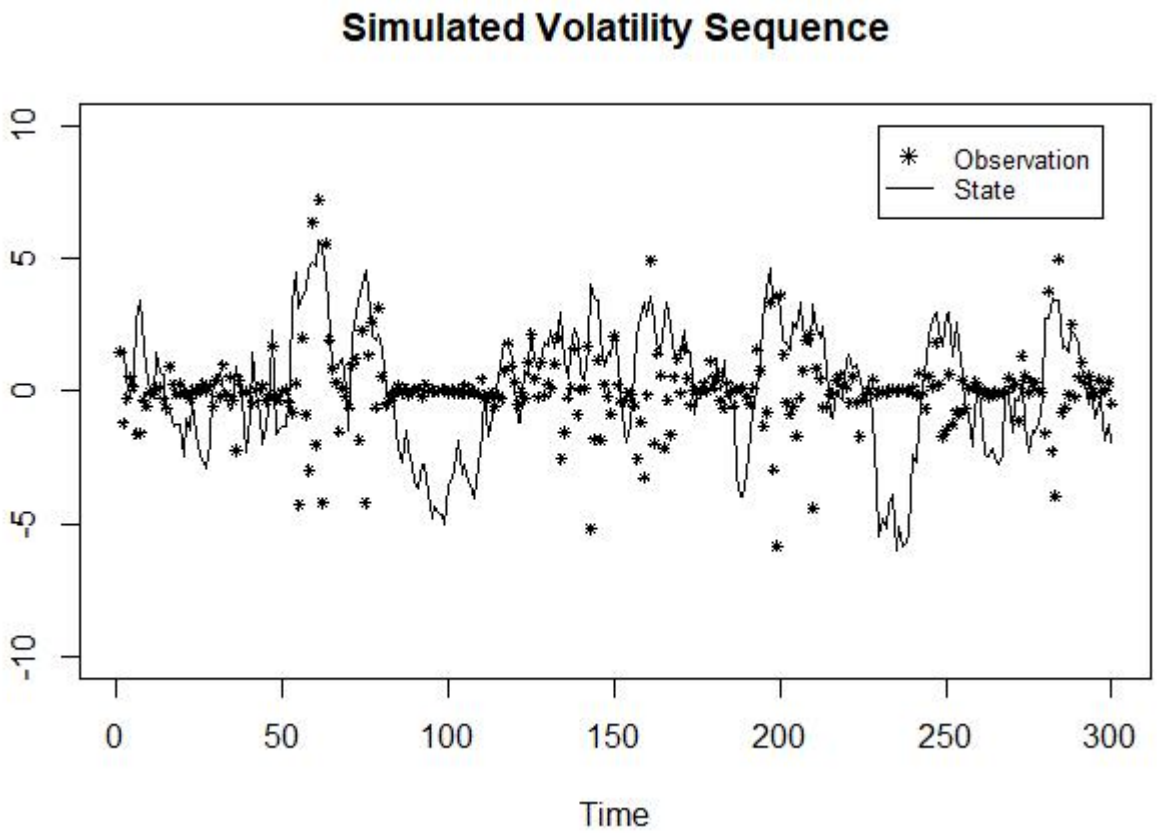

Figure 3: A simulation of the Stochastic Volatility Model

Mean Square Errors (MSEs) of these ML parameter estimates are reported in Table 3 and the Computational time of 50 replications is shown in table 4. Moreover, Figure 4 also shows the histograms of these ML parameter estimates which indicate that the distributions of these parameters are not too far from normality.

From the table 3, we can firstly notice that all the absolute biases of ML estimate for parameters α, σ, β are not significantly different from zero. With respect to the variation, in comparison with GA-EIS-MLE and EM-FFBSa-MLE, the ML estimate of these three parameters generated by our proposed EIS-based EM algorithm (R=1) demonstrates the lowest variation which can be mainly attributed to the variance minimization processs involved in the EIS principle. The performance of these algorithms is measured by Mean Square Error, and as we expected, our proposed EIS-based Monte Carlo EM algorithm (R=1) revealed its great superiority against EIS-based Gradient Ascent and FFBSa-based Monte Carlo EM algorithms. However, with the Fast-Sampling technique be applied (R=4, 10), the performance of our proposed method became slightly worse than the one without Fast-Sampling (R=1), but still better than Gradient Ascent method. This is a small price we paid for the great computational efficiency gain which is demonstrated in table 4.

Table 3: The Simulation Result

True $\alpha = 0.91$		MC mean	Abs Bias	NSEs	MSE
GA-EIS-MLE		0.8784	0.0316	0.0920	0.0095
EM-FFBSa-MLE		0.8998	0.0101	0.0409	0.0018
EIS-EM-MLE	R=1	0.9008	0.0092	0.0269	0.0008
	R=4	0.8897	0.0203	0.0373	0.0017
	R=10	0.8834	0.0266	0.0415	0.0024
True $\sigma = 1.0$		MC mean	Abs Bias	NSEs	MSE
GA-EIS-MLE		1.0321	0.0321	0.1428	0.0214
EM-FFBSa-MLE		0.9934	0.0066	0.0958	0.0092
EIS-EM-MLE	R=1	0.9809	0.0191	0.0582	0.0038
	R=4	0.9693	0.0307	0.0592	0.0044
	R=10	0.9892	0.0108	0.0755	0.0058
True $\beta = 0.5$		MC mean	Abs Bias	NSEs	MSE
GA-EIS-MLE		0.5899	0.0899	0.4463	0.2072
EM-FFBSa-MLE		0.5251	0.0251	0.1313	0.0179
EIS-EM-MLE	R=1	0.5583	0.0583	0.0734	0.0088
	R=4	0.5673	0.0673	0.0683	0.0092
	R=10	0.5642	0.0642	0.0773	0.0101

3.4 Conclusion and Future Work

In this chapter, we proposed an EIS-based Monte Carlo EM algorithm for ML parameter estimation and developed a technique known as Fast-Sampling to achieve great computational efficiency gain by paying very small price on the estimation performance. A simulation study based on a Non-linear Stochastic Volatility Model is given to demonstrate the great superiority of our proposed method. In comparison with Gradient Ascent, the ML parameter estimation performed by EM framework does not bother with the problems of likelihood evaluation, score vector calculation and step size selection leading to a more stable estimation performance. However, the implementation of EM algorithm requires the evaluation on expectation of $f(s_{0:T}, y_{0:T})$ shown in 3.2.1 and we have to rely on Monte Carlo method. To obtain the i.i.d draws from the smoothing density $f(s_{0:T}|y_{0:T})$, previous research work relies on the technique known as Forward Filtering Backward Sampling (FFBSa). However, this method suffers greatly from the problem of computational inefficiency and this problem is mainly imputed to the use of Mixture of Dirac measures for the approximation of filtering density $f(s_t|y_{0:t})$ and Backward Markov transition density $f(s_t|y_{0:t}, s_{t+1})$. Based on the EIS procedure and Constant-Weight principle, our proposed method achieved the

Table 4: Computational Time (second) of 50 replications

GA-EIS-MLE		536
EM-FFBSa-MLE		2658
EIS-EM-MLE	R=1	1317
	R=4	362
	R=10	186

Monte Carlo EM algorithm with the abandonment of Mixture of Dirac measures leading to a great efficiency gain. Moreover, our developed Fast-Sampling technique can further reduces the computational complexity with a small price be paid on the MLE performance. In the next chapter, we will introduce how our developed methods can be applied to real world applications: Dynamic Stochastic General Equilibrium modeling.

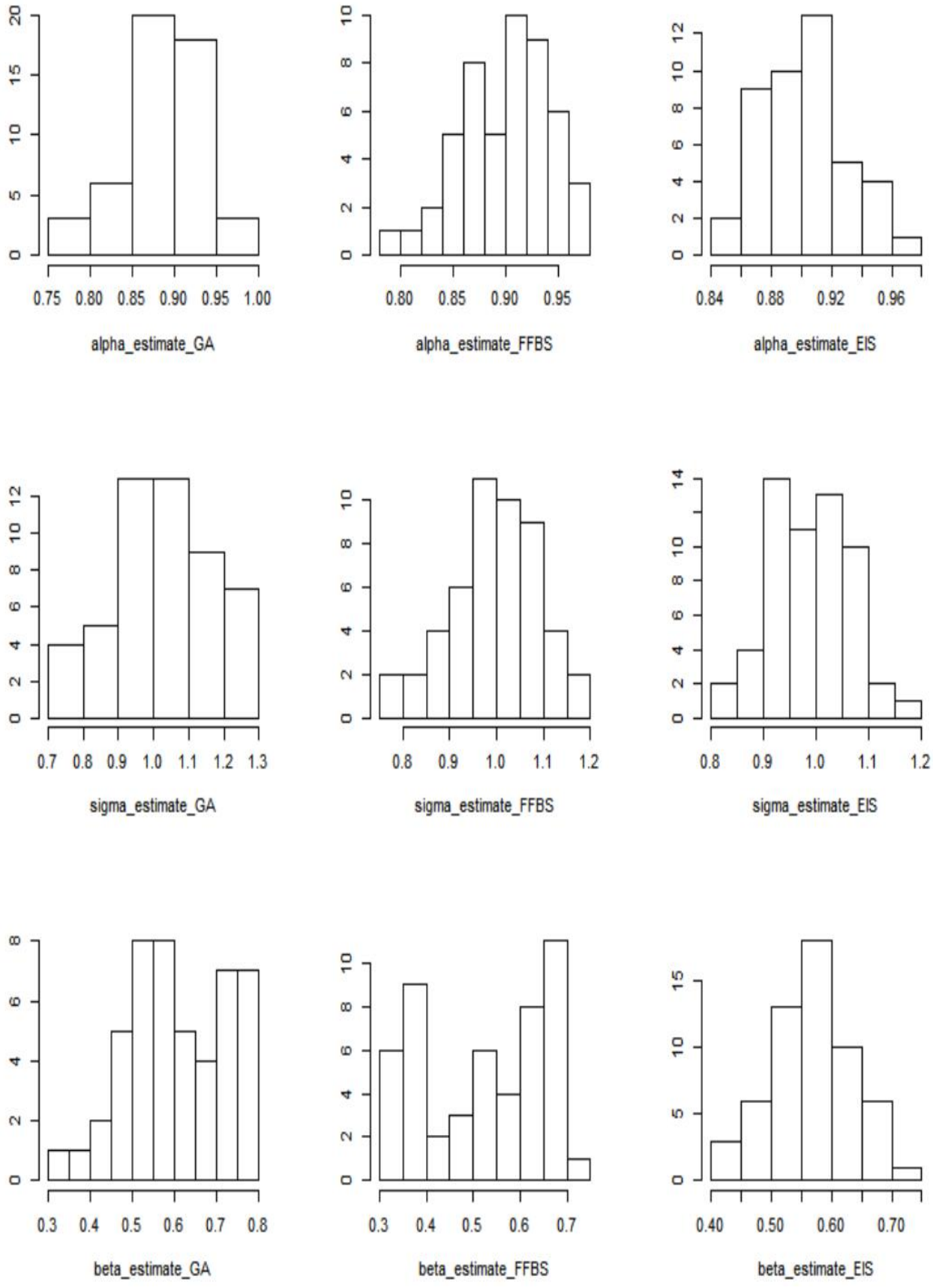

Figure 4: Histogram of ML estimate

CHAPTER IV

AN APPLICATION TO DSGE MODELS

4.1 Introduction

4.1.1 DSGE Models

The Dynamic Stochastic General Equilibrium (DSGE) modeling [18] is a well established methodology for macroeconomics. It not only takes the dominant place in academia but also plays a vital role in policy making of central banks. Based on the general equilibrium theory and microeconomics principles, the DSGE modeling attempts to explain economic phenomenons through mathematical modeling, such as, economic growth, business cycles and effect of economic policy. As its name suggests, DSGE models are dynamic which implies the property that the macroeconomic variables evolve over time, stochastic which indicates the existence of various random shocks involved in the economy, general equilibrium which illustrates the interrelation and mutual influence between economic entities and economic variables.

The cornerstone of DSGE modeling lays on the General Equilibrium theory which is first developed by the French economist Leon Walras in the late 19th century [37]. Specifically speaking, General Equilibrium theory aims to explain the behavior of supply, demand and price among a whole economic system of many interacting markets such as households, firms, banks and government, and it believes that an overall general equilibrium will be finally reached by the interaction of demand and supply.

Overall, the DSGE modeling can be mainly used in three aspects: story-telling, policy Experiments and forecasting [6]. The Story-telling refers to the exploration and discussion on the interaction and linkage relationship between various economic variables by the construction of theoretical economic models that conform to reality. The Policy Experiments carried out by government heavily rely on a good model setting to investigate the impact of different economic policies on the real economic operation. The accuracy of economic forecast is one of the criteria to test whether the model is reasonable. Moreover, better economic forecasts can enable the central bank and the government to find economic problems as early as possible and adopt appropriate economic policies in a timely manner, so as to better stabilize the economy.

With the great improvement of DSGE modeling on model setting, solution technique and parameter estimation, central banks began to widely apply DSGE models in policy analysis and economic forecasting. DSGE models can provide a strong coherent framework for policy discussion and analysis, help identify the sources of economic fluctuations, assess the potential effects of different policies, and have predictive power comparable to that of traditional macro-econometric models. Table 5 lists the most recent DSGE models used by

Table 5: DSGE models used by Central Banks

Bank	DSGE Model
Federal Reserve Board	EDO, SIGMA
Bank of Canada	ToTEM II, BoC-GEM
European Central Bank	NAWM II, EAGLE-FLI
Bank of England	COMPASS

some central banks [36].

4.1.2 A Real Business Cycle Model

A representative DSGE model is known as Real Business Cycle (RBC) model proposed by [22] which states that the fluctuations of business cycle can be explained to a large extent by real shocks and changes in productivity. It tries to explain economic fluctuations without taking currency into account, or even without the existence of monetary authorities. An example of RBC model used by [11] is given to illustrate the general framework of DSGE modeling and how we can associate it with the problem of parameter estimation on the non-linear state-space model.

To be specific, this model aims to solve a representative household problem which seeks to find the optimal household behaviour on consumption c and labour n to maximize the expected discounted stream of utility,

$$\underset{c_t,n_t}{MAX}\left\{U = E_0\sum_{t=0}^{\infty}\beta^t\frac{(c_t^{\varphi}(1-n_t)^{1-\varphi})^{1-\phi}}{1-\phi}\right\} \tag{4.1.1}$$

where β is the subjective discount factor of household, ϕ is the degree of relative risk aversion and φ is the relative importance of c_t and l_t in determining the utility at time period t. The

solution of this maximum problem leads to the following equilibrium conditions:

$$\begin{aligned}
\left(\frac{1-\varphi}{\varphi}\right)\frac{c_t}{l_t} &= (1-\alpha)z_t\left(\frac{k_t}{n_t}\right)^{\alpha} \\
c_t^{\kappa}l_t^{\lambda} &= \beta E_t\{*\} \\
\{*\} &= \left\{\left(1+\frac{g}{1-\alpha}\right)^{\kappa} c_{t+1}^{\kappa}l_{t+1}^{\lambda}\left[\alpha z_{t+1}\left(\frac{n_{t+1}}{k_{t+1}}\right)^{1-\alpha}+(1-\delta)\right]\right\} \\
x_t &= z_t k_t^{\alpha} n_t^{1-\alpha} \\
x_t &= c_t + i_t \\
\left(1+\frac{g}{1-\alpha}\right)k_{t+1} &= i_t + (1-\delta)k_t \\
1 &= n_t + l_t \\
log(z_t) &= (1-\rho)log(z_0) + \rho log(z_{t-1}) + \epsilon_t,\ \epsilon_t \sim N(0,\sigma_{\epsilon}^2) \\
\kappa &= \varphi(1-\varphi)-1 \\
\lambda &= (1-\varphi)(1-\phi)
\end{aligned} \tag{4.1.2}$$

From these equilibrium conditions, we can determine the observation to be $y_t = (x_t, c_t, i_t, n_t)$ and the measurement density $f(y_t|s_t)$ is given by the following equations,

$$\begin{aligned}
x_t &= z_t k_t^{\alpha} n(k_t, z_t)^{1-\alpha} + u_{xt},\ u_{xt} \sim N(0,\sigma_x^2) \\
c_t &= c(k_t, z_t) + u_{ct},\ u_{ct} \sim N(0,\sigma_c^2) \\
i_t &= i(k_t, z_t) + u_{it},\ u_{it} \sim N(0,\sigma_i^2) \\
n_t &= n(k_t, z_t) + u_{nt},\ u_{nt} \sim N(0,\sigma_n^2)
\end{aligned} \tag{4.1.3}$$

which can be summarized as,

$$f(y_t|s_t) \sim N_4\left(\mu(s_t), \Omega_y\right) \tag{4.1.4}$$

where Ω_y is a diagonal covariance matrix with diagonal elements $(\sigma_x^2, \sigma_c^2, \sigma_i^2, \sigma_n^2)$ and $\mu(s_t)$ can be directly found from above equations. Moreover, we can find the state variables to be $s_t = (z_t, k_t)$ and the transition density $f(s_t|s_{t-1})$ is given by the following equations,

$$\begin{aligned}
log(z_t) &= (1-\rho)log(z_0) + \rho log(z_{t-1}) + \epsilon_t,\ \epsilon_t \sim N(0,\sigma_{\epsilon}^2) \\
\left(1+\frac{g}{1-\alpha}\right)k_t &= i(k_{t-1}, z_{t-1}) + (1-\delta)k_{t-1}
\end{aligned} \tag{4.1.5}$$

which can be summarized as,

$$f(s_t|s_{t-1}) \sim N_2\left(Rs_{t-1}, \Omega_s\right) \tag{4.1.6}$$

Here, $R = (A, 0)$ and Ω_s is a diagonal covariance matrix with diagonal elements $(\sigma_z^2, 0)$ where A and σ_z^2 can be directly found from 4.1.5. The objective is to apply some methods such as maximum likelihood approach to find the estimate of parameters $(\varphi, \alpha, \rho, \delta, \lambda, \kappa, \beta, \phi)$. Since

this is a parameter estimation problem on the non-linear state space model, we can directly apply the procedures and algorithms that are proposed in previous chapters.

4.1.3 Linear Rational Expectation Models

For the purpose of illustration on the methods and techniques developed in previous chapters about likelihood evaluation and parameter estimation, we will use a simple RBC model - linear rational expectation models to demonstrate and compare the performance of those methodologies. The linear rational expectation models are originally developed by [28] and have been widely used in the field of macroeconomics. These models are based on rational expectation theory, also known as economics rational expectation hypothesis which states the fact that people always keep rational when they make the prediction on some economic phenomenon, such as the market price, and they always make the most of the information to take actions so that they can avoid systematic error. This means that all errors are random and generally, people's rational expectation is just equal to the statistical expectation.

To be specific, the model we use in this chapter is

$$\underset{c_t,k_{t+1}}{MAX}\left\{E_t\sum_{t=0}^{\infty}\beta^t ln(c_t)\right\} \tag{4.1.7}$$

subject to

$$c_t + k_{t+1} = (1-\tau_t)R_t k_t + Z_t + \Pi_t \tag{4.1.8}$$

where c is the consumption, k is capital, R is return to capital, τ_t is income tax rate, Z_t is government transfer and Π_t is the profit. The detailed model solution process is skipped here and the state space form is given by:

$$\hat{k}_t = \alpha\hat{k}_{t-1} + \hat{a}_{t-1} - \frac{\alpha\beta(1-\tau)\rho^\tau}{1-\alpha\beta(1-\tau)\rho^\tau}v_1\hat{\tau}_{t-1} \tag{4.1.9}$$

$$v_1 = \frac{1-\alpha\beta(1-\tau)}{\alpha\beta(1-\tau)} \times \frac{\tau}{1-\tau} \tag{4.1.10}$$

$$\hat{a}_t = \rho^a\hat{a}_{t-1} + \epsilon_t^a, \text{ where } \epsilon_t^a \sim N(0,\sigma_a^2) \tag{4.1.11}$$

$$\hat{\tau}_t = \rho^\tau\hat{\tau}_{t-1} + \epsilon_t^\tau, \text{ where } \epsilon_t^\tau \sim N(0,\sigma_\tau^2) \tag{4.1.12}$$

$$y_{t,1} = \hat{k}_t + \epsilon_{y1}, \text{ where } \epsilon_{y1} \sim N(0,\sigma_{y1}^2) \tag{4.1.13}$$

$$y_{t,2} = \hat{\tau}_t + \epsilon_{y2}, \text{ where } \epsilon_{y2} \sim N(0,\sigma_{y2}^2) \tag{4.1.14}$$

where $s_t = (\hat{a}_t, \hat{\tau}_t, \hat{k}_t)$ are state variables, $y_t = (y_{t,1}, y_{t,2})$ are observations, τ is the steady state income tax rate, β is the discount factor, α is the output elasticity of capital, ρ^a is the persistence in technology and ρ^τ is the persistence in tax rate. Equations 4.9 - 4.12 are transition densities. Equations 4.13 and 4.14 are measurement densities. All the error terms are Gaussian distributed. We shall notice that the state space form in this RBC model has a singular transition density $f(s_t|s_{t-1})$ in which $\hat{k}_t$ is a deterministic function of states from previous time point. This singular state space form is quite common in DSGE modeling.

As a result, we also need to make some modifications in EIS estimation to deal with this singular form.

4.2 State Space Model with Singular Transition

In DSGE models, it is quite common to see the state space form with singular transition $f(s_t|s_{t-1})$ such as equations 4.9-4.12. For this special case, David and Roman [5] illustrate the fact that this singular transition form can help us effectively reduce the dimensionality in EIS procedure so that it will take less time to reach convergence.

First of all, suppose we have a singular transition density $f(s_t|s_{t-1})$ and the state variable s_t can be divided into $s_t = (p_t, q_t)$, then, the transition of state variables can be expressed by a non-singular transition density $f(p_t|s_{t-1})$ and the following identity:

$$q_t = \phi(p_t, s_{t-1}) \tag{4.2.1}$$

which states the fact that q_t is directly determined by p_t and s_{t-1}. As we know, the conditional likelihood function $f(y_t|Y_{t-1})$ at time point t is given by:

$$\begin{aligned} f(y_t|Y_{t-1}) &= \int\int f(y_t|s_t)f(s_t|s_{t-1})f(s_{t-1}|Y_{t-1})ds_t ds_{t-1} \\ &= \int\int \frac{f(y_t|s_t)f(s_t|s_{t-1})f(s_{t-1}|Y_{t-1})}{g_t(s_t, s_{t-1})} g_t(s_t, s_{t-1})ds_t ds_{t-1} \end{aligned} \tag{4.2.2}$$

$$\varphi_t(s_t, s_{t-1}) = \frac{f(y_t|s_t)f(s_t|s_{t-1})f(s_{t-1}|Y_{t-1})}{g_t(s_t, s_{t-1})} \tag{4.2.3}$$

And in EIS procedure, we aim to find an optimal sampler $g_t(s_t, s_{t-1})$ among exponential family such that the variance of ratio $\varphi_t(s_t, s_{t-1})$ can be minimized. We still can just follow the EIS procedure introduced before even if the transition density $f(s_t|s_{t-1})$ is singular. However, if the dimension of state variable is high, then, it will takes more time for EIS procedure to reach convergence. Fortunately, the singular transition can effectively reduce this dimensionality, and the conditional likelihood function can be rewritten as,

$$\begin{aligned} f(y_t|Y_{t-1}) &= \int\int f(y_t|s_t)J(s_t, p_{t-1}) \left[f(p_t|s_{t-1})f(s_{t-1}|Y_{t-1})\right]\Big|_{q_{t-1}=\psi(s_t,p_{t-1})} ds_t dp_{t-1} \\ &= \int\int \frac{f(y_t|s_t)J(s_t, p_{t-1}) \left[f(p_t|s_{t-1})f(s_{t-1}|Y_{t-1})\right]\Big|_{q_{t-1}=\psi(s_t,p_{t-1})}}{g_t(s_t, p_{t-1})} g_t(s_t, p_{t-1})ds_t dp_{t-1} \end{aligned} \tag{4.2.4}$$

$$J(s_t, p_{t-1}) = \left|\left| \frac{\partial}{\partial q_t'}\psi(s_t, p_{t-1}) \right|\right| \tag{4.2.5}$$

where $q_{t-1} = \psi(s_t, p_{t-1})$ is derived from equation 4.14. As we can see, with this singular transition form, we no longer need to find the optimal sampler $g_t(s_t, s_{t-1})$ over two state variables any more. Instead, we just need to find the optimal sampler over s_t and p_{t-1} which

effectively reduce the convergence time in EIS procedure. The filtering density now is given by:

$$f(s_t|Y_t) = \frac{g_t(s_t)}{f(y_t|Y_{t-1})} \int w_t(s_t, p_{t-1}) g_t(p_{t-1}|s_t) dp_{t-1} \tag{4.2.6}$$

$$g_t(s_t, p_{t-1}) = g_t(s_t) g_t(p_{t-1}|s_t) \tag{4.2.7}$$

$$w_t(s_t, p_{t-1}) = \frac{f(y_t|s_t) J(s_t, p_{t-1}) \left[f(p_t|s_{t-1}) f(s_{t-1}|Y_{t-1}) \right] \Big|_{q_{t-1}=\psi(s_t, p_{t-1})}}{g_t(s_t, p_{t-1})} \tag{4.2.8}$$

Under the constant-weight approximation, we can have,

$$f(s_t|Y_t) \approx g_t(s_t) \tag{4.2.9}$$

Under our proposed method introduced in Chapter 2, this filtering density can be approximated by:

$$f(s_t|Y_t) \approx \frac{g_t(s_t)}{\hat{f}(y_t|Y_{t-1})} \frac{1}{N} \sum_{i=1}^{N} \hat{w}_t(s_t, p_{t-1}^i(s_t)) \tag{4.2.10}$$

$$\hat{w}_t(s_t, p_{t-1}) = \frac{f(y_t|s_t) J(s_t, p_{t-1}) \left[f(p_t|s_{t-1}) g_{t-1}(s_{t-1}) \right] \Big|_{q_{t-1}=\psi(s_t, p_{t-1})}}{g_t(s_t, p_{t-1})} \tag{4.2.11}$$

where $\{p_{t-1}^i(s_t)\}_{i=1}^N$ are the draws taken from $g_t(p_{t-1}|s_t)$.

In the next section, we will use linear rational expectation model to demonstrate and compare the performance of the methods and techniques we introduced before on likelihood evaluation and parameter estimation.

4.3 The EM Algorithm

The EM algorithm is an iterative method designed for the models involving hidden or missing variables to achieve the ML parameter estimation without evaluating the likelihood function. The EM iteration alternates between implementing an Expectation (E) step, in which a Q function is built up for the evaluation of the expectation of the log-likelihood function involving both observations and hidden variables, and a Maximization (M) step, in which optimal parameters will be found such that this Q function can be maximized.

In this chapter, we aim to achieve maximum likelihood parameter estimation for linear rational expectation model by using EM algorithm with two different sampling methods: Forward Filtering Backward Sampling (EM-FFBS) and our developed EIS-based sampling method (EM-EIS). For EM-FFBS and our proposed EM-EIS, both methods aim to solve the Q function in EM algorithm by taking draws from $f_{\theta_{k-1}}(s_{0:T}|y_{0:T})$:

$$Q(\theta_{k-1}, \theta) = \int ... \int log \left\{ f_\theta(s_{0:T}, y_{0:T}) \right\} \times f_{\theta_{k-1}}(s_{0:T}|y_{0:T}) ds_{0:T} \tag{4.3.1}$$

$$f_\theta(s_{0:T}, y_{0:T}) = f(s_0)\prod_{t=1}^{T} f(s_t|s_{t-1})\prod_{t=0}^{T} f(y_t|s_t) \tag{4.3.2}$$

And the estimation of this Q function is given by:

$$\hat{Q}(\theta_{k-1}, \theta) = \frac{1}{N}\sum_{i=1}^{N} log\left\{f_\theta(s^i_{0:T}, y_{0:T})\right\} \tag{4.3.3}$$

The EM-FFBS and our proposed EM-EIS methods only differ in the way they take samples from this smoothing density $f_{\theta_{k-1}}(s_{0:T}|y_{0:T})$. Specifically, this smoothing density can be written as:

$$f(s_{0:T}|y_{0:T}) = f(s_T|y_{0:T})\prod_{t=0}^{T-1} f(s_t|y_{0:t}, s_{t+1}) \tag{4.3.4}$$

$$f(s_t|y_{0:t}, s_{t+1}) \propto f(s_{t+1}|s_t)f(s_t|y_{0:t}) \tag{4.3.5}$$

In the EM-FFBS method, we firstly run the Particle Filter algorithm from time $t = 0$ to T, storing the weight values $\{W^i_t\}^N_{i=1}$ and particles $\{s^i_t\}^N_{i=1}$ at each time point. This step is known as Forward Filtering. After that, we need to take draws from $\hat{f}(s_T|y_{0:T})$ which is a probability mass function in last time point consisting of $\{W^i_T\}^N_{i=1}$ and $\{s^i_T\}^N_{i=1}$ we found from Forward Filtering step. Then, for $t = T-1, T-2, ..., 1, 0$, we need to take draws from $f(s_t|y_{0:t}, s_{t+1})$ which can be approximated by the following probability mass function:

$$\hat{f}(s_t|y_{0:t}, s_{t+1}) = \sum_{i=1}^{N} \frac{W^i_n f(s_{t+1}|s^i_t)}{\sum_{i=1}^{N} W^i_n f(s_{t+1}|s^i_t)}\delta_{s^i_t}(s_t) \tag{4.3.6}$$

This step is known as Backward Sampling and it requires $\mathcal{O}(MNT)$ operations to generate full path particles $\{s^i_{0:T}\}^M_{i=1}$, as taking draws from 4.3.6 costs $\mathcal{O}(N)$ operations and sample size is set as M.

In our proposed EM-EIS method, instead of approximating filtering density $f(s_t|y_{0:t})$ by probability mass function, we approximate it by using some continuous function from Constant-Weight EIS procedure. So, first of all, we just run EIS algorithm from time $t = 0$ to T, storing the density function $g_t(s_t)$ which can be obtained from Constant-Weight method at each time point. This step is known as Forward EIS. After that, we need to take draws from $g_T(s_T)$ which is the approximation of filtering density $f(s_T|y_{0:T})$ at the last time point. Then, for $t = T-1, T-2, ..., 1, 0$, we need to take draws from $f(s_t|y_{0:t}, s_{t+1})$. Instead of performing discrete approximation, we can actually rewrite equation 4.3.5 as follows:

$$\begin{aligned} f(s_t|y_{0:t}, s_{t+1}) &\propto f(s_{t+1}|s_t)g(s_t|y_{0:t}) \\ &\propto \frac{f(s_{t+1}|s_t)g(s_t|y_{0:t})}{q(s_t|a)}q(s_t|a) \end{aligned} \tag{4.3.7}$$

where $f(s_t|y_{0:t})$ is replaced by $g(s_t|y_{0:t})$ and we can just implement EIS procedure there to find an optimal sampler $q(s_t|\hat{a})$ among some exponential family to minimize the variance of

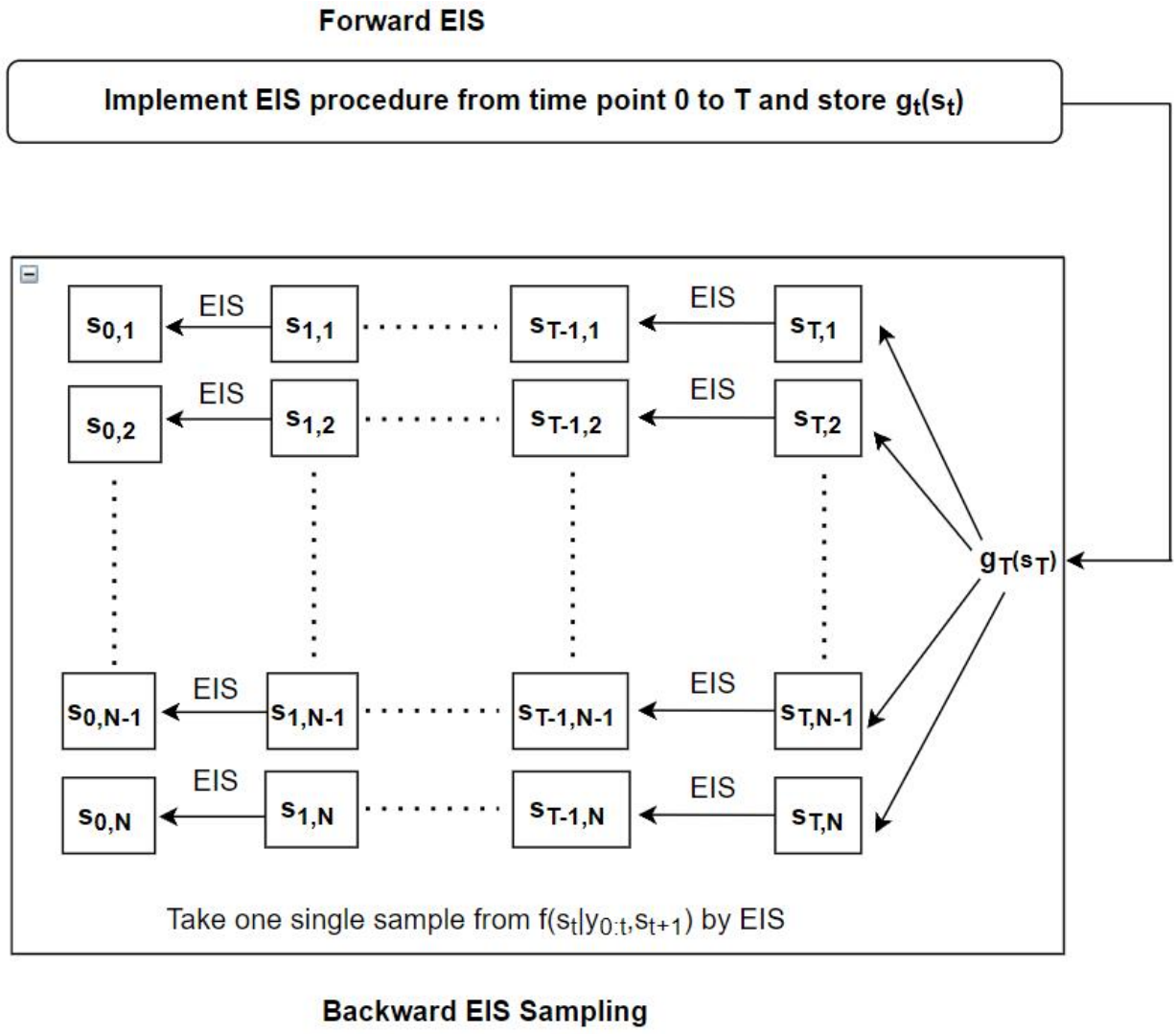

Figure 5: Flowchart of our proposed Forward-Backward EIS Sampling

$\frac{f(s_{t+1}|s_t)g(s_t|y_{0:t})}{q(s_t)}$, and by Constant-Weight principle, we can just have:

$$f(s_t|y_{0:t}, s_{t+1}) \approx q(s_t|\hat{a}) \tag{4.3.8}$$

So, we just need to take draws from $q(s_t|\hat{a})$ as the sample for $f(s_t|y_{0:t}, s_{t+1})$. This step is known as Backward EIS sampling and it only requires $\mathcal{O}(METT)$ operations to generate full path particles $\{s^i_{0:T}\}^M_{i=1}$ where E is the EIS computational complexity and usually less than 6. Moreover, Figure 5 is the flowchart which gives clearer demonstration to our proposed Forward-Backward EIS sampling process. From this flowchart, we can see that it is quite time consuming to implement EIS procedure for each single draw. To remedy this, in chapter 3, we proposed a technique known as Fast-Sampling which aims to partition these particles into several groups and use their group mean value for EIS procedure to generate samples. Since the model used in this chapter is no longer one dimensional, we adopt the K-means

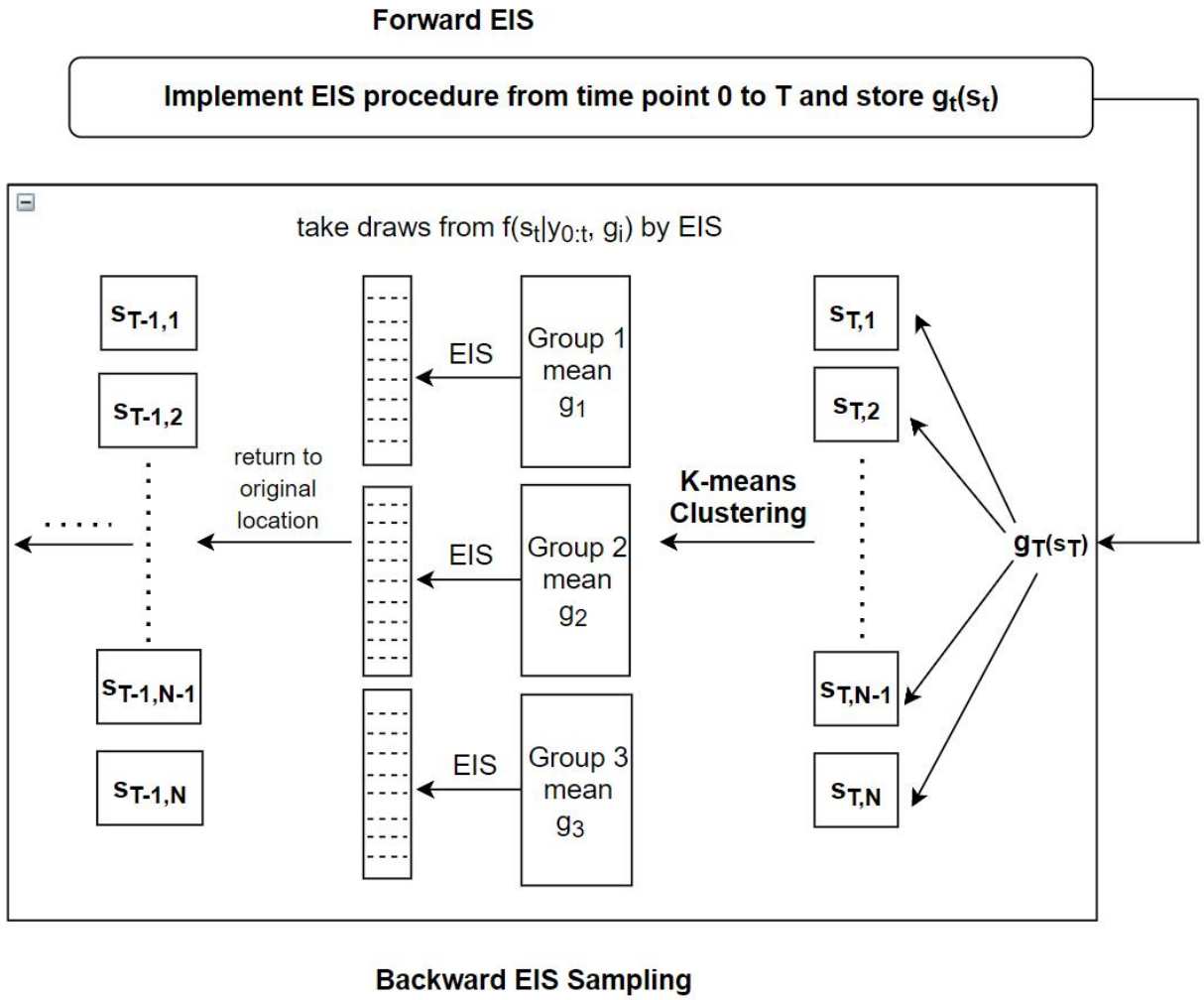

Figure 6: Flowchart of Fast-Sampling Technique

clustering method to implement this partition and Figure 6 demonstrates this technique in clearer way.

4.4 Numerical Results

In this section, we will use linear rational expectation model to demonstrate the performance of the methods and techniques we introduced in Chapter 2 and 3 in both likelihood evaluation and parameter estimation. The true parameter values for this model are set up as follows,

$$\begin{aligned} \alpha = 0.33, \tau = 0.25, \beta = 0.99 \\ \rho^a = 0.85, \rho^\tau = 0.75 \\ \sigma_a = 0.01, \sigma_\tau = 0.01, \sigma_{y1} = 0.1, \sigma_{y1} = 0.1 \end{aligned} \tag{4.4.1}$$

Let $y_t = (y_{t,1}, y_{t,2})$ and $s_t = (p_t, q_t)$ where $p_t = (\hat{a}_t, \hat{\tau}_t)$ and $q_t = \hat{k}_t$, then the distribution

Table 6: Simulation Result of Log-likelihood Estimates for DSGE model

	Sample Size	True Likelihood	MC Mean	Absolute Bias	NSE
BPF	100,000	380.9859	380.3154	0.6705	0.8188
CW-EIS	100	-	380.7585	0.2274	0.6078
PC-EIS	100	-	380.8976	0.0883	0.0648

form of Linear Rational Expectation model is given as follows:

$$y_t|s_t \sim N_2\left(\begin{bmatrix}\hat{k}_t\\ \hat{\tau}_t\end{bmatrix}, \begin{bmatrix}\sigma_{y1}^2 & 0\\ 0 & \sigma_{y2}^2\end{bmatrix}\right) \tag{4.4.2}$$

$$\hat{k}_t = \alpha\hat{k}_{t-1} + \hat{a}_{t-1} - \frac{\alpha\beta(1-\tau)\rho^\tau}{1-\alpha\beta(1-\tau)\rho^\tau} \times \frac{1-\alpha\beta(1-\tau)}{\alpha\beta(1-\tau)} \times \frac{\tau}{1-\tau}\hat{\tau}_{t-1} \tag{4.4.3}$$

$$p_t|s_{t-1} \sim N_2\left(\begin{bmatrix}\rho^a\hat{a}_{t-1}\\ \rho^\tau\hat{\tau}_{t-1}\end{bmatrix}, \begin{bmatrix}\sigma_a^2 & 0\\ 0 & \sigma_\tau^2\end{bmatrix}\right) \tag{4.4.4}$$

$$s_0 \sim N_3\left(\begin{bmatrix}0\\0\\0\end{bmatrix}, \begin{bmatrix}0.1 & 0 & 0\\ 0 & 0.1 & 0\\ 0 & 0 & 0.1\end{bmatrix}\right) \tag{4.4.5}$$

4.4.1 Likelihood Evaluation

Since the linear rational expectation model is linear and Gaussian distributed, then, its true likelihood can still be evaluated by using Fast Kalman Filter method. For this part, we will compare three different methods on likelihood estimation: Bootstrap Particle Filter (BPF), EIS with constant-weight approximation method (CW-EIS), EIS with our proposed weight computation method (PC-EIS).

Specifically speaking, a sequence of observations $\{y_t\}_{t=0}^T$ with $T = 200$ are generated by simulating this model with true parameters shown above. For the Boost Particle Filter, we will use $N = 100,000$ as the sample size, the transition density $f(p_t|s_{t-1})$ will serves as the sampler and the weight values can be just computed by measurement density $f(y_t|s_t)$. For both CW-EIS and PC-EIS, the sample size is set as 100 and common random number will be used for the purpose of fast convergence in EIS step. The replication time is set as 50 for all these three methods. The Monte Carlo (MC) mean, Numerical Standard Error (NSE), Absolute Bias, Mean Square Error and CPU time of these log-likelihood estimates by using BPF, CW-EIF and PC-EIS are reported in Table 6 and 7. Moreover, Figure 7 gives clearer MSE comparison between these three methods and Figure 8 also shows the variability of these log-likelihood estimates for a more intuitive comparison on the efficiency between BPF, CW-EIS and PC-EIS.

The first thing we note in Table 6 is that BPF has the largest absolute bias. However, we also need to note the fact that BPF is actually an unbiased method in likelihood estimation while the estimation from EIS methods is biased. If we have a look on the formula of EIS

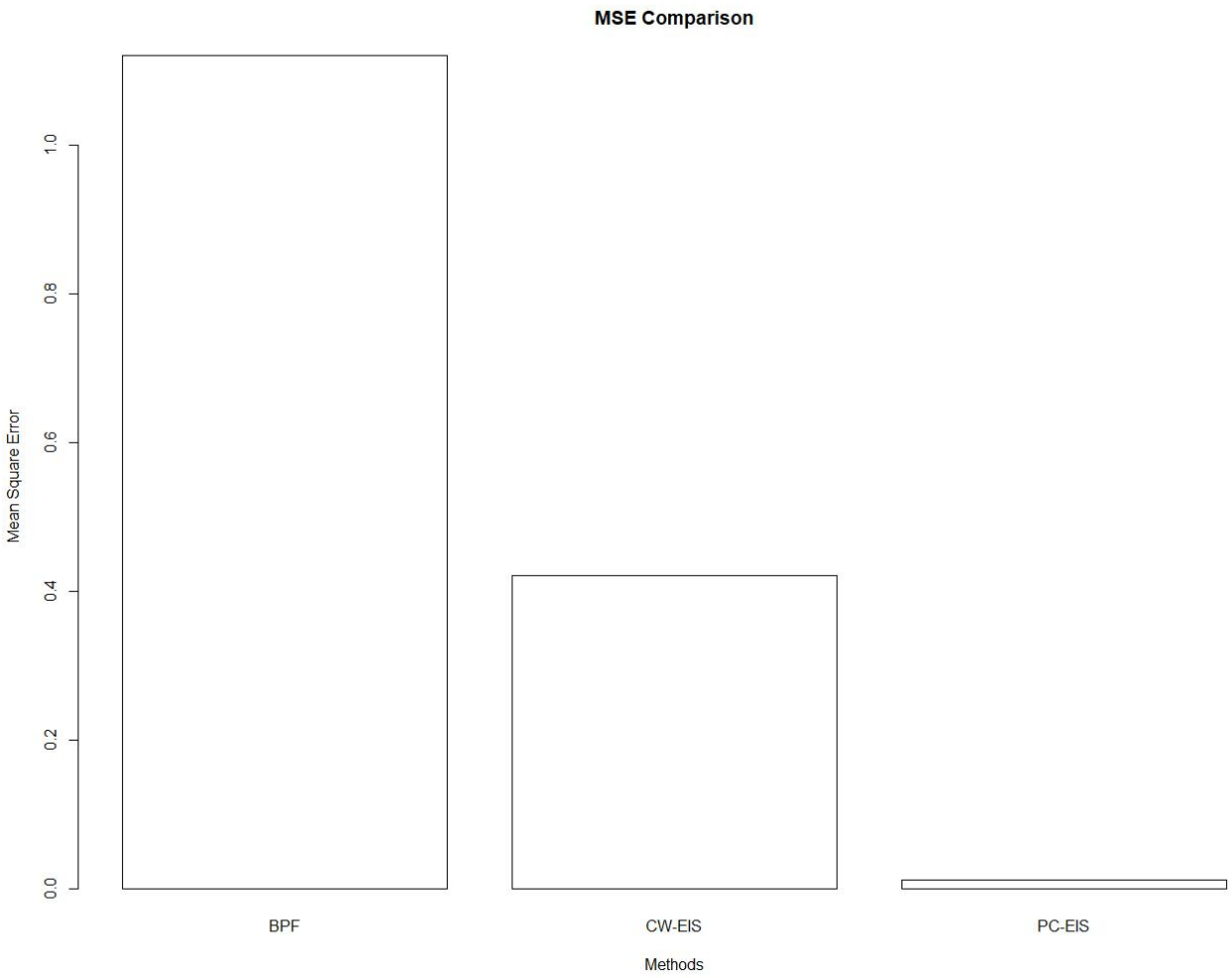

Figure 7: MSE Comparison of BPF, CW-EIS, PC-EIS

likelihood, we can have,

$$\hat{f}(y_t|Y_{t-1}) = \frac{1}{N}\sum_{i=1}^{N} \frac{f(y_t|s_t^i)J(s_t^i, p_{t-1}^i)\left[f(p_t^i|s_{t-1}^i)\hat{f}(s_{t-1}^i|Y_{t-1})\right]\Big|_{q_{t-1}=\psi(s_t,p_{t-1})}}{g_t(s_t^i, p_{t-1}^i)} \tag{4.4.6}$$

Table 7: Mean Square Error and CPU time of DSGE model

	MSE	**CPU Time**
BPF	1.1200	216 min
CW-EIS	0.4212	9 min
PC-EIS	0.0120	33 min

And its expectation is given by,

$$
\begin{aligned}
E\left(\hat{f}(y_t|Y_{t-1})\right) &= E\left\{\frac{f(y_t|s_t^i)J(s_t^i,p_{t-1}^i)\left[f(p_t^i|s_{t-1}^i)\hat{f}(s_{t-1}^i|Y_{t-1})\right]\Big|_{q_{t-1}=\psi(s_t,p_{t-1})}}{g_t(s_t^i,p_{t-1}^i)}\right\} \\
&= \int\int \frac{f(y_t|s_t)J(s_t,p_{t-1})\left[f(p_t|s_{t-1})\hat{f}(s_{t-1}|Y_{t-1})\right]\Big|_{q_{t-1}=\psi(s_t,p_{t-1})}}{g_t(s_t,p_{t-1})} g_t(s_t,p_{t-1})ds_t dp_{t-1} \\
&= \int\int f(y_t|s_t)J(s_t,p_{t-1})\left[f(p_t|s_{t-1})\hat{f}(s_{t-1}|Y_{t-1})\right]\Big|_{q_{t-1}=\psi(s_t,p_{t-1})} ds_t dp_{t-1} \\
&\neq \int\int f(y_t|s_t)J(s_t,p_{t-1})\left[f(p_t|s_{t-1})f(s_{t-1}|Y_{t-1})\right]\Big|_{q_{t-1}=\psi(s_t,p_{t-1})} ds_t dp_{t-1} \\
&= f(y_t|Y_{t-1})
\end{aligned}
\tag{4.4.7}
$$

The continuously approximated filtering density $\hat{f}(s_{t-1}|Y_{t-1})$ in EIS method is the reason for the bias estimation since $\hat{f}(s_{t-1}|Y_{t-1}) \neq f(s_{t-1}|Y_{t-1})$. However, we should also notice that this bias generated by CW-EIS is quite small and we can ignore it to some extent. Moreover, by using our proposed weight computation method, this bias can be further reduced by 61.17% which demonstrates the superiority of our PC-EIS method. In aspect of Numerical Standard Error (NSE), in comparison with BPF of 100,000 sample size, both CW-EIS and PC-EIS can achieve lower NSE with only 100 draws and the EIS procedure with our proposed method can further reduce this NSE by almost 89.34% which sufficiently demonstrate the great efficiency we gained from implementing the EIS variance minimization step and filtering density weight computation.

The comparison on Mean Square Errors (MSE) shown in Figure 7 between BPF, CW-EIS and PC-EIS further illustrate the fact that EIS procedure is superior over BPF in likelihood evaluation. And also, we should note that the MSE of PC-EIS is much smaller than that of CW-EIS. Therefore, our proposed filtering density weight computation method can help us further exploit the potentiality of EIS algorithm and realize more efficiency gains than the EIS method by Constant-Weight approximation. If we take a look on the result table 2 in Chapter 2 where a very simple one-dimensional state space model is used for simulation, we can note that the MSE results in Chapter 2 are too small to be reliable although PC-EIS is better than CW-EIS. In this chapter, we used multi-dimensional DSGE model with more

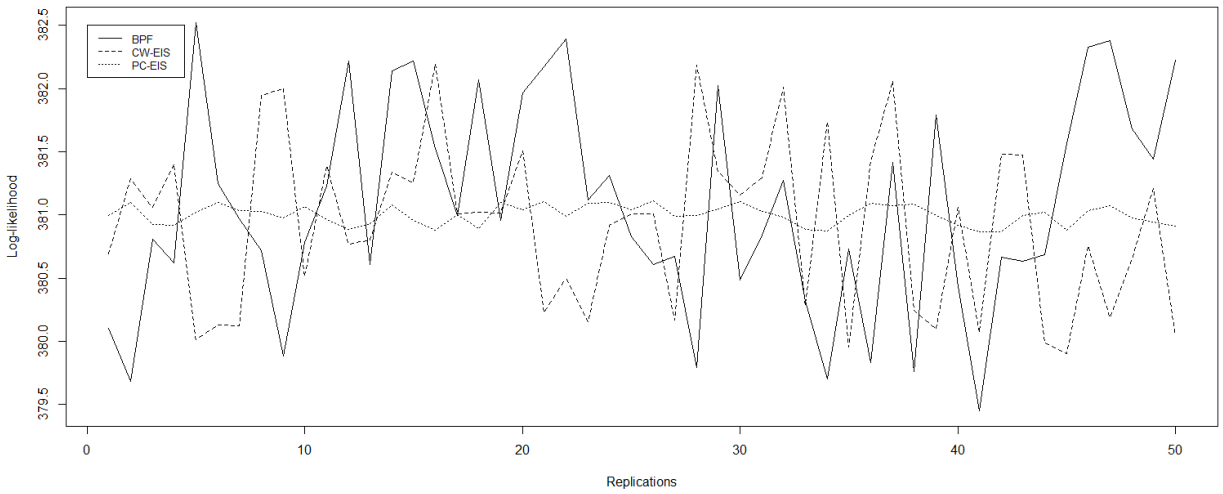

Figure 8: Log-likelihood Variability of BPF, CW-EIS, PC-EIS

complicated structure leading to more reliable MSE results and this illustrate the fact that the superiority of PC-EIS will become clearer in state space model with higher dimensionality and more complicated structure.

4.4.2 Parameter Estimation

In this part, we will explore the maximum likelihood parameter estimation on the Linear Rational Expectation model with five different methods: Gradient Ascent with Particle Filter (GA-PF), Gradient Ascent with EIS (GA-EIS), Expectation Maximization with Forward Filtering Backward Sampling (EM-FFBS), Expectation Maximization with our developed EIS-based method (EM-EIS) and EM-EIS with our proposed Fast-Sampling Technique (EM-EIS-FS).

For GA-PF and GA-EIS, the likelihood value is estimated by Bootstrap Particle Filters and EIS method respectively, and the score vector can be just evaluated by finite difference method. For EM-FFBS and our proposed EM-EIS, both methods aim to solve the Q function in EM algorithm by taking draws from $f_{\theta_{k-1}}(s_{0:T}|y_{0:T})$ in different ways.

For the simulation, specifically, a sequence of observations $\{y\}_{t=0}^{T}$ of $T = 300$ are generated and this generation process will be repeated for 50 replication times. So, a total of 50 observation sequences, denoted as $\{y_{0:T}^{i}\}_{i=1}^{50}$, are generated. In GA-PF and GA-EIS, the sample sizes for likelihood estimation are set as $100,000$ and 100 respectively, and the score vector is evaluated by finite difference method. In EM-FFBS method, the sample sizes for Forward Filtering and Q function estimation are set as $N = 1000$ and $M = 100$ respectively. In EM-EIS, the sample sizes for EIS variance minimization step and Q function estimation are both set as 100. In EM-EIS-FS, we use K-means clustering to partition these draws and the number of clusters is set to $G = 25, 20, 10$. For the gradient ascent optimization, we used R function 'optim' in which 'L-BFGS-B' method is adopted since this method can set up both lower bound and upper bound for our estimated parameters. The Monte Carlo mean, Absolute Bias, Numerical Standard Error (NSE), Mean Square Error (MSE) and

Table 8: Computational Time (min) of 50 replications of DSGE model

GA-PF		1160
GA-EIS		48
EM-FFBS		550
EM-EIS		336
EM-EIS-FS	G=25	124
	G=20	97
	G=10	36

computational time of these methods are reported in Table 8 and 4.4.2.

From the results shown in these tables, we can firstly notice that the EM-EIS method has the best performance in parameter estimation which demonstrates the superiority of EIS variance minimization technique. And overall, we can also see that the EM algorithm (EM-FFBS and EM-EIS) outperforms Gradient Ascent algorithm (GA-PF and GA-EIS) except the α estimation with EIS-FFBS which has quite large numerical standard error. In the aspect of EM-EIS-FS, it has quite good performance when the number of clusters is 25. However, as this number decreases, its performance is getting worse even though the computational time is decreasing. This is caused by the loss of sample diversity and we must be very careful with the selection of cluster size to make a trade-off between estimation performance and computational time.

True $\alpha = 0.33$		MC mean	Abs Bias	NSEs	MSE
GA-PF		0.3809	0.0509	0.1192	0.0168
GA-EIS		0.3664	0.0364	0.1187	0.0154
EM-FFBS		0.3428	0.0128	0.1538	0.0238
EM-EIS		0.3352	0.0052	0.0262	0.0007
EM-EIS-FS	G=25	0.3912	0.0612	0.0878	0.0115
	G = 20	0.4632	0.1332	0.1763	0.0488
	G = 10	0.1580	0.172	0.6324	0.4295
True $\rho^a = 0.85$		MC mean	Abs Bias	NSEs	MSE
GA-PF		0.8157	0.0343	0.0963	0.0104
GA-EIS		0.8095	0.0405	0.0894	0.0096
EM-FFBS		0.8366	0.0134	0.0499	0.0027
EM-EIS		0.8518	0.0018	0.0361	0.0013
EM-EIS-FS	G=25	0.7936	0.0564	0.1036	0.0139
	G = 20	0.7460	0.1040	0.2050	0.0528
	G = 10	0.6396	0.2104	0.4852	0.2797
True $\rho^\tau = 0.75$		MC mean	Abs Bias	NSEs	MSE
GA-PF		0.7468	0.0032	0.1008	0.0102
GA-EIS		0.7293	0.0207	0.1112	0.0128
EM-FFBS		0.7489	0.0011	0.0399	0.0016
EM-EIS		0.7806	0.0306	0.0516	0.0036
EM-EIS-FS	G=25	0.8391	0.0891	0.0863	0.0154
	G = 20	0.8758	0.1258	0.1009	0.0260
	G = 10	0.6218	0.1282	0.4573	0.2256

Table 9: The Simulation Result of DSGE model

REFERENCES

[1] Hajime Akashi and Hiromitsu Kumamoto, *Random sampling approach to state estimation in switching environments*, Automatica **13** (1977), no. 4, 429–434.

[2] B Bobrovsky and Moshe Zakai, *A lower bound on the estimation error for markov processes*, IEEE Transactions on Automatic Control **20** (1975), no. 6, 785–788.

[3] Z. Chen and E. N. Brown, *State space model*, Scholarpedia **8** (2013), no. 3, 30868, revision #189565.

[4] Piet De Jong, *The likelihood for a state space model*, Biometrika **75** (1988), no. 1, 165–169.

[5] David N DeJong, Roman Liesenfeld, Guilherme V Moura, Jean-François Richard, and Hariharan Dharmarajan, *Efficient likelihood evaluation of state-space representations*, Review of Economic Studies **80** (2013), no. 2, 538–567.

[6] Marco Del Negro and Frank Schorfheide, *Dsge model-based forecasting*, Handbook of economic forecasting, vol. 2, Elsevier, 2013, pp. 57–140.

[7] Arthur P Dempster, Nan M Laird, and Donald B Rubin, *Maximum likelihood from incomplete data via the em algorithm*, Journal of the Royal Statistical Society: Series B (Methodological) **39** (1977), no. 1, 1–22.

[8] Luc Devroye, *Nonuniform random variate generation*, Handbooks in operations research and management science **13** (2006), 83–121.

[9] Arnaud Doucet, Nando De Freitas, and Neil Gordon, *An introduction to sequential monte carlo methods*, Sequential Monte Carlo methods in practice, Springer, 2001, pp. 3–14.

[10] Arnaud Doucet and Adam M Johansen, *A tutorial on particle filtering and smoothing: Fifteen years later*, Handbook of nonlinear filtering **12** (2009), no. 656-704, 3.

[11] Jesús Fernández-Villaverde and Juan F Rubio-Ramírez, *Estimating dynamic equilibrium economies: linear versus nonlinear likelihood*, Journal of Applied Econometrics **20** (2005), no. 7, 891–910.

[12] George E Forsythe and Wolfgang R Wasow, *Finite difference methods*, Partial Differential (1960).

[13] Simon J Godsill, Arnaud Doucet, and Mike West, *Monte carlo smoothing for nonlinear time series*, Journal of the american statistical association **99** (2004), no. 465, 156–168.

[14] Neil J Gordon, David J Salmond, and Adrian FM Smith, *Novel approach to nonlinear/non-gaussian bayesian state estimation*, IEE proceedings F (radar and signal processing), vol. 140, IET, 1993, pp. 107–113.

[15] JE Handschin, *Monte carlo techniques for prediction and filtering of non-linear stochastic processes*, Automatica **6** (1970), no. 4, 555–563.

[16] Johannes Edmund Handschin and David Q Mayne, *Monte carlo techniques to estimate the conditional expectation in multi-stage non-linear filtering*, International journal of control **9** (1969), no. 5, 547–559.

[17] Adam M Johansen and Arnaud Doucet, *A note on auxiliary particle filters*, Statistics & Probability Letters **78** (2008), no. 12, 1498–1504.

[18] Michel Juillard, *Dynamic stochastic general equilibrium models*, The Oxford Handbook of Computational Economics and Finance, Oxford University Press, 2018.

[19] Genshiro Kitagawa, *Monte carlo filter and smoother for non-gaussian nonlinear state space models*, Journal of computational and graphical statistics **5** (1996), no. 1, 1–25.

[20] Daphne Koller and Nir Friedman, *Probabilistic graphical models: principles and techniques*, MIT press, 2009.

[21] Augustine Kong, Jun S Liu, and Wing Hung Wong, *Sequential imputations and bayesian missing data problems*, Journal of the American statistical association **89** (1994), no. 425, 278–288.

[22] Finn E Kydland and Edward C Prescott, *Time to build and aggregate fluctuations*, Econometrica: Journal of the Econometric Society (1982), 1345–1370.

[23] Kuo-ching Liang, Xiaodong Wang, and Dimitris Anastassiou, *Bayesian basecalling for dna sequence analysis using hidden markov models*, IEEE/ACM Transactions on Computational Biology and Bioinformatics **4** (2007), no. 3, 430–440.

[24] Jun S Liu, *Monte carlo strategies in scientific computing*, Springer Science & Business Media, 2008.

[25] Jun S Liu, Rong Chen, and Tanya Logvinenko, *A theoretical framework for sequential importance sampling with resampling*, Sequential Monte Carlo methods in practice, Springer, 2001, pp. 225–246.

[26] Sheheryar Malik and Michael K Pitt, *Particle filters for continuous likelihood evaluation and maximisation*, Journal of Econometrics **165** (2011), no. 2, 190–209.

[27] Murdoch K McAllister, Ellen K Pikitch, Andre E Punt, and Ray Hilborn, *A bayesian approach to stock assessment and harvest decisions using the sampling/importance resampling algorithm*, Canadian Journal of Fisheries and Aquatic Sciences **51** (1994), no. 12, 2673–2687.

[28] John F Muth, *Rational expectations and the theory of price movements*, Econometrica: Journal of the Econometric Society (1961), 315–335.

[29] In Jae Myung, *Tutorial on maximum likelihood estimation*, Journal of mathematical Psychology **47** (2003), no. 1, 90–100.

[30] Lior Pachter, Marina Alexandersson, and Simon Cawley, *Applications of generalized pair hidden markov models to alignment and gene finding problems*, Journal of Computational Biology **9** (2002), no. 2, 389–399.

[31] Michael K Pitt, *Smooth particle filters for likelihood evaluation and maximisation*, Tech. report, 2002.

[32] Michael K Pitt and Neil Shephard, *Filtering via simulation: Auxiliary particle filters*, Journal of the American statistical association **94** (1999), no. 446, 590–599.

[33] Lawrence R Rabiner, *A tutorial on hidden markov models and selected applications in speech recognition*, Proceedings of the IEEE **77** (1989), no. 2, 257–286.

[34] Jean-Francois Richard and Wei Zhang, *Efficient high-dimensional importance sampling*, Journal of Econometrics **141** (2007), no. 2, 1385–1411.

[35] Donald B Rubin, *Comment: A noniterative sampling/importance resampling alternative to the data augmentation algorithm for creating a few imputations when fractions of missing information are modest: The sir algorithm*, Journal of the American Statistical Association **82** (1987), no. 398, 542–543.

[36] Camilo E Tovar, *Dsge models and central banks*, Economics **3** (2009), no. 1.

[37] Leon Walras, *Elements of pure economics*, Routledge, 2013.

[38] Greg Welch, Gary Bishop, et al., *An introduction to the kalman filter*, 1995.

[39] Mike West and Jeff Harrison, *Bayesian forecasting and dynamic models*, Springer Science & Business Media, 2006.

[40] Byung-Jun Yoon, *Hidden markov models and their applications in biological sequence analysis*, Current genomics **10** (2009), no. 6, 402–415.

[41] VS Zaritskii, VB Svetnik, and LI Šimelevič, *Monte-carlo technique in problems of optimal information processing*, Avtomatika i telemekhanika (1975), no. 12, 95–103.

VITA

Shiteng Yang

Candidate for the Degree of

Doctor of Philosophy

Dissertation: MAXIMUM LIKELIHOOD ESTIMATION UNDER EFFICIENT IMPORTANCE SAMPLING FOR NON-LINEAR STATE SPACE MODELS

Major Field: Statistics

Biographical:

Education:

Completed the requirements for the Doctor of Philosophy in Statistics at Oklahoma State University, Stillwater, Oklahoma in December, 2021.

Completed the requirements for the Master of Science in Applied Statistics at Rochester Institute of Technology, Rochester, New York in 2016.

Completed the requirements for the Bachelor of Science in Statistics at Hangzhou Dianzi University, Hangzhou, China in 2013.