

## Research Article

# A Fast and Efficient Markov Chain Monte Carlo Method for Market Microstructure Model

Sun Yapeng <sup>1,2</sup>, Peng Hui <sup>1</sup>, and Xie Wenbiao <sup>3</sup>

<sup>1</sup>School of Automation, Central South University, Changsha 410083, Hunan, China

<sup>2</sup>School of Computer Science and Engineer, Hunan University of Science and Technology, Xiangtan 411201, Hunan, China

<sup>3</sup>School of Electrical & Information Engineering, Changsha University of Science and Technology, Changsha 410004, Hunan, China

Correspondence should be addressed to Peng Hui; [huipeng@csu.edu.cn](mailto:huipeng@csu.edu.cn)

Received 19 February 2021; Revised 13 May 2021; Accepted 24 September 2021; Published 27 October 2021

Academic Editor: Lijun Pei

Copyright © 2021 Sun Yapeng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The non-linear market microstructure (MM) model for financial time series modeling is a flexible stochastic volatility model with demand surplus and market liquidity. The estimation of the model is difficult, since the unobservable surplus demand is a time-varying stochastic variable in the return equation, and the market liquidity arises both in the mean term and in the variance term of the return equation in the MM model. A fast and efficient Markov Chain Monte Carlo (MCMC) approach based on an efficient simulation smoother algorithm and an acceptance-rejection Metropolis–Hastings algorithm is designed to estimate the non-linear MM model. Since the simulation smoother algorithm makes use of the band diagonal structure and positive definition of Hessian matrix of the logarithmic density, it can quickly draw the market liquidity. In addition, we discuss the MM model with Student-t heavy tail distribution that can be utilized to address the presence of outliers in typical financial time series. Using the presented modeling method to make analysis of daily income of the S&P 500 index through the point forecast and the density forecast, we find clear support for time-varying volatility, volatility feedback effect, market microstructure theory, and Student-t heavy tails in the financial time series. Through this method, one can use the estimated market liquidity and surplus demand which is much smoother than the strong stochastic return process to assist the transaction decision making in the financial market.

## 1. Introduction

The prevention and avoidance of financial risk has always been the core issue of investment theory and practice. Risk can be measured by the time-varying variance, i.e., volatility of return on investment. The main models to analyze time-varying volatility of financial time series are SV (stochastic volatility) type models (Taylor) [1] and GARCH (generalized autoregressive conditional heteroscedasticity) type models (Bollerslev) [2]. GARCH type models can describe some behavioral characteristics of volatility, but they regard volatility as a deterministic function of historical information, which makes them unable to flexibly fit financial time series. In theory, due to the introduction of stochastic volatility, SV type models are more in line with the characteristics of financial market and are then more suitable for the analysis

and prediction of financial time series. In the fields of option pricing, risk management, and portfolio, SV type models have very important theoretical and practical significance in capturing the relation of asset return and volatility and thus can provide market information and decision-making reference for investors and market managers.

The general SV models focus on modeling the dynamics of asset price itself and especially its volatility. Unlike those models, Bouchaud and Cont [3] proposed a phenomenological model called market microstructure (MM) model that describes the relationship between price and market liquidity as well as surplus demand. Based on the ideal of the MM model, Iino and Ozaki [4] proposed a continuous-time MM model that is a SV model. The estimation of the MM model is not straightforward, since the market liquidity appears both in the mean term and in the variance term of

the model, and the surplus demand is also time-varying stochastic variable. In previous works, the non-linear Kalman filter and maximum likelihood method (see, e.g., [5–10]) were used to estimate the parameters of the MM model, but this kind of method has some shortcomings in dealing with strong non-Gaussian non-linear financial data. Thus, we consider using the MCMC method to improve estimation of the model.

The MCMC method was promoted by Gelfand and Smith [11], and it is equitable to say that the MCMC method has revitalized (perhaps even revolutionized) Bayesian statistics. Bardsley and Cui [12] developed a scalable optimization-based MCMC method for solving hierarchical Bayesian inverse problems with non-linear parameter-to-observable maps and a broader class of hyperparameters. Zhou and Tartakovsky [13] proposed an adaptive MCMC method, in which a transmission model is replaced by a fast and accurate substitutive model that adopts a deep convolutional neural network to reduce calculation burden. For the shortcoming of MCMC, i.e., the huge amount of computation when the dataset is too large, Nemeth and Fearnhead [14] studied the latest development of scalable Monte Carlo algorithm for reducing computational cost of MCMC. In this paper, we focus on the stochastic gradient MCMC (SGMCMC), which uses the data subsampling technique to reduce the cost of each iteration of MCMC and to achieve good results.

In this paper, unlike the non-linear Kalman filter and maximum likelihood method-based estimation way for the MM model (see, e.g., Peng et al. [5]; Peng et al. [6]; Peng et al. [7]; and Qin et al. [9]), we develop an efficient MCMC method to estimate the market microstructure model. Note that the Kalman filter-based simulation smoother algorithm (see, e.g., De Jong [15]; Koopman [16]; De Jong and Shephard [17]; and Durbin and Koopman [18]) and the auxiliary mixture sampler (see, e.g., Kim et al. [19]; Omori et al. [20]; Nakajima and Omori [21]; Delatola and Griffin [22]; and Jensen and Maheu [23]) cannot be directly applied to the MM model, since the market liquidity enters the mean term of return equation. To make an efficient posterior inference easier using WinBUGS (Windows version of Bayesian Analysis Using Gibbs Sampler, a statistical software for Bayesian analysis using MCMC methods), Xi et al. [8] and Xi et al. [10] developed a MCMC method to estimate a MM model. But the status of the chain after a huge number of steps is then used as a sample of the desired distribution, which may lead to complex desired distribution and an inaccurate estimate result. Thus, the modification of the simulation smoother algorithm is required for the estimation of the MM model.

We extend the precision-based simulation smoother algorithm in Chan and Jeliaskov [24] and McCausland et al. [25] to the MM model and other SV models, which was originally designed for linear Gaussian state-space models. The precision-based simulation smoother includes drawing state variables or disturbances in discrete-time state-space model from their conditional distribution given parameters and observations, which can be approximate to the conditional distribution of the state in a non-Gaussian

or non-linear model by its counterpart in a linear Gaussian model. The proposed MCMC method in this paper is based on the precision-based simulation smoother algorithms instead of the conventional Kalman filter. The literatures using the former methods include Rue [26] for linear Gaussian Markov random fields, Chan and Jeliaskov [24] and McCausland et al. [25] for linear Gaussian state-space models, Rue et al. [27] for non-linear Markov random fields, and McCausland [28] and Djegn e and McCausland [29] for non-linear state space models. McCausland et al. [25] provided a detailed comparison between the Kalman filter-based algorithm and precision-based algorithm and showed that the latter is mainly more efficient and faster, since it draws state variables based on band diagonal structure and the Hessian matrix of the logarithmic density is positive definite. Our presented method inherits and sublimates the characteristics of the original precision-based method.

In contrast to the earlier works [3, 30–34], we show that the proposed model in this paper can capture the volatility feedback effect and the effect of the dynamic offset between supply and demand. The proposed method directly simulates smoother log-volatility from its posterior distribution using a proposal distribution, which approximates the target distribution and is different from the auxiliary mixture sampler [19] in which the return equation is first transformed into a linear equation. Our method takes all the market liquidity in the MM model as a single MCMC block, which is different from the single-move sampler [20–24], so it results in efficiency improvement when market liquidity is posterior serial dependence. Unlike the Kalman filter-based simulation smoother [15–18], our method achieves efficiency gains by exploiting the special structure of the problem, particularly in which the Hessian of the log-conditional density of the market liquidity given the return and the surplus demand is a diagonal band and positive matrix. Unlike the Hessian method approximation [28, 29] for the log-likelihood with respect to the log-volatility with the first five derivatives, our method obtains a Gaussian approximation with only the first two derivatives and correct the approximation error with an acceptance-rejection Metropolis–Hastings (ARMH) step [35–38].

The remainder of this paper is structured as follows. In Section 2, we introduce six SV type models including two newly proposed MM models. In Section 3, we present the estimation approach to the market liquidity in the MM model. To this end, we propose a precision-based simulation smoother to obtain a Gaussian approximation of the market liquidity density, and the acceptance-rejection Metropolis–Hastings (ARMH) step is used to improve efficiency of the posterior simulator. In Section 4, using the method proposed in Section 3, we present the efficient posterior simulators for other states and parameters of the SV type models. In Section 5, we show the simulation study result for SV model and MM model using the proposed estimation method. In Section 6, we give the modeling method application results for real data using the six SV type models that are estimated by the proposed estimation methods. Some conclusions are given in Section 7.

## 2. Preliminary: Problem Description

This paper studies the financial time series modeling issues using stochastic process models. The proposed estimation method for the stochastic models may be also used to the stochastic signal processing problems in other fields (see, e.g., Valipour [39]; Valipour et al. [40]; and Valipour [41]). Financial time series analysis is mainly to analyze statistical properties of return. In finance, one-period simple return from date  $t - 1$  to date  $t$  is  $R_t = (p_t - p_{t-1})/p_{t-1}$ , where  $p_t$  is the asset price. The one-period simple compound return is  $1 + R_t = p_t/p_{t-1}$ . For convenience, the log return  $r_t = \log(p_t/p_{t-1}) = \log(p_t) - \log(p_{t-1})$  is usually used in modeling; especially in case study, one often uses  $r_t = 100 \log(p_t/p_{t-1})$  to represent the return for clearly showing the number.

Since the avant-garde works of Box and Jenkins, the autoregressive (AR) model and autoregressive moving average (ARMA) models have become typical tools for modeling and detecting time series (see, e.g., Valipour et al. [42]). Although the theoretical proof and empirical results of such models have been verified, a large number of literatures emphasized the importance of introducing time-varying volatility into financial time series for estimation and forecasting. In finance, volatility refers to the degree of variation of transaction price series over time, which is measured by the standard deviation of return. Standard AR and ARMA models with constant variance (or constant volatility) are seemingly not flexible enough for financial time series modeling. Because the stochastic volatility (SV) model (Taylor [1]) supports time-varying stochastic volatility (exponential variance), the SV model is widely used in financial data modeling. The general form of the SV model is to treat the logarithmic volatility as a first-order autoregressive process, as shown in the following equation:

$$r_t = \mu + e^{\sigma_t/2} \varepsilon_t^r, \quad (1)$$

$$\sigma_t = \gamma + \delta(\sigma_{t-1} - \gamma) + \eta \varepsilon_t^\sigma, \quad (2)$$

where  $\varepsilon_t^r$  and  $\varepsilon_t^\sigma$  are mutually and serially uncorrelated Gaussian noise with zero mean and unit variance,  $\mu$  is the mean of return,  $\gamma$  is the mean of long-term volatility in (2),  $\delta$  is the rate where the volatility reverts toward its long-term mean, and  $\eta$  is the volatility during the volatility process in (2). In the model,  $|\delta| < 1$  and the logarithmic volatility is initialized with  $NID(\gamma, \eta^2/(1 - \delta^2))$ , where  $NID(\cdot, \cdot)$  denotes the normal and independent Gaussian distribution.

The relationship between the expected stock index returns and conditional volatility has attracted much attention in financial economic literature, in which the volatility feedback effect is an important property. Financial volatility feedback is the effect of volatility to return in financial market. Schwert et al. [30] and Campbell and Hentschel [31] gave the explanation on the volatility feedback effect as follows. If volatility is priced, an expected increase in volatility would rise at the requisite rate of return; conversely, an immediate stock-price decline is needed in

order to get higher future returns. Therefore, the potential causality of the volatility feedback effect runs from volatility to prices. A positive relation between expected returns and expected volatility conforms to the capital asset pricing model (CAPM), since rational risk-averse investors need higher expected returns during more variable periods (see, e.g., Koopman and Uspensky [32]). Unfortunately, the SV model cannot capture the financial volatility feedback effect and measure the relationship between expected return and expected volatility. To solve the problem, Koopman and Uspensky [32] proposed the stochastic volatility-in-mean (SVM) model, including the unobservable volatility square as an explanatory variable in the mean term of the return equation as follows:

$$r_t = \phi e^{\sigma_t} + e^{\sigma_t/2} \varepsilon_t^r, \quad (3)$$

$$\sigma_t = \gamma + \delta(\sigma_{t-1} - \gamma) + \eta \varepsilon_t^\sigma, \quad (4)$$

where the regression coefficient  $\phi$  measures the volatility-in-mean effect or volatility feedback effect; when  $\phi = 0$ , the SVM model becomes SV models (1) and (2). The SVM model builds a strong ex ante relationship between expected return and expected volatility, and the volatility feedback effect could be interpreted as a stronger ex ante relationship between the expected returns and expected volatility.

To extract more information from financial data, in this paper, we study the market microstructure model proposed by Bouchaud and Cont [3], which is described by the following equations:

$$dP_t = \lambda \phi_t dt, \quad (5)$$

$$\phi_t = \phi_t^+ - \phi_t^-, \quad (6)$$

where  $P_t = \log(p_t)$  is the log price of an asset,  $\phi_t$  is the market surplus demand, and  $\lambda$  is the market liquidity. At any given instant of time, there is an instantaneous demand  $\phi_t^+$  and an instantaneous supply  $\phi_t^-$  for the asset. The dynamical equations (5) and (6) describe the impact of supply-demand balance, i.e.,  $\phi_t$  tends to push the price up (if  $\phi_t^+ > \phi_t^-$ ) or down (if  $\phi_t^+ < \phi_t^-$ ) in the other case, so it is named as surplus demand.  $\lambda$  is a measurement of market depth, i.e., the surplus demand is required to remove the price by a unit. When inverse of  $\lambda$  is high, the market is able to “absorb” supply/demand offsets by the little price changes, so it is named as market liquidity, which is a market’s capability to buy or sell an asset without making big change in the asset’s price (see, e.g., Bouchaud and Cont [3]). The dynamic equation (5) assumes that the price  $P_t$  is driven by the surplus demand  $\phi_t$ , and the range of price changes depends on the market liquidity  $1/\lambda$ .

However, in the original setting, models (5) and (6) only provide an abstract description to the dynamics of stock market because the surplus demand  $\phi_t$  and the market liquidity  $1/\lambda$  cannot be easily measured directly. Iino and Ozaki [4] treated the market liquidity  $1/\lambda_t$  and the surplus demand  $\phi_t$  as stochastic processes and proposed a market microstructure model, which consists of a set of continuous

time stochastic differential equations. Peng et al. (see e.g., Peng et al. [5]; Peng et al. [6]; Peng et al. [7]; Xi et al. [8]; Qin et al. [9]; Xi et al. [10]), also extended the market microstructure model in continuous-time domain and discrete-time domain. More generally, in this paper, we build the continuous-time market microstructure model as follows:

$$\begin{aligned} dP_t &= \lambda_t \phi_t dt + \sqrt{\lambda_t} dw_{P,t}, \\ d\phi_t &= (\alpha + \beta(\phi_t - \alpha))dt + \kappa dw_{\phi,t}, \\ d\log(\lambda_t) &= (\gamma + \delta(\log(\lambda_t) - \gamma))dt + \eta dw_{\lambda,t}, \end{aligned} \quad (7)$$

where  $w_{P,t}$ ,  $w_{\phi,t}$ , and  $w_{\lambda,t}$  are the independent Wiener processes with unit variance and  $\alpha$ ,  $\gamma$ ,  $\beta$ ,  $\delta$ ,  $\kappa$ , and  $\eta$  are the constant parameters. In the related literatures, a continuous model is often discretized to make estimation easy. In mathematics, the Euler–Maruyama method is a simple and common method for solving approximate numerical solutions of stochastic differential equations. Using the Euler–Maruyama discrete-time approximation method, we can build a discrete-time market microstructure (MM) model as follows:

$$r_t = \phi_t e^{\sigma_t} + e^{\sigma_t/2} \varepsilon_t^r, \quad (8)$$

$$\phi_t = \alpha + \beta(\phi_{t-1} - \alpha) + \kappa \varepsilon_t^\phi, \quad (9)$$

$$\sigma_t = \gamma + \delta(\sigma_{t-1} - \gamma) + \eta \varepsilon_t^\sigma, \quad (10)$$

where  $\sigma_t \equiv \log(\lambda_t)$ ,  $r_t \equiv P_t - P_{t-1}$ , and  $\varepsilon_t^r$ ,  $\varepsilon_t^\phi$ , and  $\varepsilon_t^\sigma$  are mutually and serially uncorrelated Gaussian noises with zero mean and unit variance. Equation (8) characterizes the financial asset price process. It is obvious that the conditional expected mean value and the conditional variance of the financial asset price process in (8) are given by  $\phi_t e^{\sigma_t}$  and  $e^{\sigma_t}$ , respectively. Equations (9) and (10) describe the unobservable surplus demand  $\phi_t$  and market liquidity  $\sigma_t$  as an autoregressive process, respectively. The MM models (10)–(12) are similar to the SVM model, in which the unobservable logarithmic volatility or market liquidity is described by an autoregressive process.

In contrast to the SV model, MM models (10)–(12) place a strong ex ante relation between expected return and expected volatility and can capture the volatility feedback effect. Different from the general SVM model, MM models (10)–(12) place a time-varying surplus demand  $\phi_t$  instead of a constant coefficient  $\phi$  used in SVM model, and thus MM models (10)–(12) can capture the effect of the dynamic offset between supply and demand. So, MM models (10)–(12) are more flexible stochastic volatility models, and one may utilize the surplus demand process that is a more stable and predictable process, rather than the highly random return process, to assist decision making in financial investment process. It is therefore suggested that one may allocate assets by monitoring and making use of the surplus demand information. This methodology was

implemented in, e.g., Peng et al. [5], Peng et al. [6], Peng et al. [7], and Qin et al. [9].

Besides stochastic volatility, another prominent characteristic in typical financial time series is the existence of outliers. The stochastic volatility model with Student-t thick tail distribution (SVt) is a common method to study this characteristic. Evidence in favor of fat tails in financial data has been uncovered by Geweke [33], Gallant et al. [34], and Jacquier et al. [43]. Since the Student-t distribution can be written as a scale mixture of Gaussian distributions (see, e.g., Geweke [44]), the potential variable representation of the SVt model is as follows:

$$r_t = \mu + e^{\sigma_t/2} \lambda_t^{1/2} \varepsilon_t^r, \quad (11)$$

$$\sigma_t = \gamma + \delta(\sigma_{t-1} - \gamma) + \eta \varepsilon_t^\sigma, \quad (12)$$

$$(\lambda_t | v) \sim IG\left(\frac{v}{2}, \frac{v}{2}\right), \quad (13)$$

where  $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_T)^T$  are conditionally independent, given the model parameters (including  $v$ ) and data, and  $IG(\cdot, \cdot)$  denotes the inverse-gamma distribution. Because the Student-t distribution has a heavier tail than the Gaussian distribution, outliers are allowed to occur more frequently in the SVt model than in the standard SV model. In particular, by utilizing the Student-t heavy tails, we can generalize the original specification in the SVM model and MM model and obtain their counterparts, i.e., the SVM plus Student-t heavy tail (SVMt) model:

$$r_t = \phi e^{\sigma_t} + e^{\sigma_t/2} \lambda_t^{1/2} \varepsilon_t^r, \quad (14)$$

$$\sigma_t = \gamma + \delta(\sigma_{t-1} - \gamma) + \eta \varepsilon_t^\sigma, \quad (15)$$

$$(\lambda_t | v) \sim IG\left(\frac{v}{2}, \frac{v}{2}\right), \quad (16)$$

and the MM plus Student-t heavy tail (MMt) model:

$$r_t = \phi_t e^{\sigma_t} + e^{\sigma_t/2} \lambda_t^{1/2} \varepsilon_t^r, \quad (17)$$

$$\phi_t = \alpha + \beta(\phi_{t-1} - \alpha) + \kappa \varepsilon_t^\phi, \quad (18)$$

$$\sigma_t = \gamma + \delta(\sigma_{t-1} - \gamma) + \eta \varepsilon_t^\sigma, \quad (19)$$

$$(\lambda_t | v) \sim IG\left(\frac{v}{2}, \frac{v}{2}\right). \quad (20)$$

All six models studied in this paper are summarized in Table 1. It is worth noting that the structures of MM models (10)–(12) and MMt models (19)–(22) are different from those of MM and MMt models previously studied [5, 8–10]. In the return equation of the proposed MM type models in this paper, the mean term is a stochastic variable. However, in the previous works [5, 8–10], the mean term is built by a deterministic function of historical information because the MM model is estimated by a Kalman filter and maximum

TABLE 1: List of different stochastic volatility (SV) models.

Model	Description
SV (1) and (2)	Standard SV models in which $\sigma_t$ follows a stationary AR(1)
SVt (13)–(15)	SV models whose return innovation follows a Student-t distribution
SVM (3) and (4)	SV models but $\sigma_t$ enters the mean term of return equation as a covariate
SVMt (16)–(18)	Same as SVM but the return innovation follows a Student-t distribution
MM (10)–(12)	Similar to SVM but $\phi_t$ follows a stationary AR(1)
MMt (19)–(22)	Same as MM but the return innovation follows a Student-t distribution

likelihood method-based approach, which cannot cope with the stochastic mean term. Therefore, the proposed MM models in this paper can better describe the dynamic behavior between return and market liquidity as well as surplus demand of financial market.

For the estimation of the SVM model, Koopman and Hol Uspensky [32] developed a simulated maximum likelihood estimator built on the Kalman filter-based simulation smoother algorithm and approximated the Gaussian model to fit the SVM model. Since there are only one state and a constant  $\phi$  in the SVM model, it is able to maximize the likelihood numerically to get the maximum likelihood estimate of  $\sigma_t$ . However, this approach cannot be easily generalized to the time-varying surplus demand  $\phi_t$  in MM models (10)–(12) and MMt models (19)–(22) due to the likelihood evaluation in the case that involves two kinds of status,  $\phi_t$  and  $\sigma_t$ . Meanwhile, applying the Kalman filter-based smoother to obtain a smoothed estimate for  $\sigma_t$  leads to an iterative process, which needs a large computation burden. More details were given by Durbin and Koopman [45].

In this paper, we present a fast and efficient MCMC approach based on an efficient simulation smoother algorithm and an acceptance-rejection Metropolis–Hastings algorithm to estimate the models listed in Table 1. First, we directly simulate smoother log-volatility from its posterior distribution by using the proposal distribution, which approximates the target distribution. It is unlike Kalman filter-based simulation smoother and the auxiliary mixture sampler, in which the return equation is firstly transformed into a linear equation, and any non-Gaussian distributions in this transformed model may be close to Gaussian mixtures. However, through modeling the distribution of  $r_t^* = \log(r_t^2)$ , rather than  $r_t$  directly, information about the sign of the return  $r_t$  is lost. Therefore, the distribution of  $r_t$  or  $\varepsilon_t^r$  can only be recovered by assuming the distribution of the sign of the return (see, e.g., Kim et al. [19]; Omori et al. [20]; Nakajima and Omori [21]; Delatola and Griffin [22]; and Jensen and Maheu [23]). Unlike the above approach, the proposed precision-based simulation smoother has a Gaussian approximating model for the conditional return distribution, and it preserves the parametric specification of the log-volatility process.

Second, we take all the market liquidity  $\sigma = (\sigma_1, \sigma_2, \dots, \sigma_T)^T$  as a single MCMC block. It brings efficiency improvement when market liquidity is posterior serial dependence. This is achieved by constructing a Gaussian approximation of the target distribution and correcting the approximation error using the acceptance-

rejection Metropolis–Hastings step. This Gaussian approximation is closer to the target distribution than that obtained by any multivariate Gaussian approximation approach that updates the state vector but usually only for about 10–50 observations at a time (see, e.g., [35–37]).

Third, the new method can improve efficiency by exploiting the special structure of the problem, particularly in which the Hessian of the log-conditional density of the market liquidity  $\sigma_t$  given the return  $r_t$  and the excess demand  $\phi_t$  is a diagonal band and positive matrix. This feature turns out to be crucial in developing a simple, efficient, and fast precision-based simulation smoother for the market liquidity  $\sigma_t$ . Using forward- and back-substitution to simulate smoother market liquidity  $\sigma_t$ , the Cholesky decomposition diagonal band matrix Hessian can be obtained in  $\mathcal{O}(T)$  operations instead of  $\mathcal{O}(T^3)$  operations for full matrices. Meanwhile, applying the precision-based smoother to obtain a smoothed estimate for market liquidity  $\sigma_t$  leads to a very simple and fast Newton–Raphson iterative procedure, since the Hessian of the log-conditional density of the market liquidity  $\sigma_t$  is a positive matrix. These two features are essential for reducing the computational costs.

The last and most important point is the Gaussian approximation of the conditional density of logarithmic volatility. The HESSIAN method of McCausland [28] and Djegnéne and McCausland [29] provided an excellent approximation for the log-likelihood with respect to the log-volatility with the first five derivatives. However, it also needs a substantial burden on the end user, which refers to the first five derivatives, so it is not easily extended to complicated volatility-in-mean structure and the multivariable MM model. We present a simple, efficient, and fast precision-based simulation smoother algorithm to obtain a Gaussian approximation for the conditional density of the log-volatility with only the first two derivatives and to correct the approximation error with an acceptance-rejection Metropolis–Hastings (ARMH) step (see, e.g., Tierney [38]; Chib and Jeliazkov [46]; Dimitrakopoulos and Kolossiatis [47]; and Blasques et al. [48]).

### 3. Efficient Posterior Simulator for the Market Liquidity in MMt Model

In MMt models (19)–(22), since the market liquidity  $\sigma_t$  appears in both mean term and variance term of return equation (17) as volatility and the model is non-linear in the market liquidity  $\sigma_t$ , one cannot directly sample the market liquidity  $\sigma_t$  from the complicate joint conditional density. In this part, we first propose a precision-based simulation

smoother to obtain a Gaussian approximation of the market liquidity density for the non-linear non-Gaussian MMT model. Then, the acceptance-rejection Metropolis–Hastings (ARMH) step is discussed, which may significantly improve the posterior simulator efficiency, since it corrects the approximation error. The efficient posterior simulators for other parameters and other states are presented in Section 4.

### 3.1. Gaussian Approximation of Joint Conditional Density.

There are usually two approaches that are commonly used for estimating non-linear non-Gaussian state-space models. The first is the so-called sequential Monte Carlo method or more popularly known as particle filter [49] that involves sequential importance sampling and bootstrap resampling. Despite recent advances, particle filter is still quite computationally intensive and is not designed for efficient simulation smoother for the states. In fact, in posterior simulation with a particle filter, it is a common practice to generate candidate draws on parameters via a random walk sampler and then use a particle filter to compute the acceptance probability of the candidate draw. More recently, there has been work on using a particle filter to obtain candidate draws for the states (see e.g., Andrieu et al. [50]). This direction looks promising, but most applications using these methods to date are limited to univariate state-space models due to computational limitations.

The alternative approach to estimating general state space models is built on fast approximations of the conditional state density, where the approximating density is used for posterior simulation via the independence-chain Metropolis–Hastings (MH) algorithm. Of course, the main challenges are just crucial to have a fast routine to obtain a good approximation for the density, and it should be easy to generate candidate draws from the approximating density. Durbin and Koopman [45] and Shephard and Pitt [51] considered approximating the log target density around the mode by a second-order Taylor expansion. This approximation provides a Gaussian density, where its mean is the mode of the target density and its precision or inverse variance equals the negative Hessian evaluated at the mode. Candidate draws for the states are then generated via the Kalman filter-based simulation smoother. However, one problem with sampling states in a single block with the Gaussian proposal density is that the acceptance rate in the Metropolis–Hastings (MH) step can be quite low, at least in the context of stochastic volatility models. It is therefore suggested that the states are divided into blocks, and each block is sampled sequentially via the MH step. This methodology was implemented in, e.g., Watanabe and Omori [35], Omori and Watanabe [36], and Abanto-Valle et al. [37]. In addition, the research of Chan [52], Chan and Strachan [53], Zhang [54], and Zhang [55] also brings inspiration to this paper.

Departing from the obvious Gaussian approximation, McCausland [28] and Djegnéné and McCausland [29] introduced the HESSIAN method that provides an excellent approximation, and it results in a highly efficient simulation smoothing sampling algorithm. However, the HESSIAN

method requires the first five derivatives of the log-likelihood with respect to the states, which gives a substantial burden on the end user. A more severe restriction is that currently it can only be applied to univariate state-space models.

In contrast to the HESSIAN method, we present a more simple precision-based simulation smoother algorithm to obtain a Gaussian approximation for the conditional density of the log-volatility with only the first two derivatives. By exploiting the diagonal band and positive matrix structure of the precision matrix, the precision-based algorithm is more efficient than Kalman filter-based methods in general. This feature is crucial since one needs to obtain the approximating density tens of thousands times in a full Bayesian analysis via MCMC. More importantly, the marginal cost of obtaining additional draws under the precision-based algorithm is much smaller compared to Kalman filter-based methods. We exploit this important feature for two purposes: efficient simulation of the states and evaluation of the integrated likelihood. In addition, different from being divided into blocks, we sample the market liquidity in a single block and develop an acceptance-rejection Metropolis–Hastings (ARMH) algorithm to correct the approximation error.

### 3.2. Precision-Based Simulation Smoother of the Market Liquidity.

In this section, a precision-based simulation smoother is built, and a simple, efficient, and fast precision-based smoother and Newton–Raphson iterative procedure are presented to get a Gaussian approximation density for the joint conditional density of the states. For notational convenience, let  $r = (r_1, r_2, \dots, r_T)^T$ ,  $\phi = (\phi_1, \phi_2, \dots, \phi_T)^T$ ,  $\sigma = (\sigma_1, \sigma_2, \dots, \sigma_T)^T$ , and  $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_T)^T$ .

Now, we discuss a quick way to get the Gaussian approximation density for the joint conditional density  $p(\sigma|r, \phi, \lambda, \theta)$  where  $\theta = (\gamma, \delta, \eta^2)$ . By Bayes' theorem, one has  $p(\sigma|r, \phi, \lambda, \theta) \propto p(r|\sigma, \phi, \lambda)p(\sigma|\theta)$ , and the prior density  $p(\sigma|\theta)$  that characterizes the state equation (19) is Gaussian density. By approximating the observation density  $p(r|\sigma, \phi, \lambda)$ , which characterizes return equation (17) via a Gaussian density in  $\sigma$ , one obtains a Gaussian approximation of  $p(\sigma|r, \phi, \lambda, \theta)$  at once. To this end, following equation (17), the log-density  $\log(p(r_t|\sigma_t, \phi_t, \lambda_t))$  related to  $\sigma_t$  is given as follows:

$$\log(p(r_t|\sigma_t, \phi_t, \lambda_t)) = -\frac{\sigma_t}{2} - \frac{(r_t - \phi_t e^{\sigma_t})^2}{(2\lambda_t e^{\sigma_t})} + c1, \quad (21)$$

where  $c1$  is independent of  $\sigma_t$ . It is easy to examine that the first and second partial derivatives of the log-density with respect to  $\sigma_t$  are given by

$$g_t = \frac{\partial \log(p(r_t|\sigma_t, \phi_t, \lambda_t))}{\partial \sigma_t} = -\frac{1}{2} + \frac{(r_t^2 e^{-\sigma_t} - \phi_t^2 e^{\sigma_t})}{(2\lambda_t)},$$

$$h_t = -\frac{\partial^2 \log(p(r_t|\sigma_t, \phi_t, \lambda_t))}{\partial \sigma_t^2} = -\frac{\partial g_t}{\partial \sigma_t} = \frac{(r_t^2 e^{-\sigma_t} + \phi_t^2 e^{\sigma_t})}{(2\lambda_t)}. \quad (22)$$

As the first and second derivatives can be calculated quickly, we should note that when a point  $\hat{\sigma}_t$  is given, one can estimate the log conditional likelihood  $\log(p(r_t|\sigma_t, \phi_t, \lambda_t))$  by making use of a second-order Taylor expansion around  $\hat{\sigma}_t$  to obtain

$$\begin{aligned} \log(q(r_t|\sigma_t, \phi_t, \lambda_t)) &= \log(p(r_t|\hat{\sigma}_t, \phi_t, \lambda_t)) \\ &+ (\sigma_t - \hat{\sigma}_t)g_t - (\sigma_t - \hat{\sigma}_t)^2 \frac{h_t}{2}. \end{aligned} \quad (23)$$

Let  $\log(q(r|\sigma, \phi, \lambda)) = \sum_{t=1}^T \log(q(r_t|\sigma_t, \phi_t, \lambda_t))$ , which can be written in matrix form as follows:

$$\begin{aligned} \log(q(r|\sigma, \phi, \lambda)) &= \log(p(r|\hat{\sigma}, \phi, \lambda)) \\ &+ (\sigma - \hat{\sigma})^T G - (\sigma - \hat{\sigma})^T \frac{H(\sigma - \hat{\sigma})}{2} \\ &= -\sigma^T H \frac{\sigma}{2} + \sigma^T (G + H\hat{\sigma}) + c2, \end{aligned} \quad (24)$$

where  $c2$  is independent of  $\sigma$ ,  $G = (g_1, g_2, \dots, g_T)^T$ ,  $H = \text{diag}(h_1, h_2, \dots, h_T)$ , and  $\hat{\sigma} = (\hat{\sigma}_1, \hat{\sigma}_2, \dots, \hat{\sigma}_T)^T$ . Next, rewrite state equation (19) in matrix form below:

$$A_\sigma \sigma = \mu_\sigma + \varepsilon^\sigma, \varepsilon^\sigma \sim NID(0, \Lambda_\sigma), \quad (25)$$

where  $\mu_\sigma = (\gamma, (1-\delta)\gamma, \dots, (1-\delta)\gamma)^T$ ,  $\Lambda_\sigma = \text{diag}(\eta^2/(1-\delta^2), \eta^2, \dots, \eta^2)$ , and

$$A_\sigma = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ -\delta & 1 & 0 & \dots & 0 \\ 0 & -\delta & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & -\delta & 1 \end{bmatrix}. \quad (26)$$

By a change of variable, one has

$$p(\sigma|\theta) \sim NID\left(A_\sigma^{-1}\mu_\sigma, (A_\sigma^T \Lambda_\sigma^{-1} A_\sigma)^{-1}\right), \quad (27)$$

with log-density

$$\begin{aligned} \log(p(\sigma|\theta)) &= -(\sigma - A_\sigma^{-1}\mu_\sigma)^T A_\sigma^T \Lambda_\sigma^{-1} A_\sigma \frac{(\sigma - A_\sigma^{-1}\mu_\sigma)}{2} + c3 \\ &= -\sigma^T A_\sigma^T \Lambda_\sigma^{-1} A_\sigma \frac{\sigma}{2} + \sigma^T A_\sigma^T \Lambda_\sigma^{-1} \mu_\sigma + c4, \end{aligned} \quad (28)$$

where  $c3$  and  $c4$  are independent of  $\sigma$ . Then, considering the expressions in (24) and (28), one has

$$\begin{aligned} \log(q(\sigma|r, \phi, \lambda, \theta)) &= \log(q(r|\sigma, \phi, \lambda)) + \log(p(\sigma|\theta)) \\ &= -\sigma^T \left(H + A_\sigma^T \Lambda_\sigma^{-1} A_\sigma\right) \frac{\sigma}{2} + \sigma^T (G + H\hat{\sigma} + A_\sigma^T \Lambda_\sigma^{-1} \mu_\sigma) + c5, \end{aligned} \quad (29)$$

where  $c5$  is independent of  $\sigma$ . Since the expression in (29) is the log-kernel of  $NID(\hat{\mu}_\sigma, \hat{\Lambda}_\sigma)$  in  $\sigma$ , where  $\hat{\Lambda}_\sigma^{-1} = H + A_\sigma^T \Lambda_\sigma^{-1} A_\sigma$  is the precision matrix or inverse of the

variance and  $\hat{\mu}_\sigma = \hat{\Lambda}_\sigma (G + H\hat{\sigma} + A_\sigma^T \Lambda_\sigma^{-1} \mu_\sigma)$  is the mean vector,  $p(\sigma|r, \phi, \lambda, \theta)$  can be calculated by the Gaussian density

$$q(\sigma|r, \phi, \lambda, \theta) \sim NID(\hat{\mu}_\sigma, \hat{\Lambda}_\sigma). \quad (30)$$

This Gaussian approximation density is then put into use as the proposal density in the acceptance-rejection Metropolis–Hastings (ARMH) step.

Note that using forward- and back-substitution to simulate smoother market liquidity  $\sigma$ , one can calculate the (banded) Cholesky decomposition  $C_\sigma$  of  $\hat{\Lambda}_\sigma$  such as  $C_\sigma C_\sigma^T = \hat{\Lambda}_\sigma$  and obtain a random draw from  $NID(\hat{\mu}_\sigma, \hat{\Lambda}_\sigma)$  efficiently such that  $\sigma = \hat{\mu}_\sigma + C_\sigma u$  and  $u \sim NID(0, I_T)$ . A crucial result is that the Cholesky decomposition  $C_\sigma$  can be obtained in  $\mathcal{O}(T)$  operations rather than  $\mathcal{O}(T^3)$  operations for full matrices, since  $\hat{\Lambda}_\sigma$  is a band diagonal matrix (see, e.g., Chan and Jeliazkov [24]).

To this end, it remains to select the point  $\hat{\sigma}$  for the Taylor expansion in (21), which is chosen to be the mode of  $p(\sigma|r, \phi, \lambda, \theta)$  and can be quickly obtained by the numerical optimization. Using the Newton–Raphson method, one can quickly locate the maximum of  $q(\sigma|r, \phi, \lambda, \theta)$ , or equivalently that of  $\log(q(\sigma|r, \phi, \lambda, \theta))$ , or to select the point  $\hat{\sigma}$  as the mode of  $p(\sigma|r, \phi, \lambda, \theta)$  or smooth  $\sigma$ . It follows from (29) that negative Hessian of  $\log(q(\sigma|r, \phi, \lambda, \theta))$  evaluated at  $\sigma = \hat{\sigma}$  is  $\hat{\Lambda}_\sigma^{-1}$  and the gradient at  $\sigma = \hat{\sigma}$  is  $-\hat{\Lambda}_\sigma^{-1} \hat{\sigma} + \hat{\Lambda}_\sigma^{-1} \hat{\mu}_\sigma$ . Hence, one can carry out the Newton–Raphson method as follows: initializing with  $\sigma = \sigma^{(1)}$  for some lasting vector  $\sigma^{(1)}$ . For  $i = 1, 2, \dots$ , assume  $\hat{\sigma} = \sigma^{(i)}$  in the evaluation of  $\hat{\Lambda}_{\sigma^{(i)}}^{-1}$  and  $-\hat{\Lambda}_{\sigma^{(i)}}^{-1} \sigma^{(i)} + \hat{\Lambda}_{\sigma^{(i)}}^{-1} \hat{\mu}_{\sigma^{(i)}}$  and compute

$$\sigma^{(i+1)} = \sigma^{(i)} + \hat{\Lambda}_{\sigma^{(i)}} \left( -\hat{\Lambda}_{\sigma^{(i)}}^{-1} \sigma^{(i)} + \hat{\Lambda}_{\sigma^{(i)}}^{-1} \hat{\mu}_{\sigma^{(i)}} \right) = \hat{\mu}_{\sigma^{(i)}}. \quad (31)$$

Note that since  $q(\sigma|r, \phi, \lambda, \theta)$  is Gaussian density, its mode  $\hat{\sigma}$  equals its mean vector  $\hat{\mu}_\sigma$ . Repeating this procedure until some convergence criterion is reached, e.g., we get the mode  $\hat{\sigma} = \sigma^{(i+1)}$  or  $\hat{\mu}_\sigma = \hat{\mu}_{\sigma^{(i)}}$  when  $\sigma^{(i+1)} - \sigma^{(i)} < \varepsilon$  for some prefixed tolerance level  $\varepsilon > 0$ . Also, note that the negative Hessian  $\hat{\Lambda}_\sigma^{-1}$  is positive definite for all  $\sigma$ . This guarantees a fast convergence of the Newton–Raphson method.

**3.3. Acceptance-Rejection Metropolis–Hastings (ARMH) Algorithm.** As mentioned before, one problem with sampling states in a single block with the Gaussian proposal density is that the accepting rate in the MH step can be very low, and it is probably because that the Gaussian approximation density is not sufficiently precise. In this section, different from being divided into blocks, implemented in, e.g., Watanabe and Omori [35], Omori and Watanabe [36], and Abanto-Valle et al. [37], we sample the market liquidity  $\sigma$  in a single block and exploit an acceptance-rejection Metropolis–Hastings (ARMH) algorithm to correct the approximation error.

As its name implies, the ARMH algorithm (Tierney [38]) is a MCMC sampling procedure that links classic accept-reject sampling with the Metropolis–Hastings algorithm. In the setting, the target density is the joint conditional density  $p(\sigma|r, \phi, \lambda, \theta) \propto p(r|\sigma, \phi, \lambda) p(\sigma|\theta)$  of  $\sigma$  and we have the

Gaussian approximation proposal density  $q(\sigma|r, \phi, \lambda, \theta)$  from which we generate candidate draws  $\sigma$ . In the classic accepting-rejecting sampling, an important requirement is that there exists a constant  $c$ , which is selected by equation (34), such that

$$p(\sigma|r, \phi, \lambda, \theta) < cq(\sigma|r, \phi, \lambda, \theta), \quad (32)$$

for  $\sigma$  in support of  $p(\sigma|r, \phi, \lambda, \theta)$ . The ARMH releases the domination condition (32) such that when it is not satisfied for  $\sigma$ , one may resort to the MH algorithm. To show Algorithm 1, it is useful to define the set

$$\mathbb{C} = \{\sigma: p(\sigma|r, \phi, \lambda, \theta) < cq(\sigma|r, \phi, \lambda, \theta)\}, \quad (33)$$

and let  $\mathbb{C}^c$  denote the complement code. Then, the ARMH procedure is as follows.

Adjusting the original Gaussian proposal density  $q(\sigma|r, \phi, \lambda, \theta)$  by the accept-reject step, a better approximation of the target density  $p(\sigma|r, \phi, \lambda, \theta)$  is achieved, i.e., the new proposed density is identical with the target density on the set  $\mathbb{C}$  (albeit different normalizing constants  $c$ ), whereas on  $\mathbb{C}^c$ , the new proposed density is decreased to the original one.

Chib and Jeliazkov [46] presented a practical way to select the constant  $c$  and the trade-off in such a choice, which is outlined here. Notice that if a bigger  $c$  is chosen, the set  $\mathbb{C}$  is larger and one is more likely to accept the candidate  $\sigma^*$ . On the other hand, selecting a smaller  $c$  will reduce the sampling efficiency, so it will generate more rejections during the sampling steps of accepting-rejecting, resulting in increased invalid and large calculations. A practical way to strike a balance between these two conflicting considerations is to set

$$c = \frac{bp(\hat{\sigma}|r, \phi, \lambda, \theta)}{q(\hat{\sigma}|r, \phi, \lambda, \theta)}, \quad (34)$$

where  $\hat{\sigma}$  is the mode of the conditional density  $q(\sigma|r, \phi, \lambda, \theta)$  and  $b$  is, say, between 1 and 5. Such kind of choice would make sure that  $c$  is sufficiently small to decrease the required number of draws from  $q(\sigma|r, \phi, \lambda, \theta)$ , while it is also big enough so that the set  $\mathbb{C}$  contains the mode  $\hat{\sigma}$  and its neighboring points.

#### 4. Bayesian Estimation for Stochastic Volatility Models

Bayesian analysis is a model of inductive inference, which has been applied in many scientific disciplines (see, e.g., Khoshravesh et al. [57]). A typical characteristic of the Bayesian approach is that it allows researchers to use sample (data) and prior (expert judgment) information in a consistent manner when making inferences. In this section, we first discuss efficient posterior simulators for the market microstructure model plus Student-t heavy tails (MMT), which are an extra block to sample the state  $\lambda$  and the parameter  $\nu$ . Then, we outline the efficient posterior simulators for other parameters and other states, which are built

on precision-based simulation smoother algorithm. In order to finish the model specification, we presume independent prior distributions  $\gamma \sim NID(\gamma_0, V_\gamma)$ ,  $\alpha \sim NID(\alpha_0, V_\alpha)$ ,  $\delta \sim NID(\delta_0, V_\delta)$ ,  $\beta \sim NID(\beta_0, V_\beta)$ ,  $\eta^2 \sim IG(v_{\eta^2}, S_{\eta^2})$ ,  $\kappa^2 \sim IG(v_{\kappa^2}, S_{\kappa^2})$ , and  $v \sim UNI(\underline{v}, \bar{v})$ , where  $UNI(\cdot, \cdot)$  denotes the uniform distribution.

*4.1. Posterior Simulators for the State  $\lambda$  and the Parameter  $\nu$  of MMT Models (19)–(22).* Sampling the latent variable  $\lambda$  could be easily done as  $\lambda_1, \lambda_2, \dots, \lambda_T$ , which are conditionally independently, to give the parameters and data. Each  $\lambda_t$  follows an independent inverse-gamma distribution:

$$p(\lambda_t|r_t, \sigma_t, \phi_t) \sim IG\left(\frac{(v+1)}{2}, \frac{(v+d)}{2}\right), \quad (35)$$

where  $d = e^{-\sigma_t}(r_t - \phi_t e^{\sigma_t})^2$ .

Next, since  $p(\nu|\lambda) \propto p(\lambda|\nu)p(\nu)$  and the prior for  $\nu \sim UNI(\underline{\nu}, \bar{\nu})$ , one can first derive the log-density  $\log(p(\nu|\lambda))$  following equation (20) to sample  $p(\nu|\lambda)$ :

$$\begin{aligned} \log(p(\nu|\lambda)) &= \log(p(\lambda|\nu)) + \log(p(\nu)) = \frac{T\nu}{(2 \log(\nu/2))} \\ &\quad - T \log(\Gamma(\nu/2)) - (\nu/2 + 1) \sum_{i=1}^T \log(\lambda_i) \\ &\quad - (\nu/2) \sum_{i=1}^T \lambda_i^{-1} + c_6, \end{aligned} \quad (36)$$

where  $c_6$  is independent of  $\nu$  and  $\Gamma(\cdot)$  denotes the gamma function. It is easy to find out that the first and second derivatives of the log-density referring to  $\nu$  given by

$$\begin{aligned} g &= \partial \log \frac{(p(\nu|\lambda))}{\partial \nu}, \\ &= T \log\left(\frac{\nu}{2}\right) + \frac{T}{2} - T\psi\left(\frac{\nu}{2}\right) - \frac{\sum_{i=1}^T \log(\lambda_i)}{2} - \sum_{i=1}^T \frac{\lambda_i^{-1}}{2}, \quad (37) \\ h &= \partial^2 \log \frac{(p(\nu|\lambda))}{\partial \nu^2} = \frac{T}{2\nu} - T\psi_1\left(\frac{\nu/2}{4}\right), \end{aligned}$$

where  $\psi(x) = \partial \log(\Gamma(x))/\partial x$  and  $\psi_1(x) = \partial^2 \log(\Gamma(x))/\partial x^2$  are the digamma function and trigamma function, respectively.

Since the first and second derivatives can be calculated in a minute, one can construct a Gaussian proposal distribution  $q(\nu|\lambda)$  and maximize  $\log(q(\nu|\lambda))$  using numerical optimization based on the approximation of  $\log(p(\nu|\lambda))$  using second-order Taylor expansion as done in Section 3.2. One can get the mode and the negative Hessian calculated in the mode, denoted as  $\hat{\mu}_\nu$  and  $\hat{\Lambda}_\nu$ , respectively. Then, the independence-chain Metropolis–Hastings step with proposal distribution is implemented as follows:

$$q(\nu|\lambda) \sim NID(\hat{\mu}_\nu, \hat{\Lambda}_\nu), \quad (38)$$

and the proposed draw  $\nu^*$  is accepted with probability

$$\min\left(1, p(\nu^*|\lambda) \frac{q(\nu|\lambda)}{(p(\nu|\lambda)q(\nu^*|\lambda))}\right). \quad (39)$$

4.2. Posterior Simulators for the Parameters  $\theta = (\gamma, \delta, \eta^2)$  of MMt Models (19)–(22). The posterior simulator for the state  $\sigma$  can be obtained as described in Section 3. So, one can obtain posterior simulators for the parameters  $\theta = (\gamma, \delta, \eta^2)$  following equation (19) as follows.

Sampling  $p(\eta^2|\sigma, \gamma, \delta)$  is straightforward, if we assume the prior for  $\eta^2 \sim IG(\nu_\sigma, S_\sigma)$ . It yields

$$p(\eta^2|\sigma, \gamma, \delta) \sim IG(\hat{\nu}_\sigma, \hat{S}_\sigma), \quad (40)$$

where  $\hat{\nu}_\sigma = \nu_\sigma + T/2$ ,  $\hat{S}_\sigma = S_\sigma + ((\sigma_1 - \gamma)^2(1 - \delta^2) + \sum_{t=2}^T (\sigma_t - \gamma - \delta(\sigma_{t-1} - \gamma))^2)/2$ .

Sampling  $p(\gamma|\sigma, \delta, \eta^2)$  can be done easily because the conditional distribution is standard. Especially following  $p(\gamma|\sigma, \delta, \eta^2) \propto p(\gamma)p(\sigma|\gamma, \delta, \eta^2)$  and the prior for  $\gamma \sim NID(\gamma_0, V_\gamma)$ , one has

$$\begin{aligned} \log(p(\gamma|\sigma, \delta, \eta^2)) &= \log(p(\gamma)) + \sum_{t=1}^T \log(p(\sigma_t|\gamma, \delta, \eta^2)), \\ &= -\frac{(\gamma - \gamma_0)^2}{(2V_\gamma)} - (\sigma_1 - \gamma)^2 \frac{(1 - \delta^2)}{(2\eta^2)}, \\ &\quad - \frac{\sum_{t=2}^T (\sigma_t - \delta\sigma_{t-1} - \gamma(1 - \delta))^2}{(2\eta^2)} + c7, \\ &= -\gamma^2 \left( \frac{(1/V_\gamma + (1 - \delta^2)/\eta^2 + \sum_{t=2}^T ((1 - \delta)^2/\eta^2))}{2} \right), \\ &\quad + \gamma \left( \frac{\gamma_0}{V_\gamma} + \sigma_1 \frac{(1 - \delta^2)}{\eta^2} + \sum_{t=2}^T \left( (\sigma_t - \delta\sigma_{t-1}) \frac{(1 - \delta)}{\eta^2} \right) \right) + c8, \end{aligned} \quad (41)$$

where  $c7$  and  $c8$  are independent of  $\gamma$ . As the log-density is quadratic in  $\gamma$ , it is Gaussian and one has

$$p(\gamma|\sigma, \delta, \eta^2) \sim NID(\hat{\mu}_\gamma, \hat{\Lambda}_\gamma), \quad (42)$$

where

$$\begin{aligned} \hat{\Lambda}_\gamma^{-1} &= \frac{1}{V_\gamma} + \frac{(1 - \delta^2)}{\eta^2} + \sum_{t=2}^T ((1 - \delta)^2/\eta^2), \\ \hat{\mu}_\gamma &= \hat{\Lambda}_\gamma \left( \frac{\gamma_0}{V_\gamma} + \sigma_1 \frac{(1 - \delta^2)}{\eta^2} + \sum_{t=2}^T ((\sigma_t - \delta\sigma_{t-1})(1 - \delta)/\eta^2) \right), \end{aligned} \quad (43)$$

where  $\hat{\Lambda}_\gamma^{-1}$  is the precision matrix or inverse of variance and  $\hat{\mu}_\gamma$  is the mean.

Since  $p(\sigma_1|\gamma, \delta, \eta^2)$  is non-standard in  $\delta$ , the conditional density  $p(\delta|\sigma, \gamma, \eta^2)$  is non-standard, and one can carry out an independence-chain Metropolis–Hastings step to draw  $\delta$ . Firstly, following  $p(\delta|\sigma, \gamma, \eta^2) \propto p(\delta)p(\sigma|\gamma, \delta, \eta^2)$  and the prior for  $\delta \sim NID(\delta_0, V_\delta)$ , one has

$$\begin{aligned} \log(p(\delta|\sigma, \gamma, \eta^2)) &= \log(p(\delta)) + \sum_{t=1}^T \log(p(\sigma_t|\gamma, \delta, \eta^2)), \\ &\approx \log(p(\delta)) + \sum_{t=2}^T \log(p(\sigma_t|\gamma, \delta, \eta^2)), \\ &= -\frac{(\delta - \delta_0)^2}{(2V_\delta)} - \frac{\sum_{t=2}^T (\sigma_t - \gamma - \delta(\sigma_{t-1} - \gamma))^2}{(2\eta^2)} + c9, \\ &= -\delta^2 \frac{((1/V_\delta + \sum_{t=2}^T ((\sigma_{t-1} - \gamma)^2/\eta^2))}{2}, \\ &\quad + \delta \left( \frac{\delta_0}{V_\delta} + \sum_{t=2}^T \left( \frac{(\sigma_t - \gamma)(\sigma_{t-1} - \gamma)}{\eta^2} \right) \right) + c10, \end{aligned} \quad (44)$$

where  $c9$  and  $c10$  are independent of  $\delta$ . As the approximate log-density is quadratic in  $\delta$ , it is approximate Gaussian and one has

$$p(\delta|\sigma, \gamma, \eta^2) \sim NID(\hat{\mu}_\delta, \hat{\Lambda}_\delta), \quad (45)$$

where  $\widehat{\Lambda}_\delta^{-1} = 1/V_\delta + \sum_{t=2}^T ((\sigma_{t-1} - \gamma)^2/\eta^2)$  is the precision matrix or inverse of variance and  $\widehat{\mu}_\delta = \widehat{\Lambda}_\delta (\delta_0/V_\delta + \sum_{t=2}^T ((\sigma_t - \gamma)(\sigma_{t-1} - \gamma)/\eta^2))$  is the mean.

Then, using the approximate Gaussian density  $q(\delta|\sigma, \gamma, \eta^2) \sim NID(\widehat{\mu}_\delta, \widehat{\Lambda}_\delta)$  with  $|\delta| < 1$  as proposal density, one can implement an independence-chain Metropolis–Hastings in which the proposal  $\delta^*$  is accepted in the feasible region with probability

$$\min\left(1, \frac{g(\delta^*|\sigma_1, \gamma, \eta^2)}{g(\delta|\sigma_1, \gamma, \eta^2)}\right), \quad (46)$$

where

$$g(\delta|\sigma_1, \gamma, \eta^2) = -(\sigma_1 - \gamma)^2 \frac{(1 - \delta^2)}{(2\eta^2)} + \log \frac{(1 - \delta^2)}{2} - \frac{(\delta - \delta_0)^2}{(2V_\delta)} + c11, \quad (47)$$

where  $c11$  is independent of  $\delta$ .

**4.3. Posterior Simulators for the State  $\phi$  and the Parameters  $\theta = (\alpha, \beta, \kappa^2)$  of MMT Models (19)–(22).** Since  $p(\phi|r, \sigma, \lambda, \theta) \propto p(r|\sigma, \phi, \lambda)p(\phi|\theta)$ ,  $p(r|\sigma, \phi, \lambda)$ , and  $p(\phi|\theta)$  are all Gaussian distributions in  $\phi$ , one immediately obtains a Gaussian distribution  $p(\phi|r, \sigma, \lambda, \theta)$ . Now, write the return equation (17) and exceed demand equation (18) of  $\phi_t$  in matrix form

$$\begin{aligned} r &= A_r \phi + \varepsilon^r, \varepsilon^r \sim NID(0, \Lambda_r), \\ A_\phi \phi &= \mu_\phi + \varepsilon^\phi, \varepsilon^\phi \sim NID(0, \Lambda_\phi), \end{aligned} \quad (48)$$

where  $A_r = \text{diag}(e^{\sigma_1}, e^{\sigma_2}, \dots, e^{\sigma_T})^T$ ,  $\Lambda_r = \text{diag}(\lambda_1 e^{\sigma_1}, \lambda_2 e^{\sigma_2}, \dots, \lambda_T e^{\sigma_T})^T$ ,  $\mu_\phi = (\alpha, (1 - \beta)\alpha, \dots, (1 - \beta)\alpha)^T$ ,  $\Lambda_\phi = \text{diag}(\kappa^2/(1 - \beta^2), \kappa^2, \dots, \kappa^2)$ , and

$$A_\phi = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ -\beta & 1 & 0 & \dots & 0 \\ 0 & -\beta & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & -\beta & 1 \end{bmatrix}. \quad (49)$$

Then,  $p(\phi|\theta) \sim NID(A_\phi^{-1}\mu_\phi, (A_\phi^T \Lambda_\phi^{-1} A_\phi)^{-1})$  and  $p(r|\sigma, \phi, \lambda) \sim NID(A_r \phi, \Lambda_r)$  with log-density

$$\log(p(r|\sigma, \phi, \lambda)) = -\phi^T A_r^T \Lambda_r^{-1} A_r \frac{\phi}{2} + \phi^T A_r^T \Lambda_r^{-1} r + c12, \quad (50)$$

$$\log(p(\phi|\theta)) = -\phi^T A_\phi^T \Lambda_\phi^{-1} A_\phi \frac{\phi}{2} + \phi^T A_\phi^T \Lambda_\phi^{-1} \mu_\phi + c13, \quad (51)$$

where  $c12$  and  $c13$  are independent of  $\phi$ . Combining (49) and (50), one has

$$\begin{aligned} \log(p(\phi|r, \sigma, \lambda, \theta)) &= \log(p(r|\sigma, \phi, \lambda)) + \log(p(\phi|\theta)), \\ &= -\phi^T (A_r^T \Lambda_r^{-1} A_r + A_\phi^T \Lambda_\phi^{-1} A_\phi) \frac{\phi}{2} \\ &\quad + \phi^T (A_r^T \Lambda_r^{-1} r + A_\phi^T \Lambda_\phi^{-1} \mu_\phi) + c14, \end{aligned} \quad (52)$$

where  $c14$  is independent of  $\phi$ . As this log-density is quadratic in  $\phi$ , it is Gaussian and one has

$$p(\phi|r, \sigma, \lambda, \theta) \sim NID(\widehat{\mu}_\phi, \widehat{\Lambda}_\phi), \quad (53)$$

where  $\widehat{\Lambda}_\phi^{-1} = A_r^T \Lambda_r^{-1} A_r + A_\phi^T \Lambda_\phi^{-1} A_\phi$  is the precision matrix or inverse of variance and  $\widehat{\mu}_\phi = \widehat{\Lambda}_\phi (A_r^T \Lambda_r^{-1} r + A_\phi^T \Lambda_\phi^{-1} \mu_\phi)$  is the mean.

Lastly, following equations (18) and (19),  $\alpha$ ,  $\beta$ , and  $\kappa^2$  are sampled the same as those of  $\gamma$ ,  $\delta$ , and  $\eta^2$ , respectively.

For  $\kappa^2$ , it has inverse-gamma posterior density

$$p(\kappa^2|\phi, \alpha, \beta) \sim IG(\widehat{v}_\phi, \widehat{S}_\phi), \quad (54)$$

where  $\widehat{v}_\phi = v_\phi + T/2$ ,  $\widehat{S}_\phi = S_\phi + ((\phi_1 - \alpha)^2 (1 - \beta^2) + \sum_{t=2}^T (\phi_t - \alpha - \beta(\phi_{t-1} - \alpha))^2)/2$ .

For  $\alpha$ , one gets the Gaussian posterior density

$$p(\alpha|\sigma, \beta, \kappa^2) \sim NID(\widehat{\mu}_\alpha, \widehat{\Lambda}_\alpha), \quad (55)$$

where

$$\begin{aligned} \widehat{\Lambda}_\alpha^{-1} &= \frac{1}{V_\alpha} + \frac{(1 - \beta^2)}{\kappa^2} + \sum_{t=2}^T \left( \frac{(1 - \beta^2)}{\kappa^2} \right), \\ \widehat{\mu}_\alpha &= \widehat{\Lambda}_\alpha \left( \frac{\alpha_0}{V_\alpha} + \phi_1 \frac{(1 - \beta^2)}{\kappa^2} + \sum_{t=2}^T \left( (\phi_t - \beta\phi_{t-1}) \frac{(1 - \beta)}{\kappa^2} \right) \right), \end{aligned} \quad (56)$$

where  $\widehat{\Lambda}_\alpha^{-1}$  is the precision matrix or inverse of variance and  $\widehat{\mu}_\alpha$  is the mean.

For  $\beta$ , it has approximate Gaussian posterior density

$$q(\beta|\sigma, \alpha, \kappa^2) \sim NID(\widehat{\mu}_\beta, \widehat{\Lambda}_\beta), \quad (57)$$

where  $\widehat{\Lambda}_\beta^{-1} = 1/V_\beta + \sum_{t=2}^T ((\phi_{t-1} - \alpha)^2/\kappa^2)$  is the precision matrix or inverse of variance and  $\widehat{\mu}_\beta = \widehat{\Lambda}_\beta (\beta_0/V_\beta + \sum_{t=2}^T ((\phi_t - \alpha)(\phi_{t-1} - \alpha)/\kappa^2))$  is the mean.

Then, one implements an independence-chain Metropolis–Hastings in which the proposal  $\beta^*$  is accepted in the feasible region with probability

$$\min\left(1, \frac{g(\beta^*|\phi_1, \alpha, \kappa^2)}{g(\beta|\phi_1, \alpha, \kappa^2)}\right), \quad (58)$$

where  $g(\beta|\phi_1, \alpha, \kappa^2) = -(\phi_1 - \alpha)^2 (1 - \beta^2)/(2\kappa^2) + \log(1 - \beta^2)/2 - (\beta - \beta_0)^2/(2V_\beta)$ .

Now an overall description of Algorithm 2 is given below.

#### 4.4. Bayesian Estimation for Other Stochastic Volatility Models

4.4.1. For Standard Stochastic Volatility Models (1) and (2). We firstly give an overall description of Algorithm 3.

In Step 2 above, sampling  $p(\mu|r, \sigma)$  can be done easily, as the conditional distribution is standard. Since the prior for  $\mu \sim NID(\mu_0, V_\mu)$  and  $p(\mu|r, \sigma) \propto p(r|\sigma, \mu)p(\mu)$ , following (1), one has

$$\log(p(r_t|\sigma_t, \mu)) = -(r_t - \mu)^2 / (2e^{\sigma_t}) + c15, \quad (59)$$

and

$$\begin{aligned} \log(p(\mu|r, \sigma)) &= \log(p(\mu)) + \sum_{t=1}^T \log(p(r_t|\sigma_t, \mu)), \\ &= -\frac{(\mu - \mu_0)^2}{(2V_\mu)} - \sum_{t=1}^T \left( \frac{(r_t - \mu)^2}{(2e^{\sigma_t})} \right) + c16, \\ &= -\mu^2 \frac{(1/V_\mu + \sum_{t=1}^T (e^{-\sigma_t}))}{2} \\ &\quad + \mu \left( \frac{\mu_0}{V_\mu} + \sum_{t=1}^T (r_t e^{-\sigma_t}) \right) + c17, \end{aligned} \quad (60)$$

where  $c15$ ,  $c16$ , and  $c17$  are independent of  $\mu$ . As this log-density is quadratic in  $\mu$ , it is Gaussian and one has

$$p(\mu|r, \sigma) \sim NID(\hat{\mu}_\mu, \hat{\Lambda}_\mu), \quad (61)$$

where  $\hat{\Lambda}_\mu^{-1} = 1/V_\mu + \sum_{t=1}^T (e^{-\sigma_t})$  is the precision matrix or inverse of variance and  $\hat{\mu}_\mu = \hat{\Lambda}_\mu (\mu_0/V_\mu + \sum_{t=1}^T (r_t e^{-\sigma_t}))$  is the mean.

In Step 3(a), the joint conditional density  $p(\sigma|r, \mu, \gamma, \delta, \eta^2)$  is similar to that of the MMT model in which the log-density  $\log(p(r_t|\sigma_t, \mu))$  follows equation (1) instead of equation (17).

$$\log(p(r_t|\sigma_t, \mu)) = -\frac{\sigma_t}{2} - \frac{(r_t - \mu)^2}{(2e^{\sigma_t})} + c18, \quad (62)$$

where  $c18$  is independent of  $\sigma_t$ . Then, starting from equation (61) instead of equation (21),  $\sigma_t$  can be sampled as that of the MMT model.

Lastly, other steps are the same as those of the MMT model.

4.4.2. For Stochastic Volatility in Mean. We firstly give an overall description of Algorithm 4.

In Step 2, Sampling  $\phi$  can be done easily, as the conditional distribution is standard. Since the prior for  $\phi \sim NID(\phi_0, V_\phi)$  and  $p(\phi|r, \sigma) \propto p(r|\sigma, \phi)p(\phi)$ , following (3), one has

$$\log(p(r_t|\sigma_t, \phi)) = -\frac{(r_t - \phi e^{\sigma_t})^2}{(2e^{\sigma_t})} + c19, \quad (63)$$

and

$$\begin{aligned} \log(p(\phi|r, \sigma)) &= \log(p(\phi)) + \log(p(r|\sigma, \phi)), \\ &= \log(p(\phi)) + \sum_{t=1}^T \log(p(r_t|\sigma_t, \phi)), \\ &= -\frac{(\phi - \phi_0)^2}{(2V_\phi)} - \sum_{t=1}^T \left( \frac{(r_t - \phi e^{\sigma_t})^2}{(2e^{\sigma_t})} \right) + c20, \\ &= -\phi^2 \frac{(1/V_\phi + \sum_{t=1}^T (e^{\sigma_t}))}{2} \\ &\quad + \phi \left( \frac{\phi_0}{V_\phi} + \sum_{t=1}^T (r_t) \right) + c21, \end{aligned} \quad (64)$$

where  $c19$ ,  $c20$ , and  $c21$  are independent of  $\phi$ . As the log-density is quadratic in  $\phi$ , it is Gaussian and one has

$$p(\phi|r, \sigma) \sim NID(\hat{\mu}_\phi, \hat{\Lambda}_\phi), \quad (65)$$

where  $\hat{\Lambda}_\phi^{-1} = 1/V_\phi + \sum_{t=1}^T (e^{\sigma_t})$  is the precision matrix or inverse of variance and  $\hat{\mu}_\phi = \hat{\Lambda}_\phi (\phi_0/V_\phi + \sum_{t=1}^T (r_t))$  is the mean.

In Step 3(a), the joint conditional density  $p(\sigma|r, \phi, \gamma, \delta, \eta^2)$  is similar to that of the MMT model in which the log-density  $\log(p(r_t|\sigma_t, \phi))$  follows equation (6) instead of equation (17).

$$\log(p(r_t|\sigma_t, \phi)) = -\frac{\sigma_t}{2} - \frac{(r_t - \phi e^{\sigma_t})^2}{(2e^{\sigma_t})} + c22, \quad (66)$$

where  $c22$  is independent of  $\sigma_t$ . Then, starting from equation (49) instead of equation (21),  $\sigma_t$  can be sampled as that of the MMT model.

Lastly, other steps are the same as those of the MMT model.

4.4.3. For Market Microstructure Model. We firstly give an overall description of Algorithm 5.

In Step 2(a), the joint conditional density  $p(\phi|r, \sigma, \gamma, \delta, \eta^2)$  is similar to that of the MMT model in which  $p(r|\sigma, \phi)$  follows equation (8) instead of equation (17). One can write return equation (8) in matrix form:

$$r = A_r \phi + \varepsilon^r, \quad \varepsilon^r \sim NID(0, \Lambda_r), \quad (67)$$

where  $A_r = \Lambda_r = \text{diag}(e^{\sigma_1}, e^{\sigma_2}, \dots, e^{\sigma_T})^T$ . Then, starting from equation (66) instead of equation (47),  $\phi$  can be sampled as that of the MMT model.

In Step 3(a), the joint conditional density  $p(\sigma|r, \phi, \gamma, \delta, \eta^2)$  is similar to that of the MMT model in which the log-density  $\log(p(r_t|\sigma_t, \phi_t))$  follows equation (8) instead of equation (17).

$$\log(p(r_t|\sigma_t, \phi_t)) = -\sigma_t/2 - (r_t - \phi_t e^{\sigma_t})^2 / (2e^{\sigma_t}) + c22, \quad (68)$$

where  $c22$  is independent of  $\sigma_t$ . Then, starting from equation (67) instead of equation (21),  $\sigma_t$  can be sampled as that of the MMt model.

Lastly, other steps are the same as those of the MMt model.

**4.4.4. For Stochastic Volatility Model plus Student-t Heavy Tails.** We firstly give an overall description of Algorithm 6.

In Step 2, sampling  $p(\mu|r, \sigma, \lambda)$  is similar to that of the MMt model in which the log-density  $\log(p(r_t|\sigma_t, \lambda_t, \mu))$  follows equation (11) instead of equation (1).

$$\log(p(r_t|\sigma_t, \lambda_t, \mu)) = -(r_t - \mu)^2 / (2\lambda_t e^{\sigma_t}) + c23, \quad (69)$$

where  $c23$  is independent of  $\mu$ . Then, starting from equation (68) instead of equation (58),  $\mu$  can be sampled as that of the SV model.

In Step 3(a), the joint conditional density  $p(\sigma|r, \lambda, \mu, \gamma, \delta, \eta^2)$  is similar to that of the MMt model in which the log-density  $\log(p(r_t|\sigma_t, \lambda_t, \mu))$  follows equation (11) instead of equation (17).

$$\log(p(r_t|\sigma_t, \lambda_t, \mu)) = -\frac{\sigma_t}{2} - \frac{(r_t - \mu)^2}{(2\lambda_t e^{\sigma_t})} + c24, \quad (70)$$

where  $c24$  is independent of  $\sigma_t$ . Then, starting from equation (69) instead of equation (21),  $\sigma_t$  can be sampled as that of the MMt model.

In Step 4(a), to sample  $\lambda_t$  is similar to that of the MMt model, and it follows an independent inverse-gamma distribution:

$$p(\lambda_t|r_t, \sigma_t, \mu) \sim IG\left(\frac{(v+1)}{2}, \frac{(v+d)}{2}\right), \quad (71)$$

where  $d = e^{-\sigma_t} (r_t - \mu)^2$ .

Lastly, other steps are the same as those of the MMt model.

**4.4.5. For Stochastic Volatility in Mean plus Student-t Heavy Tails.** We firstly give an overall description of Algorithm 7.

In Step 2, Sampling  $\phi$  is similar to that of the SVM model in which the log-density  $\log(p(r_t|\sigma_t, \lambda_t, \phi))$  follows equation (14) instead of equation (3).

$$\log(p(r_t|\sigma_t, \lambda_t, \phi)) = -\frac{(r_t - \phi e^{\sigma_t})^2}{(2\lambda_t e^{\sigma_t})} + c25, \quad (72)$$

where  $c25$  is independent of  $\phi$ . Then, starting from equation (71) instead of equation (64),  $\phi$  can be sampled as that of the SVM model.

In Step 3(a), the joint conditional density  $p(\sigma|r, \lambda, \phi, \gamma, \delta, \eta^2)$  is similar to that of the MMt model in which the log-density  $\log(p(r_t|\sigma_t, \lambda_t, \phi))$  follows equation (14) instead of equation (17).

$$\log(p(r_t|\sigma_t, \lambda_t, \mu)) = -\frac{\sigma_t}{2} - \frac{(r_t - \mu)^2}{(2\lambda_t e^{\sigma_t})} + c26, \quad (73)$$

where  $c26$  is independent of  $\sigma_t$ . Then, starting from equation (73) instead of equation (19),  $\sigma_t$  can be sampled as that of the MMt model.

In Step 4(a), to sample  $\lambda_t$  is similar to that of the MMt model, and it follows an independent inverse-gamma distribution:

$$p(\lambda_t|r_t, \sigma_t, \phi) \sim IG\left(\frac{(v+1)}{2}, \frac{(v+d)}{2}\right), \quad (74)$$

where  $d = e^{-\sigma_t} (r_t - \phi e^{\sigma_t})^2$ .

Lastly, other steps are the same as those of the MMt model.

In summary, this paper studied the estimation problem of six typical SV models listed in Table 1, and the difference between previous research and our research in this paper is briefly explained in Table 2.

## 5. Simulation Research

With the purpose of evaluating the performance of the proposed precision-based simulation smoother algorithm and Markov Chain Monte Carlo method, we first utilize the simulated datasets to check the method. In this section, we only consider the SV model and the MM model; since the performance of the SV model and the MM model could compare to a voluminous literature, the comparison results can show the effectiveness of our method. In the simulation design, hundred datasets each composed of  $T = 1500$  observations are generated from the two models. For each dataset, we estimate the two models by constructing Markov chain of length 10000 after a burn-in period of 10000.

The parameter values are chosen to be comparable to those estimated from financial daily returns data (measured in percentage). They are also similar to those used in other simulation studies, like those in Kim et al. [19], Jacquier et al. [43], Omori et al. [20], and Pitt et al. [58]. In particular, we set  $\mu = 0$  for the SV model and  $\alpha = 0$  for the MM model. Parameters for the log-volatility or market liquidity transition are made to be  $\gamma = -0.25$ ,  $\delta = 0.97$ , and  $\eta^2 = 0.06$  for both the SV model and the MM model. Parameters for the surplus demand transition are made to be  $\beta = 0.6$  and  $\kappa^2 = 0.005$  for the MM model. These parameters above are chosen by referring to the results given in Tables 3 and 4 in Section 6 for guaranteeing the reasonability of the chosen parameters. These parameters are used to build a SV model and MM model for generating observation data for simulation study so as to evaluate the performance of the proposed precision-based simulation smoother algorithm and MCMC method. In this section, we utilize the simulated datasets generated by the SV model and MM model with the chosen parameters to check the proposed method.

The parameter prior distributions studied in Section 4 are considered. We select the same hyperparameters for the parameters that are common across two models. Moreover, the hyperparameters are selected so that the implied prior means are similar to the estimates of typical financial daily return data. Following  $\gamma \sim NID(\gamma_0, V_\gamma)$ ,  $\delta \sim NID(\delta_0, V_\delta)$ ,

(1) Accept-reject step: generate a draw  $\sigma^* \sim q(\sigma|r, \phi, \lambda, \theta) \sim NID(\widehat{\mu}_\sigma, \widehat{\Lambda}_\sigma)$  and accept  $\sigma^*$  with probability  $\min(1, p(\sigma^*|r, \phi, \lambda, \theta)/cq(\sigma^*|r, \phi, \lambda, \theta))$   
 Continue the above process until a draw  $\sigma^*$  is accepted.

(2) Metropolis–Hastings step: given the current  $\sigma$  and the proposed draw  $\sigma^*$

(a) if  $\sigma \in \mathbb{C}$ , accept  $\sigma^*$  with probability 1;

(b) if  $\sigma \in \mathbb{C}^c$  and  $\sigma^* \in \mathbb{C}$ , accept  $\sigma^*$  with probability  $\min(1, cq(\sigma|r, \phi, \lambda, \theta)/p(\sigma|r, \phi, \lambda, \theta))$

(c) if  $\sigma \in \mathbb{C}^c$  and  $\sigma^* \in \mathbb{C}^c$ , accept  $\sigma^*$  with probability  $\min(1, q(\sigma|r, \phi, \lambda, \theta)p(\sigma^*|r, \phi, \lambda, \theta)/(q(\sigma^*|r, \phi, \lambda, \theta)p(\sigma|r, \phi, \lambda, \theta)))$   
 otherwise return  $\sigma$ .

ALGORITHM 1: Acceptance-rejection Metropolis–Hastings with Gaussian proposal density [8, 24, 38, 56].

*Step 1.* Initialize  $\alpha, \beta, \kappa^2, \gamma, \delta, \eta^2, v$  and  $\sigma, \phi, \lambda$ ;

*Step 2.* Sample  $\alpha, \beta, \kappa^2, \phi|\sigma, \lambda, r$  by

(a) Sampling  $\phi|\alpha, \beta, \kappa^2, \sigma, \lambda, r$  by expression (52),

(b) Sampling  $\kappa^2|\phi, \alpha, \beta$  by expression (53),

(c) Sampling  $\alpha|\phi, \beta, \kappa^2$  by expression (54),

(d) Sampling  $\beta|\phi, \alpha, \kappa^2$  by expression (56);

*Step 3.* Sample  $\gamma, \delta, \eta^2, \sigma|\phi, \lambda, r$  by

(a) Sampling  $\sigma|\gamma, \delta, \eta^2, \phi, \lambda, r$  by expression (30),

(b) Sampling  $\eta^2|\sigma, \gamma, \delta$  by expression (39),

(c) Sampling  $\gamma|\sigma, \delta, \eta^2$  by expression (41),

(d) Sampling  $\delta|\sigma, \gamma, \eta^2$  by expression (44);

*Step 4.* Sample  $v, \lambda|\sigma, \phi, r$  by

(a) Sampling  $\lambda|v, \sigma, \phi, r$  by expression (34),

(b) Sampling  $v|\lambda$  by expression (37);

*Step 5.* Go to Step 2.

ALGORITHM 2: The MCMC algorithm for estimation of MMt models (19)–(22) [8, 10, 24, 56].

*Step 1.* Initialize  $\mu, \gamma, \delta, \eta^2$  and  $\sigma$ ;

*Step 2.* Sample  $\mu|\sigma, r$ ;

*Step 3.* Sample  $\gamma, \delta, \eta^2, \sigma|\mu, r$  by

(a) Sampling  $\sigma|\gamma, \delta, \eta^2, \mu, r$ ,

(b) Sampling  $\eta^2|\sigma, \gamma, \delta$ ,

(c) Sampling  $\gamma|\sigma, \delta, \eta^2$ ,

(d) Sampling  $\delta|\sigma, \gamma, \eta^2$ ;

*Step 4.* Go to Step 2.

ALGORITHM 3: The MCMC algorithm for estimation of SV models (1) and (2) [1, 24, 30, 31, 56].

*Step 1.* Initialize  $\phi, \gamma, \delta, \eta^2$  and  $\sigma$ ;

*Step 2.* Sample  $\phi|\sigma, r$ ;

*Step 3.* Sample  $\gamma, \delta, \eta^2, \sigma|\phi, r$  by

(a) Sampling  $\sigma|\gamma, \delta, \eta^2, \phi, r$ ,

(b) Sampling  $\eta^2|\sigma, \gamma, \delta$ ,

(c) Sampling  $\gamma|\sigma, \delta, \eta^2$ ,

(d) Sampling  $\delta|\sigma, \gamma, \eta^2$ ;

*Step 4.* Go to Step 2.

ALGORITHM 4: The MCMC algorithm for estimation of SVM model (3) and (4) [24, 32, 56].

*Step 1.* Initialize  $\alpha, \beta, \kappa^2, \gamma, \delta, \eta^2$  and  $\sigma, \phi$ ;  
*Step 2.* Sample  $\alpha, \beta, \kappa^2, \phi | \sigma, r$  by  
 (a) Sampling  $\phi | \alpha, \beta, \kappa^2, \sigma, r$ ,  
 (b) Sampling  $\kappa^2 | \phi, \alpha, \beta$ ,  
 (c) Sampling  $\alpha | \phi, \beta, \kappa^2$ ,  
 (d) Sampling  $\beta | \phi, \alpha, \kappa^2$ ;  
*Step 3.* Sample  $\gamma, \delta, \eta^2, \sigma | \phi, r$  by  
 (a) Sampling  $\sigma | \gamma, \delta, \eta^2, \phi, r$ ,  
 (b) Sampling  $\eta^2 | \sigma, \gamma, \delta$ ,  
 (c) Sampling  $\gamma | \sigma, \delta, \eta^2$ ,  
 (d) Sampling  $\delta | \sigma, \gamma, \eta^2$ ;  
*Step 4.* Go to Step 2.

ALGORITHM 5: The MCMC algorithm for estimation of MM models (10)–(12) [5, 10, 24, 56].

*Step 1.* Initialize  $\mu, \gamma, \delta, \eta^2, \nu$  and  $\sigma, \lambda$ ;  
*Step 2.* Sample  $\mu | \sigma, \lambda, r$ ;  
*Step 3.* Sample  $\gamma, \delta, \eta^2, \sigma | \mu, \lambda, r$  by  
 (a) Sampling  $\sigma | \gamma, \delta, \eta^2, \mu, \lambda, r$ ,  
 (b) Sampling  $\eta^2 | \sigma, \gamma, \delta$ ,  
 (c) Sampling  $\gamma | \sigma, \delta, \eta^2$ ,  
 (d) Sampling  $\delta | \sigma, \gamma, \eta^2$ ;  
*Step 4.* Sample  $\nu, \lambda | \sigma, \mu, r$  by  
 (a) Sampling  $\lambda | \nu, \sigma, \mu, r$ ,  
 (b) Sampling  $\nu | \lambda$ ;  
*Step 5.* Go to Step 2.

ALGORITHM 6: The MCMC algorithm for estimation of SVt models (13)–(15) [8, 24, 43, 44, 56].

*Step 1.* Initialize  $\phi, \gamma, \delta, \eta^2, \nu$  and  $\sigma, \lambda$ ;  
*Step 2.* Sample  $\phi | \sigma, \lambda, r$  by  
*Step 3.* Sample  $\gamma, \delta, \eta^2, \sigma | \phi, \lambda, r$  by  
 (a) Sampling  $\sigma | \gamma, \delta, \eta^2, \phi, \lambda, r$ ,  
 (b) Sampling  $\eta^2 | \sigma, \gamma, \delta$ ,  
 (c) Sampling  $\gamma | \sigma, \delta, \eta^2$ ,  
 (d) Sampling  $\delta | \sigma, \gamma, \eta^2$ ;  
*Step 4.* Sample  $\nu, \lambda | \sigma, \phi, r$  by  
 (a) Sampling  $\lambda | \nu, \sigma, \phi, r$ ,  
 (b) Sampling  $\nu | \lambda$ ;  
*Step 5.* Go to Step 2.

ALGORITHM 7: The MCMC algorithm for estimation of SVMt models (16)–(18) [8, 24, 43, 44, 56].

$\beta \sim NID(\beta_0, V_\beta)$ ,  $\eta^2 \sim IG(v_{\eta^2}, S_{\eta^2}) = IG(v_\sigma, S_\sigma)$ , and  $\kappa^2 \sim IG(v_{\kappa^2}, S_{\kappa^2}) = IG(v_\phi, S_\phi)$ , we set  $\gamma_0 = 0$ ,  $V_\gamma = 10$ ,  $\delta_0 = 0.97$ ,  $V_\delta = 0.01$ ,  $v_\sigma = 11$ ,  $S_\sigma = 0.6$ ,  $\beta_0 = 0.7$ ,  $V_\beta = 0.01$ ,  $v_\phi = 11$ , and  $S_\phi = 0.05$ . These parameters are the hyperparameters of the conjugate prior distributions for the model parameters, which describe the prior information of the model parameters. As a MCMC Bayesian method, our method uses the observation data and prior information of model parameters to infer the posterior information of

model parameters. The hyperparameters of the prior distributions may be given intuitively. If you set enough long MCMC and long dataset, it should result in the same hyperparameters for the posterior distribution which is only contingent on the observation data and is independent of the hyperparameters for prior distributions.

Empirically, in order to accelerate MCMC convergence, in this simulation study, as the regression coefficient  $\delta$  of the built SV models (1) and (2) and MM models (10)–(12) is set

TABLE 2: Difference between previous research and our research in this paper.

Methods	Main characteristics
Kalman filter and maximum likelihood method-based approach [5, 10, 19, 29–32]	Needs a large computation burden. Not good for non-Gaussian non-linear data.
MCMC method based on Metropolis–Hastings sampling [56]	For high-dimensional data, it has two shortcomings: one is to need quite large computation burden and another is difficulty in solving the joint distribution of variables.
MCMC method based on Gibbs sampling [8, 10]	Gibbs sampling is the application of the Metropolis–Hastings sampling in special cases. It does not need to judge the state reception. It has fast convergence speed and high sampling efficiency.
MCMC method based on auxiliary mixture sampler [23, 24, 27]	Loses precision due to transforming the return equation to a linear equation.
The proposed MCMC method in this paper	A fast and efficient MCMC approach based on a precision-based simulation smoother algorithm and an acceptance-rejection Metropolis–Hastings algorithm for estimation of non-linear non-Gaussian SV type models. Uses the band diagonal structure and positive definition of Hessian matrix of the logarithmic density to achieve a small computation burden and high precision.

TABLE 3: Parameter posterior means, standard deviations (in parentheses), inefficiency factors (in braces), and 95% intervals (in square brackets) for the SV and SVM models using Kalman filter-based simulation smoother algorithm.

	SV	SVM
M	0.1051 (0.01924) {2.149} [0.06650, 0.1431]	
$\phi$		0.041144 (0.022499) {1.172} [−0.002488, 0.084986]
$\gamma$	−0.2456 (0.2406) {1.383} [−0.7101, 0.2373]	−0.2725 (0.2553) {1.542} [−0.7593, 0.2532]
$\delta$	0.9691 (0.008464) {22.89} [0.9515, 0.9847]	0.9742 (0.007807) {20.99} [0.9578, 0.9883]
$\eta^2$	0.06961 (0.01349) {59.73} [0.04616, 0.09785]	0.05223 (0.01067) {87.73} [0.03515, 0.07667]
$Q_{LB}$	16.72 (2.212)	12.74 (2.040)
$Q_{ML}$	18.70 (4.926)	18.17 (5.047)

Note.  $Q_{LB}$  and  $Q_{ML}$  are the Ljung–Box and McLeod–Li statistics of order 20 computed on the standard residuals and squared standard residuals, and the numerical standard mistakes are in parentheses. The 0.05 and 0.01 important values are 31.41 and 37.57, respectively.

TABLE 4: Parameter posterior means, standard deviations (in parentheses), inefficiency factors (in braces), and 95% intervals (in square brackets) for the stochastic volatility models by using the precision-based simulation smoother method.

	SV	SVM	MM
M	0.1036 (0.01878) {2.411} [0.06645, 0.1409]		
$\phi$		0.04071 (0.02261) {1.177} [−0.003155, 0.08544]	
$\gamma$	−0.2772 (0.2450) {1.998} [−0.7547, 0.2248]	−0.2358 (0.2660) {1.776} [−0.7216, 0.2865]	−0.2418 (0.2663) {1.481} [−0.7388, 0.2973]
$\delta$	0.97079 (0.008393) {26.80} [0.9531, 0.9861]	0.9748 (0.007572) {19.72} [0.9588, 0.9885]	0.9749 (0.007630) {23.73} [0.95901, 0.9885]
$\eta^2$	0.06342 (0.01289) {65.44} [0.04163, 0.09177]	0.04991 (0.009473) {53.76} [0.03414, 0.07103]	0.04931 (0.01004) {75.65} [0.03220, 0.07193]
$\alpha$			0.04437 (0.02477) {50.46} [−0.004412, 0.09382]
$\beta$			0.6375 (0.06925) {27.89} [0.4976, 0.7694]
$\kappa^2$			0.004579 (0.001315) {99.00} [0.002510, 0.007710]
$Q_{LB}$	16.78 (2.167)	12.77 (1.966)	13.69 (2.299)
$Q_{ML}$	18.39 (4.904)	17.84 (4.741)	17.43 (4.809)

Note.  $Q_{LB}$  and  $Q_{ML}$  are, respectively, the Ljung–Box and McLeod–Li statistics of order 20 computed on the standard residuals and squared standard residuals, and the numerical standard mistakes are in parentheses. The 0.05 and 0.01 critical values are 31.41 and 37.57, respectively.

as  $\delta = 0.97$ , for the hyperparameters  $\delta_0, V_\delta$ , ( $\delta \sim NID(\delta_0, V_\delta)$ ), we set the mean  $\delta_0 = 0.97$  and a small variance  $V_\delta = 0.01$  for making the MCMC convergence fast.

Because the regression coefficient  $\beta$  of the built MM models (10)–(12) is set as  $\beta = 0.6$  in this simulation study, for the hyperparameters  $\beta_0, V_\beta$ , ( $\beta \sim NID(\beta_0, V_\beta)$ ), we set the

mean  $\beta_0 = 0.7$  that is close to the regression coefficient  $\beta$  and a small variance  $V_\beta = 0.01$  for making the MCMC convergence fast.

Because the regression coefficient  $\gamma$  of the built SV models (1) and (2) and MM models (10)–(12) is set as  $\gamma = -0.25$  in this simulation study, for the hyperparameters  $\gamma_0, V_\gamma$ , ( $\gamma \sim NID(\gamma_0, V_\gamma)$ ), we set the mean  $\gamma_0 = 0$  that is close to the regression coefficient  $\gamma$  and a simple large variance  $V_\gamma = 10$  since the mean of the volatility process,  $\gamma \sim NID(\gamma_0, V_\gamma)$ , has large disparity for different financial data; usually, it is small for exchange rate and is large for stock.

For the volatility of the state,  $\eta^2 \sim IG(v_\sigma, S_\sigma) = IG(11, 0.6)$  and  $\kappa^2 \sim IG(v_\phi, S_\phi) = IG(11, 0.05)$ , in financial economic viewpoint for investment arbitrage, the smaller the volatility of the state, the better, so we select sufficient simple small mean  $S/(\nu - 1)$  and variance  $S^2/[(\nu - 1)^2(\nu - 2)]$  for the  $\eta^2$  and  $\kappa^2$ , and thus we set  $\eta^2 = S_\sigma/(\nu_\sigma - 1) = 0.6/11 - 1 = 0.06$  and  $\kappa^2 = S_\phi/(\nu_\phi - 1) = 0.05/11 - 1 = 0.005$  for the built SV models (1) and (2) and MM models (10)–(12) that are used to generate the observation data for simulation study.

Figure 1 depicts the posterior densities of the parameters for the SV model and the MM model, respectively, and it shows that the most posterior samples are close to the true values, and then it is clear that all parameters are estimated very precisely with small variations across simulated samples (see e.g., Kim et al. [19]; Chib et al. [56]; Omori et al. [20]; and Nakajima and Omori [21]). The accepting rate of the acceptance-rejection Metropolis–Hastings step to sample  $\sigma$  is 89.4% and 89.3% for the SV model and the MM model, respectively. This indicates that the Gaussian proposal density is quite similar to the conditional posterior density of  $\sigma$ , and the precision-based simulation smoothing algorithm is effective. The accepting rates of the Metropolis–Hastings step to sample  $\delta$  and  $\beta$  are all larger than 90%, and it indicates the truncate Gaussian proposal density will be close to the conditional posterior density of  $\delta$  and  $\beta$ .

Tables 5 and 6 show the estimated posterior means, the standard deviations, the 95% credible intervals, and inefficiency factors calculated in the simulated datasets. The posterior means are very close to the true values, and all true values are contained in the 95% credible intervals. The posterior means are computed based on 100 independent datasets, each of which is of length 10000 after a burn-in period of 10000. Using a total of one million draws for each model, the estimated parameters are receivable and accurate.

To measure how well the chain mixes, we calculate the inefficiency factors (the inverse of inefficiency factor is also known as numerical efficiency in Geweke [59]), which are given in Tables 5 and 6. The inefficient factor (equal to the autocorrelation time) is defined as

$$\text{factor} = 1 + 2 \sum_{l=1}^L \rho_l, \quad (75)$$

where  $\rho_l$  is the sample autocorrelation at lag  $l$ , which is calculated from the sampled values.  $L$  is chosen to be large enough so that the autocorrelation tapers off, which is

calculated based on an autocorrelation rule of thumb. The inefficiency factor is also the ratio of the numerical variance of the posterior sample mean to the variance of the posterior sample mean from the hypothetical uncorrelated draws. Thus, it suggests the relative number of correlated draws necessary to attain the same variance of the posterior sample mean from the uncorrelated draws. In the ideal case where the posterior draws are independent, the corresponding inefficiency factor is 1. More generally, the inefficiency factor measures how many extra draws are needed to give results equivalent to this ideal case. For example, an inefficiency factor of 100 indicates that roughly 10000 posterior draws are required to give the same information as 100 independent draws.

## 6. Empirical Applications

**6.1. The Dataset and Hyperparameters for the Parameters.** In this section, we consider applications of the proposed method to a dataset that has been extensively analyzed in the finance literature. The data series for the study consists of daily continuously compounded returns,  $r_t = 100 \log(p_t/p_{t-1})$ , in percentage, on the Standard and Poor's 500 (S&P 500) index (computed without considering dividends) from January 2009 to December 2014, with a total of  $T = 1509$  observations. A plot of the data is given in Figure 2. The S&P 500 index time series is stable, and there is no obvious market jump which is the problem we need to study in the future. The S&P 500 index is a popular benchmark index of large-cap stocks in the United States, which is a price index, meaning it represents the stock prices of the companies within the index.

We use the priors given in Section 3.1 and set the same hyperparameters for the parameters that are common across models as follows:  $\mu_0 = 0$ ,  $V_\mu = 10$ ,  $\phi_0 = 0$ ,  $V_\phi = 10$ ,  $\gamma_0 = 0$ ,  $V_\gamma = 10$ ,  $V_\delta = 0.01$ ,  $\nu_\sigma = 11$ ,  $\alpha_0 = 0$ ,  $V_\alpha = 10$ ,  $V_\beta = 0.01$ ,  $\nu_\phi = 11$ ,  $S_\phi = 0.05$ ,  $\underline{\nu} = 2$ , and  $\bar{\nu} = 100$ . The other hyperparameters for the parameters are as follows:  $\delta_0 = 0.97$  for the SV model,  $\delta_0 = 0.975$  for the SVM model, the MM model, and the SVt model,  $\delta_0 = 0.98$  for the SVMt model and the MMt model,  $S_\sigma = 0.65$  for the SV model,  $S_\sigma = 0.5$  for the SVM model and the MM model,  $S_\sigma = 0.45$  for the SVt model,  $S_\sigma = 0.35$  for the SVMt model and the MMt model,  $\beta_0 = 0.65$  for the MM model, and  $\beta_0 = 0.80$  for the MMt model.

**6.2. Comparison Results of Methods.** To demonstrate the computation performance of the precision-based simulation smoother, a Kalman filter-based MCMC method for SV model and SVM model is also developed. The Kalman filter-based simulation smoother algorithm and the auxiliary mixture sampler (see, e.g., Kim et al. [19]) method are compared with our precision-based simulation smoother method on the SV model. For the SVM model, the details of the Kalman filter-based smoother and simulation smoother algorithm can be seen in Appendix. Note that all procedures are implemented in Matlab R2014b on Intel 4 Core i7-4790 @ 3.60 GHz CPU.

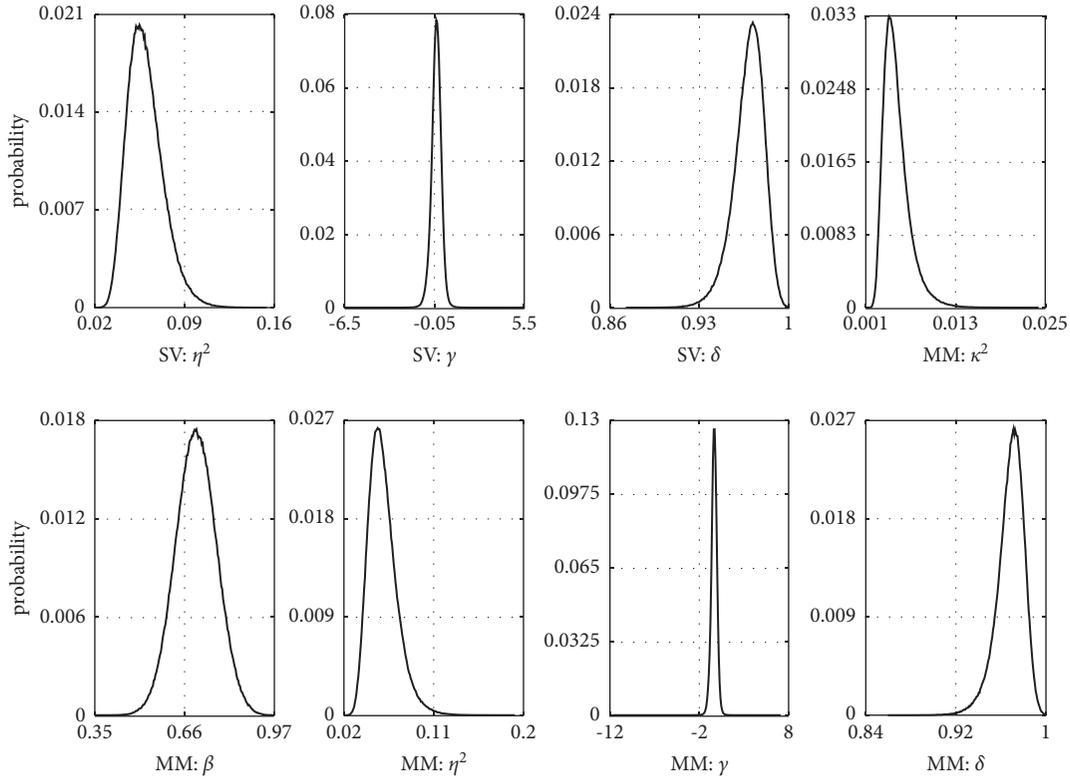


FIGURE 1: The posterior densities of parameters for SV and MM models.

TABLE 5: The estimated result for SV model using simulated datasets.

	True	Prior		Posterior		Credible intervals	Inefficiency
		Mean	Std	Mean	Std		
$\Gamma$	-0.25	0.00	3.162	-0.2930	0.3290	(-0.9744, 0.3179)	1.305
$\Delta$	0.97	0.97	0.100	0.9690	0.01191	(0.9417, 0.9885)	21.27
$\eta^2$	0.06	0.06	0.002	0.05979	0.01391	(0.03739, 0.09141)	60.33

TABLE 6: The estimated result for MM model using simulated datasets.

	True	Prior		Posterior		Credible intervals	Inefficiency
		Mean	Std	Mean	Std		
$\Gamma$	-0.25	0.00	3.162	-0.2935	0.3218	(-0.9599, 0.3082)	1.309
$\Delta$	0.97	0.97	0.100	0.9686	0.01209	(0.9404, 0.9881)	21.49
$\eta^2$	0.06	0.06	0.002	0.05954	0.01438	(0.03718, 0.09316)	61.27
B	0.70	0.07	0.100	0.7009	0.07026	(0.5629, 0.8383)	33.13
$\kappa^2$	0.005	0.005	$1.667 e-3$	0.005138	0.001748	(0.002767, 0.009502)	148.0

In Figure 3, we report the inefficiency factors corresponding to the posterior draws of  $\sigma = (\sigma_1, \sigma_2, \dots, \sigma_T)^T$  in the SV model and the SVM model using the precision-based simulation smoother method and Kalman filter-based simulation smoother method, respectively. Note that each vector is of length  $T$ , so we have a total of  $2T$  inefficiency factors. To present the information visually, boxplots are reported in Figure 3, where the middle line of the box denotes the median and the box denotes the median, while the lower and upper lines represent the 25 and the 75 percentiles, respectively. The whiskers extend to the

maximum and minimum. For example, the boxplot associated with  $\sigma$  indicates that about 50% of the log-volatilities of the precision-based method have inefficiency factors between 1 and 2, which is significantly low compared to the Kalman filter-based method.

Figures 4 and 5 report sample paths of the parameters in the SV model and the SVM model of the precision-based simulation smoother method and the Kalman filter-based simulation smoother method, respectively, and the results are similar and support that two methods are practically stable. The estimation results of the Kalman filter-based

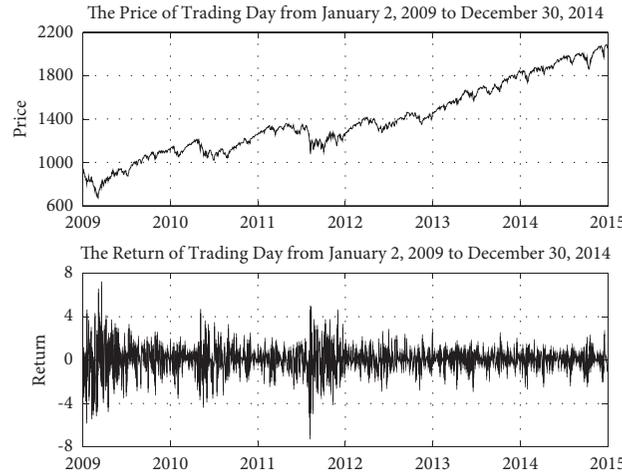


FIGURE 2: The S&P 500 index from January 2009 to December 2014.

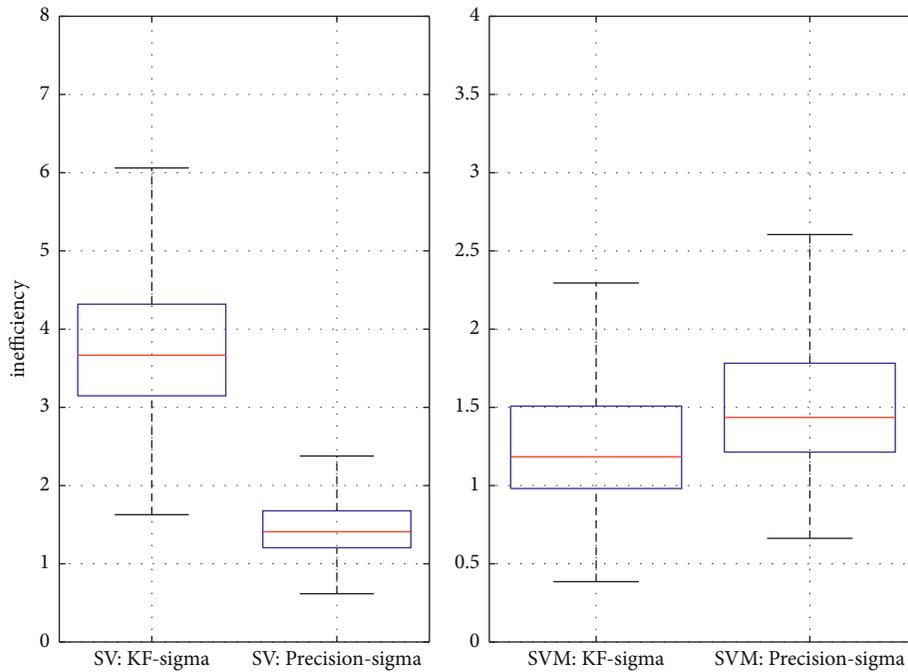


FIGURE 3: The inefficiency factor boxplots of the posterior draws  $\sigma$  in the SV model and the SVM model.

MCMC method for the SV model and the SVM model are reported in Table 3, which are also similar to the estimation results of the precision-based method (see Table 4). Last but not most important, the MCMC estimation times are 61.09 s, 1583 s, 62.76 s, and 4517 s, using the precision-based method for the SV model, the Kalman filter-based method for the SV model, the precision-based method for the SVM model, and the Kalman filter-based method for the SVM model, respectively, which are all of length 10000 after a burn-in period of 10000. These results are inspiring, since the computational cost of the proposed precision-based method is only 3.86% or 1.39% of that of the Kalman filter-based method for the SV model or the SVM model. This is consistent with the analyses in McCausland et al. [25].

6.3. *The Estimation Results.* The evolution of  $\sigma_t$  and  $\phi_t$  for MMt models (19)–(22) estimated by the proposed method is depicted in Figure 6. It is clear that the estimates of surplus demand  $\phi_t$  in the MMt model are much smoother than the return  $r_t$  itself in Figure 2, so we may take the evolution of  $\phi_t$  as the moving trend of the return  $r_t$ . On the other hand, the MM model can capture the effect of the dynamics offset between supply and demand, and it can capture the “surplus demand” and obtain the surplus yields paid over the risk return. In this way, using the market microstructure model, one may use a more stable and predictable surplus demand process rather than the strongly random return process itself to assist the trading decision making in the financial market. It is therefore suggested that one may allocate assets by

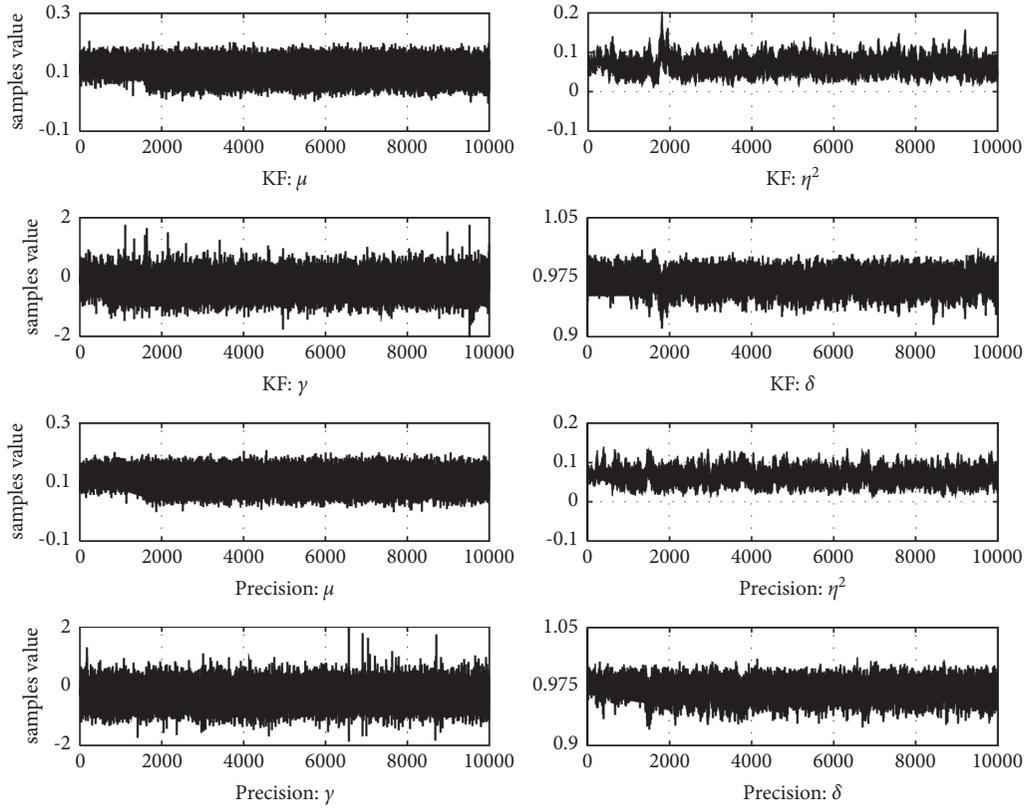


FIGURE 4: The sample paths of the parameters in the SV model.

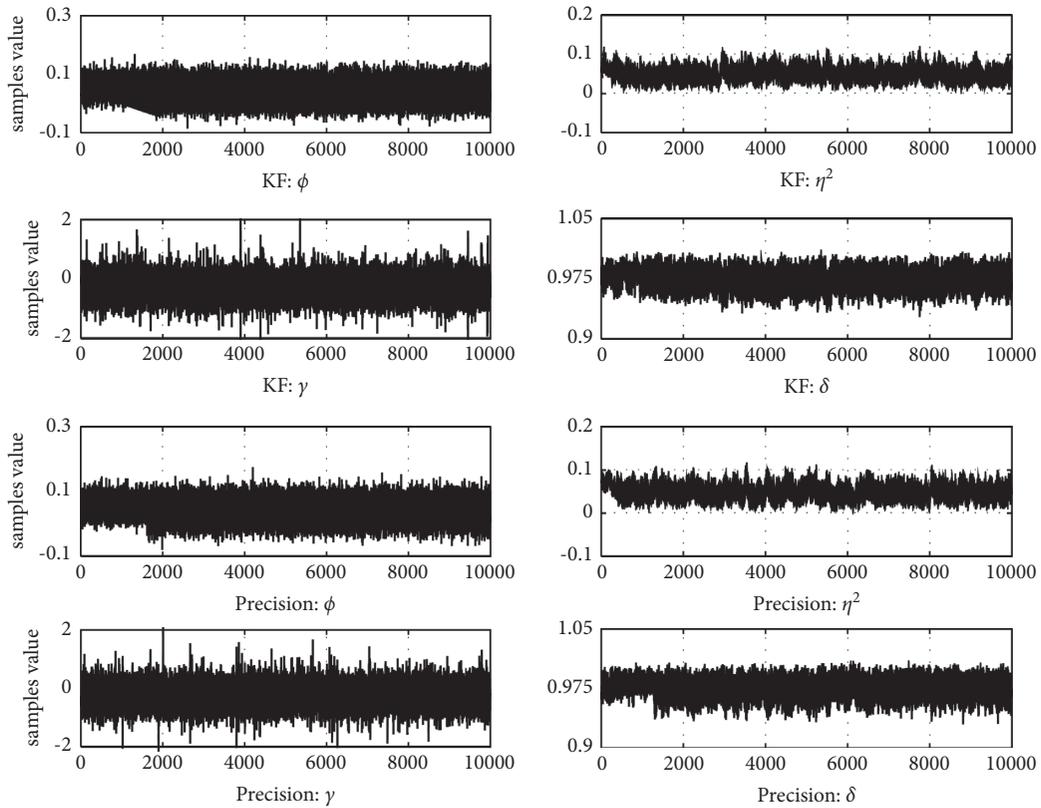


FIGURE 5: The sample paths of the parameters in the SVM model.

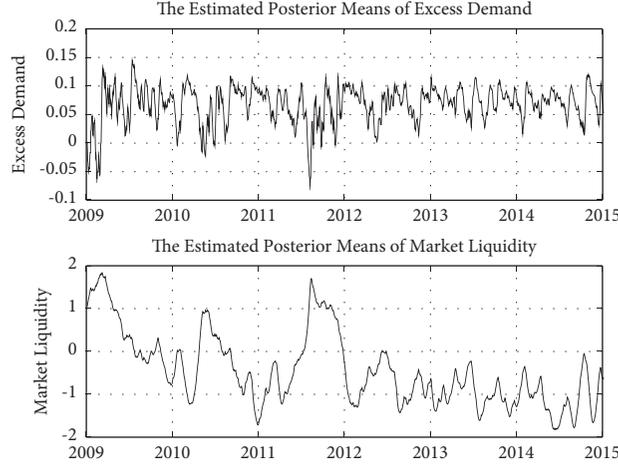


FIGURE 6: The evolution of the estimated posterior means  $\sigma_t$  and  $\phi_t$ .

monitoring and making use of the surplus demand process. This methodology was implemented in, e.g., Peng et al. [5], Peng et al. [6], Peng et al. [7], and Qin et al. [9].

Tables 4 and 7 show the estimated parameters posterior means, standard deviations, inefficiency factor, and the 95% credible intervals estimates for all six models based on 10000 posterior draws after a burn-in period of 10000. We also present the results of two diagnostic tests. Specifically, we report the Ljung–Box and McLeod–Li statistics of order 20 computed on the standardized residuals and squared standardized residuals, respectively. All the Ljung–Box tests fail to reject the null hypothesis of no serial correlation in the standardized residuals at the 5% level. This diagnostic result also supports the SVM model and the MM model, since the Ljung–Box statistics value of them is significantly lower than that of the SV model. In addition, all the McLeod–Li tests also fail to reject the null hypothesis of no serial correlation in the squared standardized residuals at the 5% level, and it indicates that the models adequately capture the time-varying volatility of the data. All in all, the Ljung–Box and McLeod–Li statistics results indicate that the estimations are successful.

**6.4. The Prediction Results.** In this part, a recursive out-of-sample forecasting exercise is performed to calculate the forecasting performance of the models listed in Table 1 for the S&P 500 index returns at various horizons. Every one of the six models is used to make both point and density  $k$ -step-ahead iterated forecasts with  $k = 1, 2, 5$ . Especially given the data up to time  $t$ , symbolized as  $r_{1:t} = (r_1, r_2, \dots, r_t)^T$ , the MCMC sampler presented in Sections 3–4 is implemented to get the posterior draws given  $r_{1:t}$ . Then, we calculate the predictive mean  $\mathbb{E}(r_{t+k}|r_{1:t})$  as the point forecast and the predictive density  $p(r_{t+k}|r_{1:t})$  as the density forecast. Next, we move one period in advance with building samples and repeat the whole exercise with data  $r_{1:t+1}$ , and so on. The entire sample beginning with the start of the asset return series is utilized to generate samples, and it is assumed that all stochastic volatility models are the stable models. These forecasts are then evaluated for  $t = t_0, \dots, T - k$ , where  $t_0$  is

January 2, 2014. An autoregressive model as standard benchmark is also considered for comparison. It is the following special case of the vector autoregressive model (VAR) which is composed of a system of more than one stochastic difference equation.

$$r_t = \mu + \beta r_{t-1} + \sigma \varepsilon_t^r, \quad (76)$$

where  $\mu$ ,  $\beta$ , and  $\sigma$  are the constants to be estimated.

In practice, neither the predictive mean nor the predictive density of  $r_{t+k}$  can be calculated analytically. Instead, they are obtained using predictive simulation. More precisely, at every MCMC iteration, given the model parameters and states up to time  $t$ , one can simulate future states from time  $t + 1$  to  $t + k$  using the relevant transition equations. The future errors  $e^{\sigma_s/2} \varepsilon_s^r$  for the SV type model or  $e^{\sigma_s/2} \lambda_s^r \varepsilon_s^r$  for the SVt type model are also simulated where  $s = t + 1, \dots, t + k$ . Given these draws,  $r_{t+k}$  is a normal random variable as specified in (1), and one can easily produce the point and density forecasts for  $r_{t+k}$ . Hence, one has a pair of forecasts (point and density) at every MCMC iteration. These forecasts are then averaged over all the posterior draws to produce estimates for  $\mathbb{E}(r_{t+k}|r_{1:t})$  and  $p(r_{t+k}|r_{1:t})$ . The whole exercise is then repeated using the data up to time  $t + 1$  to produce  $\mathbb{E}(r_{t+k+1}|r_{1:t+1})$  and  $p(r_{t+k+1}|r_{1:t+1})$ , and so on.

Let  $r_{t+k}^o$  symbolize the observed value of  $r_{t+k}$  which is known at time  $t + k$ . The metric that is used to calculate the point forecasts is the root mean squared forecast mistake.

RMSFE is defined as

$$\text{RMSFE} = \sqrt{\frac{\sum_{t=t_0}^{T-k} (r_{t+k}^o - \mathbb{E}(r_{t+k}|r_{1:t}))^2}{(T - k - t_0 + 1)}}. \quad (77)$$

To evaluate the density forecast  $p(r_{t+k}|r_{1:t})$ , one natural method is the predictive likelihood  $p(r_{t+k} = r_{t+k}^o | r_{1:t})$ , i.e., the predictive density of  $r_{t+k}$  calculated at the observed value  $r_{t+k}^o$ . Obviously, if the actual result  $r_{t+k}^o$  is impossibly under the density forecast, the value of the predictive likelihood can be small, and vice versa (see, e.g., Geweke and Amisano [60]

TABLE 7: Parameter posterior means, standard deviations (in parentheses), inefficiency factors (in braces), and 95% intervals (in square brackets) for the stochastic volatility models using the precision-based simulation smoother method.

	SVt	SVMt	MMt
$\mu$	0.1054 (0.01852) {1.913} [0.06918, 0.1410]		
$\phi$		0.06721 (0.02995) {8.675} [0.009542, 0.1287]	
$\gamma$	-0.5033 (0.2784) {6.866} [-1.038, 0.05783]	-0.4352 (0.3285) {3.655} [-1.045, 0.2024]	-0.4248 (0.3211) {4.445} [-1.020, 0.2520]
$\delta$	0.9764 (0.007722) {30.70} [0.9602, 0.9903]	0.9816 (0.006569) {20.07} [0.9676, 0.9935]	0.9817 (0.006574) {26.02} [0.9677, 0.9935]
$\eta^2$	0.04615 (0.01081) {81.64} [0.02876, 0.06980]	0.03495 (0.007486) {69.03} [0.02252, 0.05221]	0.03429 (0.007483) {76.51} [0.02308, 0.05213]
$\alpha$			0.06715 (0.03168) {27.18} [0.007429, 0.1329]
$\beta$			0.7885 (0.06333) {33.93} [0.6542, 0.8985]
$\kappa^2$			0.005679 (0.001688) {110.5} [0.003102, 0.009628]
$\nu$	7.998 (2.378) {143.6} [4.900, 14.19]	8.889 (2.634) {124.1} [5.465, 15.05]	9.551 (2.988) {123.0} [5.742, 17.22]
$Q_{LB}$	16.58 (2.877)	12.71(2.577)	14.99 (3.103)
$Q_{ML}$	18.56 (5.803)	19.23 (5.977)	18.50 (5.775)

Note.  $Q_{LB}$  and  $Q_{ML}$  are, respectively, the Ljung–Box and McLeod–Li statistics of order 20 computed on the standard residuals and squared standard residuals, and the numerical standard mistakes are in parentheses. The 0.05 and 0.01 critical values are 31.41 and 37.57, respectively.

TABLE 8: The predictive mean and the predictive density of stochastic volatility models.

	$k = 1$		$k = 2$		$k = 5$	
	RMSFE	LPS	RMSFE	LPS	RMSFE	LPS
VAR	0.71801	-1.27268	0.71803	-1.27286	0.71812	-1.27338
SV	0.71860	-1.15641	0.71852	-1.18626	0.71846	-1.27845
SVt	0.71860	-0.97835	0.71851	-0.99280	0.71853	-1.03235
SVM	0.71435	-1.13777	0.71422	-1.15935	0.71390	-1.23931
SVMt	0.71377	-0.99423	0.71360	-1.00031	0.71311	-1.03287
MM	0.71428	-1.13987	0.71355	-1.15757	0.71327	-1.24026
MMt	0.71363	-1.00282	0.71355	-1.01430	0.71372	-1.05082

for a more detailed discussion of the predictive likelihood and its combination with the marginal likelihood). The density forecasts are evaluated by using the average log predictive likelihood scores (LPSs) as follows:

$$LPS = \frac{\left(\sum_{t=t_0}^{T-k} \log(p(r_{t+k} = r_{t+k}^o | r_{1:t}))\right)}{(T - k - t_0 + 1)}. \quad (78)$$

For this metric, a larger value shows prediction performance.

Table 8 gives the point forecast and the density prediction results for seven models. One can find overwhelming support for all six stochastic volatility models compared to the VAR model from the one-step and two-step LPS, since the time-varying volatility is more flexible than const volatility. One can also find overwhelming support for the MM model and SVM model compared to standard SV model with their RMSFE values and LPS, since the volatility feedback effect is essential feature for financial time series. In addition, the stochastic volatility models plus Student-t heavy tails model may significantly improve the density forecast and the point forecast accuracy as seen in Table 8.

## 7. Conclusions and Future Research

In this paper, we studied the more general MM and MMt models compared to the MM type models researched in previous works and proposed a new approach for efficiently estimating the general MM type models, which are non-linear non-Gaussian models. Because of the general applicability of the proposed approach, it proved its usefulness in a wide range of applications for stochastic volatility models. The precision-based efficient simulation smoother algorithm and the efficient acceptance-rejection Metropolis–Hastings procedure were extended to develop a fast and efficient MCMC method for six SV models. The presented methodology was demonstrated with a simulation study and the inefficiency factors and posterior densities of parameters and states. The model estimation results showed substantial highlight for the empirical relevance of the proposed extension. In order to demonstrate computation performance of the precision-based simulation smoother, a Kalman filter-based MCMC method for SVM model was also developed. Furthermore, in a recursive forecasting exercise, the market microstructure model plus Student-t heavy tails performed

better than various standard benchmarks in both point forecast and density forecast.

For future research, it would be interesting to investigate if the proposed market microstructure model variant (plus jump, leverage, and intertemporal volatility feedback) also fits other macroeconomic or financial time series better than the standard market microstructure model. Since it is difficult to obtain an accurate estimate of marginal likelihood and deviance information criterion (DIC) through existing algorithms for the market microstructure model, due to the marginal likelihood evaluation involving “integrating out” both types of states, i.e.,  $\phi_t$  and  $\sigma_t$ , we will consider an adaptive importance sampling approach to estimate the marginal likelihood and DIC in the future, where the draws are typically costly to obtain and highly correlated in high-dimensional settings. Furthermore, in the recursive forecasting exercise, since the market microstructure model performed better than various standard benchmarks in both point forecast and density forecast, it would be interesting to do optimization portfolio and asset allocation using the market microstructure model and study the relationship between the excess demand  $\phi_t$  and the trend of return  $r_t$ .

## Appendix

In this section, we present an augmented Kalman filter-based smoother and simulation smoother and consider a common state-space model:

$$y_t = X_t\beta + Z_t\alpha_t + G_tu_t, \beta = b + B\delta, \quad (\text{A.1})$$

$$\alpha_{t+1} = W_t\beta + T_t\alpha_t + H_tu_t, \alpha_0 = 0, \quad (\text{A.2})$$

where  $\delta \sim NID(c, C)$  and  $u_t \sim NID(0, I)$  are uncorrelated,  $b$  is fixed,  $B$  has full column rank, and  $C$  is non-singular unless  $C = 0$ .  $X_t$ ,  $Z_t$ ,  $G_t$ ,  $W_t$ ,  $T_t$ , and  $H_t$  are the system matrices. The state-space models (A1) and (A2) are considered as a natural set for wide time series models to calculate parameters and to make prediction of future observations: the regression influence in time series models is put in  $X_t\beta$ ; we can find autoregressive moving average dynamics in  $T_t\alpha_t$  and  $H_tu_t$ ; a model of additive outlier and structural change interventions can be constructed through using  $X_t\beta$  and  $W_t\beta$ ; unobservable elements are put in the state vector  $\alpha_t$ ; and the covariance construction is up to the matrices  $G_t$  and  $H_t$ .

There is a stationary initial condition,  $\alpha_{1|0} = W_0\beta$  and  $P_{1|0} = H_0H_0^T$ , since  $\alpha_0 = 0$ . For notational convenience, let  $y = (y_1, y_2, \dots, y_T)^T$ .

The Kalman filter is a forward recursion and run for  $t = 1, 2, \dots, T$ .

$D_t = Z_tP_{t|t-1}Z_t^T + G_tG_t^T$ ,  $K_t = (T_tP_{t|t-1}Z_t^T + H_tG_t^T)D_t^{-1}$ ,  $J_t = H_t - K_tG_t$ ,  $L_t = T_t - K_tZ_t$ ,  $P_{t+1|t} = T_tP_{t|t-1}L_t^T + H_tJ_t^T$ ,  $e_t = y_t - X_t\beta - Z_t\alpha_{t|t-1}$ ,  $\alpha_{t+1|t} = W_t\beta + T_t\alpha_{t|t-1} + K_t e_t$ .

$\alpha_{t|t-1}$  and  $P_{t+1|t}$  are the predictor and the mean squared error of  $\alpha_t$ , respectively,  $e_t$  is the error of predictor  $y_t$ ,  $D_t$  is covariance of  $e_t$ , and  $K_t$  is Kalman gain.

For state simulation smoother, set  $r_T = 0$  and  $N_T = 0$ , and the Kalman smoother is a backward recursion and run for  $t = T, T-1, \dots, 1$ .

$n_t = D_t^{-1}e_t - K_t^T r_t$ ,  $F_t = D_t^{-1} + K_t^T N_t K_t$ ,  $M_t = H_t G_t^T K_t^T N_t H_t H_t^T$ ,  $V_t = H_t G_t^T (F_t Z_t - K_t^T N_t T_t) + H_t H_t^T N_t L_t$ ,  $C_t = H_t H_t^T - H_t G_t^T F_t G_t H_t^T - H_t H_t^T N_t H_t H_t^T + M_t + M_t^T$ ,  $\kappa_t \sim NID(0, C_t)$ ,  $r_{t-1} = Z_t^T D_t^{-1} e_t + L_t^T r_t - V_t^T C_t^{-1} \kappa_t$ ,  $N_{t-1} = Z_t^T D_t^{-1} Z_t + L_t^T N_t L_t + V_t^T C_t^{-1} V_t$ .

The disturbance  $H_t u_t$  smoothing is  $H_t G_t^T n_t + H_t H_t^T r_t + \kappa_t$ .

$$\mathbb{E}(H_t u_t | y) = H_t G_t^T n_t + H_t H_t^T r_t,$$

$$\begin{aligned} \mathbb{V}(H_t u_t | y) &= H_t H_t^T - H_t G_t^T F_t G_t H_t^T - H_t H_t^T N_t H_t H_t^T \\ &\quad + M_t + M_t^T = C_t. \end{aligned} \quad (\text{A.3})$$

The simulation smoother for states is a forward recursion and run for  $t = 1, 2, \dots, T$ .

$$\tilde{\alpha}_{t+1} = W_t \beta + T_t \tilde{\alpha}_t + H_t u_t, \quad (\text{A.4})$$

which is started off with  $\tilde{\alpha}_1 = W_0 \beta + H_0 H_0^T r_0$ .

For state smoother, set  $r_T = 0$  and  $N_T = 0$ , and the Kalman smoother is a backward recursion and run for  $t = T, T-1, \dots, 1$ .

$$n_t = D_t^{-1} e_t - K_t^T r_t,$$

$$r_{t-1} = Z_t^T D_t^{-1} e_t + L_t^T r_t, \quad (\text{A.5})$$

$$N_{t-1} = Z_t^T D_t^{-1} Z_t + L_t^T N_t L_t.$$

The quick state smoother is a forward recursion and run for  $t = 1, 2, \dots, T$ .

$$\tilde{\alpha}_{t+1} = W_t \beta + T_t \tilde{\alpha}_t + H_t \hat{u}_t, \quad (\text{A.6})$$

$$\hat{u}_t = G_t^T n_t + H_t^T r_t,$$

which is started off with  $\tilde{\alpha}_1 = W_0 \beta + H_0 H_0^T r_0$ .

The detailed procedure may be seen, e.g., in De Jong [15], Koopman [16], De Jong and Shephard [17], Koopman et al. [61], and Durbin and Koopman [18].

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This research was supported by the National Natural Science Foundation of China (71271215).

## References

- [1] S. J. Taylor, "Modeling stochastic volatility: a review and comparative study," *Mathematical Finance*, vol. 4, no. 2, pp. 183–204, 1994.
- [2] T. Bollerslev, "Generalized autoregressive conditional heteroskedasticity," *Journal of Econometrics*, vol. 31, no. 3, pp. 308–327, 1986.
- [3] J.-P. Bouchaud and R. Cont, "A Langevin approach to stock market fluctuations and crashes," *The European Physical Journal B*, vol. 6, no. 4, pp. 543–550, 1998.
- [4] M. Iino and T. Ozaki, "A nonlinear model for financial dynamics," in *Proceedings of the international symposium on frontiers of time series modeling*, Tokyo, Japan, February 2000.
- [5] H. Peng, T. Ozaki, and V. Haggan-Ozaki, "Modeling and asset allocation for financial markets based on a discrete time microstructure model," *The European Physical Journal B (EPJ B)—Condensed Matter*, vol. 31, no. 2, pp. 285–293, 2003.
- [6] H. Peng \*, Y. Tamura, W. Gui, and T. Ozaki, "Modelling and asset allocation for financial markets based on a stochastic volatility microstructure model," *International Journal of Systems Science*, vol. 36, no. 6, pp. 315–327, 2005.
- [7] H. Peng, G. Kitagawa, Y. Tamura, Y. Xi, Y. Qin, and X. Chen, "A modeling approach to financial time series based on market microstructure model with jumps," *Applied Soft Computing*, vol. 29, no. 1, pp. 40–51, 2015.
- [8] Y. Xi, H. Peng, Y. Qin, W. Xie, and X. Chen, "Bayesian analysis of heavy-tailed market microstructure model and its application in stock markets," *Mathematics and Computers in Simulation*, vol. 117, no. 3, pp. 141–153, 2015.
- [9] Y. Qin, H. Peng, Y. Xi, W. Xie, Y. Sun, and X. Chen, "An adaptive modeling and asset allocation approach to financial markets based on discrete microstructure model," *Applied Soft Computing*, vol. 43, no. 6, pp. 390–405, 2016.
- [10] Y. Xi, H. Peng, and Y. Qin, "Modeling financial time series based on a market microstructure model with leverage effect," *Discrete Dynamics in Nature and Society*, vol. 9, no. 4, pp. 78–82, 2016.
- [11] A. E. Gelfand and A. F. M. Smith, "Sampling-based approaches to calculating marginal densities," *Journal of the American Statistical Association*, vol. 85, no. 410, pp. 398–409, 1990.
- [12] J. M. Bardsley and T. Cui, "Optimization-based Markov chain Monte Carlo methods for nonlinear hierarchical statistical inverse problems," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 9, no. 1, pp. 29–64, 2021.
- [13] Z. Zhou, M. Tartakovsky, and D. M. Tartakovsky, "Markov chain Monte Carlo with neural network surrogates: application to contaminant source identification," *Stochastic Environmental Research and Risk Assessment*, vol. 35, no. 3, pp. 639–651, 2021.
- [14] C. Nemeth and P. Fearnhead, "Stochastic gradient markov chain monte carlo," *Journal of the American Statistical Association*, vol. 116, no. 533, pp. 433–450, 2021.
- [15] P. De Jong, "The diffuse Kalman filter," *The Annals of Statistics*, vol. 19, no. 2, pp. 1073–1083, 1991.
- [16] S. J. Koopman, "Disturbance smoother for state space models," *Biometrika*, vol. 80, no. 1, pp. 117–126, 1993.
- [17] P. De Jong and N. Shephard, "The simulation smoother for time series models," *Biometrika*, vol. 82, no. 2, pp. 339–350, 1995.
- [18] J. Durbin and S. J. Koopman, "A simple and efficient simulation smoother for state space time series analysis," *Biometrika*, vol. 89, no. 3, pp. 603–616, 2002.
- [19] S. Kim, N. Shephard, and S. Chib, "Stochastic volatility: likelihood inference and comparison with ARCH models," *Review of Economic Studies*, vol. 65, no. 3, pp. 361–393, 1998.
- [20] Y. Omori, S. Chib, N. Shephard, and J. Nakajima, "Stochastic volatility with leverage: fast and efficient likelihood inference," *Journal of Econometrics*, vol. 140, no. 2, pp. 425–449, 2007.
- [21] J. Nakajima and Y. Omori, "Leverage, heavy-tails and correlated jumps in stochastic volatility models," *Computational Statistics & Data Analysis*, vol. 53, no. 6, pp. 2335–2353, 2009.
- [22] E. I. Delatola and J. E. Griffin, "Bayesian nonparametric modelling of the return distribution with stochastic volatility," *Bayesian analysis*, vol. 6, no. 4, pp. 901–926, 2011.
- [23] M. J. Jensen and J. M. Maheu, "Estimating a semiparametric asymmetric stochastic volatility model with a dirichlet process mixture," *Journal of Econometrics*, vol. 178, pp. 523–538, 2014.
- [24] J. C. Chan and I. Jeliakzov, "Efficient simulation and integrated likelihood estimation in state space models," *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 1, no. 1–2, pp. 101–120, 2009.
- [25] W. J. McCausland, S. Miller, and D. Pelletier, "Simulation smoothing for state-space models: a computational efficiency analysis," *Computational Statistics & Data Analysis*, vol. 55, no. 1, pp. 199–212, 2011.
- [26] H. Rue, "Fast sampling of gaussian markov random fields," *Journal of the Royal Statistical Society: Series B*, vol. 63, no. 2, pp. 325–338, 2001.
- [27] H. Rue, S. Martino, and N. Chopin, "Approximate bayesian inference for latent Gaussian models by using integrated nested laplace approximations," *Journal of the Royal Statistical Society: Series B*, vol. 71, no. 2, pp. 319–392, 2009.
- [28] W. J. McCausland, "The hessian method: highly efficient simulation smoothing, in a nutshell," *Journal of Econometrics*, vol. 168, no. 2, pp. 189–206, 2012.
- [29] B. Djegnéné and W. J. Mccausland, "The hessian method for models with leverage-like effects," *Journal of Financial Econometrics*, vol. 13, no. 3, pp. 722–755, 2015.
- [30] K. R. French, G. W. Schwert, and R. F. Stambaugh, "Expected stock returns and volatility," *Journal of Financial Economics*, vol. 19, no. 1, pp. 3–29, 1987.
- [31] J. Y. Campbell and L. Hentschel, "No news is good news," *Journal of Financial Economics*, vol. 31, no. 3, pp. 281–318, 1992.
- [32] S. J. Koopman and E. Hol Uspensky, "The stochastic volatility in mean model: empirical evidence from international stock markets," *Journal of Applied Econometrics*, vol. 17, no. 6, pp. 667–689, 2002.
- [33] J. Geweke, "[Bayesian analysis of stochastic volatility models]: comment," *Journal of Business & Economic Statistics*, vol. 12, no. 4, pp. 397–399, 1994.
- [34] A. R. Gallant, D. Hsieh, and G. Tauchen, "Estimation of stochastic volatility models with diagnostics," *Journal of Econometrics*, vol. 81, no. 1, pp. 159–192, 1997.
- [35] T. Watanabe and Y. Omori, "A multi-move sampler for estimating non-Gaussian time series models: comments on Shephard & Pitt (1997)," *Biometrika*, vol. 91, no. 1, pp. 246–248, 2004.
- [36] Y. Omori and T. Watanabe, "Block sampler and posterior mode estimation for asymmetric stochastic volatility models," *Computational Statistics & Data Analysis*, vol. 52, no. 6, pp. 2892–2910, 2008.
- [37] C. A. Abanto-Valle, H. S. Migon, and V. H. Lachos, "Stochastic volatility in mean models with scale mixtures of normal distributions and correlated errors: a Bayesian

- approach,” *Journal of Statistical Planning and Inference*, vol. 141, no. 5, pp. 1875–1887, 2011.
- [38] L. Tierney, “Markov chains for exploring posterior distributions,” *The Annals of Statistics*, vol. 22, pp. 1701–1728, 1994.
- [39] M. Valipour, “Optimization of neural networks for precipitation analysis in a humid region to detect drought and wet year alarms,” *Meteorological Applications*, vol. 23, no. 1, pp. 91–100, 2015.
- [40] M. Valipour, M. A. G. Sefidkouhi, and S. Eslamian, “Surface irrigation simulation models: a review,” *International Journal of Horticultural Science and Technology*, vol. 5, no. 1, pp. 51–70, 2015.
- [41] M. Valipour, “Sprinkle and trickle irrigation system design using tapered pipes for pressure loss adjusting,” *Journal of Agricultural Science*, vol. 4, no. 12, p. 125, 2012.
- [42] M. Valipour, M. E. Banihabib, and S. M. R. Behbahani, “Comparison of the ARMA, ARIMA, and the autoregressive artificial neural network models in forecasting the monthly inflow of Dez dam reservoir,” *Journal of Hydrology*, vol. 476, pp. 433–441, 2013.
- [43] E. Jacquier, N. G. Polson, and P. E. Rossi, “Bayesian analysis of stochastic volatility models with fat-tails and correlated errors,” *Journal of Econometrics*, vol. 122, no. 1, pp. 185–212, 2004.
- [44] J. Geweke, “Bayesian treatment of the independent student-t linear model,” *Journal of Applied Econometrics*, vol. 8, no. S1, pp. S19–S40, 1993.
- [45] J. Durbin and S. J. Koopman, “Monte Carlo maximum likelihood estimation for non-Gaussian state space models,” *Biometrika*, vol. 84, no. 3, pp. 669–684, 1997.
- [46] S. Chib and I. Jeliazkov, “Accept-reject Metropolis-Hastings sampling and marginal likelihood estimation,” *Statistica Neerlandica*, vol. 59, no. 1, pp. 30–44, 2005.
- [47] S. Dimitrakopoulos and M. Kolossiatis, “Bayesian analysis of moving average stochastic volatility models: modeling in-mean effects and leverage for financial time series,” *Econometric Reviews*, vol. 39, no. 4, pp. 319–343, 2019.
- [48] F. Blasques, S. J. Koopman, and A. Lucas, “Nonlinear autoregressive models with optimality properties,” *Econometric Reviews*, vol. 39, no. 6, pp. 559–578, 2020.
- [49] D. Creal, “A survey of sequential Monte Carlo methods for economics and finance,” *Econometric Reviews*, vol. 31, no. 3, pp. 245–296, 2012.
- [50] C. Andrieu, A. Doucet, and R. Holenstein, “Particle Markov chain Monte Carlo methods,” *Journal of the Royal Statistical Society: Series B*, vol. 72, no. 3, pp. 269–342, 2010.
- [51] N. Shephard and M. K. Pitt, “Likelihood analysis of non-Gaussian measurement time series,” *Biometrika*, vol. 84, no. 3, pp. 653–667, 1997.
- [52] J. Chan, “The stochastic volatility in mean model with time-varying parameters: an application to inflation modeling,” *Journal of Business & Economic Statistics*, vol. 35, no. 1, 2017.
- [53] C. Joshua, “The stochastic volatility in mean model with time-varying parameters: an application to inflation, modeling,” *Journal of Business & Economic Statistics*, vol. 35, pp. 17–28, 2017.
- [54] S. Zhang, X. Mao, K. Kwang Raymond, T. Choo, and G. Wang, “A trajectory privacy-preserving scheme based on a dual-K mechanism for continuous location-based services,” *Information Sciences*, vol. 527, pp. 406–419, 2020.
- [55] W. Liang, D. Zhang, X. Lei, M. Tang, K.-C. Li, and A. Y. Zomaya, “Circuit copyright blockchain: blockchain-based homomorphic encryption for IP circuit protection,” *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 3, pp. 1410–1420, 2021.
- [56] S. Chib, F. Nardari, and N. Shephard, “Markov chain Monte Carlo methods for stochastic volatility models,” *Journal of Econometrics*, vol. 108, no. 2, pp. 281–316, 2002.
- [57] M. Khoshravesh, M. A. G. Sefidkouhi, and M. Valipour, “Estimation of reference evapotranspiration using multivariate fractional polynomial, Bayesian regression, and robust regression models in three arid environments,” *Applied Water Science*, vol. 7, no. 4, pp. 1911–1922, 2015.
- [58] M. K. Pitt, S. Malik, and A. Doucet, “Simulated likelihood inference for stochastic volatility models using continuous particle filtering,” *Annals of the Institute of Statistical Mathematics*, vol. 66, no. 3, pp. 527–552, 2014.
- [59] J. Geweke, *Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments*, Federal Reserve Bank of Minneapolis, Research Department, Minneapolis, MN, USA, 1991.
- [60] J. Geweke and G. Amisano, “Hierarchical Markov normal mixture models with applications to financial asset returns,” *Journal of Applied Econometrics*, vol. 26, no. 1, pp. 1–29, 2011.
- [61] S. J. Koopman, N. Shephard, and J. A. Doornik, “Statistical algorithms for models in state space using SsfPack 2.2,” *The Econometrics Journal*, vol. 2, no. 1, pp. 107–160, 1999.