Adaptive Filtering for Stochastic Volatility by using Exact Sampling

ShinIchi Aihara[1] , Arunabha Bagchi[2] and Saikat Saha[3]

[1]Department of Mechanical Systems, Tokyo University of Science Suwa, 5000-1 Toyohira, Chino, Nagano, Japan

[2]Department of Applied Mathematics, Twente University, P.O.Box 217,7500AE, Ensched, The Netherlands

[3]Department of Electrical Engineering, Liköping University, SE-58183 Liköping Sweden

Keywords: Particle Filter, Stochastic Volatility, Parameter Identification, Adaptive Filter.

Abstract: We study the sequential identification problem for Bates stochastic volatility model, which is widely used as the model of a stock in finance. By using the exact simulation method, a particle filter for estimating stochastic volatility is constructed. The systems parameters are sequentially estimated with the aid of parallel filtering algorithm. To improve the estimation performance for unknown parameters, the new resampling procedure is proposed. Simulation studies for checking the feasibility of the developed scheme are demonstrated.

1 INTRODUCTION

In the early 1960s, the linear filtering theory is formulated by Kalman and Bucy (Kalman and Bucy, 1961) and nonlinear filtering has already been well developed by many researchers, see Bensoussan (Bensoussan, 1992) and the bibliography therein. The realization problem for the nonlinear filter is still not easy. The recent development of particle filtering theory (Doucet et al., 2000) enable us to realize the nonlinear filtering in an easy way with the aid of the digital computer.

In this paper we consider the Bates model which is used in the fiance problem. In this model, we observe the tick value of stock price and need to estimate the movement of the volatility process for trading the stock and/or options. It is not possible to apply the nonlinear filtering theory to this volatility estimation problem, because this is out of the usual filtering problem in the continuous stochastic systems (Aihara and Bagchi, 2006). To circumvent this difficulty, the particle filter theory is usually applied in (Aihara et al., 2008; Cappé et al., 2005; Javaheri, 2005). The Bates model is given by

$$dS_t = \mu_S S_t dt + \sqrt{v_t} S_t dB_t + S_t dZ_t^J - \lambda m^J S_t dt, \quad (1)$$

$$dv_t = \kappa(\theta - v_t)dt + \xi\sqrt{v_t} dZ_t \quad (2)$$

where B_t and Z_t are standard Brownian motion processes with correlation ρ and Z_t^J denotes the pure-jump process. Noting that the process S_t denotes the stock value, the observation data $y_t = \log S_t/S_0$ is given by

$$dy_t = (\mu_S - \lambda m^J - \frac{1}{2}v_t)dt + \sqrt{v_t}dB_t + dq_t^J, \quad (3)$$

where q_t^J is a compound Poisson process with intensity λ and Gaussian distribution of jump size,i.e., $N(\mu_J, \sigma_J^2)$, and the mean relative jump size is given by $m^J = E(\exp(U^s) - 1) = \exp(\mu_J + \sigma_J^2/2) - 1$ and where the $\lambda m^J S_t$ term in (1) compensates for the instantaneous change in expected stock introduced by the pure-jump process Z_t^J. The particular properties of this model are

1. The observation mechanism (3) contains the signal dependent noise.
2. The observation noise has a correlation with the systems noise.

From the first property, the estimation of stochastic volatility becomes out of filtering problem. To circumvent this difficulty, all systems are discretized and the particle filter is applied in (Aihara et al., 2008; Johannes and Polson, 2006). However the usual discretization method transformed the original continuous non-Gaussian system into the conditional Gaussian. Recently, Brodie and Kaya (Broadie and Kaya, 2006; Smith, 2008; van Haastrecht and Pelsser, 2010) proposed the exact simulation method from the fact that the original system has a non-central chi-square distribution and we use this technique to the particle filtering (Aihara et al., 2012). Introducing the new Brownian motion

$$\tilde{Z}_t = \frac{1}{\sqrt{1-\rho^2}}(Z_t - \rho B_t), \quad (4)$$

Aihara S., Bagchi A. and Saha S..
Adaptive Filtering for Stochastic Volatility by using Exact Sampling.
DOI: 10.5220/0004454703260335
In *Proceedings of the 10th International Conference on Informatics in Control, Automation and Robotics* (ICINCO-2013), pages 326-335
ISBN: 978-989-8565-70-9

from the second property, (2) becomes

$$dv_t = \kappa(\theta - v_t)dt + \xi\sqrt{v_t}\sqrt{1-\rho^2}d\tilde{Z}_t$$
$$+\xi\rho(dy_t - (\mu_S - \lambda m^J - \frac{1}{2}v_t)dt - dq_t^J). \tag{5}$$

Although in (Aihara et al., 2012) the exact particle filtering procedure has been applied to the systems (5) and (3), the priority property of the exact simulation method can not be guaranteed and the final parameter estimation results are not satisfactory. In this paper, we introduce the rejection and acceptance method for enjoying the exact simulation method and construct the parallel filtering algorithm with an new resampling procedure for parameter identification.

In Sec. 2, we review the exact particle filtering with the new use of rejection and resampling procedure. For the real application to the finance problem, the market price of risk terms are included in the original dynamics in Sec. 3. Simulation studies for filtering and smoothing are presented in Sec. 5. The new parallel filter algorithm is developed for estimating the systems unknown parameters sequentially in Sec. 4. Finally some simulation studies for parameter estimation are presented in Sec. 6.

2 EXACT PARTICLE FILTERING

2.1 Exact Sampling

In order to perform the particle filter, the original system is usually approximated to the discrete-time one by using the Euler method. This approximation easily causes bias from the original continuous system. For example, the discrete-time volatility process v_k often becomes negative value. To avoid this bias, we propose the exact sampling method which is developed by Broadie and Kaya (Broadie and Kaya, 2006),Smith (Smith, 2008) and (van Haastrecht and Pelsser, 2010) for simulating the Heston process. In this paper, from (5) we can obtain the optimal importance function $p(v_{t_2}|v_{t_1}, y_{t_2}, y_{t_1})$. Hence we generate samples from this optimal importance function. Now we shall present the exact sampling procedure. For simplicity we consider the time interval $t_1 < t_2$ and set the following assumption: At most one jump occurs in this time interval and we observe y_{t_2} and y_{t_1}.

2.1.1 Exact Sampling from $p(v_{t_2}|v_{t_1}, y_{t_2}, y_{t_1})$

From (1), the volatility process v_{t_2} is represented by

$$v_{t_2} = \tilde{v}_{t_1} + \int_{t_1}^{t_2} \tilde{\kappa}(\tilde{\theta} - v_s)ds$$
$$+\int_{t_1}^{t_2} \xi\sqrt{v_s}\sqrt{1-\rho^2}d\tilde{Z}_s, \tag{6}$$

where

$$\begin{aligned}
\tilde{v}_{t_1} &= v_{t_1} + \rho\xi\{y_{t_2} - y_{t_1} \\
&\quad -(\mu_S - \lambda m^J)(t_2 - t_1) - \Delta q_{t_1}^i\} \\
\tilde{\kappa} &= \kappa - \frac{\rho\xi}{2} \\
\tilde{\theta} &= \frac{\kappa\theta}{\tilde{\kappa}} \\
\Delta q_{t_1}^i &= \text{jump sample from } q_t^J \text{ for } t_1 < t < t_2.
\end{aligned}$$

Now assuming that $\tilde{v}_{t_1} \geq 0$, we find that the transition law of v_{t_2} given by v_{t_1}, y_{t_1} and y_{t_2} is expressed as the non-central chi-square random variable $\chi_d^2(\lambda_\chi)$ with d degrees of freedom and non-centrality parameter λ_χ,

$$\frac{\xi^2(1-\rho^2)(1-e^{-\tilde{\kappa}(t_2-t_1)})}{4\tilde{\kappa}}\chi_d^2(\lambda_\chi), \tag{7}$$

where

$$d = \frac{4\tilde{\theta}\tilde{\kappa}}{\xi^2(1-\rho^2)}$$

and

$$\lambda_\chi = \frac{4\tilde{\kappa}e^{-\tilde{\kappa}(t_2-t_1)}}{\xi^2(1-\rho^2)(1-e^{-\tilde{\kappa}(t_2-t_1)})}\tilde{v}_{t_1}.$$

Hence by using MATLAB code ”ncx2rnd.m”, we can get a sample v_{t_2} .

For the case that $\tilde{v}_{t_1} < 0$, this event may occur when v_{t_1} is very small in generating particles by using the data $y_{t_2} - y_{t_1}$. In the real world, we have already get the value $y_{t_2} - y_{t_1}$. Hence v_{t_1} should satisfy

$$v_{t_1} + \rho\xi\{y_{t_2} - y_{t_1} - (\mu_S - \lambda m^J)(t_2 - t_1) - \Delta q_{t_1}^i\}$$
$$+\tilde{\kappa}\tilde{\theta}(t_2 - t_1) \geq 0. \tag{8}$$

2.1.2 $\tilde{v}_{t_1} < 0$ Case

We use the rejection and resampling procedure. At the time t_1, we already get many particles say $v_{t_1}^{(i)}$. Hence we check the above inequality (8) for each $v_{t_1}^{(i)}$. If the particles which do not satisfy (8) are found, we ignore these and perform a resampling procedure.

2.2 Construction of Probability Density Function

If we use the Euler scheme for discretization, the generated sample becomes the conditionally Gaussian. However for the exact sampling scheme , the processes generated are governed by the non-central chi-square distribution. Although the explicit function form of this distribution is not possible, we can numerically evaluate the pdf by using the MATLAB code, "ncx2pdf.m".

- $p(v_{t_2}|v_{t_1}, y_{t_2}, y_{t_1})$ form
  Noting that the jump occurs at most one time during the time interval $[t_2, t_1]$, i.e, the probability that the jump occurs is $\lambda e^{-\lambda(t_2-t_1)}$ and no jump becomes $1-\lambda e^{-\lambda(t_2-t_1)}$, and the jump size $U^s_\cdot$ is Gaussian with mean μ_J and variance σ_J^2, we have

$$\begin{aligned}
&p(v_{t_2}|v_{t_1}, y_{t_2}, y_{t_1}) \\
&= (1-e^{-\lambda(t_2-t_1)}\lambda(t_2-t_1)) \\
&\times p(v_{t_2}|v_{t_1}, y_{t_2}, y_{t_1}, \Delta q^j_{t_1}=0) \\
&+ e^{-\lambda(t_2-t_1)}\lambda(t_2-t_1)\int_{-\infty}^{\infty} p(v_{t_2}|v_{t_1}, y_{t_2}, y_{t_1}, U^s) \\
&\quad \frac{1}{\sqrt{2\pi\sigma_J^2}}\exp\left(-\frac{(U^s-\mu_J)^2}{2\sigma_J^2}\right)dU^s \\
&= (1-e^{-\lambda(t_2-t_1)}\lambda(t_2-t_1)) \\
&\times \text{pdf of } \left\{\frac{\xi^2(1-\rho^2)(1-e^{-\tilde{\kappa}(t_2-t_1)})}{4\tilde{\kappa}}\chi^2_{\tilde{d}}(\tilde{\lambda}_\chi)\right\} \\
&+ e^{-\lambda(t_2-t_1)}\lambda(t_2-t_1) \\
&\times \int_{-\infty}^{\infty} \text{pdf of } \left\{\frac{\xi^2(1-\rho^2)(1-e^{-\tilde{\kappa}(t_2-t_1)})}{4\tilde{\kappa}}\right. \\
&\quad \left.\times\chi^2_{\tilde{d}}\left(\tilde{\lambda}_\chi - \frac{4\tilde{\kappa}e^{-\tilde{\kappa}(t_2-t_1)}\rho}{\xi(1-\rho^2)(1-e^{-\tilde{\kappa}(t_2-t_1)})}U^s\right)\right\} \\
&\times \frac{1}{\sqrt{2\pi\sigma_J^2}}\exp\left(-\frac{(U^s-\mu_J)^2}{2\sigma_J^2}\right)dU^s \qquad (9)
\end{aligned}$$

  where

$$\begin{aligned}
\tilde{\lambda}_\chi &= \frac{4\tilde{\kappa}e^{-\tilde{\kappa}(t_2-t_1)}}{\xi^2(1-\rho^2)(1-e^{-\tilde{\kappa}(t_2-t_1)})} \\
&\quad \times\{v_{t_1}+\rho\xi\{y_{t_2}-y_{t_1}-(\mu_S-\lambda m^J)(t_2-t_1)\}\}
\end{aligned}$$

  In (9), the first term implies that we have no jump and the second term is caused from the jump size $U^s \in N(\mu^J, \sigma_J^2)$.

- $p(v_{t_2}|v_{t_1}, y_{t_1})$ form
  It follows from (2) that

$$p(v_{t_2}|v_{t_1}, y_{t_1}) = \text{pdf of } \frac{\xi^2(1-e^{-\kappa(t_2-t_1)})}{4\kappa}\chi^2_{\tilde{d}}(\lambda^v_\chi),$$

  where

$$\tilde{d} = \frac{4\theta\kappa}{\xi^2},$$

  and

$$\lambda^v_\chi = \frac{4\kappa e^{-\kappa(t_2-t_1)}}{\xi^2(1-e^{-\kappa(t_2-t_1)})}v_{t_1}.$$

- $p(y_{t_2}|y_{t_1}, \int_{t_1}^{t_2} v_s ds)$ form
  In this case, from

$$\begin{aligned}
dy_t = (\mu_S - \lambda m^J - \frac{1}{2}v_t)dt + \frac{\rho}{\xi}(dv_t - \kappa(\theta - v_t)dt) \\
+ \sqrt{1-\rho^2}\sqrt{v_t}d\tilde{Z}_t + dq^J_t
\end{aligned}$$

  we easily get

$$\begin{aligned}
&p(y_{t_2}|y_{t_1}, \int_{t_1}^{t_2} v_s ds) = \frac{1-e^{-\lambda(t_2-t_1)}\lambda(t_2-t_1)}{\sqrt{2\pi(1-\rho^2)\int_{t_1}^{t_2} v_s ds}} \\
&\times \exp\left[-\frac{1}{2(1-\rho^2)\int_{t_1}^{t_2} v_s ds}\{y_{t_2}\right. \\
&\quad - [y_{t_1} + (\mu_S - \lambda m^J - \frac{\kappa\rho\theta}{\xi})(t_2-t_1) \\
&\quad \left. - (\frac{1}{2} - \frac{\kappa\rho}{\xi})\int_{t_1}^{t_2} v_s ds + \frac{\rho}{\xi}(v_{t_2} - v_{t_1})]\}^2\right] \\
&+ \frac{e^{-\lambda(t_2-t_1)}\lambda(t_2-t_1)}{\sqrt{2\pi((1-\rho^2)\int_{t_1}^{t_2} v_s ds + \sigma_J^2)}} \\
&\times \exp\left[-\frac{1}{2((1-\rho^2)\int_{t_1}^{t_2} v_s ds + \sigma_J^2)}\{y_{t_2}\right. \\
&- [y_{t_1} + (\mu_S - \lambda m^J - \frac{\kappa\rho\theta}{\xi})(t_2-t_1) \\
&\left. - (\frac{1}{2} - \frac{\kappa\rho}{\xi})\int_{t_1}^{t_2} v_s ds + \mu_J + \frac{\rho}{\xi}(v_{t_2} - v_{t_1})]\}^2\right] \qquad (10)
\end{aligned}$$

2.3 Exact Particle Filter Algorithm

Now we can perform the exact particle filter. The weight $w^{(i)}_\cdot$ is given by the following recursive form: for $i = 1, \cdots, N$ and $k = 1, \cdots, m$

$$w^{(i)}_{t_k} = w^{(i)}_{t_{k-1}} \frac{p(y_{t_k}|y_{t_{k-1}}, \int_{t_{k-1}}^{t_k} v^{(i)}_s ds)p(v^{(i)}_{t_k}|v^{(i)}_{t_{k-1}})}{p(v^{(i)}_{t_k}|v^{(i)}_{t_{k-1}}, y_{t_k}, y_{t_{k-1}})}. \qquad (11)$$

Of course we need to perform the resampling scheme in the above filtering algorithm.

It is also possible to construct the smoothing algorithm by using forward filtering-backward sampling scheme by Doucet et. al. (Doucet et al., 2000)

Algorithm(Sample realization)**.**

- Run the particle filter to obtain $(v^{(i)}_{t_k}, \omega^{(i)}_{t_k})_{1\le k\le m, 1\le i\le N}$.
- By using the systematic resampling method, we generate new index J_m from $\{\omega^{(i)}_{t_m}\}_{1\le i\le N}$.

- Set $\tilde{v}_{t_m}^{(i)}$ as $v_{t_m}^{J_m}$.
- For $k = m-1$ to 1;
  Resample to get J_k from $\{\omega_{t_k}^{(i)} p(\tilde{v}_{t_{k+1}}^{(i)} | v_{t_k}^{(i)})\}_{1 \le i \le N}$.
  Set $\tilde{v}_{t_k}^{(i)}$ as $v_k^{J_k}$.
- $\tilde{v}^{(i)} = [\tilde{v}_{t_1}^{(i)}, \tilde{v}_{t_2}^{(i)}, \cdots, \tilde{v}_{t_m}^{(i)}]$ is the realized particles for smoothing with $1/N$ probability.

3 MARKET PRICE OF RISK

For estimation we also need the dynamics of the state S_t and v_t under the actual probability measure $\mathcal{P}$. We specify the market price of risk for B_t and Z_t as

$$d\begin{pmatrix} B_t^{\mathcal{P}} \\ Z_t^{\mathcal{P}} \end{pmatrix} = d\begin{pmatrix} B_t \\ Z_t \end{pmatrix} - \begin{pmatrix} \lambda_S\sqrt{v_t} & 0 \\ 0 & \lambda_v\sqrt{v_t} \end{pmatrix} dt.$$

We ignore the jump-timing risk premium and the jump-size risk is assumed to be included in μ_J. Now the dynamics of y_t and v_t under $\mathcal{P}$ is given by

$$\begin{aligned} dy_t &= (\mu_S - \lambda m^J + (\lambda_S - \frac{1}{2})v_t)dt + \sqrt{v_t}dB_t^{\mathcal{P}} + dq_t^J \\ dv_t &= (\kappa\theta - (\kappa - \lambda_v\xi)v_t)dt + \xi\sqrt{v_t}\sqrt{1-\rho^2}d\tilde{Z}_t^{\mathcal{P}} \\ &\quad + \xi\rho(dy_t - (\mu_S - \lambda m^J - (\frac{1}{2} - \lambda_S)v_t)dt - dq_t^J). \end{aligned}$$

Hence it is possible to apply our particle filer algorithm to this world measure dynamics. The corresponding dynamics is transformed to

$$v_{t_2} = \tilde{v}_{t_1} + \int_{t_1}^{t_2} \tilde{\kappa}(\tilde{\theta} - v_s)ds + \int_{t_1}^{t_2} \xi\sqrt{v_s}\sqrt{1-\rho^2}d\tilde{Z}_s,$$

where

$$\begin{aligned} \tilde{v}_{t_1} &= v_{t_1} + \rho\xi\{y_{t_2} - y_{t_1} \\ &\quad -(\mu_S - \lambda m^J)(t_2 - t_1) - \Delta q_{t_1}^i\} \\ \tilde{\kappa} &= \kappa - \frac{\rho\xi}{2} + \xi(\rho\lambda_S - \lambda_v) \\ \tilde{\theta} &= \frac{\kappa\theta}{\tilde{\kappa}}. \end{aligned}$$

$p(v_{t_2}|v_{t_1})$ becomes

$$p(v_{t_2}|v_{t_1}) = \text{pdf of } \frac{\xi^2(1 - e^{-(\kappa-\xi\lambda_v)(t_2-t_1)})}{4(\kappa - \xi\lambda_v)}\chi_d^2(\lambda_\chi^v),$$

where

$$d = \frac{4\theta\kappa}{\xi^2},$$

and

$$\lambda_\chi^v = \frac{4(\kappa - \xi\lambda_v)e^{-(\kappa-\xi\lambda_v)(t_2-t_1)}}{\xi^2(1 - e^{-(\kappa-\xi\lambda_v)(t_2-t_1)})}v_{t_1}.$$

Noting that

$$\begin{aligned} dy_t &= (\mu_S - \lambda m^J + (\lambda_S - \frac{1}{2})v_t)dt \\ &+ \frac{\rho}{\xi}(dv_t - \kappa\theta dt + (\kappa - \lambda_v\xi)v_t)dt) \\ &+ \sqrt{1-\rho^2}\sqrt{v_t}d\tilde{Z}_t + dq_t^J, \end{aligned}$$

we also have

$$\begin{aligned} p(y_{t_2}|y_{t_1}, \int_{t_1}^{t_2} v_s ds) &= \frac{1 - e^{-\lambda(t_2-t_1)}\lambda(t_2 - t_1)}{\sqrt{2\pi(1-\rho^2)\int_{t_1}^{t_2} v_s ds}} \\ &\times \exp[-\frac{1}{2(1-\rho^2)\int_{t_1}^{t_2} v_s ds}\{y_{t_2} \\ &-[y_{t_1} + (\mu_S - \lambda m^J - \frac{\kappa\rho\theta}{\xi})(t_2 - t_1) \\ &-(\frac{1}{2} - \frac{\kappa\rho}{\xi} + \rho\lambda_v - \lambda_S)\int_{t_1}^{t_2} v_s ds + \frac{\rho}{\xi}(v_{t_2} - v_{t_1})]\}^2] \\ &+ \frac{e^{-\lambda(t_2-t_1)}\lambda(t_2 - t_1)}{\sqrt{2\pi((1-\rho^2)\int_{t_1}^{t_2} v_s ds + \sigma_J^2)}} \\ &\times \exp[-\frac{1}{2((1-\rho^2)\int_{t_1}^{t_2} v_s ds + \sigma_J^2)}\{y_{t_2} \\ &-[y_{t_1} + (\mu_S - \lambda m^J - \frac{\kappa\rho\theta}{\xi})(t_2 - t_1) \\ &-(\frac{1}{2} - \frac{\kappa\rho}{\xi} + \rho\lambda_v - \lambda_S)\int_{t_1}^{t_2} v_s ds + \mu^J \\ &+ \frac{\rho}{\xi}(v_{t_2} - v_{t_1})]\}^2] \end{aligned}$$

4 SIMULATION STUDIES FOR FILTERING AND SMOOTHING

In this section, we simulate the proposed filtering and smoothing equations with known systems parameters for Heston model without any market price of risk terms for simplicity.

The systems parameters are listed in Table-1 and in this case the point ”zero” is acceptable.

Table 1: Model parameters.

	κ	θ	μ_S	ρ	ξ
True	0.8638	1.0000	0.0100	-0.1050	2.5017

We present the log price in Fig.1. As shown in Fig.2, the volatility process hits ”zero” at many times because the degree of freedoms is less than 2. The rejection and resampling rate is shown in Fig.3.

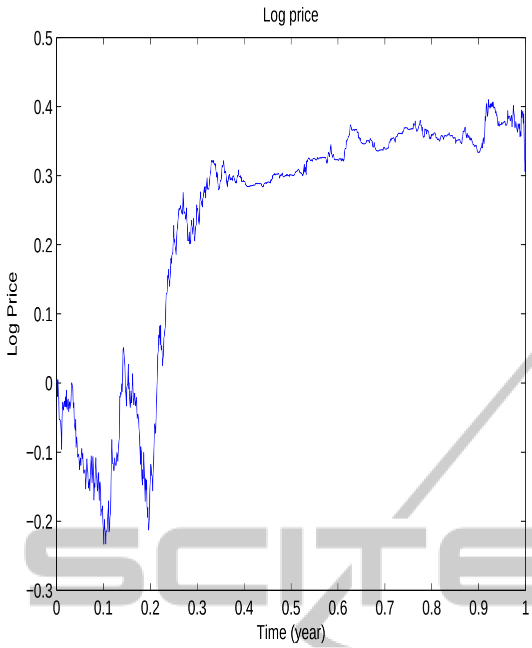

Figure 1: Observation data y_t.

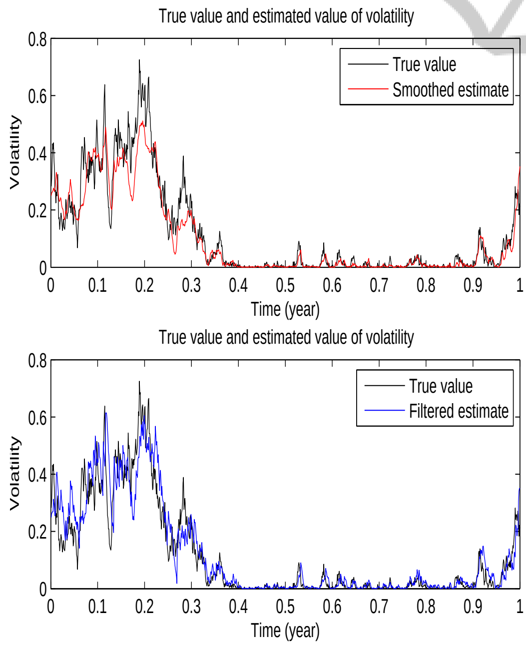

Figure 2: True and estimated v_t.

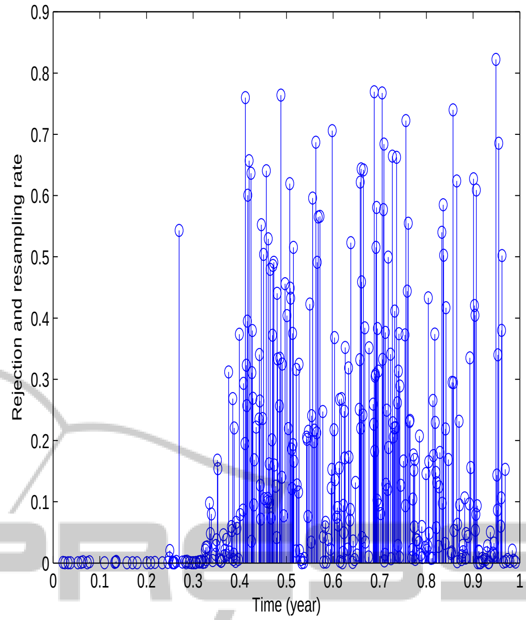

Figure 3: Rejection and resampling rate.

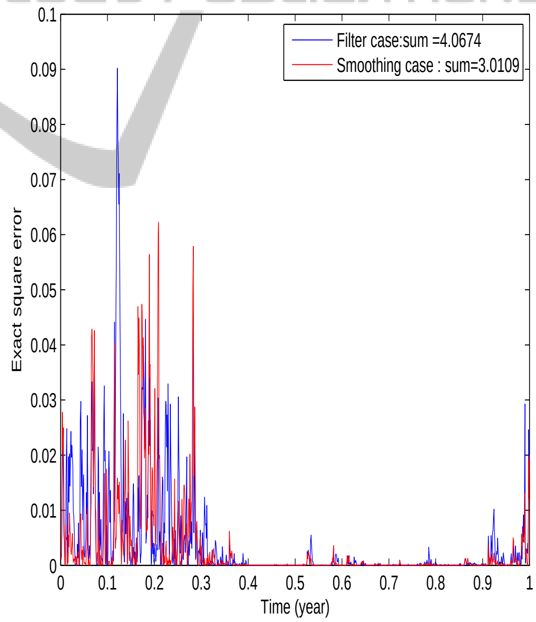

Figure 4: Mean square errors.

5 PARALLEL FILTERING ALGORITHM

In a market, traders buy or sell stocks from their feeling of the volatility movement of the traded stock. Form this fact, we need to estimate the volatility itself rather than the parameters in the model. The estimate of the volatility should be online. Hence, in this section, we construct the recursive online estimate for the volatility. Of course to obtain the estimate of volatility, we also get the estimate of systems parameters at the same time. Here the unknown parameters are denoted by

$$\alpha = [\kappa, \theta, \nu_S, \rho, \xi, \lambda, \nu^J, \sigma^J, \lambda_v, \lambda_S].$$

Now we set candidates of unknown parameter α such that

$\alpha^{(j)} \in$ uniformly random vectors in $\Theta, j = 1, \cdots, M_p$

If we know an *a-priori* information for Θ, we may set the pdf $p_o(\alpha)$. For each $\alpha^{(j)}$, we solve the particle filter $\hat{v}_{t_k}(\alpha^{(j)})$ from Section 2.3. Hence from (B.D.O.Anderson and J.B.Moore, 1979), we get the posteriori density given by

$$p(\alpha^{(j)}|y_{t_0:t_k}) = \frac{\{\Sigma_{i=1}^{N} w_{t_{k-1}}^{(i)}(\alpha^{(j)}) LF_{k,i}\} p(\alpha^{(j)}|y_{t_0:t_{k-1}})}{\Sigma_{j=1}^{M_p}\left\{\{\Sigma_{i=1}^{N} w_{t_{k-1}}^{(i)}(\alpha^{(j)}) LF_{k,i}\} p(\alpha^{(j)}|y_{t_0:t_{k-1}})\right\}}$$

where

$$LF_{k,i} = p(y_k|y_{k-1}, \int_{t_{k-1}}^{t_k} v_s^{(i)}(\alpha^{(j)})ds).$$

The estimates of volatility and parameters are given by

$$\hat{v}_k = \Sigma_{j=1}^{M_p} \hat{v}_{t_k}(\alpha^{(j)}) p(\alpha^{(j)}|y_{t_0:t_k}) \quad (12)$$

$$\hat{\alpha}_k = \Sigma_{j=1}^{M_p} \alpha^{(j)} p(\alpha^{(j)}|y_{t_0:t_k}). \quad (13)$$

5.1 New Resampling Procedure

The sample of parameter $\{\alpha^{(j)}\}_{j=1}^{M_P}$ is drawn only from the initial information (in this paper we set the uniform distribution). Hence for a long time period the estimates of parameters are sometimes stacked with some biases. This may cause from the fact that there are so many unknown parameters while we get a scaler observation data. In order to improve this property, a resampling for the candidates for parameters $\alpha^{(j)}$ is usually performed in MCMC algorithm in (Johannes and Polson, 2006). In the parallel filtering algorithm, we already get the posterior probability $p(\alpha^{(j)}|y_{t_0:t_k})$ and from this distribution, we propose to get new samples for $\alpha^{(j)}$ by using the following procedure:

1. We set the resampling time t_p^r if

$$(\sum_{j=1}^{M_P} p^2(\alpha^{(j)}|y_{t_0:t_p^r}))^{-1} \leq \frac{2M_P}{3},$$

   we generate new sample $\alpha^{(j)}$ from the step 2) to 6).

2. Calclulate

$$\hat{\alpha}_{t_p^r} = \Sigma_{j=1}^{M_p} \alpha^{(j)} p(\alpha^{(j)}|y_{t_0:t_p^r})$$

$$\hat{\sigma}_{\alpha(i)} = \Sigma_{j=1}^{M_p} (\alpha^{(j)}(i))^2 p(\alpha^{(j)}(i)|y_{t_0:t_k^r}) - (\hat{\alpha}_{t_p^r}^{(j)}(i))^2$$

   for $i = 1, 2, \cdots, 10$.

3. We denote the parameter range at the resampling time point $t = t_p^r$ as

$$\text{lb}(i, t_p^r) \leq \alpha(i) \leq \text{ub}(i, t_p^r) \ \text{ for } i = 1, 2, \cdots, 10,$$

   where $\text{lb}(i,t) = \text{lb}(i)$ and $\text{ub}(i,t) = \text{ub}(i)$ for $t <$ the first resampling time.

4. From the calculated $\hat{\alpha}_{t_p^r}$ and $\hat{\sigma}_\alpha$, we reset the parameter range from t_{p-1}^r as

$$\text{lb}(i, t_p^r) = \max(\text{lb}(i, t_{p-1}^r), \hat{\alpha}_{t_k^r}(i) - 3\hat{\sigma}_{\alpha(i)})$$

   and

$$\text{ub}(i, t_p^r) = \min(\text{ub}(i, t_{p-1}^r), \hat{\alpha}_{t_p^k} + 3\hat{\sigma}_{\alpha(i)}).$$

5. Construct the candidates of parameter k;

$$\alpha_k(i) = \text{lb}(i, t_p^r) + \frac{\text{ub}(t_p^r) - \text{ub}(t_p^r)}{M_p - 1}(i-1).$$

   for $k = 1, 2, \cdots, M_p$.

6. Construct the posterior distribution for each parameter $\alpha(i)$ by using the Gaussian approximation:

$$P(\alpha(i)|y_{t_0:t_p^r}) \sim \mathcal{N}(\alpha(i); \hat{\alpha}_{t_p^r}, \varepsilon_i \hat{\sigma}_{\alpha(i)}^{1/2}),$$

   where ε_i is a user defined parameter to increase diversity.

7. Allocate n_k copies of the particle $\alpha_k(i)$ from

$$n_i = \text{the number of } \frac{(k-1)+\tilde{u}}{M_p} \in (F_G(\alpha_{k-1}(i)), F_G(\alpha_k(i))]$$

   for $\tilde{u}$= uniform random number, where F_G is an approximated Gaussian distribution (step 6)):

$$F_G(\alpha_k(i)) = \frac{\int_{-\infty}^{\alpha_k(i)} \frac{1}{\sqrt{2\pi}\varepsilon_i \hat{\sigma}_{\alpha(i)}} \exp[-\frac{1}{2(\varepsilon_i \hat{\sigma}_\alpha(i))^2}(\alpha(i) - \hat{\alpha}_{t_p^r})^2] d\alpha(i)}{F_G(\text{lb}(i, t_p^r))}.$$

8. Construct new candidate; for $j = 1, 2, \cdots, M_P$

$$\alpha^{(j)} = [\alpha^j(1), \alpha^j(2), \cdots, \alpha^j(10)].$$

9. Reset $p(\alpha^{(j)}|y_{y_0:t_p^r}) = 1/M_P$.

6 SIMULATION STUDIES

We set the following parameters in Table 2.
The lower and upper bounds for parameters are set as Here we set $dt = 0.001, T = 1, M = 100, M_P = 60$ and $t_r = 20dt, \varepsilon_1 = 1.1, \varepsilon_2 = 1.1, \varepsilon_3 = 1.15, \varepsilon_4 = 1.01, \varepsilon_5 = 1.01, \varepsilon_6 = 1.15, \varepsilon_7 = 1.15, \varepsilon_8 = 1.15, \varepsilon_9 = 1.15$, and $\varepsilon_{10} = 1.15$. In Fig.5, we show the true volatility state

Table 2: Model parameters.

	κ	θ	μ_S	ρ
True	0.8638	1.1000	0.6000	-0.1500
	ξ	λ	μ^J	σ^J
True	2.1017	5.4000	-0.3000	0.2500
	λ_v	λ_S		
True	0.1882	0.1723		

Table 3: Lower and upper bounds of model parameters.

	κ	θ	μ_S	ρ
Upper	1.6412	2.0900	1.140	-0.001
Lower	0.0086	0.0011	0.006	-0.300
	ξ	λ	μ^J	σ^J
Upper	3.9932	10.260	-0.0030	0.4570
Lower	0.0210	0.0540	-0.5700	0.0025
	λ_v	λ_S		
Upper	0.35758	0.62174		
Lower	0.0018	0.0032		

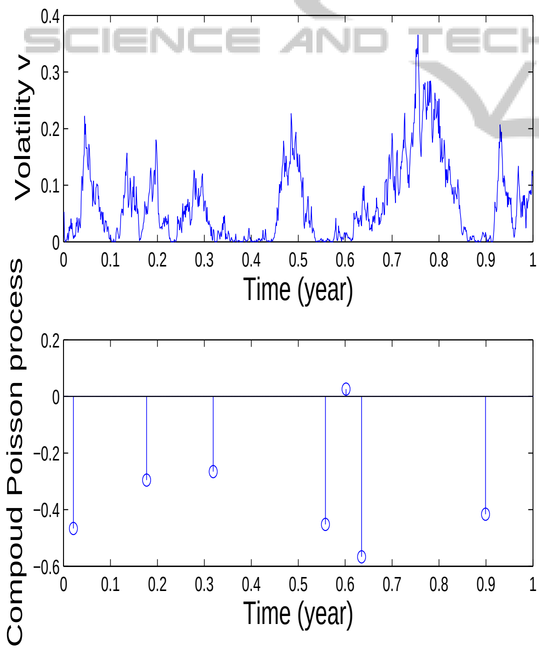

Figure 5: The true volatility state and compound Poisson process.

and compound Poisson process. The observed log price is also shown in Fig.6. The estimated volatility is shown in Fig.7 with the square error in Fig.8.

We also present the resampling rates of the particle filter of this algorithm in Fig. 9.

The estimates of unknown parameters are demonstrated from Figs 10 to 19 with the corresponding histogram for $0 \leq t \leq 1$.

The true and estimated parameters at $t = 1$ are shown in Figs. 20-24 where the green line indicated

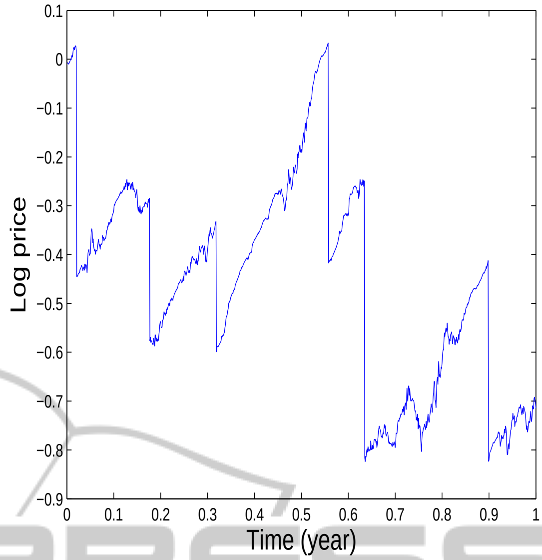

Figure 6: Observed log price.

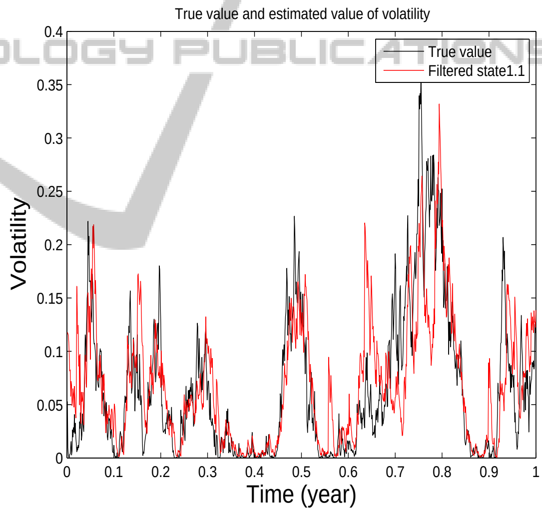

Figure 7: True and estimated volatility.

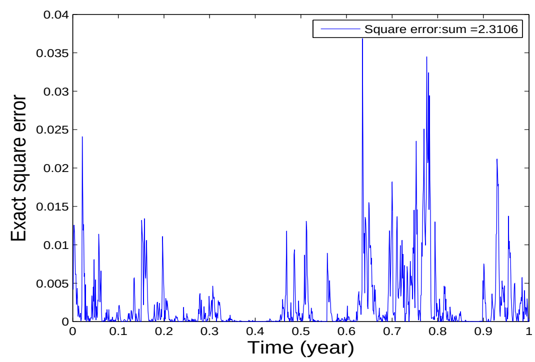

Figure 8: Square error of estimated volatility.

the upper and lower bounds for each parameters at $t = 0$.

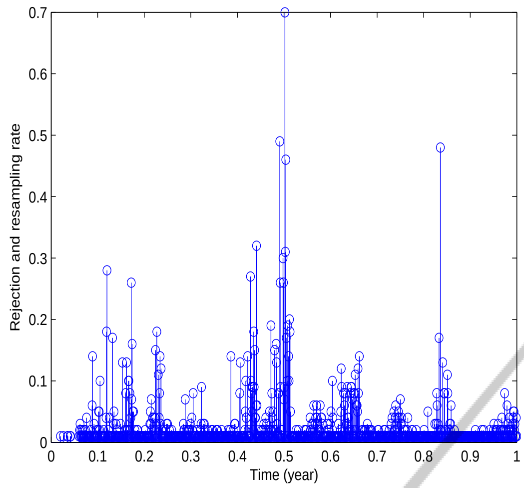

Figure 9: Square error of estimated volatility

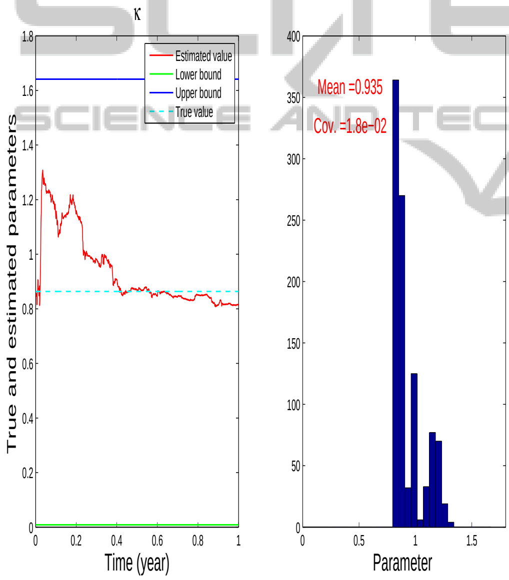

Figure 10: Estimated κ and histogram for $0 \leq t \leq 1$

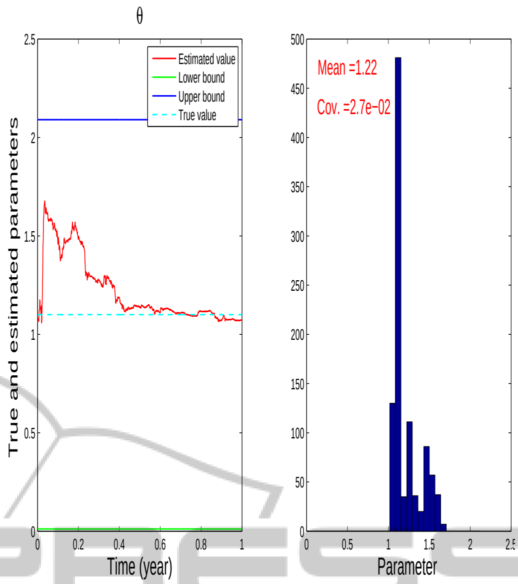

Figure 11: Estimated θ and histogram for $0 \leq t \leq 1$

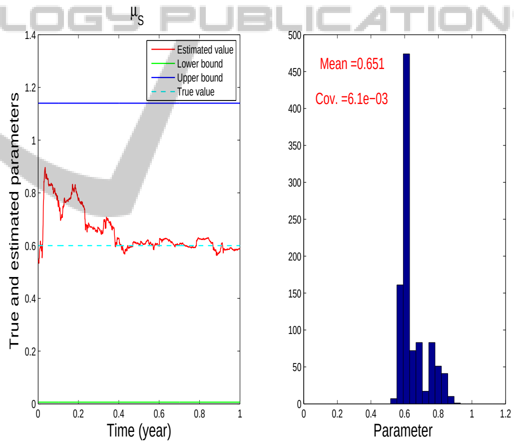

Figure 12: Estimated μ and histogram for $0 \leq t \leq 1$.

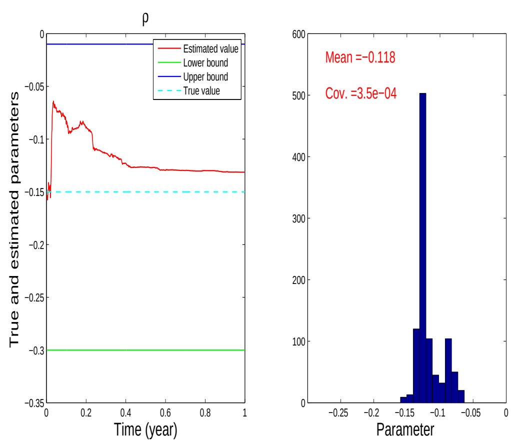

Figure 13: Estimated ρ and histogram for $0 \leq t \leq 1$.

7 CONCLUSIONS

By using the non-central chi-square random generation method, we developed the particle filter for estimating the stochastic volatility process. The sequential estimation for the systems unknown parameters are performed with the aid of the new resampling procedure. In this procedure, we need to choose the resampling time t_r and the user defined parameter ε_i to obtain the good numerical results. This turning problem is still an open problem.

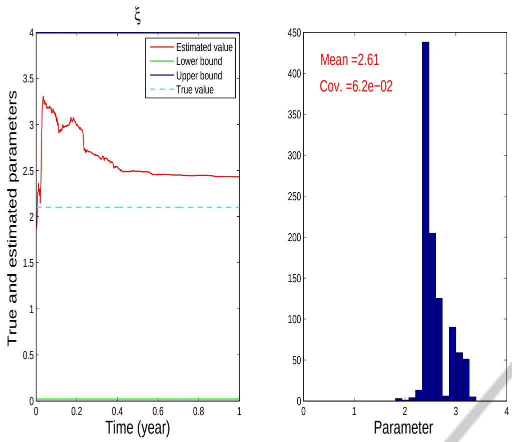

Figure 14: Estimated ξ and histogram for $0 \leq t \leq 1$.

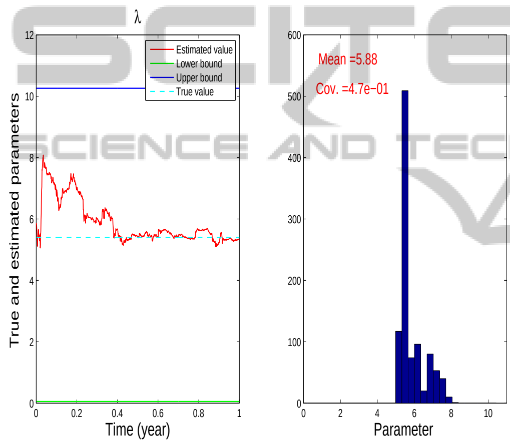

Figure 15: Estimated λ and histogram for $0 \leq t \leq 1$

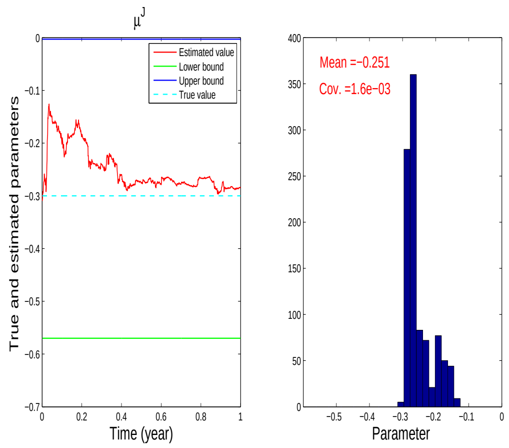

Figure 16: Estimated μ^J and histogram for $0 \leq t \leq 1$

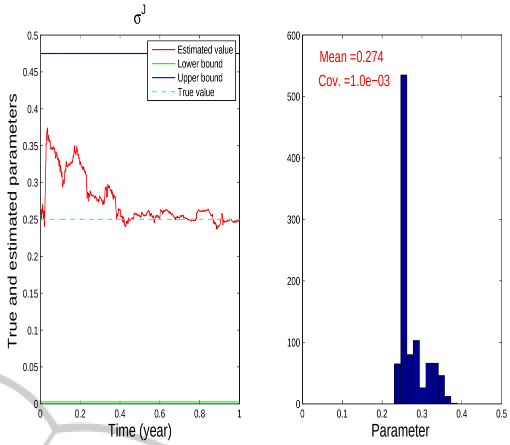

Figure 17: Estimated σ^J and histogram for $0 \leq t \leq 1$

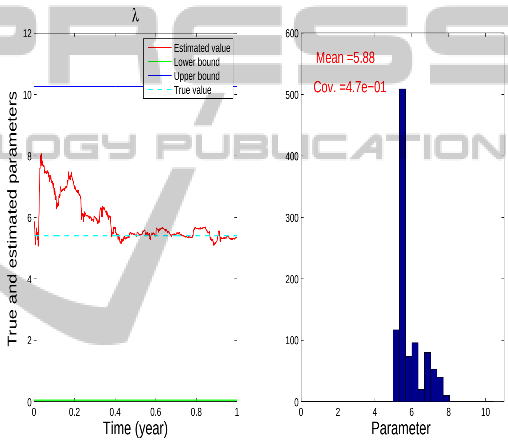

Figure 18: Estimated λ_v and histogram for $0 \leq t \leq 1$

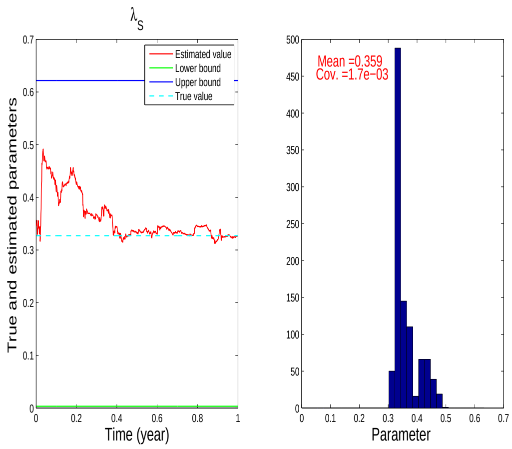

Figure 19: Estimated λ_S and histogram for $0 \leq t \leq 1$

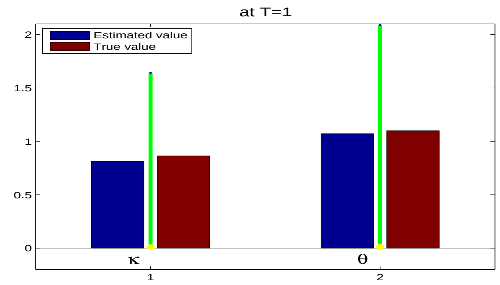

Figure 20: Estimated κ and θ at $t = 1$.

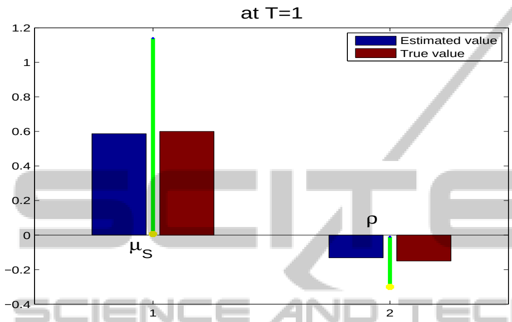

Figure 21: Estimated μ_S and ρ at $t = 1$.

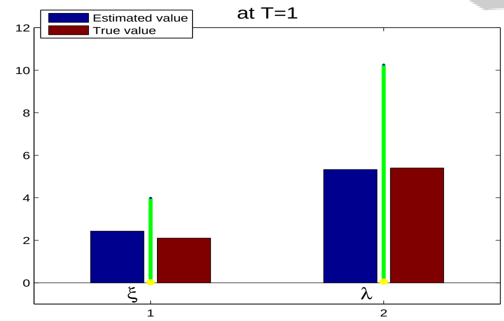

Figure 22: Estimated ξ and λ at $t = 1$.

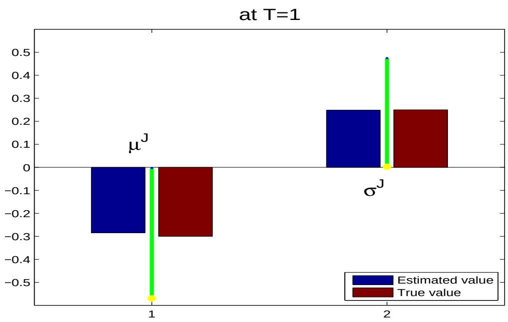

Figure 23: Estimated ξ and λ at $t = 1$.

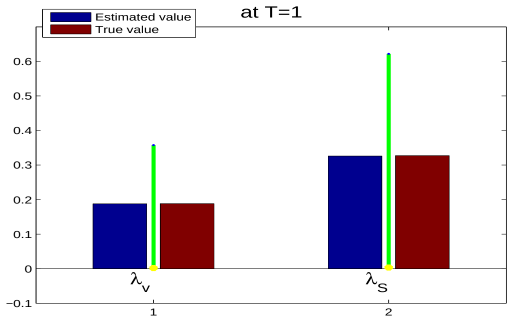

Figure 24: Estimated λ_V and λ_S at $t = 1$.

REFERENCES

Aihara, S. and Bagchi, A. (2006). Filtering and identification of heston's stochastic volatility and its market risk. *J. Economical Dynamics and Control*, 30:2363–2388.

Aihara, S., Bagchi, A., and S.Saha (2008). Estimating volatility and model parameters of stochastic volatility models with jumps using particle filter. *Proc. of 17th IFAC World Congress*.

Aihara, S., Bagchi, A., and S.Saha (2012). Identification of bates stochastic volatility model by using non-central chi-square random generation method. *Proc. of IEEE ICASSP 2012*.

B.D.O.Anderson and J.B.Moore (1979). *Optimal Filtering*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Bensoussan, A. (1992). *Stochastic Control of Partially Observable Systems*. Cambridge University Press, Cambridge.

Broadie, M. and Kaya, O. (2006). Exact simulation of stochastic volatility and other affine jump diffusion processes. *Operations Research*, 54(2):217–231.

Cappé, O., E.Moulines, and Rydén, T. (2005). *Inference in Hidden Markov Models*. Springer Science+Business Media, Inc., New York.

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

Javaheri, A. (2005). *Inside Volatility Arbitrage*. John Wiley & Sons, Inc, Hoboken.

Johannes, M. and Polson, N. (2006). MCMC method for financial econometrics. In Y.Ait-Sahalia and Hansen, L., editors, *Handbook of Financial Econometrics*. Elsevier.

Kalman, R. and Bucy, R. (1961). New results in linear filtering and prediction theory. *Trans. ASME - Journ. Basic Engineering*, 83 (Series D):95–108.

Smith, R. (2008). An almost exact simulation method for the heston model. *Journal of Computational Finance*, 11(1):115–125.

van Haastrecht, A. and Pelsser, A. (2010). Efficient, almost exact simulation of the heston stochastic volatility model. *IJTAF*, 13:1–43.