TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

Hanxue Yang

# Markov Chain Monte Carlo Estimation of Stochastic Volatility Models with Finite and Infinite Activity Lévy Jumps

Evidence for Efficient Models and Algorithms

Julkaisu 1331 • Publication 1331

Tampere 2015

Tampereen teknillinen yliopisto. Julkaisu 1331
Tampere University of Technology. Publication 1331

Hanxue Yang

# Markov Chain Monte Carlo Estimation of Stochastic Volatility Models with Finite and Infinite Activity Lévy Jumps

Evidence for Efficient Models and Algorithms

Thesis for the degree of Doctor of Philosophy degree to be presented with due permission for public examination and criticism in Festia Building, Auditorium Pieni Sali 1, at Tampere University of Technology, on the 13th of November 2015, at 12 noon.

Tampereen teknillinen yliopisto - Tampere University of Technology
Tampere 2015

ISBN 978-952-15-3597-0 (printed)
ISBN 978-952-15-3617-5 (PDF)
ISSN 1459-2045

# Abstract

A financial model plays a key role in the valuation and risk management of financial derivatives, and it serves as an important tool for investors to measure the risk exposure of their portfolios and make predictions and decisions. However, the popular affine stochastic volatility models without jumps, such as the Heston model, have been questioned in the finance literature in terms of their appropriateness for modelling stock prices and pricing derivatives. Many alternative model specifications have been proposed in recent decades, including the specification of non-affine variance dynamics and the inclusion of Lévy jumps. However, the complexity introduced by further model specifications leads to poor probabilistic properties, and hence most popular estimation methods are not applicable. The Bayesian estimation method is among the few that work. In this thesis, I discuss the role of new model specifications and investigate the performance of Bayesian estimation methods.

First, I use an extensive empirical data set to study how the use of infinite-activity Lévy jumps in stock returns and variance improves model performance. The stock returns and variance are driven by diffusions and different Lévy jumps, including the finite-activity compound Poisson jump and infinite-activity Variance Gamma and Normal Inverse Gaussian (NIG) jumps. Moreover, the non-affine linear variance process is compared to the affine square-root stochastic process. With the conventional Markov Chain Monte Carlo (MCMC) algorithms, including the Gibbs sampler and Metropolis-Hastings (MH) methods, and the Damien-Wakefield-Walker method to cope with complicated posteriors, eighteen different model specifications are estimated using the joint information of the S&P 500 index and the VIX index for 1996 – 2009. There is clear evidence that in terms of the goodness of fit and option pricing performance, a relatively parsimonious model with infinite-activity NIG jumps in returns and non-affine variance dynamics is particularly competitive.

In the second part of the thesis, I examine the performance of advanced MCMC algorithms. The efficiency of the MH algorithm has been questioned because of its slow mixing speed, especially in the presence of high dimensions and a strong dependence between model parameters and state variables. Generally, a class of algorithms seeks to improve the MH by constructing more effective proposals, and another combines the MCMC with the Sequential Monte Carlo algorithms. To investigate, I first conduct simulation studies to compare the estimation performance of seven advanced Bayesian estimation methods against the MH. Specifically, I use the affine Heston model, the affine Bates model, and an affine model with NIG return jumps, and examine whether the different jump structures affect the estimation results. Second, I estimate the non-affine model with NIG return jumps using the joint information of the S&P 500 index and the VIX for 2002 – 2005 with selected algorithms that perform well in the simulation studies. The results of the simulation and empirical studies are mixed about the performance of the algorithms. The

Fast Universal Self-tuned Sampler algorithms are particularly competitive in generating virtually independent samples and achieving the fastest mixing with a fixed number of MCMC runs, and their performance is stable regardless of the model specifications. However, they are computationally expensive. The computational costs of the Particle Markov Chain Monte Carlo (PMCMC) methods are much cheaper and also efficient in mixing, and they perform best when estimating the models without jumps/with NIG jumps in the simulation studies, as well as in the fit to the VIX in the empirical studies. However, the PMCMC methods are more vulnerable to model specifications than the other algorithms; in particular, the rare large compound Poisson jumps in the Bates model significantly reduce the acceptance rate and worsen the estimation performance of the PMCMC methods.

# Preface

The research presented in this thesis was carried out at the Department of Industrial Management at Tampere University of Technology between 2013 and 2015, within the Marie Curie Initial Training Network HPCFinance. During the writing of this thesis, I was financially supported by the European Union Seventh Framework Programme under grant agreement No. 289032.

I would like to express my sincere gratitude to my supervisor, Professor Juho Kanniainen, for giving me such an invaluable opportunity to work on the HPCFinance project, and more importantly, for his insightful guidance and consistent support that helped me manage many difficult problems. In my research, I have always been encouraged by his enthusiasm and honest criticism and advice. I am also especially grateful for the inspiring cooperation with, and support from, Professor Luca Martino, who proposed and developed the powerful FUSS algorithms and guided me in applying the algorithms to financial problems. In addition, I would like to express my gratitude to the pre-examiners, Professor Michael Johannes and Professor Andreas Kaeck, for their helpful comments and suggestions.

Furthermore, I wish to thank my colleagues at the Department of Industrial Management and in the HPCFinance project. In particular, I thank Milla Siikanen, Sindhuja Ranganathan, Jaakko Valli, and Binghuan Lin in the finance group for the helpful discussions and activities outside of work. Special thanks go to my officemates, Jun Hu and Ye Yue, for providing a fun and stimulating work environment. I am also thankful to the project managers, Dr. Santiago Velasquez and Dr. Hanna-Riikka Sundberg, and the department secretary, Ms. Heli Kiviranta, for their assistance and administrative support.

I want to thank Jukka Annala and his wife Tuija for inviting me to their beautiful home and summer cottage. I will never forget the journey into the deep forest and the experiences of taking a sauna and jumping into the cold lake, picking mushrooms, and tasting karjalanpiirakka. A special mention goes to the lovely couple Huiling Wang and Tinghuai Wang. I am glad to have friends like you, who have made my life here unforgettable.

Last but not least, I owe my deepest thanks to my parents and my dearest grandmother for their endless encouragement and love. Thank you for sharing your lives with me. You remain my best friends, no matter how far away you are.

# Contents

**Abstract** **i**

**Preface** **iii**

**Acronyms** **vii**

**1 Introduction** **1**
1.1 Background and Motivation . . . . . 1
1.2 Questions and Research Methods . . . . . 6
1.3 Outline and Contributions of the Thesis . . . . . 8

**2 Jumps and Volatility Dynamics for the S&P 500 Index: Conventional MCMC** **13**
2.1 Description of Models and Notations . . . . . 13
2.2 Change of Measure . . . . . 16
2.3 VIX . . . . . 19
2.4 Estimation Method . . . . . 21
2.5 Data Description and Option Pricing Settings . . . . . 25
2.6 Estimation Results . . . . . 28
2.7 Discussion . . . . . 45

**3 Model Estimation with Advanced MCMC Algorithms: Simulation Studies** **47**
3.1 Motivation . . . . . 47
3.2 Review: Conventional MCMC Methods . . . . . 48
3.3 Particle Filters . . . . . 50
3.4 Advanced MCMC Methods . . . . . 51
3.5 Simulation Studies: Data Generation . . . . . 56
3.6 Implementation of Algorithms . . . . . 58
3.7 Results . . . . . 62
3.8 Discussion . . . . . 76

**4 Model Estimation with Advanced MCMC Algorithms: Empirical Studies** **79**
4.1 Motivation . . . . . 79
4.2 Data . . . . . 80
4.3 Implementation of Algorithms . . . . . 81
4.4 Results . . . . . 82
4.5 Discussion . . . . . 96

**5 Discussion and Conclusion** **97**

**Bibliography** **101**

**Appendices** **109**

**A Simulation Studies: Extracted Variance** **111**

**B Simulation Studies: Extracted Jumps** **117**

**C Empirical Studies: Extracted Variance of the SV-NIG-1 Model** **121**

**D Empirical Studies: Model-implied VIX of the SV-NIG-1 Model** **125**

# Acronyms

| | |
|---|---|
| ACF | Autocorrelation Function |
| AM | Adaptive Metropolis |
| APF | Auxiliary Particle Filter |
| AR | Acceptance Rate |
| ARMS | Adaptive Rejection Metropolis-Sampling |
| ARS | Adaptive Rejection Sampling |
| BS | Black-Scholes |
| CBOE | Chicago Board Options Exchange |
| CEV | Constant Elasticity of Variance |
| DIC | Deviance Information Criterion |
| DWW | Damien-Wakefield-Walker |
| FUSS | Fast Universal Self-tuned Sampler |
| LL | Log-Likelihood |
| MC | Monte Carlo |
| MCMC | Markov Chain Monte Carlo |
| MH | Metropolis-Hastings |
| MLE | Maximum Likelihood Estimation |
| MSE | Mean Squared Error |
| NIG | Normal Inverse Gaussian |
| PDF | Probability Density Function |
| PF | Particle filter |
| PG | Particle Gibbs |
| PGAS | Particle Gibbs with Ancestor Sampling |
| PMCMC | Particle Markov Chain Monte Carlo |
| PMMH | Particle Marginal Metropolis Hastings |
| RC | Rejection Chain |
| RMSE | Root Mean Squared Error |
| RS | Rejection Sampling |
| SMC | Sequential Monte Carlo |
| SQR | Square Root |
| VG | Variance Gamma |

# 1 Introduction

## 1.1 Background and Motivation

A financial model can be regarded as an approximation of the true dynamics of a financial asset. It is a mathematical model that uses variables to abstract parameters of interest and is designed to capture some of the most important empirical features of the financial market. It enables us to introduce randomness and make assumptions on the basis of market observations to link the variables representing different features of the market and different types of risk, and more importantly, to quantify and evaluate the risk and make a useful analysis. The financial model is a key element of the valuation and risk management of complicated financial derivatives and an important tool for investors to estimate future returns, measure the risk exposure of their portfolios, and make predictions and decisions.

### 1.1.1 Financial Models

The empirical evidence has shown that the Black-Scholes (BS) model proposed by Black and Scholes (1973) fails to provide a satisfactory description of the dynamics of real-life stock prices. That is, the stock prices are not log-normally distributed; rather, the distribution of stock returns is usually negatively skewed and has long and fat tails and high peaks. More importantly, jumps, especially downside jumps, may occur randomly in the path of the stock price. If investors base their investment and risk management decisions on a model that misses the fundamental features of stock prices, they will be exposed to a substantial risk of loss.

In response, many more flexible and realistic models have been proposed for stock prices, such as the stochastic volatility model and the model with Lévy jumps.

*Stochastic volatility models*

As its name suggests, the stochastic volatility model replaces the constant volatility in the BS model with another random variable described by some stochastic process. The Heston model (Heston, 1993) is one of the most popular stochastic volatility models. It describes the spot variance as a mean-reverting square-root (SQR) process that is driven by a Brownian motion correlated with the Brownian innovation in the process of the stock returns. By assuming a negative correlation between two Brownian innovations in returns and variance, the Heston model takes account of the volatility feedback effect, that is, changes in stock returns and variance are negatively correlated. Moreover, the introduction of stochastic volatility can alter the skewness and kurtosis of the distribution of stock prices. Yet the performance of the Heston model has been questioned in the literature. Jones (2003) argues that the stochastic volatility model with an SQR variance process is unable to generate realistic variance dynamics, and he concludes that the

Constant Elasticity of Variance (CEV) model is more consistent with index returns and option prices and is, hence, a better choice for describing the variance process. Similar observations have been made in many finance studies. For instance, Christoffersen et al. (2010a) suggest that the stochastic volatility model with linear diffusion for variance, compared to SQR diffusion, is much better in the aspects of modelling the realised volatility and index returns and pricing options.

Moreover, in the historical time series of stock returns and realised variance, many frequent small jumps and some large jumps typically occur randomly, and while stochastic volatility models may resolve some empirical biases of the BS model, they do not account for unexpected frequent jumps. The Bates model proposed by Bates (1996) augments the Heston model by adding a compound Poisson jump component to the dynamics of stock returns. However, the results in both Bates (2000) based on options and Pan (2002) based on the joint information of stock prices and options suggest that the Bates model fails to trace systematic variations in option prices, and that a model with jump components in the variance process is highly useful. Duffie et al. (2000) propose the double-Poisson-jump model as a generalisation of the Heston, Bates, and Merton jump diffusion models. This model is constructed by adding another compound Poisson jump component to the variance process, whose jump times and sizes are simultaneous and correlated, respectively, with those in stock returns. Eraker et al. (2003) examine the double-Poisson-jump model using return data, and their empirical results are clearly in favour of this model rather than the Heston and Bates models, because the double-Poisson-jump model successfully captures unexpected large jumps. Consequently, Eraker et al. (2003) argue that the inclusion of a double-Poisson-jump specification might be indispensable, especially under extreme market conditions. However, when models are estimated and compared using the joint information of index returns and option prices, the inclusion of jumps in variance seems less important, as Eraker (2004) and Kaeck and Alexander (2012) point out.

*Lévy processes*

Most of the finance literature relies on the use of Brownian motion to capture normal asset price variations and on compound Poisson jumps to capture large price movements in returns and variance. In fact, both Brownian motion and compound Poisson jump belong to the class of Lévy processes. By definition, a Lévy process is a stochastic process, the distribution of whose increments is stationary and independent of its past track.

According to the behaviour of the Lévy measure, a Lévy process can be classified in terms of jump activity: finite activity and infinite activity. A finite-activity Lévy process can generate only a finite number of small and large jumps in a finite time interval. Brownian motion and the compound Poisson process are two examples of the finite-activity Lévy process. In contrast, the infinite-activity Lévy process can generate an infinite number of small jumps; however, the number of large jumps can only be finite because the Lévy process is often assumed with càdlàg paths. Importantly, infinite-activity jumps are more flexible to generate distributions with desired kurtosis, skewness, or tails.

Over the years, a number of option pricing models with infinite-activity Lévy jumps have been proposed, such as the Variance Gamma (VG) model (Madan et al., 1998), the Normal Inverse Gaussian (NIG) model (Barndorff-Nielsen, 1997b,a), the CGMY model (Carr et al., 2002), and the finite moment log stable process (Carr and Wu, 2003). Carr et al. (2003) propose the time-changed Lévy model and incorporate stochastic volatility into the Lévy process through an instantaneous time change, and the resultant model is still a tractable affine model, according to Duffie et al. (2003) and Kallsen (2006).

Moreover, Wu (2005) conducts an extensive analysis of the empirical performance of different types of time-changed Lévy models discussed in Carr et al. (2003).

*Combination of stochastic volatility and jumps*

Combining stochastic volatility models and infinite-activity Lévy jumps may further improve model performance, but this specification has not been widely studied. Two notable exceptions are Li et al. (2008) and Yu et al. (2011). Specifically, they compare affine SQR stochastic volatility models with infinite-activity VG and Log Stable (LS) jumps in the return process, without jumps in the variance process, against the Bates model and the double-Poisson-jump model. The models are estimated using the S&P 500 index returns in Li et al. (2008) and the joint information of the S&P 500 index returns and short-term ATM SPX option prices in Yu et al. (2011). Their empirical results show that models with infinite-activity Lévy jumps significantly outperform the Bates and double-Poisson-jump models in goodness of fit and achieve better performance in the option pricing test in Yu et al. (2011). More recently, with a discrete-time affine GARCH model, Ornthanalai (2014) estimates five types of infinite-activity Lévy jumps using the joint information of index returns and option prices by the maximum likelihood estimation (MLE) method, and suggests that infinite-activity jumps, rather than Brownian motion, should be "the default modelling choice" for option valuation. The difference between my research and theirs lies mainly in the use of non-affine variance dynamics and infinite-activity variance jumps.

Previous empirical studies have not focused on whether the use of jumps in both the return and variance processes can obviate the non-affine variance process, or vice versa. One interesting paper is Kaeck and Alexander (2012), which uses an extensive data set to estimate and compare the model specifications of the non-affine linear variance process and the CEV process with the inclusion of a finite-activity compound Poisson jump structure in the return and variance processes. Kaeck and Alexander (2012) conclude that the inclusion of compound Poisson jumps is less important than allowing for non-affine dynamics. The main difference between my research and Kaeck and Alexander (2012) lies in the use of infinite-activity Lévy jumps.

Surprisingly, very little research focuses on the empirical option pricing performance of stochastic volatility models with infinite-activity Lévy jumps in both variance and returns, especially with non-affine variance dynamics, perhaps due to computational challenges. The first part of this thesis aims to fill this gap.

Overall, empirical research with long-term data sets, including booms and crises, is necessary for making robust conclusions on the empirical performance of different types of infinite-activity jumps and non-affine variance dynamics. The recent literature contains only a few papers that use the joint information of returns and options to study model performance with infinite-activity Lévy jumps. Some exceptions include Yu et al. (2011), Li (2011), and Ornthanalai (2014). In this thesis, I use the S&P 500 index returns and option prices via the 30-day VIX index from January 1996 to December 2009, a 14-year period in which several financial crises occurred, to estimate the models. The VIX is computed by averaging the weighted prices of SPX call and put options with the target maturity over a wide range of strike prices, and under certain model assumptions, it measures the square root of expected integrated variance over the horizon defined by the target maturity. Since the VIX index is computed from a portfolio of option prices, it contains aggregated information about option prices and can be used to derive the risk-neutral dynamics of a model. It is worth mentioning that the optimal strategy is

using option prices directly to estimate models, because the VIX index averages out the information contained in option prices. However, because I consider both the affine and non-affine models in this thesis, I employ the VIX index, instead of option prices, to estimate the risk-neutral model parameters. Specifically, since the characteristic functions of non-affine models are not available in closed form, the efficient Fast Fourier Transform (FFT) proposed in Carr and Madan (1999) is not applicable. Therefore, with option prices, time-consuming Monte Carlo methods should be used to estimate the models. However, since the closed-form formula of the VIX index can be derived in all the affine and non-affine models discussed in this thesis, using the VIX enables me to estimate the non-affine models efficiently. Several previous studies, such as Ait-Sahalia and Kimmel (2007), Duan and Yeh (2010), Kaeck and Alexander (2012), and Kanniainen et al. (2014), have already confirmed the effectiveness of using the VIX for estimating the risk-neutral model dynamics.

### 1.1.2 Bayesian Estimation Methods

Many financial research papers apply the classic MLE method to estimate and compare models (see, for example Ait-Sahalia and Kimmel, 2007; Bates, 2006). However, the application of the MLE requires that the form of the joint probability density function (PDF) of the data and the specifications of the moments of the joint PDF should be known. It also requires that the joint PDF be evaluated for all parameter values. Therefore, although the MLE has proved excellent for model estimation, its application is limited. Many other estimation methods with relaxed assumptions are employed in finance studies, such as the Generalised Method of Moments in Andersen and Sørensen (1996) and Chacko and Viceira (2003), the Efficient Method of Moments in Chernov and Ghysels (2000), and the Method of Moments techniques used in Pan (2002). However, these estimation techniques require that the moments be fully specified, although the assumption of a fully-specified distribution in the MLE is removed.

In contrast, Bayesian estimation methods do not have the above requirements. Their implementations by means of sophisticated Monte Carlo techniques (Liu, 2004; Robert and Casella, 2004) have become very popular over the last two decades. In particular, Bayesian methods have recently gained popularity in the finance research and have been applied to estimate complicated financial models (see, for example, Christoffersen et al., 2010a; Eraker et al., 2003; Kaeck and Alexander, 2013a; Li, 2011, and reference therein). First, Bayesian methods are very flexible and can combine the information of index returns and options prices. Second, Bayesian methods allow for not only estimating the parameters but also extracting the latent state variables of models, such as jumps and variance. Third, as general and flexible tools for simulating complicated stochastic processes, Bayesian methods can be combined with other estimation methods. For example, when integrated with the MLE, they can solve the problem of a likelihood that is not known in closed form (see, for example, Jacquier et al., 2007; Johansen et al., 2008).

One shortcoming of Bayesian methods is that they are typically implemented on the basis of approximation. In most cases where Bayesian methods are employed, we cannot directly sample from the true distribution that we are interested in, that is, $p(\Theta, X|Y)$, where $\Theta$ is the parameter set, $X$ represents the latent state variable, and $Y$ represents the data. Bayesian methods try to approximate the true distribution by generating approximate samples, and this has plagued designers of Bayesian estimation algorithms since the beginning. Although the literature contains powerful mathematical theory for assessing the mixing and accuracy of Bayesian methods, such as the asymptotic and limit

theories, in many problems, it is difficult to assess the approximating performance of Bayesian algorithms.

Generally, there are two classes of Bayesian estimation methods: Sequential Monte Carlo (SMC) and Markov Chain Monte Carlo (MCMC). MCMC algorithms generate samples from a target distribution by drawing from a simpler proposal distribution (Liang et al., 2010; Liu, 2004) and by generating a Markov chain. SMC algorithms, or particle filters, use a finite set of particles to represent the target distribution and compute probabilistic properties on the basis of the particle set. SMC algorithms are on-line estimation methods, which means that when a new observation arrives, the particles can be updated to account for the information brought by the new observation; in contrast, off-line MCMC algorithms have to be restarted. However, since particle filters were originally designed for extracting latent state variables in a model, they were not capable of dealing with estimation involving unknown, fixed model parameters. In the past two decades, although there have been various adaptations of particle filters aimed at simultaneously handling the estimation of state variables and unknown fixed parameters, their performance must be tested further. Conversely, MCMC algorithms are typically flexible in dealing with problems of estimating parameters and state variables, but the mixing speed of the Markov chain, which is the key to the success of MCMC algorithms, can be very slow, especially when the proposal is poorly selected.

In the past decade, numerous new Bayesian estimation methods have been proposed, targeted at solving the problems of conventional MCMC algorithms in the presence of high dimensions, complicated target distributions, and complex patterns of dependence between stable variables and parameters. The second part of this thesis seeks to apply these methods to estimate different financial models and compare their estimation performance.

The Adaptive Metropolis (AM) algorithm, introduced by Haario et al. (1999a, 2001), aims to develop an effective *Gaussian* proposal by updating its variance on the basis of past samples generated in the MCMC chain. According to Haario et al. (1999a, 2001, 2006), the MCMC chain generated by this on-line tuning proposal is no longer Markovian or reversible. However, according to Haario et al. (2001), under some regularity conditions about how the adaptation is conducted, the chain is ergodic and retains the desired stationary distribution.

Moreover, several automatic and self-tuned samplers have been proposed for dealing with more general proposals, such as Adaptive Rejection Sampling (ARS) (Gilks, 1992; Gilks and Wild, 1992), Adaptive Rejection Metropolis Sampling (ARMS) (Gilks et al., 1995, 1997; Meyer et al., 2008), Independent Doubly Adaptive Rejection Metropolis Sampling (Martino et al., 2012, 2014), and Adaptive Sticky Metropolis (Martino et al., 2013). Generally, these algorithms construct the proposals with a number of support points, and adaptively modify the proposals by adding new points. However, the ARS cannot dealing with log-concave target distributions, since it is based on the rejection sampling technique and the proposal must always be above the target. In the ARMS, support points cannot be added inside regions where the proposal is below the target. Moreover, the performance of the above algorithms is dependent on the choice of initial support points, and it is difficult to ensure their ergodicity, especially in applications within the Gibbs sampler (Gilks et al., 1997; Robert and Casella, 2004). Recently, the Fast Universal Self-tuned Sampler (FUSS) algorithm (Martino et al., 2015) is proposed to overcome the above drawbacks. The FUSS algorithms construct effective self-tuned proposals, starting with a large number of support points, and then remove many of them according to some pruning scheme combining relevant information. The numerical

experiments in Martino et al. (2015) show that FUSS algorithms are able to generate virtually independent samples in the presence of high dimensions and spiky distributions.

Alternatively, it is possible to make use of an SMC algorithm as the proposal distribution within an MCMC algorithm. More specifically, an SMC algorithm can be used to approximate the likelihood to be used within a standard Metropolis-Hastings (MH) algorithm, and the resultant algorithm can be regarded as an approximation to a marginal MH that targets the marginal posterior $p(\Theta|Y)$ of the joint posterior $p(\Theta, X|Y)$, where $Y$ represents the data, $\Theta$ represents the set of parameters in a model, and $X$ represents the latent state variables. This approach, called Particle Marginal Metropolis-Hastings (PMMH) in Andrieu et al. (2010), is used in some literature (see, for example, An and Schorfheide, 2007; Fernández-Villaverde and Rubio-Ramírez, 2005), and Andrieu et al. (2010) prove the convergence of the algorithm, such that to ensure the convergence of the generated chain, the resampling algorithm should be unbiased, so that the estimation error produced by the approximation does not change the equilibrium distribution.

Andrieu et al. (2010) also propose a new scheme called Particle Gibbs (PG) sampler, which can be regarded as an approximation of the Gibbs sampler that targets the joint posterior $p(\Theta, X|Y)$. In Andrieu et al. (2010), the PMMH and PG sampler are collectively called Particle Markov Chain Monte Carlo (PMCMC) methods. The PG sampler uses a so-called conditional SMC update to ensure that the PG kernel leaves the exact target distribution invariant. In a conditional SMC update, a pre-specified reference trajectory of latent state variables with an ancestral lineage survives after all the resampling steps. After a complete run of the SMC update, a new reference trajectory is selected from the particle set with probabilities given by the importance weights of the particles. However, as underlined in Lindsten and Schön (2013) and Chopin and Singh (2013), a potential problem with the PG sampler is that in the presence of high dimensions, the path degeneracy is inevitable and the mixing of the chain may be poor. This problem is addressed by the Particle Gibbs with Ancestor Sampling (PGAS), proposed in Lindsten et al. (2014), by adding a backward sampling step to the PG sampler. Numerical experiments show that this backward sampling step can significantly improve the mixing speed of the chain, even with a small number of particles.

While some finance research applies the conventional MCMC or SMC algorithms to estimate stochastic volatility models with jumps, few attempts have been made to apply the above newly proposed algorithms. Indeed, the problem of estimating stochastic volatility models with jumps is particularly difficult because of the strong dependence between parameters and state variables and the model complexity brought by jump components. Therefore, it would be interesting to examine how the advanced estimation methods perform in estimating complicated financial models compared to the conventional MCMC algorithms.

## 1.2 Questions and Research Methods

This research is divided into two parts. The first part of the research focuses on estimating and comparing different model specifications, namely, the inclusion of return jumps of finite or infinite activity to the stochastic volatility model, affine/non-affine variance dynamics, and the inclusion of variance jumps correlated with return jumps. Eighteen model specifications are estimated from an extensive data set using conventional MCMC algorithms and are then compared in terms of goodness of fit and option pricing performance. The second part focuses on the application and comparison of advanced

Bayesian estimation methods, with the aim to identifying a method that is capable of efficiently estimating the unknown fixed model parameters and latent state variables in the presence of high dimensions, strong dependence between parameters and state variables, and complicated target distributions.

In the first part of my research, I study the models with infinite-activity return jumps, augmented with correlated infinite-activity variance jumps and specifications of affine and non-affine variance dynamics. I use both VG and NIG jumps, which are popular examples of infinite-activity pure jump processes; the VG of finite variation with a relatively low arrival rate of small jumps; and the NIG of infinite variation with a sample path that may have infinite total variation in any bounded time interval, almost surely. These models are compared against the benchmark Heston model without any jump component and the Bates and double-Poisson-jump models with finite-activity compound Poisson jumps.

This research aims to answer the following questions:

- Do inclusions of infinite-activity jump components in returns and variance and the use of a non-affine variance process improve the goodness of fit and option pricing performance, as compared to the standard SQR models with or without finite-activity compound Poisson jumps (the Heston, Bates, and double-Poisson-jump models)?
- In particular, which Lévy jump, VG or NIG, in returns and variance, better describes the return and variance dynamics and prices options?
- Moreover, how do Lévy jumps of finite activity and infinite activity differ in capturing jumps in returns and variance?

These are important questions because there is a trade-off between model accuracy and a real computational challenge due to the need for Monte Carlo methods in the option pricing that use double-infinite-activity-jump models or non-affine models.

In the first part of the research, I estimate the models using the joint information of the S&P 500 index returns and option prices via the 30-day VIX index from January 1996 to December 2009, a 14-year period with several financial crises. The models are estimated with conventional MCMC algorithms, including the Gibbs sampler, the MH algorithm, and the Damien-Wakefield-Walker (DWW) method to increase efficiency. After the estimation, I compare the model performance in terms of goodness of fit and option pricing errors in two sample periods: January 1996 – December 2009 and January 2010 – December 2010. The option prices are computed with the Monte Carlo method, and instead of using the realised return time series or a simulation method to predict the daily spot variance when pricing multiple daily cross-sections of options, I use the VIX-based technique to extract the time series of daily spot variance, as in Kanniainen et al. (2014). The results in Kanniainen et al. (2014) clearly show that this technique can improve the option pricing performance across different models, including the NGARCH and the Heston-Nandi models, over the traditional approach of estimating spot variance from realised returns.

In the second part of the research, I examine the performance of advanced MCMC algorithms. The algorithms comprise the AM, the FUSS, the PMMH, and the PGAS.

I aim to answer the following questions:

- Do inclusions of jump components of finite or infinite activity affect the estimation performance?
- Can the problems of conventional MCMC algorithms be solved, or at least alleviated, by using advanced MCMC algorithms?
- Which algorithm performs best with a fixed length of chain?
- What are the advantages and disadvantages of each algorithm in estimating the financial models?

I first conduct simulation studies to compare the performance of algorithms. In the simulation studies, I simulate one-year data of index returns and variance from the affine Heston and Bates models and the affine model with NIG return jumps and then use the simulated index returns as observations to estimate the model parameters under the physical measure. Then I examine the performance of algorithms with different chain lengths and numbers of particles in the PMMH and PGAS algorithms and compare the algorithms in terms of the parameter estimates, extraction of variance and jumps, acceptance rate, and the likelihood of the estimation result. Next, I apply the advanced MCMC algorithms to more complex problems with a large volume of empirical data. Both the physical and risk-neutral dynamics of the non-affine model with NIG return jumps, which performs best in the model comparison part, are estimated using the joint information of the S&P 500 index returns and option prices via the 30-day VIX index from January 2002 to December 2005. Besides the aspects compared in the simulation studies, I employ the estimated model to price the index options in 2002 – 2010, and compare the option pricing performance.

## 1.3 Outline and Contributions of the Thesis

This thesis is divided into five chapters. The contents of each chapter are summarised as follows. Chapter 1 briefly introduces the financial models and Bayesian estimation methods and describes the motivation, objectives, and contributions of the thesis. Chapter 2 describes the physical and risk-neutral dynamics of the models studied in this thesis, explains the application of the conventional MCMC estimation methods, and presents the empirical results. Chapter 3 introduces the general ideas and designs of advanced MCMC algorithms, explains their applications to selected financial models, and presents the estimation results using the simulated data. Chapter 4 focuses on the application of advanced MCMC algorithms to model estimation using the empirical data. Chapter 5 concludes the thesis.

The empirical results in the first part are summarised as follows. First and most importantly, the inclusion of infinite-activity return jumps is critical for non-affine (linear) variance specification. In particular, the non-affine model with infinite-activity NIG return jumps significantly outperforms the affine and non-affine models with and without finite-activity return jumps in both goodness of fit and option pricing. Its performance is also clearly better than that of the non-affine model with VG return jumps. Interestingly, the performance of infinite-activity VG and NIG return jumps is mixed with the affine (SQR) variance specification, suggesting that using unrealistic affine variance dynamics may negatively affect the identification of the models' jump components. Overall, it is the combination of infinite-activity NIG jumps in the return process and non-affine variance specification that makes the model efficient and robust.

Second, the role of infinite-activity variance jumps is less important than that of infinite-activity return jumps. Obviously, in terms of goodness of fit, the relatively parsimonious non-affine model with NIG return jumps is almost as good as the more complex models with infinite-activity jumps in both the return and variance processes. In terms of option pricing, the non-affine model with NIG return jumps performs best among all the models that I test, including the complex models with return and variance jumps. Moreover, variance jumps are insignificant for the models with NIG jumps in returns, since they worsen the goodness of fit and cause a small improvement only in the option pricing test with affine models. However, according to the goodness-of-fit results, if one still prefers using a model with infinite-activity jumps in both returns and variance, the inclusion of NIG variance jumps, rather than VG variance jumps, in the models with VG return jumps is more appropriate. This is plausible because the NIG is of infinite variation and is more capable of capturing the frequent small jumps in the variance dynamics than the finite-variation VG.

The above findings provide two financial economic insights. First, although the Bates and double-Poisson-jump models may capture rare, large jumps with the finite-activity compound Poisson jumps, they miss a large number of frequent small jumps that may cause potential substantial losses in investment and risk management. On the other hand, the inclusion of infinite-activity jumps in the return process better captures the uncertainty of future jumps, generates more realistic return dynamics and option prices, and obviates the complex models with variance jumps. In particular, the infinite-variation NIG process as return jumps produces a larger jump risk premium than the finite-variation VG process and outperforms the VG in goodness of fit and option pricing.

Second, despite the popularity of tractable affine models in derivative pricing, my research results clearly demonstrate the dominance of non-affine models over affine ones in goodness of fit and option pricing, which is in line with the recent literature (see, for example, Christoffersen et al., 2010b; Kaeck and Alexander, 2012). More importantly, the non-affine variance specification improves model robustness, whereas the affine variance specification may lead to unstable option pricing performance between different samples. It may be dangerous to use affine models for making investment and risk management decisions; therefore, decision makers must seriously consider the trade-off between computation time and model robustness.

In the second part of my research, I note that different jump structures significantly affect the estimation performance of algorithms. The results of the simulation studies show that the algorithms fail to distinguish the randomness created by the infinite-activity jumps and Brownian diffusions in the return process of the affine model with NIG return jumps. This makes parameter estimation and variance extraction very challenging, and the extra complexity introduced by the NIG jumps requires the FUSS and PMCMC methods to increase the numbers of MCMC runs and particles, while the MH and AM are less capable of dealing with the complicated model specification. Compound Poisson jumps create rare but significant changes in the return process of the affine Bates model, which are distinctive of the diffusions; therefore, all the algorithms perfectly extract the compound Poisson jumps when estimating the affine Bates model. However, the inclusion of rare, large return jumps significantly reduces the acceptance rates of the PMCMC methods, and thus their estimation performance deteriorates. In contrast, the other algorithms are less vulnerable to the specification of compound Poisson jumps and performs consistently in estimating different models.

Considering the complexity and computational costs of different algorithms, I conclude

that for models with a simple specification, the MH is very competitive owing to its low computational cost. However, if the target distribution is complicated or the proposal of the MH is inappropriate, the acceptance rate may be very low, and most generated draws are wasted. The AM makes use of the previous samples and dynamically tunes the proposal, and this online-tuned adaptive proposal can significantly raise the acceptance rate and speed up the convergence of the chain. Moreover, in the presence of a spiky and complicated target distribution and high-dimensional state variables, numerous MCMC iterations may be required for the chain generated by the MH and AM to converge. The FUSS algorithms can tackle this problem by constructing an efficient proposal and producing virtually independent samples, as noted in Martino et al. (2015). A shortcoming of the FUSS algorithms is that with a fixed number of MCMC iterations, they are slower than the very fast MH and AM algorithms. However, if one focuses on achieving a good estimation performance, despite the relatively high computational cost, the FUSS algorithms are very competitive due to their fast and good mixing properties. In addition, the PMCMC methods can deal with complicated target distributions and strong dependence between parameters and state variables, and their computational cost is significantly lower than that of the FUSS algorithms, making them very competitive. However, when the other algorithms achieve a stable estimation performance across different model specifications, the performance of the PMCMC methods depends largely on the properties of the specific problem, and their performance may deteriorate when the dependence between state variables is weak. Moreover, as pointed out in Lindsten et al. (2014), the relative performances of the PMMH and PGAS depend on whether the ideal marginal MH or the Gibbs sampler, that is, the samplers that PMMH and PGAS approximate, respectively, has the better mixing property for the specific problem. In estimating the Heston-0 and SV-NIG-0 models, the PGAS outperforms the PMMH and the other algorithms with a small number of MCMC iterations and particles, suggesting the fast and good mixing of the PGAS kernel in estimating these two models.

In the empirical studies, the adaptive proposal used in the AM increases the acceptance rate of the MH and reduces the autocorrelation between samples of the spot variance. The FUSS-RC and PGAS further improves the extraction of the spot variance in that the autocorrelation functions (ACFs) of the spot variance extracted by the FUSS-RC and PGAS drop sharply. Furthermore, the FUSS-RC even reduces the ACF with negative autocorrelation, suggesting that the chain is efficient in generating good representatives of the target distribution. In contrast, the acceptance rates of the PMMH algorithms are very low owing to the model specification of independent NIG return jumps, and jump-related parameters are poorly identified by the PGAS. This suggests that compared to the MH, AM, and FUSS-RC, the PMCMC methods are less capable of extracting independent jumps; moreover, the inclusion of independent jumps may negatively affect the estimation performance of the PMCMC methods. Furthermore, the choice of estimation methods affects the option pricing performance. In particular, the model-implied VIX based on models estimated with the PMMH-PF1 almost completely coincides with the market VIX; therefore, since the option pricing performance is highly correlated with the fit to the VIX, the PMMH-PF1 outperforms the other algorithms in the option pricing test. In contract, the PGAS performs the worst in all three option samples owing to the poorly identified jump-related parameters under the risk-neutral measure. The MH is the second best algorithm in predicting option prices due to the use of realised variance as the initial variance, which leads to an estimation result consistent with the market conditions in 2006 – 2010.

According to the results of the empirical studies, the PGAS can cope with strong

dependence between parameters and state variables, and if one emphasises the fit to observations that can be represented as a function of model parameters and state variables, the PMMH-PF1 is very competitive. However, the performance of the PMMH-PF1 and the PGAS may be weakened by the inclusion of independent jumps. In contrast, the other algorithms are less vulnerable to the inclusion of independent jumps, and they achieve a stable performance in parameter estimation, extraction of state variables, fit to the VIX, and option pricing. In particular, an appropriate choice of initial values for state variables can significantly improve the performance of the MH; moreover, the FUSS-RC is very competitive because it can improve the estimation performance of the MH and AM and generate good representatives of the target distribution of state variables in the presence of high dimensions and a strong dependence structure.

Finally, I declare that I am the sole author of this thesis. Although this PhD thesis is a monograph and is not presented as a collection of papers, it relates to three papers that I have co-authored. First, Chapter 2 is closely related to Yang and Kanniainen (2015).[1] For this paper, I coded the MCMC algorithms, estimated the models, and mainly wrote the paper. Prof. Kanniainen mainly coded the option pricing framework and participated in the writing.

The estimation results in Chapters 3 and 4 have not been reported in any previous paper, but part of the results are obtained by the FUSS algorithms proposed by Martino et al. (2015).[2] This paper presents the key idea and design of the FUSS algorithms and uses numerical experiments to illustrate their estimation performance compared to other MCMC methods. For this paper, I coded the external MCMC algorithm to estimate a stochastic volatility model with VG return jumps, estimated the model using simulated data, and wrote the financial example. I did not code the FUSS algorithms that were applied inside the external MCMC.

Kanniainen et al. (2014)[3] present an efficient way to employ information of the VIX index in option valuation, which I also employ in this thesis. For this paper, I participated in implementing the MLE method and I had a minor role in the writing.

[1] Yang, H. and J. Kanniainen (2015), "Jump and Volatility Dynamics for the S&P 500: Evidence for Infinite-Activity Jumps with Non-Affine Volatility Dynamics from Stock and Option Markets", submitted to *Review of Finance* (under revision).

[2] Martino, L., H. Yang, D. Luengo, J. Kanniainen, J. Corander (2015),"A Fast Universal Self-tuned Sampler within Gibbs Sampling", forthcoming in *Digital Signal Processing*.

[3] Kanniainen, J., B. Lin, and H. Yang (2014), "Estimating and Using GARCH Models with VIX Data for Option Valuation", *Journal of Banking and Finance*, 43, 200-211.

# 2 Jumps and Volatility Dynamics for the S&P 500 Index: Conventional MCMC

In this chapter, I compare 18 stochastic volatility models with finite/infinite-activity Lévy jumps in returns, or in both returns and variance, and with affine/non-affine variance dynamics in terms of goodness of fit and option pricing errors. The models are estimated with conventional MCMC algorithms using the joint information of the S&P 500 index returns and 30-day VIX index reformulated from option data during 1996 – 2009.

This chapter seeks to answer the following questions:

- Do the inclusion of infinite-activity jump components in returns and variance and the use of a non-affine variance process improve the goodness of fit and the option pricing performance, compared to the standard SQR models with or without finite-activity compound Poisson jumps (the Heston, Bates, and double-Poisson-jump models)?
- In particular, which Lévy jump, VG or NIG, in returns and variance, better describes the return and variance dynamics and prices options?
- Moreover, how do Lévy jumps of finite/infinite activity differ in capturing jumps in returns and variance?

The outline of this chapter is as follows. Section 2 introduces the affine and non-affine models with finite/infinite-activity jumps and describes the change of measure. In Section 3, I derive the formulae of the model-implied VIX index and assume a relation between the market VIX and model-implied VIX. Section 4 describes the estimation method. Section 5 describes the data set and option pricing methods. Section 6 presents the empirical results, including parameter estimation, extracted jumps, goodness of fit, and option pricing errors. Section 7 concludes the chapter.

## 2.1 Description of Models and Notations

In this part, I briefly describe the models to be compared in this chapter. First, the types of Lévy jumps considered here comprise the compound Poisson jump, which is of finite activity, and the VG and NIG jumps, which are of infinite activity. Second, stochastic volatility is generated by diffusion and Lévy jumps that are correlated with jumps in the return process. Moreover, apart from the conventional SQR variance process, the non-affine variance process with the linear diffusion term is used.

Lévy processes are used because of their flexible distributions, whereas Brownian motion, as a special type of Lévy process with continuous paths, restricts itself to following the symmetric normal distribution. The Lévy process may generate any path as long as the distribution of the increments of the path is stationary and independent of its past track, and the path does not have to be continuous. When used in a model to fit a real-life stock price, the Lévy process may help resolve some known empirical biases of the BS model, such as the realised skewness and excess kurtosis in the distribution of stock returns, and may capture the jump risks that are missed by the BS and Heston models.

As in Carr and Wu (2004) and Tankov (2003), the Lévy process can be classified in terms of jump activity according to the behaviour of the Lévy measure $\pi$: if $\pi$ satisfies

$$\int_{-\infty}^{\infty} \pi(dx) < \infty,$$

the Lévy process is of finite activity. This implies that $\int_{|x|<1} \pi(dx) < \infty$ and that $\int_{|x|\geq 1} \pi(dx) < \infty$; therefore, the Lévy process can have only a finite number of both small and large jumps per unit of time. However, if $\pi$ satisfies

$$\int_{-\infty}^{\infty} \pi(dx) = \infty,$$

the Lévy process is of infinite activity. This means that this Lévy process can generate an infinite number of small jumps and a finite number of large jumps per unit of time, because $\int_{|x|<1} \pi(dx) = \infty$ and $\int_{|x|\geq 1} \pi(dx) < \infty$, where the latter inequality is entailed by the definition of the Lévy process. As pointed out in Li et al. (2008), the infinite-activity Lévy process can generate small jumps for the path of stock returns that are too big for Brownian motion, and too small and frequent for the finite-activity compound Poisson process, to capture.

Let $\{Y_t\}$ be the continuously compounded return of the S&P 500 index, and let $\{v_t\}$ be the instantaneous squared volatility of the return. Then under the physical measure $P$, I assume that the return dynamics of the S&P 500 index is described by the following stochastic differential equations:

$$dY_t = \left(\mu_t - \frac{1}{2}v_t + \phi_J^P(-i)\right)dt + \sqrt{v_t}dW_{Y,t} + dJ_{Y,t}, \tag{2.1}$$

$$dv_t = \kappa^P(\theta^P - v_t)dt + \alpha v_t^{\beta}(\rho dW_{Y,t} + \sqrt{1-\rho^2}dW_{v,t}) + dJ_{v,t}, \tag{2.2}$$

where $W_{Y,t}$ and $W_{v,t}$ are mutually independent Brownian motions. $J_{Y,t}$ and $J_{v,t}$ are Lévy jumps in the return and variance processes, respectively, of finite or infinite activity. $\phi_J^P(u)$ is calculated from $E^P[e^{iuJ_{Y,t}}] = e^{-t\phi_J^P(u)}$, and it measures the expectation of $e^{J_{Y,t}}$ because $E^P[e^{J_{Y,t}}] = e^{-t\phi_J^P(-i)}$; therefore, $E^P[S_t]$, the expectation of the index price $S_t = e^{Y_t}$, equals $S_0 E^P[e^{\int_0^t \mu_s ds}]$, as in Madan et al. (1998). $\mu_t$ is the physical time-varying drift of the index return, $\kappa^P$ is the physical rate of the mean reversion of $v_t$, $\theta^P$ represents the physical long-term mean of $v_t$, $\alpha$ controls the diffusion term in the variance process, and $\rho$ controls the correlation between $Y_t$ and $v_t$ and captures the leverage effect. The parameter $\beta$ determines whether the variance dynamics is an affine SQR process or a non-affine linear process. The linear specification is often called the continuous-time GARCH model (see, for example, Ait-Sahalia and Kimmel, 2007).

### 2.1.1 Benchmark Models: Heston, Bates, and Double-Poisson-Jump Models

The models considered as benchmarks are the Heston, Bates, and double-Poisson-jump models. The last model has contemporaneous jumps in both returns and variance occurring at random times, depending on the increments of the Poisson process $N_t$ with intensity $\lambda$:

$$J_{Y,t} = \sum_{i=1}^{N_t} \xi_i^Y, \quad J_{v,t} = \sum_{i=1}^{N_t} \xi_i^v, \quad N_t \sim \text{Poisson}(\lambda t). \tag{2.3}$$

The size of jumps in the variance process follows an exponential distribution: $\xi_t^v \sim \exp(\mu_v)$, where $\mu_v$ is the scale parameter. The reason for choosing the exponential distribution for variance jumps is to ensure the positiveness of these jumps given the economic observation that realised variance typically demonstrates large, positive jumps. The size of jumps in the return process follows a normal distribution with the mean correlated with the jumps in variance: $\xi_t^Y | \xi_t^v \sim N(\mu_y + \rho_J \xi_t^v, \sigma_y^2)$, where $\rho_J$ models the correlation between return and variance jumps. The double-Poisson-jump model nests the Bates model by removing the variance jumps, $J_{v,t}$, and the Heston model by removing both $J_{Y,t}$ and $J_{v,t}$.

In this thesis, the double-Poisson-Jump models are labelled SV-P-P-0 when $\beta = 0.5$ and SV-P-P-1 when $\beta = 1$ (the notation comes from **S**tochastic **V**olatility, compound **P**oisson jumps in returns, and compound **P**oisson jumps in variance). Similarly, the Heston and Bates models are labelled SV and SV-P, respectively, but for simplicity, their existing names are used and the models are labelled Heston-0, Heston-1, Bates-0, and Bates-1, depending on the values of $\beta$.

### 2.1.2 Infinite-Activity Lévy Jumps

First, the VG and NIG jumps are added only in the return process, and the resultant models are labelled SV-VG-$i$ or SV-NIG-$i$, according to the choice of return jumps and values of $\beta$. The VG process can be defined by subordinating a Brownian motion with drift $\gamma$ and variance $\sigma$ by an independent Gamma process $G_Y$ with a unit mean rate and a variance rate $\nu$:

$$J_{Y,t}(\sigma, \gamma, \nu) = \gamma G_{Y,t}(\nu) + \sigma W_{Y, G_{Y,t}(\nu)} \tag{2.4}$$

and its Lévy measure is

$$\pi_{VG}(dx) = \frac{e^{-Mx}dx}{\nu x}, x > 0; \quad \pi_{VG}(dx) = \frac{e^{-N|x|}dx}{\nu|x|}, x < 0, \tag{2.5}$$

where

$$M = \left( \frac{1}{2}\gamma\nu + \sqrt{\frac{1}{4}\gamma^2\nu^2 + \frac{1}{2}\sigma^2\nu} \right)^{-1}, \tag{2.6}$$

$$N = \left( -\frac{1}{2}\gamma\nu + \sqrt{\frac{1}{4}\gamma^2\nu^2 + \frac{1}{2}\sigma^2\nu} \right)^{-1}. \tag{2.7}$$

Similarly, an NIG process can be obtained by subordinating a Brownian motion with drift $\gamma$ and variance $\sigma$ by an independent inverse Gaussian process $G_Y$. $G_Y$ is the first

time that a Brownian motion with drift $\nu$ reaches the positive level $t$. The Lévy measure of the NIG is

$$\pi_{NIG}(dx) = \frac{\sigma\alpha}{\pi}\frac{e^{\beta x}K_1(\alpha|x|)}{|x|}dx, \tag{2.8}$$

where $\alpha = \nu^2/\sigma^2 + \gamma^2/\sigma^4$, $\beta = \gamma/\sigma^2$, and $K_1$ is the modified Bessel function of the third kind with index 1. The NIG is also of infinite activity; moreover, it is of infinite variation. A Lévy process is of infinite variation if the total variation of its sample path in any bounded time interval is infinite, almost surely, for any partition of the time interval. When the VG is of finite variation and hence can only have jump-type discontinuities and is bounded in any bounded interval, the infinite-variation NIG is more flexible and, therefore, a potential candidate for good performance in modelling financial time series.

Further, an infinite-activity Lévy jump component, VG or NIG, is added to the variance process, motivated by the findings in Eraker et al. (2003), which indicate the necessity of jumps in the variance process to account for extreme market conditions. The models obtained with infinite-activity jumps in returns and variance are labelled SV-VG-VG-$i$, SV-VG-NIG-$i$, SV-NIG-VG-$i$, and SV-NIG-NIG-$i$, $i = 0, 1$, according to the choices of return and variance jumps and the values of $\beta$.

The structure of the jumps $J_v$ in variance is similar to that of $J_Y$ in returns, regardless of the jump subordinator:

$$J_{v,t}(\sigma_v, \gamma_v, \nu_v) = \gamma_v G_{v,t}(\nu_v) + \sigma_v W_{v,G_{v,t}(\nu_v)}. \tag{2.9}$$

Here, $G_v(\nu_v)$ can be an inverse Gaussian or Gamma process, independent of the other random sources. $W_{v,G_v}$ is the base Brownian motion correlated with $W_{Y,G_Y}$ by $\rho_v$, as in Eberlein and Madan (2009).

Overall, there are *18 different model specifications*, and they are summarised in Table 2.1.

## 2.2 Change of Measure

In this section, I describe the risk-neutral dynamics and the change of measure.

I follow the procedure described in Pan (2002) to define the change of the probability measure for Brownian motions in the return and variance processes in all the models characterised above. Briefly, assuming two risk premia, $\eta_s$ and $\eta_v$, the change of measure for Brownian motions is as follows:

$$dW_{Y,t}(Q) = dW_{Y,t} - \eta_s\sqrt{v_t}dt \tag{2.10}$$

$$dW_{v,t}(Q) = dW_{v,t} + \frac{1}{\sqrt{1-\rho^2}}\left(\rho\eta_s + \frac{\eta_v}{\alpha v_t^{\beta-0.5}}\right)\sqrt{v_t}dt. \tag{2.11}$$

In defining the change of measure for the Lévy jumps in returns, I follow the practice in Yu et al. (2011). Under the Sato theorem, which states the relation between the physical and risk-neutral measures of the Lévy process, and the restriction that the jump $J_Y(Q)$ under $Q$ is still a VG jump, a change of measure for $J_Y$ exists if $\nu^P = \nu^Q$, while $\gamma^Q$ and $\sigma^Q$ may change freely. For a compound Poisson jump, all jump-related parameters can change freely between $P$ and $Q$; however, for simplicity and econometric meaning, only $\mu_y^Q$ is assumed to change freely. For models with an NIG jump component in returns, the Esscher transform is used to define the risk-neutral measure $Q$, under which the jump in returns

**Table 2.1:** Model acronyms and specifications

| **Model name** | **Heston-0** | **Heston-1** |
|---|---|---|
| | 0.5 | 1 |
| Jump in return | None | |
| Jump in variance | None | |

| **Model name** | **Bates-0** | **Bates-1** |
|---|---|---|
| | 0.5 | 1 |
| Jump in return | compound Poisson | |
| Jump in variance | None | |

| **Model name** | **SV-P-P-0** | **SV-P-P-1** |
|---|---|---|
| $\beta$ | 0.5 | 1 |
| Jump in return | compound Poisson | |
| Jump in variance | compound Poisson | |

| **Model name** | **SV-VG-0** | **SV-VG-1** |
|---|---|---|
| $\beta$ | 0.5 | 1 |
| Jump in return | Variance Gamma | |
| Jump in variance | None | |

| **Model name** | **SV-VG-VG-0** | **SV-VG-VG-1** |
|---|---|---|
| $\beta$ | 0.5 | 1 |
| Jump in return | Variance Gamma | |
| Jump in variance | Variance Gamma | |

| **Model name** | **SV-VG-NIG-0** | **SV-VG-NIG-1** |
|---|---|---|
| $\beta$ | 0.5 | 1 |
| Jump in return | Variance Gamma | |
| Jump in variance | Normal inverse Gaussian | |
| **Model name** | **SV-NIG-0** | **SV-NIG-1** |
| $\beta$ | 0.5 | 1 |
| Jump in return | Normal inverse Gaussian | |
| Jump in variance | None | |

| **Model name** | **SV-NIG-VG-0** | **SV-NIG-VG-1** |
|---|---|---|
| $\beta$ | 0.5 | 1 |
| Jump in return | Normal inverse Gaussian | |
| Jump in variance | Variance Gamma | |

| **Model name** | **SV-NIG-NIG-0** | **SV-NIG-NIG-1** |
|---|---|---|
| $\beta$ | 0.5 | 1 |
| Jump in return | Normal inverse Gaussian | |
| Jump in variance | Normal inverse Gaussian | |

remains of the NIG type, with the same $\sigma^P$ and $\sigma^Q$ and the same $\alpha^P_{NIG} = \frac{(\nu^P)^2}{(\sigma^P)^2} + \frac{(\gamma^P)^2}{(\sigma^P)^4}$ and $\alpha^Q_{NIG} = \frac{(\nu^Q)^2}{(\sigma^Q)^2} + \frac{(\gamma^Q)^2}{(\sigma^Q)^4}$, but a different $\beta^Q_{NIG} = \frac{\gamma^P}{(\sigma^P)^2}$ and $\beta^Q_{NIG} = \frac{\gamma^Q}{(\sigma^Q)^2}$. It can be easily worked out that the restriction on the values of risk-neutral jump-related parameters $\sigma^Q$, $\gamma^Q$, and $\nu^Q$ is $\sigma^P = \sigma^Q$, $\left(\nu^P\right)^2 = \frac{(\nu^Q)^2(\sigma^Q)^2+(\gamma^Q)^2-(\gamma^P)^2}{(\sigma^Q)^2}$, and $\gamma^Q$ can change freely. In fact, under a pure NIG jump model for stock price $dS_t = S_{t-}dJ_t$, where $J_t$ is an NIG jump process, $\beta^Q_{NIG}$ and $\gamma^Q$ can be worked out from the other jump parameters and a constant risk-free interest rate. However, in reality, the risk-free interest rate may vary, and as the model structure becomes complicated, it is difficult to compute $\beta^Q_{NIG}$. Consequently, in the estimation, $\gamma^Q$ is identified from the VIX data.

Given the change of measure for the jumps and diffusion terms, the Radon-Nikodym derivative for the models is

$$\frac{dQ}{dP}|_t = \exp\left\{-\int_0^t \zeta_{Y,s}dW_{Y,s} - \int_0^t \zeta_{v,s}dW_{v,s} - \frac{1}{2}\left[\int_0^t \left(\zeta^2_{Y,s} + \zeta^2_{v,s}\right)ds\right]\right\} e^{U_t}, \quad (2.12)$$

where $e^{U_t}$ is defined in the Sato theorem (details can be found in Sato, 1999) and $\zeta_Y$ and $\zeta_v$ are the market prices of risks of Brownian innovations in returns and variance, respectively, defined as

$$\zeta_{Y,t} = -\eta_s\sqrt{v_t}, \quad (2.13)$$

$$\zeta_{v,t} = \frac{1}{\sqrt{1-\rho^2}}\left(\rho\eta_s + \frac{\eta_v}{\alpha v_t^{\beta-0.5}}\right)\sqrt{v_t}. \quad (2.14)$$

Therefore, under the risk-neutral measure $Q$, the return and variance follow

$$dY_t = \left(r_t - \frac{1}{2}v_t + \phi^Q_J(-i)\right)dt + \sqrt{v_t}dW_{Y,t}(Q) + dJ_{Y,t}(Q), \quad (2.15)$$

$$dv_t = \kappa^Q(\theta^Q - v_t)dt + \alpha v_t^\beta\left(\rho dW_{Y,t}(Q) + \sqrt{1-\rho^2}dW_{v,t}(Q)\right) + dJ_{v,t}(Q), \quad (2.16)$$

Here, $\phi^Q_J(-i)$ is the jump compensator for $J_Y$ under measure $Q$, and its form depends on the specific type of $J_Y$:

$$\phi^{Q,CPoisson}_J(u) = \lambda\left(1 - \frac{e^{iu\mu^Q_y - \frac{1}{2}\sigma^2_y u^2}}{1 - iu\mu_v\rho_J}\right), \quad (2.17)$$

$$\phi^{Q,VG}_J(u) = \frac{\log\left(1 - iu\gamma^Q\nu + \frac{1}{2}u^2\nu(\sigma^Q)^2\right)}{\nu}, \quad (2.18)$$

$$\phi^{Q,NIG}_J(u) = -\nu^Q + \sigma\sqrt{u^2 - 2iu\frac{\gamma^Q}{\sigma^2} + \frac{(\nu^Q)^2}{\sigma^2}}. \quad (2.19)$$

The form of $\phi^P_J$ in Equation (2.1) is the same as $\phi^Q_J$, except that the risk-neutral parameters should be changed to the corresponding physical ones.

Furthermore,

$$\eta_s v_t = r_t - \mu_t + \phi^Q_J(-i) - \phi^P_J(-i), \quad (2.20)$$

$$\eta_v = \kappa^Q - \kappa^P, \quad (2.21)$$

where $r_t$ is the risk-free interest rate and $\kappa^Q$ is the mean-reverting speed under the risk-neutral measure $Q$.

In the variance dynamics under $Q$,

$$\kappa^Q(\theta^Q - v_t) = \kappa^P(\theta^P - v_t) - \eta_v v_t. \tag{2.22}$$

This implies that $\theta^Q\kappa^Q = \theta^P\kappa^P$. In the estimation, the physical $\theta^P$ is estimated from the data, and $\theta^Q$ is computed by $\theta^Q = \kappa^P\theta^P/\kappa^Q$.

## 2.3 VIX

In this section, I briefly introduce the construction of the VIX index from the option data (market VIX) and derive its pricing formula in different model settings (model-implied VIX). Finally, I quote an assumption that suggests a relation between the market and model-implied VIX.

The VIX measures the square root of expected integrated variance over the horizon defined by the target maturity, and it is computed by averaging the weighted prices of SPX puts and calls with the target maturity over a wide range of strike prices. Importantly, there are two reasons why I have chosen the VIX index, not option prices directly, to estimate the risk-neutral model parameters. First, the VIX index, computed from a portfolio of option prices, contains aggregated information about option prices[1] and can be used to derive the risk-neutral dynamics of a model. Second, the efficient FFT is not applicable to non-affine models because their characteristic functions are not available in closed form. Therefore, with option prices, one should use the Monte Carlo method to estimate the models, an extremely time-consuming process, or approximation methods (see, for example, Lewis, 2000). However, since the formula of the VIX index under all the affine and non-affine models discussed in this chapter can be derived in closed form, using the VIX can make the model estimation efficient. Several previous studies, such as Ait-Sahalia and Kimmel (2007), Duan and Yeh (2010), Kaeck and Alexander (2012), and Kanniainen et al. (2014), have confirmed the effectiveness of using the VIX, instead of option prices, for estimating the risk-neutral model dynamics.

Specifically, according to the definition published in 2003 by the Chicago Board Options Exchange (CBOE), the theoretical value of the squared VIX is

$$\mathrm{VIX}_{t,T}^2 \times 10^{-4} = \frac{2}{T}e^{rT}\Pi_t\left(F_t(t+T), t+T\right), \tag{2.23}$$

where $F_t(t+T)$ is the forward price of the stock with maturity $t+T$ at time $t$, and where

$$\Pi_t(K_0, t+T) = \int_0^{K_0} \frac{P_t(K, t+T)dK}{K^2} + \int_{K_0}^{\infty} \frac{C_t(K, t+T)dK}{K^2}. \tag{2.24}$$

Here, $C_t(K, t+T)$ and $P_t(K, t+T)$ are the European call and put option prices at time $t$, with maturity $t+T$ and strike price $K$. As in Duan and Yeh (2010), by the generic payoff expansion result of Carr and Madan (2001), the theoretical value of the squared VIX can be reduced to

$$\mathrm{VIX}_{t,T}^2 \times 10^{-4} = \frac{2}{T}\left(-\log\frac{S_t}{F_t(t+T)} - E^Q\left[\log\frac{S_{t+T}}{S_t}\right]\right). \tag{2.25}$$

[1]It is worth mentioning that the optimal strategy for estimating the risk-neutral dynamics is using option prices directly to estimate models, because the VIX index averages out the information contained in option prices.

According to the model assumption in my setting,

$$\begin{aligned} \mathrm{VIX}_{t,T}^2 \times 10^{-4} &= \frac{2}{T}\int_t^{t+T} r_s ds - \frac{2}{T}\Big(\int_t^{t+T} r_s ds + T(M_J + \phi_J^Q(-i)) && (2.26)\\ &\qquad -\frac{1}{2}E^Q\left[\int_t^{t+T} v_s ds\right]\Big), && (2.27)\\ &= -2(M_J + \phi_J^Q(-i)) + \frac{1}{T}E^Q\left[\int_t^{t+T} v_s ds\right]. && (2.28) \end{aligned}$$

Here, the jump compensators $\phi_J^Q(-i)$ for compound Poisson, VG, and NIG jumps are specified in Section 2.2. $M_J$ is the expected mean size of the return jumps under $Q$. For the VG jump, $M_J = \gamma^Q$, for the NIG jump, $M_J = \gamma^Q/\nu^Q$, and for the compound Poisson jump, $M_J = \lambda(\mu_y^Q + \rho_J\mu_v)$.

The average integrated variance under the risk-neutral measure $Q$ is

$$E_t^Q\left(\int_t^{t+T} v_s ds\right) = v_t\frac{1-e^{-\kappa^Q T}}{\kappa^Q} + \frac{\theta^P\kappa^P + M_v}{\kappa^Q}\left(T - \frac{1-e^{-\kappa^Q T}}{\kappa^Q}\right), \tag{2.29}$$

where $M_v$ is the expected mean size of the variance jumps under $Q$. If $J_v$ is a VG jump, $M_v = \gamma_v^Q$. If $J_v$ is an NIG jump, $M_v = \gamma_v^Q/\nu_v^Q$. In the models SV-P-P-$i$, $M_v = \lambda\mu_v$.

Moreover, the target maturity $T$ is annualised. When a 30-day VIX is computed and if the calendar day count convention is used, obviously $T = 30/365$. However, I use the trading day count convention, because the return data are recorded on a trading day basis (see also Kanniainen et al., 2014). Thus, in my setting, $T = 22/252$, as there are 252 trading days per year and 22 trading days per month. In addition, since the CBOE uses calendar days to calculate the VIX (see CBOE's documentation), squared observations on the VIX must be multiplied by $(30/365)(252/22)$ when the trading day count convention is applied.

From the VIX index, we can estimate model parameters under the risk-neutral measure $Q$ and use the estimated model for option pricing. Since all the models in this chapter have closed-form formulae for the VIX, I need not resort to the time-consuming Monte Carlo method to estimate the non-affine models. If the option pricing error computed using the estimates from the VIX is small, the VIX can be regarded as a reliable source of data from which to derive the risk-neutral dynamics.

Following Amengual (2009) and Kaeck and Alexander (2012), I assume that the theoretical value of the VIX is related to its market value as follows:

$$\mathrm{VIX}_{t,T}^{\mathrm{Market}} = \mathrm{VIX}_{t,T}^{\mathrm{Model}} \times e^{\varepsilon_t}, \tag{2.30}$$

where $\mathrm{VIX}_{t,T}^{\mathrm{Market}}$ denotes the market observation of the $T$-day VIX index at time $t$, and where $\mathrm{VIX}_{t,T}^{\mathrm{Model}}$ is obtained by Equation (2.26). The multiplicative pricing error $e^{\varepsilon}$ for maturity $T$ is supposed to follow a first-order autoregressive process as follows:

$$\varepsilon_{t+1} = \rho_\varepsilon\varepsilon_t + \sigma_\varepsilon\epsilon_{\varepsilon,t+1}, \epsilon_{\varepsilon,t+1} \sim N(0,1). \tag{2.31}$$

The parameter $\rho_\varepsilon$ measures the correlation between pricing errors on neighbouring trading days. For example, assuming $\rho_\varepsilon$ is positive and high today, the VIX pricing error is likely to be high tomorrow.

## 2.4 Estimation Method

I use the Bayesian MCMC methods to estimate the model parameters and state variables. In this section, I illustrate the model discretisation and derive the likelihood inference, which is used in the estimation.

*Discretisation*

As in Eraker (2004) and Yu et al. (2011), I apply the first-order Euler discretisation, whose bias has been shown to be very small, to the continuous-time models. After discretisation at daily frequency, the discrete version of the joint dynamics of the daily index returns and variance under the physical measure is as follows:

$$\begin{aligned} Y_{t+1} - Y_t &= (r_t - \frac{1}{2}v_t + \phi_J^Q(-i) - \eta_s v_t)\Delta + \sqrt{v_t\Delta}\epsilon_{Y,t+1} + J_{Y,t+1}, \\ v_{t+1} - v_t &= \kappa^P(\theta^P - v_t)\Delta + \alpha v_t^\beta\sqrt{\Delta}\epsilon_{v,t+1} + J_{v,t+1}, \end{aligned} \tag{2.32}$$

where $\Delta = 1/252$, $\epsilon_{v,t}$, $\epsilon_{Y,t} \sim N(0,1)$, $corr(\epsilon_{v,t}, \epsilon_{Y,t}) = \rho$.

In the double-Poisson-jump model,

$$J_{Y,t} = \xi_t^Y N_t, J_{v,t} = \xi_t^v N_t, P(N_t = 1) = \lambda\Delta. \tag{2.33}$$

In the models with VG or NIG jumps in returns and variance,

$$\begin{aligned} J_{Y,t} &= \gamma_Y G_{Y,t} + \sigma_Y\sqrt{G_{Y,t}}\epsilon_t^{J_Y}, \\ J_{v,t} &= \gamma_v G_{v,t} + \sigma_v\sqrt{G_{v,t}}\epsilon_t^{J_v}. \end{aligned} \tag{2.34}$$

Here, $\epsilon^{J_Y}$ and $\epsilon^{J_v}$ follow a bivariate standard normal distribution with the correlation parameter $\rho_v$. $G$ is the subordinator independent of any other stochastic sources. If $J_Y$ is VG, $G_{Y,t} \sim \Gamma(\Delta/\nu, \nu)$, and if $J_Y$ is NIG, $G_{Y,t} \sim IG(\Delta/\nu, \Delta^2)$. Similarly, if $J_v$ is VG, $G_{v,t} \sim G(\Delta/\nu_v, \nu_v)$, and if $J_v$ is NIG, $G_{v,t} \sim IG(\Delta/\nu_v, \Delta^2)$.

*Likelihood Inference*

Let $\mathbf{J}$ represent the set of jump variables, consisting of $\mathbf{J_Y} = \{J_{Y,t}\}$, $\mathbf{J_v} = \{J_{v,t}\}$, $\xi_\mathbf{v} = \{\xi_{v,t}\}$, $\xi_\mathbf{Y} = \{\xi_{Y,t}\}$, and $\mathbf{N} = \{N_t\}$ in the SV-P-P-$i$, and $\mathbf{J_Y} = \{J_{Y,t}\}$ and $\mathbf{G_Y} = \{G_{Y,t}\}$ in the SV-VG/NIG-$i$, and, in addition, $\mathbf{J_v} = \{J_{v,t}\}$ and $\mathbf{G_v} = \{G_{v,t}\}$ if there is a VG or an NIG jump component in variance. Moreover, $\mathbf{vix} = \{\text{vix}_t^{\text{Market}}\}$ represents the market-observed value vector of the VIX, and $\mathbf{vix}^{\text{Model}} = \{\text{vix}_t^{\text{Model}}\}$ represents the model-implied value vector of the VIX. $\mathbf{Y} = \{Y_t\}$ is the return of the S&P 500 index, and $\mathbf{v} = \{v_t\}$ is the spot variance. $\Theta$ represents the whole set of physical and risk-neutral parameters. Then, according to the Bayes' rule, the joint posterior of latent variables and parameters is

$$\begin{aligned} p(\Theta, \mathbf{v}, \mathbf{J}|\mathbf{vix}, \mathbf{Y}) &\propto p(\mathbf{vix}, \mathbf{Y}, \mathbf{v}, \mathbf{J}, \Theta) \\ &= p(\mathbf{vix}|\mathbf{Y}, \mathbf{v}, \mathbf{J}, \Theta)p(\mathbf{Y}, \mathbf{v}|\mathbf{J}, \Theta)p(\mathbf{J}|\Theta)p(\Theta). \end{aligned}$$

The last term, $p(\Theta)$, is the product of priors of all the model parameters. I choose uninformative priors as in Yu et al. (2011) or flat priors to ensure that choice of priors does not distort the estimation results and that the posteriors are dominated by the likelihoods rather than the priors.

The first likelihood of the market-VIX given parameters and latent variables can be derived from the relation between the model-implied VIX and market VIX:

$$p(\mathbf{vix} \mid \mathbf{Y}, \mathbf{v}, \mathbf{J}, \Theta) = \prod_{t=0}^{T-1} \frac{1}{\sqrt{2\pi}\sigma_\varepsilon} \exp\left\{ -\frac{\left[\log\left(\frac{\text{vix}_{\mathbf{t+1}}^{\text{Market}}}{\text{vix}_{\mathbf{t+1}}^{\text{Model}}}\right) - \rho_\varepsilon \log\left(\frac{\text{vix}_{\mathbf{t}}^{\text{Market}}}{\text{vix}_{\mathbf{t}}^{\text{Model}}}\right)\right]^2}{2\sigma_\varepsilon^2} \right\}. \tag{2.35}$$

The remaining likelihoods in the joint posterior can be calculated on the basis of the model assumptions mentioned above.

Let us take the models SV-VG-NIG-$i$ as an example. Since for every time step $t$, the joint dynamics of the model residuals of returns and variance is a bivariate normal distribution with the correlation factor $\rho$, the likelihood $p(\mathbf{Y}, \mathbf{v}|\mathbf{J}, \Theta)$ is

$$p(\mathbf{Y}, \mathbf{v}|\mathbf{J}, \Theta) \propto \prod_{t=0}^{T-1} \frac{1}{\alpha v_t^{1/2+\beta}\Delta\sqrt{1-\rho^2}} \exp\left\{ -\frac{\epsilon_{Y,t+1}^2 - 2\rho\epsilon_{Y,t+1}\epsilon_{v,t+1} + \epsilon_{v,t+1}^2}{2(1-\rho^2)} \right\}, \tag{2.36}$$

where

$$\epsilon_{Y,t} = \frac{Y_t - Y_{t-1} - \left(r_t - \frac{1}{2}v_{t-1} + \phi_J^Q(-i) - \eta_s v_{t-1}\right)\Delta - J_{Y,t}^{VG}}{\sqrt{\Delta v_{t-1}}} \tag{2.37}$$

and

$$\epsilon_{v,t} = \frac{v_t - v_{t-1} - \kappa^P(\theta^P - v_{t-1})\Delta - J_{v,t}^{NIG}}{\alpha v_{t-1}^\beta \sqrt{\Delta}}. \tag{2.38}$$

Moreover, $p(\mathbf{J}_Y^{VG}, \mathbf{J}_v^{NIG}|\mathbf{G}_Y, \mathbf{G}_v, \Theta) \propto$

$$\prod_{t=0}^{T-1} \frac{1}{\sigma\sigma_v\sqrt{(1-\rho_v^2)G_{Y,t+1}G_{v,t+1}}} \exp\left\{ -\frac{\epsilon_{Y,t+1}^2 - 2\rho_v\epsilon_{Y,t+1}\epsilon_{v,t+1} + \epsilon_{v,t+1}^2}{2(1-\rho_v^2)} \right\}, \tag{2.39}$$

where

$$\epsilon_{Y,t}^J = \frac{J_{Y,t+1} - \gamma G_{Y,t+1}}{\sigma\sqrt{G_{Y,t+1}}}, \tag{2.40}$$

$$\epsilon_{v,t}^J = \frac{J_{v,t+1} - \gamma_v G_{v,t+1}}{\sigma_v\sqrt{G_{v,t+1}}}. \tag{2.41}$$

and

$$p(\mathbf{G}_Y|\Theta) \propto \prod_{t=0}^{T-1} \frac{1}{\nu^{\Delta/\nu}\Gamma\left(\frac{\Delta}{\nu}\right)} G_{Y,t+1}^{\Delta/\nu-1} \exp\left\{ -\frac{G_{Y,t+1}}{\nu} \right\}, \tag{2.42}$$

$$p(\mathbf{G}_v|\Theta) \propto \prod_{t=0}^{T-1} \frac{\Delta}{\sqrt{2\pi}} G_{v,t+1}^{-3/2} \exp\left\{ -\frac{\nu_v^2\left(G_{v,t+1} - \frac{\Delta}{\nu_v}\right)^2}{2G_{v,t+1}} \right\}, \tag{2.43}$$

$$\tag{2.44}$$

Since for all the models discussed in this thesis, the joint posterior is not a known distribution, it is impossible to jointly update the parameters and state variables. Instead, the joint posterior must be broken down into lower-dimensional conditional posteriors of individual parameters and state variables. In each MCMC run, the state variables and parameters are sequentially updated. When the conditional posterior is a known distribution that can be directly simulated, the Gibbs sampling is used to update the parameters or state variables. Otherwise, the MH algorithm is employed as in Eraker et al. (2003), Eraker (2004), and Ignatieva et al. (2015) and others, or the DWW method proposed by Damlen et al. (1999), as in Yu et al. (2011).

The conditional posteriors of some state variables and parameters, such as $\kappa^P$ and $\theta^P$, can be worked out as in the appendices of Eraker et al. (2003), Li et al. (2008), and Yu et al. (2011). One difference between the posteriors used in my research and theirs comes from the different data used in the estimation. In Eraker et al. (2003) and Li et al. (2008), the models are estimated from the index data, and therefore only physical parameters are estimated, and in Yu et al. (2011), the models are estimated under both physical and risk-neutral measures, using the joint information of index return and option data, whereas I use the VIX to derive the risk-neutral dynamics. The other difference is the value of $\beta$: in the above three papers, only the affine models are studied, and hence $\beta$ is set at 0.5; in my research, $\beta = 1$ is also considered, which may cause adjustment in the posteriors, as in Equation (2.36).

To illustrate, I report the posteriors of parameters $\eta_s$, $\gamma^Q$, and $\sigma$ in the model SV-NIG-$i$, as they represent parameters that are estimated only from the index data, only from the VIX, and from the joint information of index and VIX, respectively. The conditional posteriors $\pi(\cdot)$ of $\eta_s$, $\gamma^Q$, and $\sigma$ are as follows:

$$\pi(\eta_s) = p(\eta_s|\mathbf{vix}, \mathbf{Y}, \mathbf{v}, \mathbf{J}, \Theta - \{\eta_s\}) \propto p(\mathbf{Y}, \mathbf{v}|\mathbf{J}, \Theta)p(\eta_s) \propto N\left(\tfrac{S}{W}, \tfrac{1}{W}\right), \quad (2.45)$$

$$S = \tfrac{a}{b} + \tfrac{1}{1-\rho^2}\sum_{t=0}^{T-1}\left(A_t - \tfrac{\rho}{\alpha}v_t^{0.5-\beta}B_t\right), W = \tfrac{1}{b} + \tfrac{\Delta}{1-\rho^2}\sum_{t=0}^{T-1} v_t, \quad (2.46)$$

where

$$A_t = Y_{t+1} - Y_t - \left(r - \frac{1}{2}v_t + \phi_J^Q(-i)\right)\Delta - J_{Y,t+1}, \quad (2.47)$$

$$B_t = v_{t+1} - v_t - \kappa^P(\theta^P - v_t)\Delta. \quad (2.48)$$

Here, $p(\eta_s) = N(a, b)$ is the prior of $\eta_s$, with $a$ being the mean and $b$ the variance of the normal distribution. In the estimation, I set $a = 0$ and $b = 5$.

$$\pi(\gamma^Q) = p(\gamma^Q|\mathbf{vix}, \mathbf{Y}, \mathbf{v}, \mathbf{J}, \Theta - \{\gamma^Q\}) \propto p(\mathbf{vix}|\mathbf{v}, \Theta)p(\gamma^Q) \propto \quad (2.49)$$

$$\prod_{t=0}^{T-1}\exp\left\{-\frac{\left[\log\left(\frac{\mathrm{vix}_{\mathbf{t+1}}^{\mathrm{Market}}}{\mathrm{vix}_{\mathbf{t+1}}^{\mathrm{Model}}}\right) - \rho_\varepsilon \log\left(\frac{\mathrm{vix}_{\mathbf{t}}^{\mathrm{Market}}}{\mathrm{vix}_{\mathbf{t}}^{\mathrm{Model}}}\right)\right]^2}{2\sigma_{\tilde{\varepsilon}}^2}\right\}\exp\left(-\frac{(\gamma^Q - a)^2}{2b}\right) \quad (2.50)$$

$p(\gamma^Q) = N(a, b)$ is the prior of $\gamma^Q$, and in the estimation, I set $a = 0$ and $b = 1$. Considering reasonable value intervals of $\eta_s$ and $\gamma^Q$, both $N(0, 5)$ for $\eta_s$ and $N(0, 1)$ for $\gamma^Q$ are flat priors. Therefore, when they are used in the estimation, the posteriors of $\eta_s$ and $\gamma^Q$ are dominated by the likelihoods, instead of the priors.

Moreover,

$$\pi(\sigma) = p(\sigma|\mathbf{vix}, \mathbf{Y}, \mathbf{v}, \mathbf{J}, \Theta - \{\sigma\}) \propto p(\mathbf{vix}|\mathbf{v}, \Theta)p(\mathbf{J}|\Theta)p(\sigma), \quad (2.51)$$

where

$$p(\mathbf{vix}|\mathbf{v},\Theta) \propto \prod_{t=0}^{T-1} \exp\left\{ -\frac{\left[\log\left(\frac{\text{vix}_{\mathbf{t+1}}^{\text{Market}}}{\text{vix}_{\mathbf{t+1}}^{\text{Model}}}\right) - \rho_\varepsilon \log\left(\frac{\text{vix}_{\mathbf{t}}^{\text{Market}}}{\text{vix}_{\mathbf{t}}^{\text{Model}}}\right)\right]^2}{2\sigma_\varepsilon^2} \right\}, \quad (2.52)$$

$$p(\mathbf{J}|\Theta) \propto \prod_{t=1}^{T} \frac{1}{\sigma\sqrt{G_{Y,t}}} \exp\left(-\frac{(J_{Y,t}-\gamma G_{Y,t})^2}{2\sigma^2 G_{Y,t}}\right), \quad (2.53)$$

$p(\sigma)$ is the prior of $\sigma$, and in the estimation, an uninformative prior is used: $p(\sigma) = \frac{1}{\sigma}$.

In addition, some state variables and parameters, such as $v$, $\theta^P$, and $\sigma$, should be positive, either because of economic interpretation or model assumption. Therefore, in the estimation, it would be reasonable to generate only positive samples for these variables and parameters to avoid numerical problems and unrealistic estimation results. This issue is addressed as follows: if the state variable or parameter is to be updated from a normal distribution with the Gibbs sampling or DWW method, then an auxiliary uniform random number $u$ is generated first, and the new sample (in the Gibbs sampling) or candidate (in the DWW method) is computed by $\pi^{-1}((1-\pi(0))u + \pi(0))$, as in Li et al. (2008). For other parameters or state variables that are updated with the Gibbs sampling or DWW method, the forms of their posteriors or proposals, including Gamma distribution, inverse Gamma distribution, and Beta distribution, already ensure the positiveness of new samples or candidates. For parameters and variables that are updated with the MH method, such as state variables $\mathbf{G}$ and $\mathbf{v}$, new candidates are repeatedly drawn until they are positive, before being forwarded to the accept-reject step.

To study the goodness of fit, I calculate the Deviance Information Criterion (DIC) for each model. The DIC, developed by Spiegelhalter et al. (2002), employs log-likelihoods to measure goodness of fit, and the model with a lower DIC is preferred to that with a higher DIC.

Specifically, suppose $X$ is a random variable and $Y$ is the data. Then DIC is computed by

$$DIC = \overline{D(X)} + p_D, \quad (2.54)$$

where

$$D(X) = -2\log(p(Y|X)) + 2\log(f(Y)), \quad p_D = \overline{D(X)} - D(\overline{X}). \quad (2.55)$$

$f(Y)$ is some fully specified standardising term that is a function of the data alone.[2] $\overline{X}$ is the Bayesian estimator of $X$, and I set it to be the posterior mean $E(X|Y)$, which is approximated by $\frac{1}{M-M_0}\sum_{i=M_0+1}^{M} X^{(i)}$, where $M$ is the length of the complete Markov chain of $X$, $M_0$ is the length of the burn-in period, and $X^{(i)}$ represents the i$^{th}$ sample in the chain.

The DIC is made of two components. The first term, $\overline{D(X)}$, measures the goodness of fit, since the better the goodness of fit, the larger the log-likelihood $\log(p(Y|X))$. The second term, $p_D$, measures the complexity of the models; consequently, it penalises complex models with too many parameters.

In the estimation, the DIC is calculated as

$$DIC = 2\overline{D(X)} - D(\overline{X}) = -4E_{X|Y}\left(\log(p(Y|X))\right) + 2\log(p(Y|\overline{X})). \quad (2.56)$$

[2] In the estimation, I follow Berg et al. (2004) and assume $f(Y) = 1$.

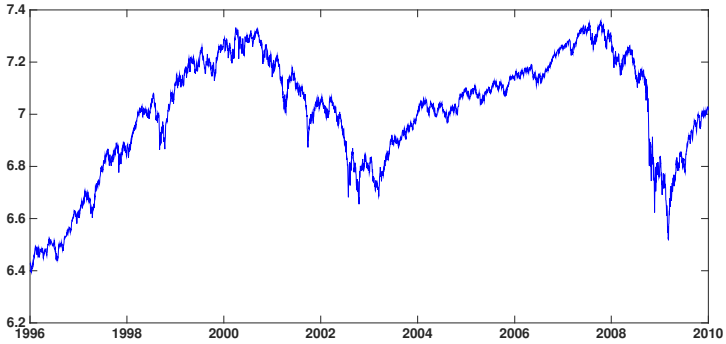


**Figure 2.1:** Daily returns of the S&P 500 index in 1996-2009.

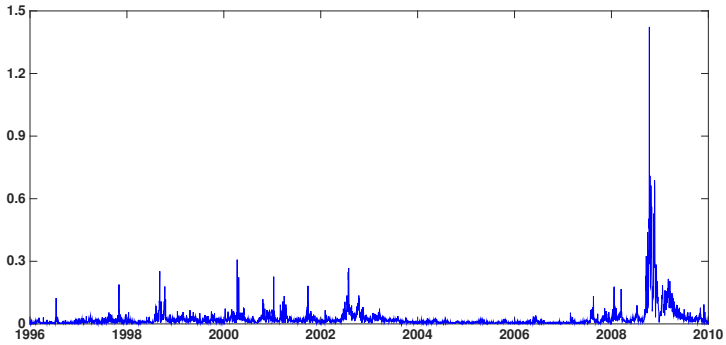


**Figure 2.2:** Annualised realised variance in 1996-2009.

## 2.5 Data Description and Option Pricing Settings

The models are estimated using the joint information of the S&P 500 index returns and daily 30-day VIX data from January 1996 to December 2009, covering a total of 14 years and 3,507 trading days. The daily returns of the S&P 500 index, annualised realised variance of the index, and 30-day VIX index are plotted in Figures 2.1 – 2.3.

The daily VIX index is reformulated from the volatility surface provided by OptionMetrics. This approach differs from the CBOE's methodology, and as pointed out in Kaeck and Alexander (2012), it can reduce the systemic biases in the VIX index provided by the CBOE. In the estimation, I set 100,000 as the number of MCMC iterations for each model, take the first 60,000 runs as the burn-in period, estimate the state variables and parameters as the mean of posteriors, and compute the standard deviation as the standard error.

To test the option pricing performance, I apply the estimated models to price options in the long-term sample of 1996 – 2009 and short-term sample of 2010, obtained from OptionMetrics. The data files comprise the expiration date, call or put identifier, strike price, best bid, best offer, implied volatility, interest rates, and future dividend estimates

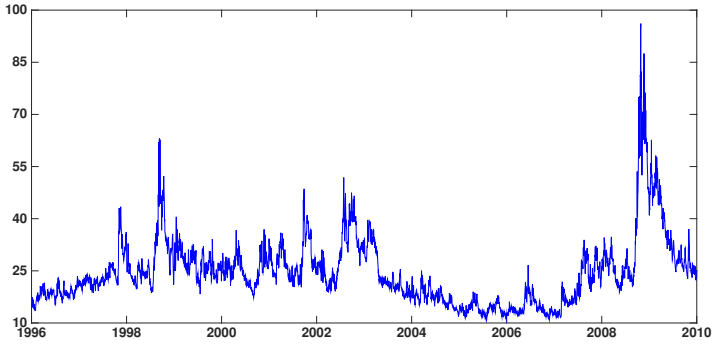


**Figure 2.3:** 30-day VIX index in 1996-2009.

based on put-call parity. As in Heston and Nandi (2000), Christoffersen and Jacobs (2004), and Christoffersen et al. (2010b), I first use the Wednesday call and put options in 1996 – 2009 (Sample A) to best avoid weekend effects and then enhance the sample by including the option data on Thursday in 1996 – 2009 (Sample B). To test the option pricing performance in the short-term sample, I use both the Wednesday and Thursday option data (Sample C). Moreover, as suggested by Bakshi et al. (1997), I filter out options with prices below 3/8 dollar. Then I remove options with maturities shorter than one week or longer than one year to reduce the liquidity bias and focus on the short-run option pricing performance. After filtering, there are 68509, 68322, and 19327 options left in Samples A, B, and C, respectively (details of the option samples are shown in Table 2.2).

When $\beta = 0.5$, the models with the SQR variance process are affine; consequently, the efficient FFT can be used to price options. However, since the other models with linear variance diffusion are non-affine, and the FFT is not applicable, for fairness, I use the Monte Carlo method with the technique of antithetic variates to price options. The number of Monte Carlo paths is $50,000$, and the models are discretised at daily frequency.

Another issue to address is how to predict spot variance at each sample date, since the spot variance is necessary as the starting value for the variance process in Monte Carlo simulations. Conventionally, the variance is computed from the past return series; however, as pointed out in Kanniainen et al. (2014), this approach reflects only the physical dynamics, whereas for pricing options, I need information under the risk-neutral measure $Q$. Furthermore, past return data provide no information about future stock movements, whereas the VIX index is computed from implied volatilities and, thus, naturally indicates the investor's expectation of the future stock market. Kaeck and Alexander (2012) use the particle filter to generate samples of $v_t$ at each out-of-sample date from $p(v_t|vix_t, Y_t)$ using the fixed parameters estimated from the in-sample data; however, this approach uses the information at time $t$ that, in practice, is not known when option prices are predicted at time $t-1$. In this thesis, I follow the same practice as in Kanniainen et al. (2014): the spot variance at every sample date is extracted from the VIX index value on the previous date according to Equations 2.26 – 2.29. Since the squared VIX index can be regarded as a linear function of variance, this approach is extremely fast. According to the results in Kanniainen et al. (2014), this variance extraction method can improve the

**Table 2.2:** Properties of Option Samples A, B, and C. Sample A: Wednesday call and put options in 1996−2009. Sample B: Thursday call and put options in 1996−2009. Sample C: Wednesday and Thursday call and put options in 2010. The options with prices below 3/8 dollar or maturities shorter than one week or longer than one year are filtered out. This table shows the number of contracts, the average price (in parentheses), and the average bid-ask spread {in braces} across the moneyness and maturity of the option data, reported by dividing the data into three groups according to maturity (calendar days) and five groups according to moneyness (S/K).

Sample A: Wed, 1996-2009

| Moneyness | Maturity (days to expiration) | | | |
|---|---|---|---|---|
| S/K | 7-90 | 91-180 | 181-365 | Total |
| <0.90 | 9889 | 3330 | 3448 | 16667 |
| | (4.05) | (11.11) | (18.79) | (8.51) |
| | {0.83} | {1.26} | {1.61} | {1.08} |
| 0.90-0.97 | 16425 | 2993 | 2384 | 21802 |
| | (8.23) | (25.67) | (44.02) | (14.54) |
| | {0.98} | {1.76} | {2.15} | {1.22} |
| 0.97-1.03 | 20054 | 2878 | 2504 | 25436 |
| | (24.32) | (50.21) | (73.14) | (32.05) |
| | {1.57} | {2.08} | {2.31} | {1.70} |
| 1.03-1.10 | 2594 | 510 | 506 | 3610 |
| | (68.98) | (91.84) | (111.61) | (78.18) |
| | {2.35} | {2.38} | {2.51} | {2.38} |
| >1.10 | 638 | 161 | 195 | 994 |
| | (236.47) | (242.98) | (216.34) | (233.58) |
| | {2.92} | {2.61} | {2.76} | {2.84} |
| Total | 49600 | 9872 | 9037 | 68509 |
| | (20.01) | (34.87) | (49.97) | (26.11) |
| | {1.29} | {1.73} | {2.03} | {1.45} |

Sample B: Thu, 1996-2009

| Moneyness | Maturity (days to expiration) | | | |
|---|---|---|---|---|
| S/K | 7-90 | 91-180 | 181-365 | Total |
| <0.90 | 9299 | 3310 | 3320 | 15929 |
| | (4.27) | (11.29) | (19.59) | (8.93) |
| | {0.86} | {1.33} | {1.70} | {1.13} |
| 0.90-0.97 | 16403 | 3034 | 2320 | 21757 |
| | (8.30) | (26.43) | (43.95) | (14.63) |
| | {0.99} | {1.80} | {2.20} | {1.23} |
| 0.97-1.03 | 20426 | 3081 | 2346 | 25853 |
| | (24.03) | (50.09) | (72.11) | (31.50) |
| | {1.58} | {2.09} | {2.34} | {1.71} |
| 1.03-1.10 | 2705 | 536 | 517 | 3758 |
| | (68.16) | (93.63) | (110.62) | (77.63) |
| | {2.41} | {2.65} | {2.56} | {2.47} |
| >1.10 | 656 | 191 | 178 | 1025 |
| | (222.45) | (231.63) | (205.08) | (221.15) |
| | {3.17} | {2.88} | {2.94} | {3.08} |
| Total | 49489 | 10152 | 8681 | 68322 |
| | (20.15) | (36.08) | (49.52) | (26.25) |
| | {1.32} | {1.80} | {2.08} | {1.49} |

Sample C: Wed & Thu, 2010

| Moneyness | Maturity (days to expiration) | | | |
|---|---|---|---|---|
| S/K | 7-90 | 91-180 | 181-365 | Total |
| <0.90 | 4009 | 1423 | 1043 | 6475 |
| | (3.42) | (10.29) | (22.86) | (8.06) |
| | {0.90} | {1.59} | {2.46} | {1.30} |
| 0.90-0.97 | 4681 | 869 | 569 | 6119 |
| | (7.74) | (28.29) | (52.98) | (14.86) |
| | {1.24} | {2.55} | {3.44} | {1.63} |
| 0.97-1.03 | 4252 | 847 | 625 | 5724 |
| | (25.16) | (54.54) | (85.90) | (36.14) |
| | {2.20} | {3.09} | {3.82} | {2.51} |
| 1.03-1.10 | 589 | 116 | 105 | 810 |
| | (69.42) | (98.82) | (122.91) | (80.57) |
| | {3.47} | {3.53} | {4.87} | {3.66} |
| >1.10 | 120 | 37 | 42 | 199 |
| | (221.48) | (223.20) | (231.20) | (223.85) |
| | {6.77} | {3.90} | {6.16} | {6.11} |
| Total | 13651 | 3292 | 2384 | 19327 |
| | (16.44) | (31.94) | (54.65) | (23.79) |
| | {1.58} | {2.32} | {3.22} | {1.91} |

option pricing performance of GARCH models beyond that of traditional return-based variance updating methods.

As in Christoffersen et al. (2013), Kaeck and Alexander (2012), and Kanniainen et al. (2014), I compute the root mean square errors (RMSE) from the implied volatility RMSE from the sum of the squared difference between the theoretical and market option prices scaled by the inverse Black-Scholes vegas of the options as follows:

$$\mathcal{V}\text{RMSE}(\Theta) = 100 \times \sqrt{\frac{1}{N_M} \sum_{t,i} \left( \frac{c_{i,t}(\Theta, v_t; T_i, K_i) - \hat{c}_{i,t}(T_i, K_i)}{\hat{\mathcal{V}}_t} \right)^2}. \tag{2.57}$$

Here, $K_i$ is the strike of the $i$th option at time $t$, $T_i$ is the maturity, $c_{i,t}$ is the model price of the option, and $\hat{c}_{t,i}$ is the market price. $\hat{\mathcal{V}}_{t,i}$ is the inverse Black-Scholes vega computed on the true market prices of options. Moreover, $N_M = \sum_{t=1}^{M} N_t$, where $N_t$ is the number of option prices in the sample at time $t$, and $M$ is the total number of days in the sample. The use of the implied volatility RMSE is motivated by the findings of Broadie et al. (2007) in that the arithmetic RMSE places more weight on expensive in-the-money and long-maturity options and, conversely, the implied volatility RMSE can add reasonable weights to options in different moneyness and maturity groups.

## 2.6 Estimation Results

### 2.6.1 Parameter Estimates

Tables 2.3 – 2.5 report the parameter estimation results obtained using the joint information of the S&P 500 index returns and the 30-day VIX from January 1996 to December 2009.

The values of $\kappa^P$ vary significantly across different models. In the model group SV-P-P-$i$, the largest $\kappa^P$, estimated at 6.1991 (0.5844 as the standard deviation), appears in the model SV-P-P-1. Compared to the results in previous papers estimating the SV-P-P-0 model, my estimation of $\kappa^P$ is somewhat higher than that in most papers where it ranges from 2 to 5. For instance, in Yu et al. (2011), $\kappa^P$ is 3.3627 (0.6452), estimated with the MCMC from the S&P 500 index and daily prices of a short-term ATM SPX option from January 1993 to December 1993. In Duan and Yeh (2010), $\kappa^P$ is 1.9449 (0.6987), estimated with the MLE from the S&P 500 index and the CBOE's 30-day VIX index from January 1990 to August 2007. Moreover, in Kaeck and Alexander (2012), $\kappa^P$ is 3.395 (0.328) with the MCMC, estimated from the data of the S&P 500 index and 30-day and 360-day VIX indices from January 1990 to December 2004, whereas my sample period is from January 1996 to December 2009. One possible explanation relates to the sample data and estimation method that I use. For instance, Yu et al. (2011) use the index returns of 1993, when the market was relatively quiet, whereas I adopt longer sample period in which there were several big financial crises; this difference may lead to the higher $\kappa^P$ for a more volatile variance path. Another explanation could be the different estimates of the jump intensity $\lambda$ in different papers. In Duan and Yeh (2010), $\lambda$ is 43.9476 (6.4716), suggesting a quite frequent occurrence of large jumps in a year. Moreover, in Kaeck and Alexander (2012), $\lambda$ is 5.894 (1.273), almost three times my estimate, and frequent large jumps may obviate a strong mean-reversion to generate large movements.

On the other hand, the inclusion of VG or NIG jumps in returns and variance in the stochastic volatility model can lead to quite different estimates of $\kappa^P$. In my estimation,

**Table 2.3:** Parameter estimates and DICs of model parameters of the Heston, Bates, and SV-P-P with $\beta = 0.5, 1$. The parameters are estimated using the daily spot returns of the S&P 500 index and the reformulated 30-day VIX from January 1996 to December 2009. The parameter values are the mean of the posteriors as annual decimals. The standard errors are the standard deviations of the posteriors, reported in parentheses.

| Models | Heston-0 | Heston-1 | Bates-0 | Bates-1 | SV-P-P-0 | SV-P-P-1 |
|---|---|---|---|---|---|---|
| $\beta$ | 0.5 | 1 | 0.5 | 1 | 0.5 | 1 |
| $\theta^P$ | 0.0403 | 0.0347 | 0.0326 | 0.0212 | 0.0375 | 0.0289 |
| | (0.0057) | (0.0048) | (0.0029) | (0.0023) | (0.0054) | (0.0064) |
| $\kappa^P$ | 2.9688 | 3.4357 | 4.7392 | 4.0800 | 5.8687 | 6.1991 |
| | (0.4056) | (0.1671) | (0.2450) | (0.1891) | (0.4267) | (0.5844) |
| $\kappa^Q$ | 1.8501 | 2.0683 | 4.1184 | 3.7637 | 4.9907 | 5.7795 |
| | (0.9416) | (1.1267) | (1.7880) | (1.6250) | (1.4718) | (1.8249) |
| $\alpha$ | 0.3181 | 1.8546 | 0.6047 | 1.8230 | 0.4152 | 1.3972 |
| | (0.0060) | (0.0369) | (0.0888) | (0.1226) | (0.7627) | (0.7241) |
| $\rho$ | -0.7435 | -0.7993 | -0.8493 | -0.8113 | -0.8078 | -0.8471 |
| | (0.0056) | (0.0103) | (0.0702) | (0.0097) | (0.0603) | (0.0133) |
| $\eta_s$ | -2.1615 | -2.0133 | -1.6083 | -0.5095 | -1.2440 | -1.3763 |
| | (0.5494) | (0.6097) | (0.6784) | (0.5444) | (1.2248) | (0.9610) |
| $\mu_y$ | | | -0.0057 | 0.0002 | -0.0002 | -0.0022 |
| | | | (0.0018) | (0.0003) | (0.0005) | (0.0010) |
| $\mu_y^Q$ | | | 0.0212 | -0.0023 | -0.0172 | -0.0152 |
| | | | (0.0250) | (0.0163) | (0.0098) | (0.0084) |
| $\lambda$ | | | 31.5728 | 16.6611 | 2.0916 | 2.6678 |
| | | | (3.2097) | (2.2038) | (1.3919) | (1.4048) |
| $\sigma_y$ | | | 0.0074 | 0.0213 | 0.0469 | 0.0348 |
| | | | (0.0009) | (0.0012) | (0.0084) | (0.0052) |
| $\rho_J$ | | | | | -0.1975 | -0.3185 |
| | | | | | (0.0529) | (0.0876) |
| $\mu_v$ | | | | | 0.0556 | 0.0609 |
| | | | | | (0.0156) | (0.0104) |
| DIC | -39149 | -39258 | -39383 | -43617 | -41992 | -43874 |

$\kappa^P$ in SV-VG-VG-$i$ is the strongest (6.2600 in the SV-VG-VG-0), while in the SV-VG-NIG-$i$, SV-NIG-VG-$i$, SV-NIG-$i$, and SV-NIG-NIG-$i$, $\kappa^P$ is much less significant (the lowest $\kappa^P$ is 1.0120 in the SV-NIG-0). This is an interesting observation because, intuitively, the inclusion of jumps in variance reduces the need for a strong $\kappa^P$ to create large changes. This result may suggest that the VG may not be a proper choice for the variance jump, as supported by the insignificant change in the DIC caused by the inclusion of VG variance jumps, compared to that of the SV-VG-$i$ without variance jumps. Furthermore, the figures of extracted variance jumps in the SV-VG-VG-$i$ confirm that the VG does not very actively generate frequent small jumps in variance, and that the magnitude of large variance jumps generated by the VG is lower than that by the NIG, especially during periods of market turbulence. Moreover, I examine how long-term physical and risk-neutral variance levels are identified in different model groups. In fact, according to the VIX index and realised variance in 1996 – 2009, the long-term physical and risk-neutral annual volatilities are about 16% and 20%, respectively. Overall, the SV-NIG-$i$, $i = 0, 1$, and the SV-VG-NIG-$i$ seem more consistent with the data to project realised and expected

**Table 2.4:** Parameter estimates and DICs of model parameters of the SV-VG, SV-VG-VG, and SV-VG-NIG with $\beta = 0.5, 1$. The parameters are estimated using the daily spot returns of the S&P 500 index and the reformulated 30-day VIX from January 1996 to December 2009. The parameter values are the mean of the posteriors as annual decimals. The standard errors are the standard deviations of the posteriors, reported in parentheses.

| Model | SV-VG-0 | SV-VG-1 | SV-VG-VG-0 | SV-VG-VG-1 | SV-VG-NIG-0 | SV-VG-NIG-1 |
|---|---|---|---|---|---|---|
| $\beta$ | 0.5 | 1 | 0.5 | 1 | 0.5 | 1 |
| $\theta^P$ | 0.0486 | 0.0450 | 0.0175 | 0.0232 | 0.0296 | 0.0301 |
| | (0.0014) | (0.0008) | (0.0025) | (0.0052) | (0.0037) | (0.0022) |
| $\kappa^P$ | 5.3469 | 2.6819 | 6.2600 | 4.9766 | 2.3209 | 1.7369 |
| | (0.6851) | (0.2757) | (0.6878) | (0.4331) | (0.2465) | (0.1607) |
| $\kappa^Q$ | 5.3896 | 1.1632 | 5.2927 | 3.6522 | 1.2462 | 1.7498 |
| | (0.6562) | (0.2221) | (0.5958) | (0.2917) | (0.1986) | (0.2512) |
| $\alpha$ | 0.6493 | 2.6504 | 0.6395 | 2.7354 | 0.4489 | 2.6262 |
| | (0.0280) | (0.0538) | (0.0292) | (0.1276) | (0.0309) | (0.1649) |
| $\rho$ | -0.9435 | -0.9225 | -0.9334 | -0.9288 | -0.9434 | -0.9261 |
| | (0.0062) | (0.0085) | (0.0078) | (0.0093) | (0.0060) | (0.0089) |
| $\eta_s$ | -2.2801 | -1.1632 | -2.2931 | -2.3841 | 2.3746 | 1.2730 |
| | (1.3211) | (1.1854) | (2.1134) | (1.0236) | (0.6148) | (0.6001) |
| $\gamma^Q$ | -0.0182 | -0.0003 | -0.0900 | 0.0036 | -0.1820 | -0.1839 |
| | (0.1447) | (0.1939) | (0.1427) | (0.0135) | (0.1402) | (0.1419) |
| $\gamma$ | -0.0307 | 0.0344 | 0.2264 | 0.3263 | 0.1604 | 0.2248 |
| | (0.1394) | (0.1863) | (0.1303) | (0.1458) | (0.0548) | (0.1260) |
| $\nu$ | $< 0.0001$ | $< 0.0001$ | 0.0229 | 0.0228 | 0.0229 | 0.0230 |
| | $(< 0.0001)$ | $(< 0.0001)$ | (0.0004) | (0.0004) | (0.0005) | (0.0005) |
| $\sigma$ | 0.0606 | 0.0566 | 0.1303 | 0.1217 | 0.1043 | 0.1091 |
| | (0.0020) | (0.0022) | (0.0120) | (0.0101) | (0.0095) | (0.0098) |
| $\sigma^Q$ | 0.1189 | 0.0910 | 0.1446 | 0.1011 | 0.0755 | 0.0715 |
| | (0.0067) | (0.0050) | (0.0159) | (0.0233) | (0.0415) | (0.0400) |
| $\rho_v$ | | | -0.1907 | -0.1242 | -0.0482 | -0.0401 |
| | | | (0.0952) | (0.0896) | (0.1022) | (0.1039) |
| $\gamma_v^Q$ | | | -0.0007 | -0.0031 | 0.0005 | -0.0008 |
| | | | (0.0097) | (0.0099) | (0.0031) | (0.0034) |
| $\gamma_v$ | | | 0.0396 | 0.1000 | 0.0063 | 0.0363 |
| | | | (0.0135) | (0.0983) | (0.0074) | (0.0681) |
| $\nu_v^Q$ | | | 0.0287 | 0.0287 | 0.0939 | 0.0936 |
| | | | (0.0005) | (0.0005) | (0.0179) | (0.0184) |
| $\sigma_v$ | | | 0.1953 | 0.2052 | 0.0312 | 0.0680 |
| | | | (0.0057) | (0.0227) | (0.0163) | (0.0025) |
| DIC | -44792 | -48144 | -43098 | -48750 | -47202 | -53696 |

variance levels, the SV-P-P-$i$ and the SV-VG-$i$ imply higher variance levels, and the SV-VG-VG-$i$ imply slightly lower levels.

Consistent with the previous literature, the estimates of the physical mean-reverting parameters $\kappa^P$ and $\kappa^Q$ imply a negative variance risk premium $\eta_v = \kappa^Q - \kappa^P$ in most cases. In the SV-P-P-0, the variance risk premium is as large as -0.8780, mainly because of the large $\kappa^P$. In the Heston-$i$ and SV-VG-VG-$i$, the variance risk premia are even more significant: -1.1187 and -1.3674 for $i = 0, 1$ in the Heston-$i$ and -0.9673 and -1.3244 for $i = 0, 1$ in the SV-VG-VG-$i$, respectively. Exceptions of a positive $\eta_v$ arise in models SV-VG-0, SV-VG-NIG-1, SV-NIG-0, and SV-NIG-VG/NIG-$i$, with 2.9173 being the largest in the SV-NIG-VG-1 and 0.0129 in the SV-VG-NIG-1. However, most of them, except for those in the SV-NIG-VG-1 and the SV-NIG-0, are insignificant compared to the negative risk premium values in the other models. A similar result of a small positive $\eta_v$ also appears in the results of Kaeck and Alexander (2012), who argue that the estimation of the variance process is plausible because the VIX-implied long-term variance level in their sample period is successfully estimated, which is also done in my estimation. They

**Table 2.5:** Parameter estimates and DICs of model parameters of the SV-NIG, SV-NIG-VG, and SV-NIG-NIG with $\beta = 0.5, 1$. The parameters are estimated using the daily spot returns of the S&P 500 index and the reformulated 30-day VIX from January 1996 to December 2009. The parameter values are the mean of the posteriors as annual decimals. The standard errors are the standard deviations of the posteriors, reported in parentheses.

| Model | SV-NIG-0 | SV-NIG-1 | SV-NIG-VG-0 | SV-NIG-VG-1 | SV-NIG-NIG-0 | SV-NIG-NIG-1 |
|---|---|---|---|---|---|---|
| $\beta$ | 0.5 | 1 | 0.5 | 1 | 0.5 | 1 |
| $\theta^P$ | 0.0115 | 0.0266 | 0.0167 | 0.0451 | 0.0128 | 0.0452 |
| | (0.0059) | (0.0041) | (0.0011) | (0.0048) | (0.0027) | (0.0100) |
| $\kappa^P$ | 1.0120 | 2.3512 | 3.0015 | 2.2694 | 1.1531 | 1.3045 |
| | (0.4810) | (0.4118) | (1.0950) | (0.2257) | (0.1570) | (0.3451) |
| $\kappa^Q$ | 2.6163 | 0.9687 | 3.7093 | 5.1867 | 1.6379 | 1.7273 |
| | (1.5738) | (0.6880) | (1.7368) | (1.8835) | (1.6176) | (1.3073) |
| $\alpha$ | 0.3376 | 2.5347 | 0.4891 | 2.5039 | 0.3154 | 2.2123 |
| | (0.0101) | (0.0649) | (0.0122) | (0.1305) | (0.0224) | (0.1127) |
| $\rho$ | -0.9495 | -0.9326 | -0.9479 | -0.9313 | -0.9489 | -0.9367 |
| | (0.0011) | (0.0088) | (0.0026) | (0.0087) | (0.0018) | (0.0078) |
| $\eta_s$ | 0.1511 | -0.0699 | -1.3819 | -0.5883 | -1.2290 | -1.3166 |
| | (1.0707) | (1.0341) | (0.6887) | (0.6835) | (0.7557) | (0.6827) |
| $\gamma^Q$ | -0.2031 | -0.1947 | -0.1854 | -0.1768 | -0.3740 | -0.1762 |
| | (0.1425) | (0.1412) | (0.1373) | (0.1416) | (0.2299) | (0.1361) |
| $\gamma$ | 0.1954 | 0.0722 | 0.6571 | 0.5233 | 0.4280 | 0.2207 |
| | (0.5910) | (0.1293) | (0.9566) | (0.9952) | (1.0015) | (0.9343) |
| $\nu^Q$ | 6.7768 | 3.3669 | 11.5623 | 12.3703 | 46.4268 | 12.7099 |
| | (0.7422) | (0.5523) | (0.9656) | (1.1066) | (2.5926) | (1.2125) |
| $\sigma$ | 0.2154 | 0.1506 | 0.4582 | 0.3091 | 0.2584 | 0.1944 |
| | (0.0151) | (0.0245) | (0.0326) | (0.0645) | (0.1546) | (0.0847) |
| $\rho_v$ | | | -0.0327 | -0.1554 | -0.1352 | -0.1110 |
| | | | (0.0352) | (0.1186) | (0.0683) | (0.1353) |
| $\gamma_v^Q$ | | | -0.0027 | -0.0008 | 0.0427 | -0.0029 |
| | | | (0.0976) | (0.0958) | (0.2941) | (0.0922) |
| $\gamma_v$ | | | 0.3753 | 0.0559 | 0.0380 | 0.0619 |
| | | | (0.0521) | (0.0973) | (0.0432) | (0.0903) |
| $\nu_v^Q$ | | | 0.0287 | 0.0287 | 0.2250 | 0.2196 |
| | | | (0.0005) | (0.0005) | (0.1221) | (0.0997) |
| $\sigma_v$ | | | 0.0300 | 0.1002 | 0.0566 | 0.1223 |
| | | | (0.0009) | (0.0200) | (0.0036) | (0.0129) |
| DIC | -50861 | -51700 | -49835 | -50314 | -45757 | -49526 |

also postulate that the insignificant $\eta_v$ may result from a relatively flat variance term structure, which increases the difficulty of correctly estimating the variance risk premium $\eta_v$, as pointed out in Broadie et al. (2007). The estimation of the variance risk premium in returns, that is, $\eta_s$, in all models except for the SV-VG-NIG-$i$ and the SV-NIG-0, supports the economic interpretation of a negative $\eta_s$. However, in most models, the standard errors of $\eta_s$ are large, suggesting that $\eta_s$ is quite difficult to estimate.

The leverage effect is captured by all the models, because the correlation factor $\rho$ is estimated to be negative. However, compared to $\rho$ in the Bates-$i$ and SV-P-P-$i$, in other models with VG or NIG return jumps, $\rho$ is much closer to -1, ranging from $-0.9225$ in the SV-VG-1 to $-0.9495$ in the SV-NIG-0 and suggesting a stronger negative interaction between changes in spot returns and variance. This can be explained by the different jump structures: in models without jumps in variance, the correlation between changes in spot variance and returns can only be captured by the correlation between diffusion terms; in models with VG/NIG jumps in variance, the correlation between $J_y$ and $J_v$ is relatively weak, whereas in the SV-P-P-$i$, $i = 0, 1$, the variance jumps directly affect the jump in returns by $J_Y \sim N(\mu_y + \rho_J J_v, \sigma_y^2)$.

In models with VG/NIG variance jumps, $\rho_v$ remains negative, from -0.2 to -0.03, suggesting a negative correlation of jumps in returns and variance, but the correlation between the jumps is not as strong as that between diffusions. In models with different jump types in returns and variance, $\rho_v$ is negligible, which is plausible because different types of Lévy jumps with distinctive features and behaviour are used in returns and variance.

Let us discuss now how these models seek to account for the long-memory effect. Here, two quantities specific to the models are considered, namely, conditional variance $E_t^P[v_{t+d}]$ and conditional variance of variance $var_t^P[v_{t+d}]$, to illustrate how under the physical measure $P$, the current variance affects the future variance after $d$ years in different models. $E_t[v_{t+d}]$ and $var_t[v_{t+d}]$ are calculated as follows:

$$E_t^P[v_{t+d}] = e^{-\kappa^P d}v_t + \frac{\kappa^P\theta^P + M_v}{\kappa^P}\left(1 - e^{-\kappa^P d}\right), \tag{2.58}$$

$$var_t^P[v_{t+d}] = \frac{\alpha^2}{\kappa^P}\left(e^{-\kappa^P d} - e^{-2\kappa^P d}\right)v_t + \frac{var^P(J_{v,t})}{\kappa^P}\left(1 - e^{-2\kappa^P d}\right) \tag{2.59}$$

$$+\frac{\alpha^2(\kappa^P\theta^P + M_v)}{2(\kappa^P)^2}(1 - e^{-\kappa^P d})^2, \tag{2.60}$$

where $var^P(J_{v,t})$ is the variance of the variance jumps. The above equations show that the conditional expectation of future variance is affected by the current level with the coefficient $e^{-\kappa^P d}$; thus, a higher $\kappa^P$ leads to a smaller correlation between $v_t$ and $v_{t+d}$. Furthermore, with other conditions being the same, a low $\kappa^P$ increases the conditional variance of variance $var_t[v_{t+d}]$.

Although the choice of $J_v$ and $\beta$ does not directly affect the long-memory effect, I notice that $\kappa^P$ is quite sensitive to $J_v$ and $\alpha$ varies with $\beta$. Generally, in the SV-VG-VG-$i$ and the SV-P-P-$i$, $\kappa^P$ is the highest, followed by that in the Bates-$i$, SV-VG-$i$, and Heston-$i$, and the lowest in the SV-VG-NIG-$i$, SV-NIG-$i$, and SV-NIG-VG/NIG-$i$, suggesting that with the same $\beta$, models with an NIG jump in returns or variance show the strongest long-memory effect, and the SV-VG-VG-$i$ and the SV-P-P-$i$ show the weakest. To illustrate, the coefficient $e^{-\kappa^P d}$ when $d = 1/12$ is 0.5935 in the SV-VG-VG-0, and after one year, it drops to 0.0019; in the SV-NIG-0, the one-month $e^{-\kappa^P d}$ is 0.9191, and the one-year $e^{-\kappa^P d}$ is still high, reaching 0.3635. Furthermore, across different model groups, a higher $\beta$ leads to a significantly higher $\alpha$, and although the value of $\alpha$ does not affect the conditional variance, it raises the conditional variance of variance as it increases.

Next, I move to the behaviour of jumps in returns and variance. Extracted jumps in returns and variance for each model are plotted in Figures 2.4 – 2.15. Since there are apparent differences between different model groups and almost the same patterns within the same model group, I include only the figures of jumps in models with $\beta = 1$ in each model group. The figures of extracted jumps in returns and variance show clear differences in jump behaviour between finite-activity and infinite-activity jumps. The SV-P-P-1 has 26 jumps with an absolute size of over 0.005; in fact, they are almost all downside jumps, which is consistent with the observations. More interestingly, when jumps with absolute size larger than 0.01 are extracted, 17 jumps are left with several waves of large jumps occurring in 1998, 2000, 2007, and 2008, covering the several financial crises in the sample period of 1996 – 2009. Furthermore, there is a clear negative correlation between the jumps in returns and in variance.

Notably, in the SV-P-P-$i$, both $\mu_y$ and $\mu_y^Q$ are estimated with large standard errors. As pointed out in Eraker (2004), the difficulty of estimating $\mu_y$ lies in that this parameter

can be identified only from the index return; however, if only a few return jumps are observed, it may be hard to estimate $\mu_y$ as the expected mean of the jumps. Moreover, as shown in Kaeck and Alexander (2012), since only the squared value $(\mu_y^Q)^2$ enters the formula of the model-implied VIX, $\mu_y^Q$ is bi-modal, and the algorithm cannot determine its sign.

To deal with the difficulty of estimating jump-related parameters, Kaeck and Alexander (2012) assume a zero jump risk premium by setting the risk-neutral jump-related parameters identical to the physical ones, because they notice that the jump premium $\phi_J^Q(-i) - \phi_J^P(-i)$ is very insignificant. In the estimation, I first use different jump-related parameters under $P$ and $Q$, and my estimation results of $\mu_y$ are similar to those in Kaeck and Alexander (2012). However, the absolute sizes of the risk-neutral parameters $\mu_y^Q$ in the SV-P-P-$i$ are much larger than the physical $\mu_y$. The difference is also found in, for example, Eraker (2004) and Yu et al. (2011), implying from the investors' point of view a strong uncertainty about future jump size.

To further probe into the problem of estimating $\mu_y$ and $\mu_y^Q$, I follow the practice of Kaeck and Alexander (2012) by setting $\mu_y^Q = \mu_y$ and updating other jump-related parameters, including $\lambda$, $\sigma_y$, $\rho_J$, and $\mu_v$, only according to the information about index returns. The results reveal that for the SV-P-P-$i$, when $\mu_v$ is close to its previous estimates, $\mu_y$ is more significant and $\sigma_y$ is more than 10 times its previous estimate. The larger expected mean and variance of return jumps can be explained by a smaller $\lambda$: about 0.25 jumps per year, estimated under the uniform prior, daily $\lambda \sim U(0, 1)$. When this new estimation result is used, both the DICs and option pricing performance deteriorate. Therefore, in this chapter, I only report the estimates obtained on the basis of the joint information of index returns and VIX.[3]

The behaviour of return jumps in the models SV-NIG-$i$ resembles closely that of models SV-VG-$i$, except that there are many more small jumps, which is expected because the NIG is of infinite variation. For models with VG return jumps, the expected jump size $\gamma^Q$ is bi-modal, because only $(\gamma^Q)^2$ enters the formula of the VIX index. Thus, to obtain more plausible results in the estimation, I choose priors for $\gamma^Q$ by restricting its value to either positive or negative. However, $\gamma$ and $\gamma^Q$ are still estimated with relatively large standard errors. One could argue that the VIX does not contain useful information in this regard, and that option data should be used instead of the VIX. However, in Yu et al. (2011), where the model SV-VG-0 is estimated from the index and option data, these parameters continue to be estimated with quite large standard errors: 0.0256(0.0315) for $\gamma$ and 0.0030(0.0056) for $\gamma^Q$ in the SV-VG-0. When the other jump-related parameters are estimated with reasonable standard errors, the standard errors of parameters $\mu_y$, $\mu_y^Q$, $\gamma$, and $\gamma^Q$ are relatively large, and this suggests that the parameters measuring the expected mean of jumps are difficult to identify. One possible solution could be to make these parameters time varying, and they may depend on the level of variance, as assumed in Bates (2012) and Eraker (2004).

[3]The estimates of $\lambda$, $\sigma_y$, $\rho_J$, and $\mu_v$ reported in Table 2.3 are computed as the mean of the posteriors, estimated from the joint information of the VIX and index returns, and when they are estimated only from index data, as in Kaeck and Alexander (2012), their estimates are quite different. The difference in the estimates of jump-related parameters may suggest that the models are not consistent with the market data: the SV-P-P-$i$ fail to jointly capture the physical dynamics implied by the index and the risk-neutral dynamics implied by the VIX. Then I conduct a simple simulation study to verify this, simulating five-year data of the index and VIX under the Bates-1 and SV-P-P-1 models and first estimating the models with simulated index data and then with the simulated index and VIX data. I notice that in both estimation experiments, the extracted jumps in returns and variance almost coincide with the simulated jumps, and the estimation results of $\lambda$, $\sigma_y$, $\rho_J$, and $\mu_v$ are very close to the corresponding true values.

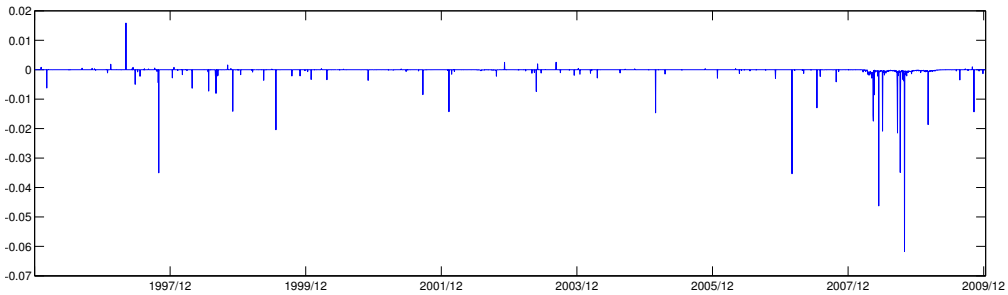


**Figure 2.4:** Extracted jumps in returns in the SV-P-P-1

Moreover, I compute the jump risk premium $\phi_J^Q(-i) - \phi_J^P(-i)$. In the SV-P-P-$i$, it is 0.03568 and 0.14728 with $i = 0, 1$. However, the jump risk premium becomes more indispensable for models with infinite-activity VG jumps. Indeed, apart from the SV-VG-0, all the other models with VG return jumps demonstrate positive jump premia: 0.0322 in the SV-VG-1, 0.3151 and 0.3263 in the SV-VG-VG-0, 1, and 0.3449 and 0.3794 in the SV-VG-NIG-0, 1. The jump risk premium is -0.0125 in the SV-VG-0, coming from the large $\gamma^Q$. The jump premium is 0.0588 in the SV-NIG-0 and 0.0775 in the SV-NIG-1, both larger than that in SV-VG-1. However, the jump premia of models with NIG return jumps and VG/NIG variance jumps are less significant than those of models with VG return jumps and VG/NIG variance jumps.

The figures of extracted jumps show that, compared to models with infinite-activity variance jumps, the models SV-VG/NIG-1 generate many more large return jumps with a jump size larger than 0.01. The jump patterns of models SV-VG-VG-1 and SV-VG-NIG-1 are similar; the slight difference between them, compared to the return jumps in the SV-VG-NIG-1, is that the return jumps in SV-VG-VG-1, upside or downside, are larger in general. In particular, the SV-VG-VG-1 has more positive jumps with a jump size larger than 0.01. However, the frequency of both large and small jumps drops significantly after the VG/NIG variance jumps are added to the SV-NIG-1, suggesting that the inclusion of variance jumps reduces the NIG's effectiveness in capturing return jumps, and that it does not improve the current model specification of the SV-NIG.

Next, let us compare the extracted variance jumps. Compared to the SV-VG-VG-1, the SV-VG-NIG-1 exhibits more frequent small jumps in the variance path, and its large jumps are more significant. In particular, to capture the sudden market changes during the 2007−2008 financial crisis, both models exhibit a very large jump; however, the size of the NIG variance jump is about two times that of the VG variance jump. In the models with NIG return jumps, the variance jumps in the SV-NIG-VG-1 are similar to those in the SV-VG-VG-1; however, the SV-NIG-NIG-1 captures only several large variance movements and does not actively generate small variance jumps. The model SV-P-P-1 creates large variance jumps, and generally its jumps and the infinite-activity variance jumps are contemporaneous, at which the market became volatile.

### 2.6.2 Goodness of Fit

The goodness-of-fit statistics DIC shown in Tables 2.3−2.5 clearly confirm the importance of non-affine models. In all the model groups, models with $\beta = 1$ significantly outperform models with $\beta = 0.5$. In terms of the impact of jumps in returns and variance, the

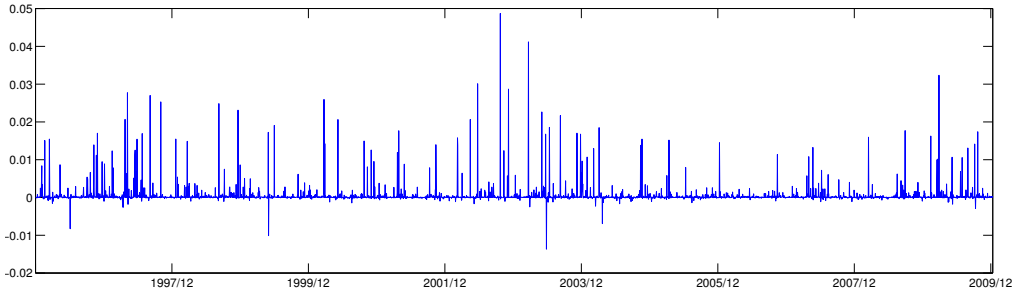


**Figure 2.5:** Extracted jumps in returns in the SV-VG-1

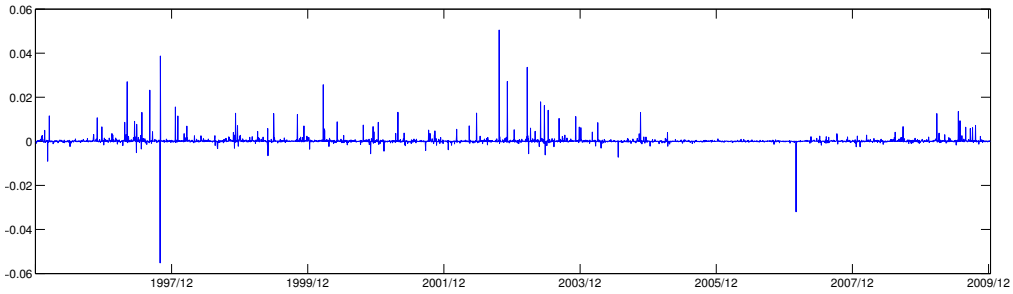


**Figure 2.6:** Extracted jumps in returns in the SV-VG-VG-1

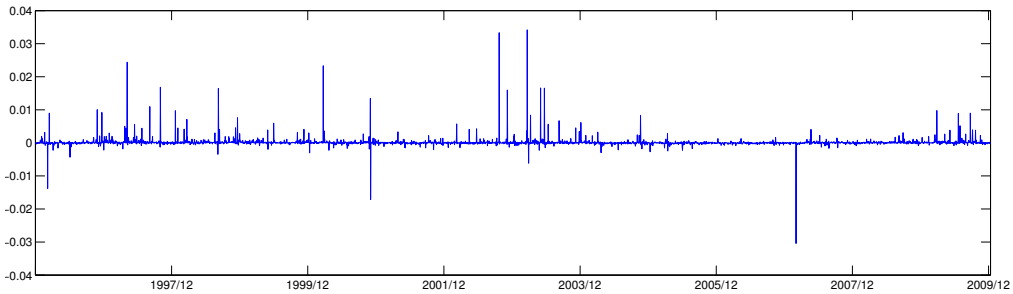


**Figure 2.7:** Extracted jumps in returns in the SV-VG-NIG-1

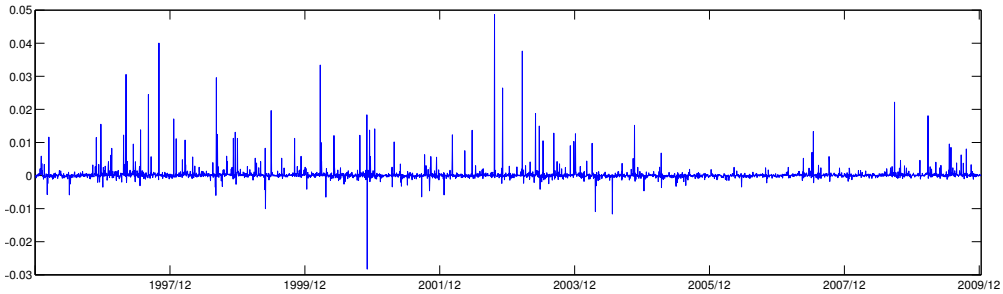


**Figure 2.8:** Extracted jumps in returns in the SV-NIG-1

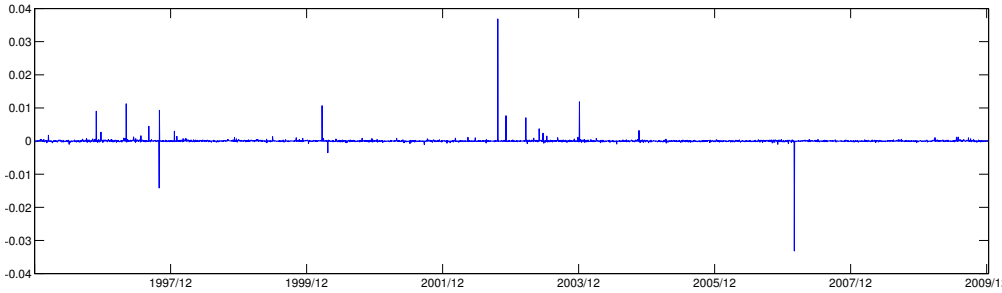


**Figure 2.9:** Extracted jumps in returns in the SV-NIG-VG-1

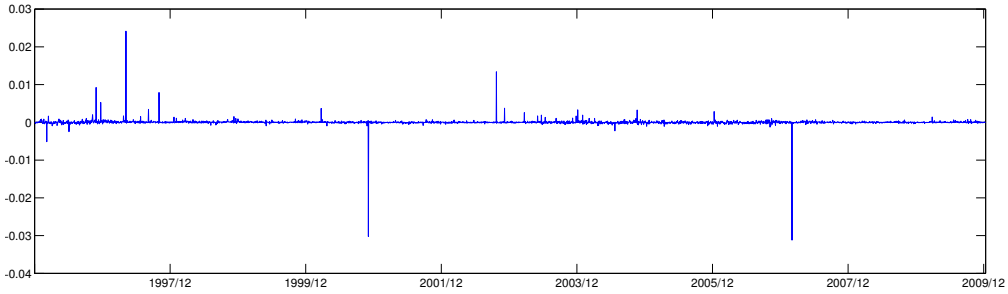


**Figure 2.10:** Extracted jumps in returns in the SV-NIG-NIG-1

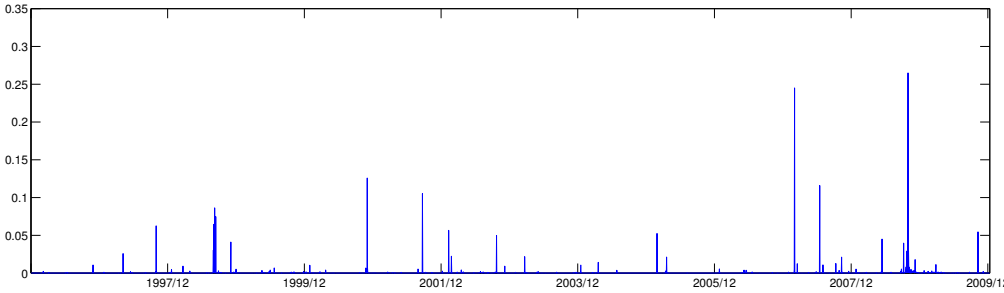


**Figure 2.11:** Extracted jumps in variance in the SV-P-P-1

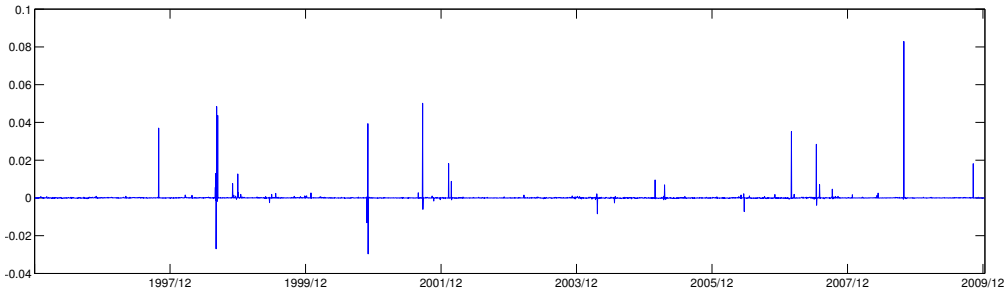


**Figure 2.12:** Extracted jumps in variance in the SV-VG-VG-1

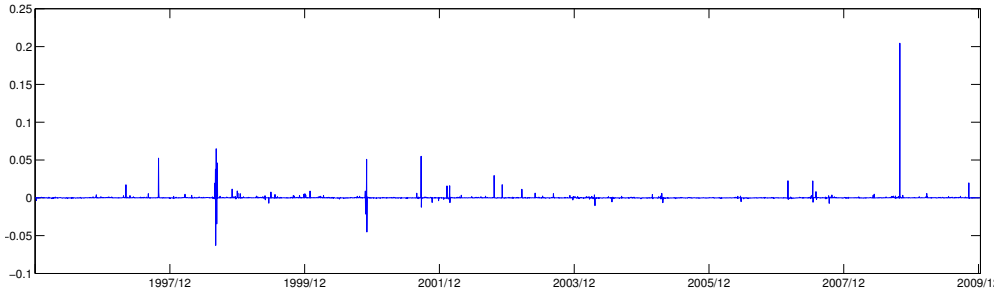


**Figure 2.13:** Extracted jumps in variance in the SV-VG-NIG-1

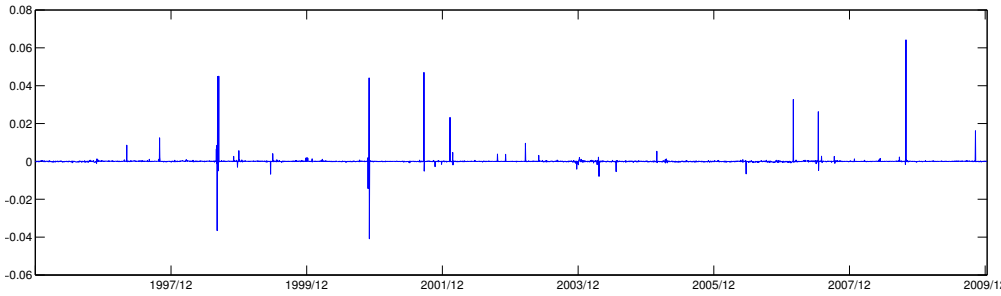


**Figure 2.14:** Extracted jumps in variance in the SV-NIG-VG-1

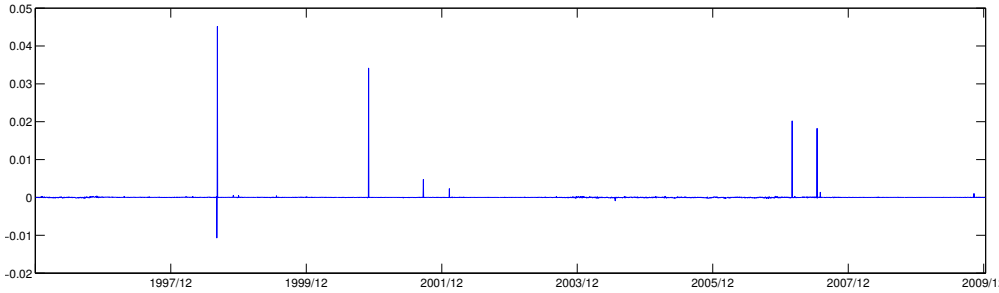


**Figure 2.15:** Extracted jumps in variance in the SV-NIG-NIG-1

Heston-$i$, Bates-$i$, and SV-P-P-$i$ perform the worst, with the best model class being the SV-VG-NIG-$i$, followed by model classes SV-NIG-$i$, SV-NIG-VG/NIG-$i$, SV-VG-$i$, and SV-VG-VG-$i$. The large difference in the DIC between the benchmark models (Heston-$i$, Bates-$i$, and SV-P-P-$i$) and the other models provides strong evidence of the inclusion of VG/NIG jumps in the return process, which produces a better fit than not only the diffusion model but also models with compound Poisson jumps. In particular, the inclusion of NIG variance jumps in the SV-VG can further improve its goodness of fit. A surprising observation is that the inclusion of VG variance jumps (SV-VG-VG-$i$) does not result in an obviously lower DIC than the SV-VG-$i$. This implies that the VG-type jump may not be an appropriate choice for the variance process, at least compared to the NIG jump. In the models with NIG return jumps, the inclusion of variance jumps, in contrast, worsens the goodness of fit due to over-fitting, suggesting that no extra infinite-activity jumps are necessary in variance for the SV-NIG models.

However, if we take a closer look at the difference in DICs caused by the linear variance process and inclusion of jumps, there is mixed evidence about the relative importance of these two specifications. The goodness-of-fit performance becomes much better after the compound Poisson jumps in returns are replaced with VG/NIG jumps, even if a comparison is made between the Bates-1/SV-P-P-1 and the SV-VG-0/SV-NIG-0. The inclusion of proper NIG-type variance jumps can further improve the fit, but this can be achieved only after changing the variance process from SQR to the linear type. In terms of modelling variance dynamics, the VG jump has proved not to be the best choice, as it fails to markedly improve the SV-VG models. In general terms, linear variance specification can improve the fit within the same magnitude, but the inclusion of appropriate jumps in returns and variance can significantly lower the DICs. In other words, the inclusion of VG/NIG return jumps is the first priority, followed by the specification of the linear variance process, and then the inclusion of the "right" jump in variance.

I have one comment about the difference in DICs. The difference between the highest and lowest DICs across all models is 14,547. Within the same model group, the differences in DICs caused by the different $\beta$ range from 109 in the Heston-$i$ to 7,354 in the SV-VG-NIG-$i$, and the differences caused by the use of infinite-activity jumps in returns or variance are about 3,000-5,000. However, as pointed out in Spiegelhalter et al. (2002), a small difference in the DICs can be significant and, by contrast, my result is surprisingly strong. I also compare the DICs with Kaeck and Alexander (2012), because we discuss some common models, and because our sample data sets are quite similar. In their paper, they compare one-factor and two-factor models with $\beta = 0.5$ or 1 or flexible $\beta$, and the largest DIC difference is 44,511, caused by the use of a second factor. However, the difference caused by $\beta$ in one-factor models, which is about 600, is quite small compared to mine, whereas in two-factor models, the difference caused by a different $\beta$ is an average of 3,000. In one-factor models, the difference in DICs caused by the use of compound Poisson jumps in returns and variance is about 900, and in two-factor models it is about 3,000, on average, derived from the penalty of overfitting with jump components. Considering that Kaeck and Alexander (2012) use the VIX term structure of 30 and 360 days, the second factor plays an important role in capturing the long-term behaviour of variance, whereas in the short term, jumps can help create frequent fluctuations in returns and variance, which may explain the relatively large difference in DICs caused by jumps in my estimation. Another possible reason is errors in the Monte Carlo sampling. To study this, I re-estimate the SV-P-P-0, 1 and notice that the Monte Carlo sampling seems to cause small differences in the DIC, but they are insignificant compared to those between different models.

### 2.6.3 Fit to the VIX

Table 2.6 reports the estimates of the correlation factor $\rho_\varepsilon$ and the scaled factor $\sigma_\varepsilon$. Among all models, the difference in $\rho_\varepsilon$ and $\sigma_\varepsilon$ is negligible; therefore, no inference can be drawn regarding the performance of fit to the VIX. Across all the models, $\rho_\varepsilon$ is close to 1, suggesting a significant positive autocorrelation between the VIX pricing errors on neighbouring days. A similar observation is made in Kanniainen et al. (2014), where an additive error model with auto-regressive disturbances is used for VIX pricing errors, and also in Yu et al. (2011), which, like me, use the multiplicative error model with auto-regressive disturbances to model the errors between the market prices and the theoretical values of options on neighbouring days. The autocorrelation factor in both papers is close to 1. With the same error model, but for variance swap rates, the result

**Table 2.6:** Estimates of $\rho_\varepsilon$ and $\sigma_\varepsilon$ in the Heston, Bates, SV-P-P, SV-VG, SV-VG-VG, SV-VG-NIG, SV-NIG, SV-NIG-VG, and SV-NIG-NIG with different choices of $\beta$. The parameters are estimated using daily spot returns of the S&P 500 index and the reformulated 30-day VIX from January 1996 to December 2009. The parameter values are the mean of the posteriors as annual decimals. The standard errors are the standard deviations of the posteriors, reported in parentheses.

| Models | Heston-0 | Heston-1 | Bates-0 | Bates-1 | SV-P-P-0 | SV-P-P-1 |
|---|---|---|---|---|---|---|
| $\rho_\varepsilon$ | 0.9870 | 0.9919 | 0.9800 | 0.9832 | 0.9988 | 0.9968 |
| | (0.0032) | (0.0025) | (0.0034) | (0.0040) | (0.0020) | (0.0015) |
| $\sigma_\varepsilon$ | 0.0371 | 0.0344 | 0.0514 | 0.0339 | 0.0273 | 0.0305 |
| | (0.0008) | (0.0010) | (0.0017) | (0.0009) | (0.0012) | (0.0031) |

| Models | SV-VG-0 | SV-VG-1 | SV-VG-VG-0 | SV-VG-VG-1 | SV-VG-NIG-0 | SV-VG-NIG-1 |
|---|---|---|---|---|---|---|
| $\rho_\varepsilon$ | 0.9861 | 0.9816 | 0.9898 | 0.9835 | 0.9840 | 0.9828 |
| | (0.0033) | (0.0041) | (0.0043) | (0.0045) | (0.0221) | (0.0085) |
| $\sigma_\varepsilon$ | 0.0347 | 0.0333 | 0.0352 | 0.0343 | 0.1636 | 0.0377 |
| | (0.0008) | (0.0009) | (0.0009) | (0.0011) | (0.2039) | (0.0039) |

| Model | SV-NIG-0 | SV-NIG-1 | SV-NIG-VG-0 | SV-NIG-VG-1 | SV-NIG-NIG-0 | SV-NIG-NIG-1 |
|---|---|---|---|---|---|---|
| $\rho_\varepsilon$ | 0.9955 | 0.9780 | 0.9789 | 0.9789 | 0.9923 | 0.9786 |
| | (0.0020) | (0.0034) | (0.0052) | (0.0050) | (0.0055) | (0.0086) |
| $\sigma_\varepsilon$ | 0.0410 | 0.0342 | 0.0428 | 0.0350 | 0.0435 | 0.0387 |
| | (0.0008) | (0.0011) | (0.0007) | (0.0016) | (0.0068) | (0.0023) |

obtained in Amengual (2009) differs slightly from mine. In their paper, the values of $\rho_\varepsilon$, regardless of the choice of models, are U-shaped: close to zero in three months and much higher in one month or two years. In particular, their one-month $\rho_\varepsilon$ ranges from 0.437 to 0.684. However, using the same error model, Kaeck and Alexander (2012) obtain quite different results: $\rho_\varepsilon$ is almost zero for the 30-day VIX index, and for the 360-day VIX index, it shows large positive values for one-factor models and again nearly zero values for two-factor models, suggesting weak autocorrelation in the error terms. This mixed difference between some papers and mine may result from our different sources of information or different choices of models. In particular, I compare the model-implied VIX and the market VIX and notice that there tends to be strong, positive autocorrelation between daily VIX pricing errors when there are significant jumps in the path of the market VIX, that is, when the stock market became volatile. This positive autocorrelation may come from a variance path different from the pattern of the VIX index,[4] and suggests that if the data sample for estimation covers periods when the market was turbulent, the autocorrelation parameter $\rho_\varepsilon$ could be more significant than that estimated from the data sample that comes from a period during which the market was relatively quiet. Moreover, the large positive autocorrelation between daily errors may also suggest that models do not perform well in fitting the VIX and the index at the same time. Therefore, a near random-walk error process with a large positive autocorrelation factor could help

[4]It may be because I extract the variance using the *joint* information of the index return and VIX index.

to capture the deviation between the market VIX and model-implied VIX.[5] Furthermore, the choice of estimation method counts. In my research, the variance path is extracted with the random-walk MH algorithm and in Kaeck and Alexander (2012) with the ARMS algorithm to estimate the variance. In my other research with the MCMC, I observe that methods of extracting the variance dynamics might affect the model-implied VIX. If a method tends to extract a more volatile volatility curve, the model-implied VIX is more volatile; consequently, daily errors between the model-implied and market VIX might vary widely from day to day, implying a lower autocorrelation coefficient.

### 2.6.4 Option Pricing Performance

In this section, I address the question of whether infinite-activity Lévy jumps can improve the option pricing performance. I answer this question under both affine (SQR) and non-affine (linear) volatility diffusions. Table 2.7 reports the option RMSEs of all the models for three samples: Sample A comprising the Wednesday options from January 1996 to December 2009, Sample B comprising the Thursday options from January 1996 to December 2009,[6] and Sample C comprising the Wednesday and Thursday options from January 2010 to December 2010. The test[7] allows me to analyse the role of different model specifications in option pricing, and it further verifies the effectiveness of the use of the VIX index in estimating the models' risk-neutral dynamics.

As for the role of jumps, I first examine option performance with affine models and then non-affine models.

*Importance of Infinite-Activity Jumps under Affine (SQR) Volatility Diffusion*

In pricing options from the long-term Samples A and B, the SV-NIG-0 achieves the best result with Sample A, and its result with Sample B is very close to the best one obtained by the SV-NIG-VG-0. The superior performance of the affine model with NIG return jumps over the Heston-0, Bates-0, and SV-P-P-0 suggests that the inclusion of NIG return jumps can significantly improve the performance of the diffusion model in pricing options in the long-term sample period, and that it is more appropriate than the inclusion of compound Poisson jumps. However, there is mixed evidence about the role of variance jumps. On the one hand, the inclusion of VG variance jumps improves the pricing performance of the SV-NIG-0 with Sample B, and the SV-VG-NIG-0 achieves the third best performance with Samples A and B, better than the SV-VG-0; on the other hand, the model specifications of the SV-VG-VG-0 and SV-NIG-NIG-0 make the results deteriorate, and this suggests that more complex models do not necessarily lead to better results. The SV-VG-0 achieves a result comparable to that of the Heston-0 and Bates-0 with Samples A and B, but it achieves the best option pricing performance with the short-term Sample C. Surprisingly, the SV-NIG-0, which is among the two best models with Samples A and B, produces the worst result with Sample C, while the SV-VG-NIG-0 and SV-NIG-VG-0 maintain their performance. Another interesting observation is that

[5] In future research, it would be interesting to preset the autocorrelation factor $\rho_\varepsilon$ to a lower value, which may lead to the model-implied VIX further coinciding with the market VIX.

[6] The use of Thursday option data is aimed at enhancing the evidence of model performance, motivated by Christoffersen et al. (2010a), where the Wednesday options are used for estimation and the Thursday options for out-of-sample analysis.

[7] As argued in Christoffersen et al. (2010b), the option pricing test is out-of-sample even if the data sample used for estimation is from the same period as the option sample. Therefore, by following Christoffersen et al. (2010b), all option Samples A, B, and C can be considered out-of-sample because no option data are directly used for the estimation.

**Table 2.7:** Option $\mathcal{V}$RMSEs for affine ($i = 0$) and non-affine ($i = 1$) models. Sample A: Options on Wednesdays from January 1996 to December 2009. Sample B: Options on Thursdays from January 1996 to December 2009. Sample C: Options on Wednesdays and Thursdays from January 2010 to December 2010. Options with prices below 3/8 dollar or maturities shorter than one week or longer than one year are filtered out. I compute the option prices using the Monte Carlo method with the technique of antithetic variates. The number of Monte Carlo paths is $50,000$, and the models are discretised at daily frequency. In each sample, the bolded and italic numbers are the lowest and highest $\mathcal{V}$RMSEs, respectively, and the models are ranked according to their performance in Sample A.

| model | $i$ | A | B | C |
|---|---|---|---|---|
| SV-NIG | 0 | **4.4506** | 5.0538 | 5.3543 |
| SV-NIG-VG | 0 | 4.5488 | **5.0458** | 4.8854 |
| SV-VG-NIG | 0 | 4.8204 | 5.1926 | 4.7558 |
| Heston | 0 | 4.8397 | 5.4596 | *5.5918* |
| Bates | 0 | 4.8491 | 5.3973 | 5.0740 |
| SV-VG | 0 | 4.8855 | 5.3167 | **4.5432** |
| SV-NIG-NIG | 0 | 5.1724 | 5.6865 | 5.4251 |
| SV-P-P | 0 | 5.1917 | 5.6332 | 5.0015 |
| SV-VG-VG | 0 | *6.4875* | *6.6092* | 5.3300 |

| model | $i$ | A | B | C |
|---|---|---|---|---|
| SV-NIG | 1 | **3.2988** | **3.8806** | **3.4939** |
| SV-VG-NIG | 1 | 3.6579 | 4.1553 | 3.7446 |
| SV-VG | 1 | 3.7863 | 4.2534 | 3.7698 |
| Heston | 1 | 3.8896 | 4.4093 | 4.5014 |
| Bates | 1 | 3.9211 | 4.4643 | *4.7496* |
| SV-NIG-NIG | 1 | 4.0437 | 4.4377 | 4.0404 |
| SV-VG-VG | 1 | 4.1851 | 4.6071 | 4.0656 |
| SV-NIG-VG | 1 | 4.3431 | 4.4265 | 4.1746 |
| SV-P-P | 1 | *4.6311* | *4.9499* | 4.4725 |

the SV-P-P-0 outperforms the Heston-0 and Bates-0, and ranks fourth best with Sample C. This observation shows that with the affine variance dynamics, the performance of different return jump specifications is not stable with different samples. Overall, the performance of infinite-activity jump models is mixed between the different option price data samples. The models' mixed, inconsistent performance in pricing options in the long-term and short-term samples suggests that the use of (unrealistic) affine variance dynamics may negatively affect the identification of the other model components and lead to an unstable pricing performance between different samples.

*Importance of Infinite-Activity Jumps under Non-Affine (linear) Volatility Diffusion*

In contrast to the mixed performance of the affine models with different samples, with the non-affine variance dynamics, the SV-NIG-1, SV-VG-NIG-1, and SV-VG-1 with infinite-activity jumps are always the best three models for all the samples, and their option pricing performance is much better than the best results obtained by the affine models. This observation shows that infinite-activity jumps, especially NIG return jumps, should be used with non-affine variance dynamics, instead of finite-activity compound Poisson jumps. Similar to those with affine models, the results with non-affine models imply that

**Table 2.8:** Option $\mathcal{V}$RMSEs for models in different maturity groups. Sample: Options on Wednesdays from January 2010 to December 2010. Options with prices below 3/8 dollar or maturities shorter than one week or longer than one year are filtered out. The option prices are computed using the Monte Carlo method with the technique of antithetic variates. The number of Monte Carlo paths is 50,000, and the models are discretised at daily frequency.

| Maturity | # Options | Heston-0 | Heston-1 | Bates-0 | Bates-1 | SV-P-P-0 | SV-P-P-1 |
|---|---|---|---|---|---|---|---|
| 7 - 90 | 6926 | 6.1884 | 5.0640 | 5.7554 | 4.8434 | 5.4191 | 4.7530 |
| 91 - 180 | 1663 | 4.9269 | 3.2591 | 4.2301 | 3.7321 | 4.6767 | 3.9233 |
| 181 - 365 | 1187 | 3.6833 | 2.5166 | 3.2569 | 4.9606 | 4.1484 | 3.6546 |
| Maturity | # Options | SV-VG-0 | SV-VG-1 | SV-VG-VG-0 | SV-VG-VG-1 | SV-VG-NIG-0 | SV-VG-NIG-1 |
| 7 - 90 | 6926 | 5.0464 | 4.1744 | 5.6661 | 4.3562 | 5.3376 | 4.0894 |
| 91 - 180 | 1663 | 3.5736 | 2.3744 | 4.4217 | 2.7691 | 4.1223 | 2.1997 |
| 181 - 365 | 1187 | 3.0746 | 2.6338 | 4.2983 | 2.8392 | 3.4604 | 2.8178 |
| Maturity | # Options | SV-NIG-0 | SV-NIG-1 | SV-NIG-VG-0 | SV-NIG-VG-1 | SV-NIG-NIG-0 | SV-NIG-NIG-1 |
| 7 - 90 | 6926 | 5.7412 | 3.9579 | 5.5780 | 4.4826 | 5.7293 | 4.5392 |
| 91 - 180 | 1663 | 4.3054 | 2.4266 | 3.9773 | 3.4209 | 5.1226 | 2.6424 |
| 181 - 365 | 1187 | 5.7584 | 2.9047 | 4.0908 | 4.0810 | 5.8349 | 2.4138 |

the inclusion of jumps in variance does not necessarily lead to an improved option pricing performance. The choice of jump type matters, and the impact of the variance jumps is not as significant as that of the return jumps. Indeed, the parsimonious SV-NIG-1 clearly outperforms all the other models, including the more complex models with variance jumps, and the inclusion of either VG or NIG variance jumps in the SV-NIG-1 distorts the pricing results. This implies that the NIG return jump is sufficient for capturing the jump risks under the risk-neutral measure without variance jumps, and that extra model specifications seem to destabilise the model estimation.

Overall, complicated models with extra parameters do not necessarily yield a better option pricing performance. For example, in Eraker (2004), with the parameters directly estimated from returns and option data, the in-sample option pricing performance of the Bates-0 model is comparable to that of the augmented SV-P-P-0, and even better than SV-P-P-0 in terms of out-of-sample performance. Importantly, I do not directly calibrate the models to the option data, which makes it possible for the SV-NIG-1 to have better results with Samples A and B than the SV-NIG-NIG/VG-1. However, if one prefers to include jumps in variance, the specification of the SV-VG-NIG is best for both affine and non-affine variance dynamics.

In all samples, the specification of the non-affine linear variance process can significantly reduce the option $\mathcal{V}$RMSE across different model groups, up to 35% in the SV-VG-VG-$i$ in Sample A, and 23% and 36% in the SV-NIG-$i$ in Samples B and C, respectively. In general, the superior performance of the linear variance process across different model groups strongly suggests that to capture the risk-neutral dynamics, the variance process with $\beta = 1$ is preferable to the SQR variance process. More importantly, as highlighted above, with the non-affine variance dynamics, the performance of return jumps is consistent with different option samples, whereas with the affine dynamics, the results are mixed. This suggests that the use of non-affine dynamics both improves model performance and stabilises it.

**Table 2.9:** Option $\mathcal{V}$RMSEs for models in different moneyness groups. Sample: Options on Wednesdays from January 2010 to December 2010. Options with prices below 3/8 dollar or maturities shorter than one week or longer than one year are filtered out. The option prices are computed using the Monte Carlo method with the technique of antithetic variates. The number of Monte Carlo paths is 50,000, and the models are discretised at daily frequency.

| moneyness | # options | Heston-0 | Heston-1 | Bates-0 | Bates-1 | SV-P-P-0 | SV-P-P-1 |
|---|---|---|---|---|---|---|---|
| $< 0.9$ | 3390 | 6.6727 | 4.8862 | 5.7688 | 4.5984 | 5.8705 | 4.8242 |
| 0.9 - 0.97 | 3111 | 5.9981 | 4.9288 | 5.6582 | 5.2043 | 5.2291 | 5.0293 |
| 0.97 - 1.03 | 2846 | 3.8318 | 3.4482 | 3.3859 | 3.9205 | 3.5410 | 3.4043 |
| 1.03 - 1.1 | 350 | 6.0552 | 5.0312 | 5.5314 | 5.5568 | 5.4663 | 5.3315 |
| $> 1.1$ | 79 | 7.3114 | 6.7341 | 8.5083 | 6.6527 | 8.3880 | 6.4026 |
| moneyness | # options | SV-VG-0 | SV-VG-1 | SV-VG-VG-0 | SV-VG-VG-1 | SV-VG-NIG-0 | SV-VG-NIG-1 |
| $< 0.9$ | 3390 | 4.9290 | 3.7040 | 5.3411 | 3.7786 | 5.4945 | 3.2580 |
| 0.9 - 0.97 | 3111 | 4.9593 | 4.4165 | 6.0141 | 4.4602 | 5.0199 | 4.3377 |
| 0.97 - 1.03 | 2846 | 3.5035 | 3.1907 | 4.1737 | 3.2588 | 3.5742 | 3.3264 |
| 1.03 - 1.1 | 350 | 5.9050 | 4.1624 | 5.9083 | 4.5042 | 5.3796 | 4.4608 |
| $> 1.1$ | 79 | 6.8483 | 6.5403 | 7.7055 | 6.6761 | 6.9993 | 6.5765 |
| moneyness | # options | SV-NIG-0 | SV-NIG-1 | SV-NIG-VG-0 | SV-NIG-VG-1 | SV-NIG-NIG-0 | SV-NIG-NIG-1 |
| $< 0.9$ | 3390 | 5.9946 | 3.4747 | 5.5608 | 4.4506 | 6.5896 | 3.7888 |
| 0.9 - 0.97 | 3111 | 5.7392 | 3.7611 | 5.4397 | 4.5384 | 5.4890 | 4.5728 |
| 0.97 - 1.03 | 2846 | 4.1765 | 3.1182 | 3.5329 | 3.5091 | 3.5008 | 3.4467 |
| 1.03 - 1.1 | 350 | 6.1278 | 4.0621 | 5.6845 | 4.8642 | 5.6951 | 4.6430 |
| $> 1.1$ | 79 | 7.2112 | 6.2430 | 6.8266 | 7.0017 | 8.8259 | 6.8553 |

Table 2.8 reports the option $\mathcal{V}$RMSEs of models in different maturity groups. The option sample consists of options on Wednesdays over the short-term sample period of January 2010 to December 2010, and the options are divided into three maturity groups: short maturity (7-90 calendar days), intermediate maturity (91-180 calendar days), and long maturity (181-365 calendar days).

According to Table 2.8, the difference in $\mathcal{V}$RMSEs caused by a change in $\beta$ is considerable in all model groups. Especially for models with infinite-activity Lévy jumps, the reduction ranges from 17.3% to 31.1% in pricing short-dated options, from 33.6% to 48.4% in pricing intermediate-maturity options, and from 14.3% to 58.6% in pricing long-maturity options.

Moreover, using infinite-activity Lévy jumps in returns can reduce pricing errors in all maturity groups. Specifically, I first compare the $\mathcal{V}$RMSEs of the Heston-1 and SV-VG-1: the $\mathcal{V}$RMSE is reduced by 17.6% for short-maturity options, 27.1% for intermediate-maturity options, and is increased by 4.7% for long-maturity options, which is not surprising, because jumps mainly affect the prices of short-term options. The inclusion of NIG variance jumps can further improve the model SV-VG-1 in pricing options with short and intermediate maturities. The model SV-NIG-1 outperforms the SV-VG-1 in pricing short-dated options; however, in pricing options with intermediate and long maturities, the SV-VG-1 achieves the lowest $\mathcal{V}$RMSEs. Moreover, although the overall $\mathcal{V}$RMSEs of the SV-NIG-NIG-1 in Samples A, B, and C are greater than those of the SV-NIG-1, the SV-NIG-NIG-1 yields a comparable result in pricing intermediate-dated options and a 17% improvement in pricing long-dated options.

Table 2.9 reports option $\mathcal{V}$RMSEs in five moneyness groups. Here, moneyness is computed as the ratio of spot price and strike price $S_0/K$, and options are divided into groups of deep in-the-money (ITM), ITM, at-the-money (ATM), out-of-money (OTM), and deep

OTM. Each moneyness group consists of both call and put options. For example, the deep OTM option group consists of options with $S_0/K < 0.9$ for calls and $S_0/K > 1.1$ for puts, and the OTM options refer to those with $0.9 < S_0/K < 0.97$ for calls and $1.03 < S_0/K < 1.1$ for puts.

Uniformly, in different model and moneyness groups, the models with $\beta = 1$ significantly outperform those with $\beta = 0.5$. Next, I compare the $\mathcal{V}$RMSEs between the Heston-1 and SV-NIG-1 in five moneyness groups to study the role of infinite-activity jumps. The errors are reduced by 28.9%, 23.7%, 9.6%, 19.3%, and 7.3% in pricing deep OTM, OTM, ATM, ITM, and deep ITM, respectively. This suggests that the model with NIG return jumps is preferable to a simple diffusion model in pricing ITM, OTM, and deep OTM options, although in the other two moneyness groups, the improvement in pricing options may be trivial. On the other hand, the inclusion of infinite-activity jumps in variance seems to have no significant impact on reducing pricing errors, except that there is an improvement of 12.0% caused by the inclusion of NIG variance jumps in the SV-VG-1 in pricing deep OTM options. In all the other four moneyness groups, the SV-NIG-1 achieves the lowest $\mathcal{V}$RMSEs.

Thus, overall, in terms of option pricing in different moneyness and maturity groups, using linear variance has a strong impact, especially on options with short or intermediate maturities, regardless of moneyness. Moreover, adding VG/NIG return jumps can significantly reduce pricing errors in all moneyness and maturity groups, except for deep ITM options. Including NIG variance jumps in the SV-VG-1 can further improve the pricing of options with short and intermediate maturities or deep OTM options, but on the whole, the role of NIG variance jumps is less important than that of the VG/NIG return jumps. The model SV-NIG-1 outperforms the other models in almost all moneyness groups, except that the SV-VG-NIG-1 achieves a slightly better result in pricing deep OTM options. The SV-NIG-1 performs best when pricing short-dated options; however, in the other two maturity groups, it underperforms models with VG return jumps with and without variance jumps.

### 2.6.5 Comparison of Models with $\beta = 1$ and Flexible $\beta$

It is worth mentioning that I also estimate models with the CEV-type variance process; that is, $\beta$ is a flexible parameter and is estimated from the data, although the estimation results of the models are not reported. In particular, I follow Ait-Sahalia and Kimmel (2007) and impose the restriction that $0.5 \leq \beta \leq 1$. The restriction is realised by choosing $N(0.8, 0.3)$ as the prior of $\beta$, and in the step of updating $\beta$, the samples of $\beta$ are repeatedly drawn until they satisfy the restriction. Interestingly, the estimation results in Duan and Yeh (2010) and Kaeck and Alexander (2012) without such a restriction suggest that the elasticity of variance of volatility is between 0.5 and 1.[8]

I notice that in all model groups except the SV-P-P, the values of $\beta$ are very close to 1. Specifically, $\beta$ is 0.7663 in the SV-P-P, which is close to the estimation of $\beta = 0.778$ in Kaeck and Alexander (2012). In the SV-VG, $\beta$ rises to 0.8162, followed by 0.8893 in the SV-VG-VG, and then to around 0.999 in the SV-VG-NIG, very close to 1. Apparently, the value of $\beta$ rises as the effectiveness of the jumps increases in generating frequent small jumps. It is reasonable because, as mentioned above, the infinite-activity Lévy jump can capture jumps that are too small for the compound Poisson jump and too

[8] Ait-Sahalia and Kimmel (2007) use the VIX as a volatility proxy, whereas Duan and Yeh (2010) and Kaeck and Alexander (2012) estimate models from index and VIX data.

big for Brownian motion to capture, so the presence of VG or NIG variance jumps is sufficient to capture small jumps, and there is no need to have a small $\beta$ to "enlarge" the diffusion term. However, the inclusion of compound Poisson jumps in variance, as in the model SV-P-P, does not produce a $\beta$ close to 1, since the inclusion of compound Poisson jumps only accounts for large jumps rather than frequent small jumps. Surprisingly, $\beta$ is estimated at 0.9222 in the SV-NIG, although this model has no variance jump. One possible explanation is that with NIG return jumps, the return process shows sufficient fluctuation, which obviates an active variance process for generating extra volatility.

Then I compare the DICs and option pricing performance of the models with $\beta = 1$ and a flexible $\beta$ and observe no significant difference and, overall, the specification of $\beta = 1$ leads to better results. Similar results can be found in Kaeck and Alexander (2012), where in all three groups of the Heston, Bates, and SV-P-P, the models with $\beta = 1$ achieve lower DICs than those with a flexible $\beta$. Considering the length and readability of this chapter, I omit the estimation results of models with a flexible $\beta$, but will provide them on request.

## 2.7 Discussion

In this chapter, I use an extensive data set from stock and option markets to estimate and compare several model specifications, including the use of infinite-activity Lévy jumps in returns, or in both returns and variance, and the specification of the linear variance process, against the benchmark Heston, Bates, and double-Poisson-jump models with the SQR variance process and with/without finite-activity compound Poisson jumps. The models are estimated with the MCMC, using the joint information of daily S&P 500 index returns and the reformulated 30-day VIX index from January 1996 to December 2009. Very little research has examined the empirical option pricing performance of stochastic volatility models with infinite-activity Lévy jumps. Notable exceptions include Li et al. (2008), Yu et al. (2011), and Ornthanalai (2014). The difference between my research and theirs lies mainly in the use of non-affine variance dynamics and infinite-activity variance jumps.

The empirical results clearly confirm the importance of the linear variance process in terms of both goodness of fit and option pricing, as noted in the literature (see, for example, Christoffersen et al., 2010a; Kaeck and Alexander, 2012). More importantly, the improvement provided by infinite-activity Lévy jumps, a fact not widely recognised, seems more significant, especially for non-affine models. Models with VG/NIG return jumps markedly outperform the benchmark models in option pricing and goodness of fit, and the SV-VG-NIG-1 model achieves the best performance in goodness of fit, and the SV-NIG-1 achieves the best performance in option pricing. Overall, the models SV-VG-NIG-1 and SV-NIG-1 are comparable, and considering the extra model complexity of the SV-VG-NIG-1, I conclude that the parsimonious model SV-NIG-1 is the more competitive one.

With modern acceleration technologies (e.g., clouds, FPGAs, GPUs) closed-form solutions are not necessarily a critical requirement for option valuation. Therefore, my results strongly suggest that the infinite-activity NIG jumps in the return process should be used with non-affine volatility diffusion, which obviates jumps in variance.

# 3 Model Estimation with Advanced MCMC Algorithms: Simulation Studies

## 3.1 Motivation

In this chapter, I apply a variety of advanced MCMC algorithms to estimate three financial models with different jump specifications and compare their estimation performance to that of the random-walk MH algorithm. The algorithms comprise the AM (Haario et al., 2006), the FUSS (Martino et al., 2015), the PMMH (Andrieu et al., 2010), and the PGAS (Lindsten et al., 2014).

The models estimated are the affine Heston and Bates models (the Heston-0 and Bates-0 models) and the affine model with NIG jumps in the log price process (the SV-NIG-0 model). These three models are selected because they represent stochastic volatility models without jumps, with finite-activity jumps, and with infinite-activity jumps, respectively. I seek to study whether the inclusion of jumps and the type of jumps affect the estimation performance of algorithms, and what can be learned in practice to better estimate models with high-dimensional latent state variables, including strong-autocorrelated variance and independent jumps.

This chapter focuses on simulation studies. First, I simulate one-year data of log asset prices and spot variance under the Heston-0, Bates-0, and SV-NIG-0 models, respectively. Second, I use the log prices as observations to estimate the *physical* dynamics of these models and extract the jumps and paths of the spot variance.

The objectives of this chapter are:

- To apply advanced MCMC methods to estimate complex models with a number of unknown parameters and high-dimensional latent state variables;
- To examine the mixing properties of different algorithms and to determine the optimal settings for the algorithms to improve the estimation performance and to keep the computational costs reasonable;
- To compare the estimation performance of different algorithms and analyse the possible reasons for the success and failure of each algorithm, and thereby determine the strengths and weaknesses of the algorithms.

This chapter is organised as follows. Sections 2 and 3 briefly review the conventional MCMC and SMC methods. In Section 4, I introduce the advanced MCMC algorithms in

detail, including the general idea, algorithm design, and features. Section 5 describes the simulated data. Section 6 presents the implementation of each algorithm in the estimation experiments of the Heston-0, Bates-0, and SV-NIG-0 models. Section 7 presents the estimation results and compares the estimation performance of the algorithms. Section 8 briefly summarises this chapter.

## 3.2 Review: Conventional MCMC Methods

Recently, MCMC methods have gained popularity in finance research, and quite a few papers have applied the MCMC methods to the estimation of financial models (see, for example, Eraker et al., 2003; Kaeck and Alexander, 2013b; Yu et al., 2011, and reference therein). The general idea of MCMC algorithms is to generate a Markov chain, whose equilibrium distribution is a given target distribution, that is, the joint posterior $p(\Theta, X|Y)$, where $\Theta$ represents the set of model parameters, $X$ represents the state variable, and $Y$ represents the observation. However, in most cases, a direct sampling from $p(\Theta, X|Y)$ is not possible because it is not of a known form. Instead, the joint posterior is broken into its complete set of conditional distributions $p(\Theta|X, Y)$ and $p(X|\Theta, Y)$, from which samples of $\Theta$ and $X$ are generated sequentially. As justified by the Hammersley-Clifford theorem (Hammersley and Clifford, 1971), the joint posterior $p(\Theta, X|Y)$ can be completely characterised by its complete set of conditional distributions $p(\Theta|X, Y)$ and $p(X|\Theta, Y)$.

Two of the most widely-used MCMC approaches are the MH algorithm and the Gibbs sampler (Johannes and Polson, 2003; Liu, 2004; Robert and Casella, 2004). Suppose the conditional distributions $p(\Theta|X, Y)$ and $p(X|\Theta, Y)$ are known in closed form and can be directly sampled, the Gibbs sampler generates the chain as follows:

Step 1 **Initialisation**: Assume the initial state: $(\Theta^{(0)}, X^{(0)})$,

Step 2 **Recursion**: For $i = 1, ..., M$,

1. Sample $X^{(i)}$ from $p(X^{(i)}|\Theta^{(i-1)}, Y)$,
2. Sample $\Theta^{(i)}$ from $p(\Theta^{(i)}|X^{(i)}, Y)$,

where $M$ is the pre-specified length of the chain. This algorithm generates a sequence of random variables $\{\Theta^{(i)}, X^{(i)}\}_{i=1}^{M}$, and the equilibrium distribution of the chain converges to $p(\Theta, X|Y)$, as justified by the Hammersley-Clifford theorem.

We can further factorise the multivariate conditional distribution $p(\Theta|X, Y)$ into its univariate conditional distributions. When a multivariate target distribution can be easily factorised into univariate conditional distributions, the key point for successfully applying the Gibbs sampler is the ability to draw efficiently from these univariate distributions (Koch, 2007; Liu, 2004; Robert and Casella, 2004). The best scenario for the Gibbs sampler occurs when exact samplers are available for each full conditional.

However, in many situations, it is not possible to directly sample from one or more of the conditional distributions, and one has to use the MH algorithms. The idea of MH algorithms is to generate a candidate draw from a simpler proposal density $q$, and then accept or reject the candidate draw according to an acceptance criterion.

Consider the case in which the posterior of $X$, $\pi(X)$, can be evaluated, however, it is not possible to generate a sample directly from the distribution $\pi(X)$. The MH algorithms update $X$ as follows:

**1**. Sample $X^{(i)}$ from the proposal density $q(X^{(i)}|X^{(i-1)})$,

**2**. Accept $X^{(i)}$ with probability $\alpha(X^{(i)}, X^{(i-1)}) = min\left(\frac{q(X^{(i-1)}|X^{(i)})\pi(X^{(i)})}{q(X^{(i)}|X^{(i-1)})\pi(X^{(i-1)})}, 1\right)$.

The acceptance criterion ensures that the distribution of the samples generated by the MH converges to $\pi(X)$.

The success of the MH lies in an appropriate choice of the proposal $q$ and, ideally, a good proposal should lead to a reasonable acceptance rate and a small correlation between the generated samples. The MH algorithms are usually applied inside another external MCMC (the Gibbs sampler). Therefore, typical problems of the external MCMC (such as a long "burn-in" period and a large correlation) may dramatically deteriorate if the internal MCMC is not efficient.

One popular special case of MH algorithms is the random-walk MH algorithm. As its name suggests, the random-walk MH generates a candidate sample from a random-walk model $X^{(i)} = X^{(i-1)} + \epsilon$, where $\epsilon$ is an independent mean zero error term. Typically, $\epsilon_t$ is assumed to be symmetric, and this assumption leads to a symmetric $q$ and, therefore, the algorithm is simplified as follows:

**1**. Sample $X^{(i)}$ from the proposal density $q(X^{(i)}|X^{(i-1)})$,

**2**. Accept $X^{(i)}$ with probability $\alpha(X^{(i)}, X^{(i-1)}) = min\left(\frac{\pi(X^{(i)})}{\pi(X^{(i-1)})}, 1\right)$.

Further, the DWW method proposed by Damlen et al. (1999) seeks to simplify the sampling from complicated posteriors. The DWW method introduces an auxiliary variable $U$ and extends the Gibbs sampler to include the extra full conditional. Then, all the full conditionals are of a known form and, thus, can be directly sampled. Specifically, suppose the posterior of $X$ given by $\pi(X)$ is of the form:

$$\pi(X) \propto t(X)l(X), \tag{3.1}$$

where $t(X)$ is a density of a known form, and $l(X)$ is a non-negative invertible function. Then, the update of $X$ is as follows:

**1**. Sample $X^{(i)}$ from $t(X^{(i)})$,

**2**. Draw an auxiliary variable $u$ from Uniform$(0, l(X^{(i-1)}))$,

**3**. Accept $X^{(i)}$ if $l(X^{(i-1)}) > u$, or keep $X^{(i-1)}$.

With a number of examples, Damlen et al. (1999) show that the DWW method is fast and efficient. In the simulation studies, the DWW method is not used because the conditional posteriors that are of unknown forms cannot be factorised as in Equation 3.1, but the DWW method is applied in the empirical studies in Chapter 4, where a number of conditional posteriors can be represented as products of $t(X)l(X)$.

## 3.3 Particle Filters

The general idea of the SMC, or particle filters, is to represent the target posterior with a finite set of particles with weights, and as a new observation arises, the particles are updated in order to represent the new posterior. A variety of techniques can be used to update the particle set. The bootstrap filter, which is the original particle filter (Djuric et al., 2003; Doucet et al., 2001; Gordon et al., 1993), uses the sampling/importance re-sampling; however, the BF blindly samples $X_{t+1}^{(i)}, i = 1, ..., N$, with respect to the observation $Y_{t+1}$. The auxiliary particle filter (APF), proposed by Pitt and Shephard (2001), seeks to tackle the problem by introducing an auxiliary index $k$, which is discarded at each time step, to combine the information of the likelihood of $Y_{t+1}$ when sampling $X_{t+1}^{(i)}$. Other update techniques include, for example, the rejection sampling as in Müller (1991) and the importance sampling as in Liu and Chen (1998). Some finance research applies particle filters to estimate stochastic volatility models (see, for example, Creal, 2008; Duan and Yeh, 2011; Li, 2011), and Creal (2012) introduces and surveys particle filters applicable to finance and economics in depth.

In this thesis, I consider only the BF and APF as representatives of SMC methods to be used in the PMCMC methods; hence, the algorithms of only the BF and APF are described in detail as follows.

*Bootstrap Filter*

Step 1 **Initialisation**: Draw $X_0^i$ from the prior $p(X_0)$ and set the weights $w_0^i = 1/N$, for $i = 1, ..., N$,

Step 2 **Recursion**: For $t = 1, ..., T-1$,

1. Draw $X_{t+1}^i \sim \sum_{j=1}^N p(X_{t+1}|X_t^j)w_t^j$, for $i = 1, ..., N$,
2. Compute the weights $w_{t+1}^i \propto p(Y_{t+1}|X_{t+1}^i)$, for $i = 1, ..., N$,

where $N$ is the number of particles and $T$ is the number of observations.

*Auxiliary Particle Filter*

Step 1 **Initialisation**: Draw $X_0^i$ from the prior $p(X_0)$ and set the weights $w_0^i = 1/N$, for $i = 1, ..., N$,

Step 2 **Recursion**: For $t = 1, ..., T-1$,

1. Compute $\hat{X}_{t+1}^k = \int X_{t+1} p(X_{t+1}|X_t^k) dX_{t+1}$, for $k = 1, ..., N$,
2. Draw $k^i \sim q_{t+1}(k) \propto p(Y_{t+1}|\hat{X}_{t+1}^k)w_t^k$, for $i = 1, ..., N$,
3. Draw $X_{t+1}^i \sim p(X_{t+1}|X_t^{k^i})$,
4. Compute the weights $w_{t+1}^i \propto \frac{p(Y_{t+1}|X_{t+1}^i)}{p(Y_{t+1}|\hat{X}_{t+1}^{k(i)})}$, for $i = 1, ..., N$.

Originally, particle filters were applicable only to state-variable models with *known* parameters. In the presence of unknown parameters, Berzuini et al. (1997) treat the parameters as latent variables. However, as pointed out in Storvik (2002), the values of

some parameters may become very unlikely when a new observation arises, leading to a shrinking set of distinct parameter values. Gordon et al. (1993) introduce diversity to the particle set by adding random noise to the particles; however, the "diversity" procedure is tricky to choose. Moreover, this approach tends to overweight the information introduced by recent observations; in other words, the parameters are more consistent with recent observations although, ideally, the non-dynamic parameters should account for the information introduced by all observations.

Storvik (2002) considers an alternative approach. The only assumption made in Storvik (2002) is that the joint posterior distribution of parameters is analytically tractable. At each time point $t$, the particle set of parameters are updated to a new one using all the information available up to time $t$; therefore, this approach does not face the problems that appear in some others. For example, it avoids the problem of choosing diversity, and it does not have the drawback that the recent observations are overweighted and affect the parameter estimates much more than the old ones, since the parameters are updated on the basis of all the information available up to time $t$.

## 3.4 Advanced MCMC Methods

The key to the success of MH algorithms is the appropriate choice of proposal distributions. Ideally, the proposal distribution should be easy to sample from and lead to a reasonable acceptance rate and a small correlation between the generated samples. However, in the presence of high dimensions, complicated target distributions, and complex patterns of dependence between parameters and state variables, the simple fixed proposal in MH algorithms may not be a good approximation of the target distribution. Hence, the acceptance rate of MH algorithms can be very low, leading to a poor mixing of the chain. In this section, I introduce some newly proposed MCMC algorithms that attempt to deal with the problems of MH algorithms.

### 3.4.1 Adaptive Metropolis

After an initial non-adaptation period of $i_0$ iterations, the proposed AM algorithm updates the variance of the *Gaussian* proposal according to the values of previously generated samples. Specifically, suppose $c_i$ is the variance of the proposal in the $i^{th}$ iteration, $i > i_0$; then, $c_i$ is adapted according to the past samples of the chain as follows:

$$c_i = s \times Cov(x^{(1)}, x^{(2)}, ..., x^{(i-1)}) + s * \epsilon.$$

Here, $s$ is a parameter that scales the covariance of past samples, and according to Gelman et al. (1996), the optimal value of $s$ is $2.4^2$, because this value optimises the mixing of the Metropolis search with Gaussian targets and Gaussian proposals. $\epsilon > 0$ is a constant to make sure that $c_i$ will not become singular, and it can be set very small. $c_0$ is the initial variance of the Gaussian proposal, and it should be a strictly positive value, chosen according to some prior information.

Importantly, as discussed in Haario et al. (2006), in practice, the choice of $i_0$ affects the performance of the adaptation: the bigger it is, the slower the adaptation takes effect. Moreover, earlier in Haario et al. (1999a) with the non-ergodic version of the AM algorithm, it is found that it is not optimal to conduct the adaptation in each iterations, instead, the adaptation should be done only in given iterations to further improve the mixing of the chain. So the index $i_0$, in fact, can be used to define the length

of non-adaptation during the whole chain. In the simulation studies, the AM is applied to the update of spot variance $\mathbf{v}$ in all three models.

### 3.4.2 Particle Markov Chain Monte Carlo

The key idea of the PMCMC is to use SMC algorithms to approximate the performance of MCMC algorithms, which is achieved by generating a particle set to construct the proposal distribution used in the MH algorithm. As pointed out in Andrieu et al. (2010), for any fixed number of particles $N \geq 1$, the transition kernels leave the target density of interest invariant and lead to the convergence of the algorithms under a mild standard assumption. For further details, please refer to Andrieu et al. (2010).

In the simulation studies, I employ two schemes of PMCMC methods, namely, the PMMH (Andrieu et al., 2010) and the PGAS (Lindsten et al., 2014). In particular, the PMMH can be regarded as an approximation of the marginal MH that targets $p(\Theta|Y_{1:T})$ of $p(\Theta, X_{1:T}|Y_{1:T})$, whereas the PGAS approximates the Gibbs sampler that targets $p(\Theta, X_{1:T}|Y_{1:T})$, where $Y_{1:T}$ and $X_{1:T}$ represent the observations and state variables at time steps $t = 1, ..., T$, respectively.

#### 3.4.2.1 Particle Marginal Metropolis Hasting

The PMMH jointly updates $\Theta$ and $X_{1:T}$ with the proposal $q$:

$$q(\Theta^*, X^*_{1:T}|\Theta, X_{1:T}) = q(\Theta^*|\Theta)p_{\Theta^*}(X^*_{1:T}|Y_{1:T}).$$

In my model setting, the structure of the PMMH algorithms is as follows: consider a model with state variables $v_{1:T}$ and $J_{1:T}$,

Initialisation: $i = 0$,

1. Arbitrarily choose $\Theta^{(0)}$,
2. Run an SMC algorithm targeting $p_{\Theta^{(0)}}(v_{1:T}, J_{1:T}|Y_{1:T})$, sample $v^{(0)}_{1:T}$ and $J^{(0)}_{1:T} \sim \hat{p}_{\Theta^{(0)}}(\cdot|Y_{1:T})$, and denote $\hat{p}^{(0)}_{\Theta^{(0)}}(Y_{1:T})$ the corresponding marginal likelihood estimate.

Recursion: For $i \geq 1$,

1. sample $\Theta^* \sim q(\cdot|\Theta^{(i-1)})$,
2. run an SMC algorithm targeting $p_{\Theta^*}(v_{1:T}, J_{1:T}|Y_{1:T})$, sample $v^*_{1:T}$ and $J^*_{1:T} \sim \hat{p}_{\Theta^*}(\cdot|Y_{1:T})$, and denote $\hat{p}_{\Theta^*}(Y_{1:T})$ the corresponding marginal likelihood estimate,
4. set $\Theta^{(i)} = \Theta^*$, $v^{(i)}_{1:T} = v^*_{1:T}$, $J^{(i)}_{1:T} = J^*_{1:T}$, and set $\hat{p}^{(i)}_{\Theta^{(i)}}(Y_{1:T}) = \hat{p}_{\Theta^*}(Y_{1:T})$ with probability

$$\min(1, \frac{\hat{p}_{\Theta^*}(Y_{1:T})p(\Theta^*)q(\Theta^{(i-1)}|\Theta^*)}{\hat{p}^{(i-1)}_{\Theta^{(i-1)}}(Y_{1:T})p(\Theta^{(i-1)})q(\Theta^*|\Theta^{(i-1)})}). \tag{3.2}$$

The MH acceptance ratio in Equation 3.2 suggests that the PMMH actually approximates the marginal posterior $p(\Theta|Y_{1:T}) \propto p_\Theta(Y_{1:T})p(\Theta)$.

In the PMMH, the SMC update is used to generate samples of $v_{1:T}$ and $J_{1:T}$, and to compute the approximate marginal likelihood $p_\Theta(Y_{1:T})$. The SMC algorithms that can be

used in the PMMH without destroying the ergodicity of the Markov chain could be the original particle filter (PF), that is, the BF, and the APF. In this thesis, the algorithms combined with the PF or APF are labeled PMMH-PF and PMMH-APF, respectively.

#### 3.4.2.2 Particle Gibbs with Ancestor Sampling

Like the PG sampler, the PGAS approximates the ideal Gibbs sampler that targets $p(\Theta, X_{1:T}|Y_{1:T})$. The algorithm iteratively samples from $p(\Theta|X_{1:T}, Y_{1:T})$, which is usually feasible, and from $p_\Theta(X_{1:T}|Y_{1:T})$, which is approximated by an SMC update with the approximate likelihood $\hat{p}_\Theta(X_{1:T}|Y_{1:T})$.

In order to admit $p(\Theta, X_{1:T}|Y_{1:T})$ as an invariant density, the implementation of the PGAS requires using the so-called conditional SMC update. In the PG sampler, this type of update ensures that a pre-specified path $X'_{1:T}$ with an ancestral lineage survives all the re-sampling steps, whereas the remaining $N-1$ particles are generated as usual. However, in the PGAS, at time $t \geq 2$, a part of the original reference trajectory, $X'_{t:T}$ of $X'_{1:T}$, is connected with one of the particles $X^i_{1:t-1}, i \in \{1, ..., N\}$, and the new trajectory $\{X^i_{1:t-1}, X'_{t:T}\}$ is regarded as the reference trajectory for the next step in the SMC update. The new trajectory assigns a new value to the variable $a^N_t = i \in \{1, ..., N\}$ that corresponds to the particle $X^i_{1:t-1}$, whereas in the PG sampler, $a^N_t$ is always set to $N$.

The structure of the SMC update in the PGAS is as follows: let $X'_{1:T}$ be the reference trajectory that is associated with the ancestral lineage $\{N, ..., N\}$,

Step 1: for $t = 1$,

1. Sample $X^i_1 \sim q_{\Theta,1}(X_1)$, for $i = 1, ..., N-1$.
2. Set $X^N_1 = X'_1$,
3. Compute $w^i_1 = \frac{p_{\Theta,1}(X^i_1)}{q_{\Theta,1}(X^i_1)}$, for $i = 1, ..., N$.

Step 2: for $t = 2, ..., T$,

1. Sample $\{a^i_t, X^i_t\} \sim \frac{w^{a_t}_{t-1}}{\sum_j w^j_{t-1}} q_{\Theta,t}(X_t|X^{a_t}_{1:t-1})$, for $i = 1, ..., N$,
2. Set $X^N_t = X'_t$,
3. Compute $\hat{w}^i_{t-1|T} = w^i_{t-1} \frac{p_{\Theta,T}((X^i_{1:t-1}, X'_{t:T}))}{p_{\Theta,t-1}(X^i_{1:t-1})}$, for $i = 1, ..., N$,
4. Draw $a^N_t$ with $P(a^N_t = i) \propto \hat{w}^i_{t-1|T}$,
5. Set $X^i_{1:t} = (X^{a^i_t}_{1:t-1}, X^i_t)$, for $i = 1, ..., N$,
6. Set $w^i_t = \frac{p_{\Theta,t}(X^i_{1:t})}{p_{\Theta,t-1}(X^i_{1:t-1}) q_{\Theta,t}(X^i_t|X^i_{t-1})}$, for $i = 1, ..., N$.

Here $q_{\Theta,t}$ is the proposal conditional on $\Theta$ at time $t$.

### 3.4.3 Fast Universal Self-tuned Sampler Algorithms

As pointed out in Martino et al. (2015), previous algorithms that construct a proposal with support points usually start with a small number of support points and then add new points in case a candidate draw is rejected. However, Martino et al. (2015) present a novel algorithm that constructs effective proposals, starting with a large number of support points and then removing many of them according to some pruning scheme combining relevant information.

#### 3.4.3.1 General Structure of FUSS

The FUSS is an MCMC approach based on an independent proposal distribution, which is built through a simple interpolation procedure. Its general structure is as follows:

1. **Initialisation:** Choose a set of support points, $\mathcal{S}_M = \{s_1, \ldots, s_M\}$, such that $s_1 < s_2 < \ldots < s_M$.

2. **Pruning:** Remove support points according to a pre-specified criterion, attaining a final set $\mathcal{S}_m$, with $m < M$.

3. **Construction:** Build a proposal function $p(x|\mathcal{S}_m)$ given $\mathcal{S}_m$, using some appropriate pre-defined mechanism.

4. **MCMC algorithm:** Perform $K$ steps of the MCMC method using $p(x|\mathcal{S}_m)$ as a proposal PDF, thus yielding a set of samples $\{x_1, \ldots, x_K\}$.

The first three steps are optimisation steps to obtain a good proposal density, tailored to the shape of the target distribution, while step 4 contains the MCMC iterations repeated at $K$ times. In particular, Martino et al. (2015) propose two techniques for step 4, namely the MH algorithm and the Rejection Chain (RC) algorithm, and the resultant FUSS algorithms are named the FUSS-MH and FUSS-RC, respectively. The MH algorithm has been described in Section 3.2, and the structure of the RC algorithm used in step 4 is as follows:

1. Set $k = 0$ and choose $x_0$.

2. Draw $x' \sim \bar{p}(x) \propto p(x|\mathcal{S}_m)$ and $u' \sim \mathcal{U}([0, 1])$.

3. If $u' > \frac{\pi(x')}{p(x'|\mathcal{S}_m)}$, then go back to step 2.

4. If $u' \leq \frac{\pi(x')}{p(x'|\mathcal{S}_m)}$, set $x_{k+1} = x'$ with probability

$$\alpha_{RC} = 1 \wedge \frac{\pi(x')\,[\pi(x_k) \wedge p(x_k|\mathcal{S}_m)]}{\pi(x_k)\,[\pi(x') \wedge p(x'|\mathcal{S}_m)]}. \tag{3.3}$$

   Otherwise, set $x_{k+1} = x_k$ with probability $1 - \alpha_{RC}$.

5. If $k \leq K$, set $k = k + 1$ and repeat from step 2. Otherwise, stop.

The difference between the FUSS-MH and FUSS-RC is that the FUSS-RC contains a rejection sampling (RS) test, and an MH step is applied only when a sample is accepted, thus ensuring that the samples are drawn from the target distribution. Generally, the FUSS-RC is slower than the FUSS-MH, since the chain does not move forward when a sample is rejected in the RS test, but it yields samples with a lower correlation because of the RS test.

It is worth noting that the initial support points in $\mathcal{S}_M$ should cover all the high probability regions of the target $\pi(x)$. In the simulation studies, when the FUSS algorithms are applied to the update of spot variance $\mathbf{v}$, I choose a uniform initial grid of support points:

$$\mathcal{S}_M = \{s_1, s_2 = s_1 + \epsilon, \ldots, s_M = s_1 + (M-1)\epsilon\},$$

where $\epsilon$ is a small constant, and $s_1$ and $s_M$ decide the initial region of the proposal.

In step 3, the FUSS algorithms construct the proposal with the support points. Suppose a set of support points $\mathcal{S}_m = \{s_1, s_2, \ldots, s_m\}$ is obtained after the pruning step, where $s_1 < \ldots < s_m$. We can define the intervals $\mathcal{I}_0 = (-\infty, s_1]$, $\mathcal{I}_j = (s_j, s_{j+1}]$, for $j = 1, ..., m-1$ and $\mathcal{I}_m = (s_m, +\infty)$. The unnormalised proposal is then given by:

$$p(x|\mathcal{S}_m) = e^{W(x)},$$

where $W(x)$ is built using a piecewise constant approximation for all the intervals, with the exception of the first and last intervals (that is, the tails), where a non-constant function is used. Mathematically,

$$W(x) = w_i(x) = \max\left[V(s_i), V(s_{i+1})\right] \mathbb{I}_{\mathcal{I}_i}(x), \tag{3.4}$$

where $1 \leq i \leq m-1$ and

$$\mathbb{I}_{\mathcal{I}_i}(x) = \begin{cases} 1, & x \in \mathcal{I}_i = (s_i, s_{i+1}], \\ 0, & x \neq \mathcal{I}_i = (s_i, s_{i+1}]. \end{cases} \tag{3.5}$$

In the first and last intervals, $\mathcal{I}_0$ and $\mathcal{I}_m$, respectively,

$$W(x) = w_j(x), \quad j \in \{0, m\}, \quad x \in \mathcal{I}_j,$$

where $w_j(x)$ represents a generic log-tail function.

The proposal $\bar{p}(x) \propto p(x|\mathcal{S}_m)$ is composed of $m+1$ pieces, including the two tails. Therefore, $p(x|\mathcal{S}_m)$ can be seen as a finite mixture,

$$p(x|\mathcal{S}_m) = \sum_{i=0}^{m} \eta_i \phi_i(x),$$

with $\sum_{i=0}^{m} \eta_i = 1$, whereas $\phi_i(x) = \exp\left(w_i(x)\right)$ for all $x \in \mathcal{I}_i$, and $\phi_i(x) = 0$ for $x \notin \mathcal{I}_i$. Hence, in order to draw a sample from $\bar{p}(x) \propto p(x|\mathcal{S}_m)$, it is necessary to perform the following steps:

1. Compute the area $A_i$ below each piece, $i = 0, \ldots, m$ and then normalise them,

$$\eta_i = \frac{A_i}{\sum_{j=1}^{m} A_j}, \quad \text{for } i = 0, \ldots, m.$$

2. Choose a piece (that is, an index $j^* \in \{0, \ldots, m\}$) according to the weights $\eta_i$ for $i = 0, \ldots, m$,

3. Given the index $j^*$, draw $x' \sim \bar{\phi}_{j^*}(x) \propto \phi_{j^*}(x) = \exp(w_{j^*}(x))$.

Another issue to address is the pruning step. Martino et al. (2015) propose four different pruning types, and after several tries, I find that the pruning type 4 is the best fit for the problems considered in this thesis. Considering the length of this thesis, I report only the estimation results obtained by the FUSS algorithms with the pruning type 4. Specifically, the algorithm of the pruning type 4 is as follows:

1. Choose a value $\delta > 0$. Given $\mathcal{S}_M = \{s_1, ..., s_M\}$, set $\mathcal{S}^{(0)} = \mathcal{S}_M$, $m = M$, $n = 0$, and

$$L = \max_{1 \le j \le \lfloor \frac{m-1}{2} \rfloor} (s_{2j+1} - s_{2j-1}) |\pi(s_{2j+1}) - \pi(s_{2j-1})|.$$

2. For $r = 1, \ldots, R = \lfloor \frac{m-1}{2} \rfloor$:
   a) Compute $b_r = (s_{2r+1} - s_{2r-1}) |\pi(s_{2r+1}) - \pi(s_{2r-1})|$,
   b) If $b_r \le \delta L$, set $\mathcal{S}^{(r)} = \mathcal{S}^{(r-1)} \setminus \{s_{2r}\}$ and $n = n + 1$,
   c) Otherwise, if $b_r > \delta L$, set $\mathcal{S}^{(r)} = \mathcal{S}^{(r-1)}$.

3. If $n > 0$ set $n = 0$, $\mathcal{S}^{(0)} = \mathcal{S}^{(R)}$, $m = |\mathcal{S}^{(R)}|$ and repeat from step 2.

4. Otherwise, if $n = 0$, return $\mathcal{S}_m = \mathcal{S}^{(R)}$.

## 3.5 Simulation Studies: Data Generation

In the simulation studies, I estimate three models with different jump structures in the log price process by all the above Bayesian methods. The input for the estimation is the simulated one-year log asset prices,[1] and considering the computational cost, I follow Andrieu et al. (2010) and Martino et al. (2015) and simulate one path[2] for each model. Since typically there are 252 trading days a year, I set $T = 252$, and assume the initial log price $Y_0$ and variance $v_0$ as known, and further assume no jump in the log price at $t = 0$.

The true values of parameters and initial conditions are reported as follows:

Initial conditions: $Y_0 = \log(1000)$, $v_0 = 0.02$.

Common parameters: $\kappa = 3$, $\theta = 0.02$, $\rho = -0.8$, $\alpha = 0.3$ in the Heston-0 and Bates-0 models and $\alpha = 0.1$ in the SV-NIG-0 model.[3]

[1] I simulate one-year data with 252 observations for two reasons. First, it is computationally economical. Second, for the Bates-0 model with $\lambda = 0.1$, with the expected number of jumps per year of approximately 25, and for the SV-NIG-0 model with infinite-activity jumps in the log price process, the observations of jumps are sufficient to estimate the jump-related parameters. This is later justified by the reasonable estimation results of the jump-related parameters in the Bates-0 and SV-NIG-0 models, as shown in Tables 3.5 and 3.6.

[2] In further research, it would be important to simulate more paths and base the algorithm comparison on more extensive data sets.

[3] The reason for choosing a smaller $\alpha$ in the SV-NIG-0 is that quite significant volatility has been generated by the infinite-activity jumps in the path of the log asset price, and a higher $\alpha$ may make the variance process and the log price process very volatile with many spiky peaks, and thus too challenging for the algorithms to estimate the models.

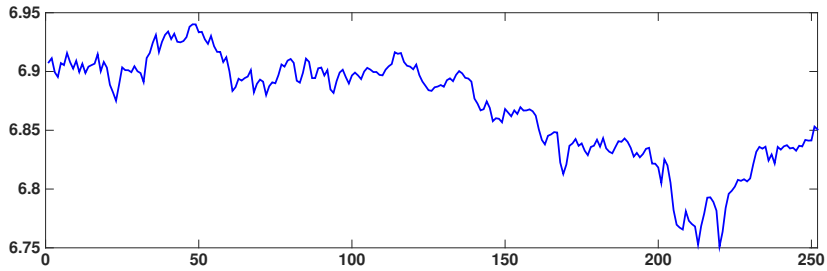


**Figure 3.1:** Simulated one-year path of the daily log asset price under the Heston-0 model.

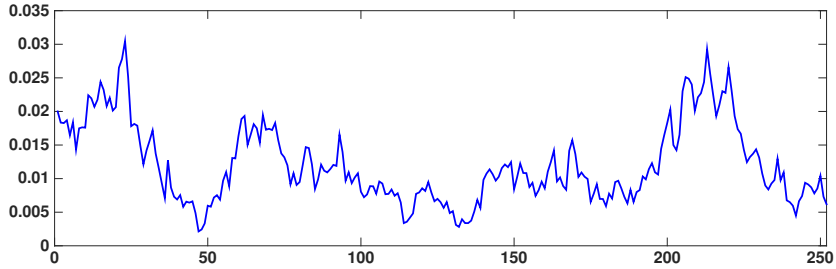


**Figure 3.2:** Simulated one-year path of spot variance under the Heston-0 model.

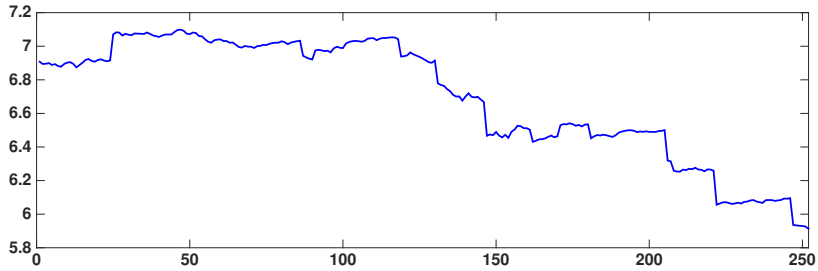


**Figure 3.3:** Simulated one-year path of the daily log asset price under the Bates-0 model.

Parameters specific to the Bates-0 model: $\mu_y = -0.05$, daily $\lambda = 0.1$, $\sigma_y = 0.1$.

Parameters specific to the SV-NIG-0 model: $\gamma = -0.1$, $\sigma = 0.2$, $\nu = 5$.

The paths of the daily log asset price and spot variance[4] simulated under the Heston-0, Bates, and SV-NIG-0 models are plotted in Figures 3.1–3.6.[5]

[4] In the simulation, negative variance processes are discarded.

[5] The set of random numbers is not specified when the paths of the log asset price and variance are simulated. In future research, it would be interesting to use the same set of random numbers such that the simulated paths of variance in all three models are the same.

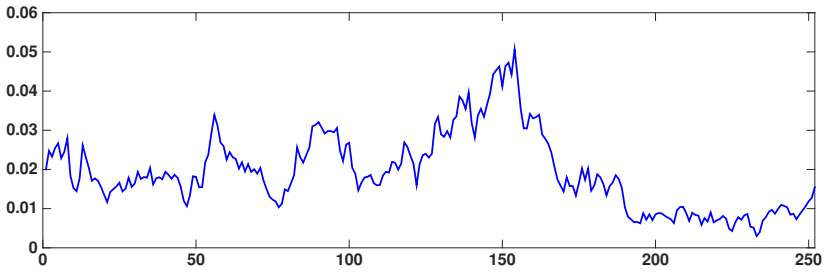


**Figure 3.4:** Simulated one-year path of spot variance under the Bates-0 model.

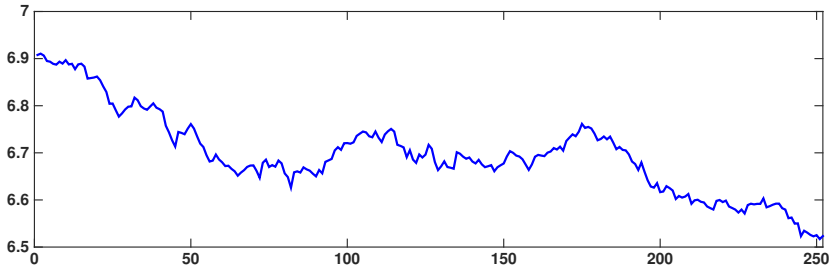


**Figure 3.5:** Simulated one-year path of the daily log asset price under the SV-NIG-0 model.

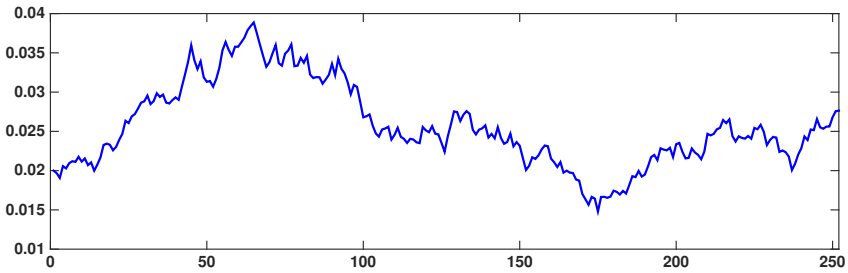


**Figure 3.6:** Simulated one-year path of spot variance under the SV-NIG-0 model.

## 3.6 Implementation of Algorithms

### 3.6.1 Likelihood Inference

Let the observations $\mathbf{Y} = \{Y_t\}_{t\geq 0}$ be the log asset price, and let $\mathbf{v} = \{v_t\}_{t\geq 0}$ be the instantaneous squared volatility of log prices. After the Euler discretisation, under the physical measure $P$, I assume that the transition of $\mathbf{Y}$ and $\mathbf{v}$ is described by the following stochastic differential equations:

$$\begin{aligned} Y_{t+1} - Y_t &= \left(\mu - \frac{1}{2}v_t\right)\Delta + \sqrt{v_t\Delta}\epsilon_{Y,t+1} + J_{Y,t+1}, \\ v_{t+1} - v_t &= \kappa(\theta - v_t)\Delta + \alpha\sqrt{v_t\Delta}\epsilon_{v,t+1}, \end{aligned} \tag{3.6}$$

where $\Delta = 1/252$, $\epsilon_{v,t}$, $\epsilon_{Y,t} \sim N(0,1)$, $corr(\epsilon_{v,t}, \epsilon_{Y,t}) = \rho$.

In the Heston-0 model, $J_{Y,t} = 0, \forall t$. The parameters $\Theta = \{k, \theta, \alpha, \rho, \mu\}$ are unknown to be estimated, and $\mathbf{v}$ is the latent state variable to be extracted.

In the Bates-0 model,

$$J_{Y,t} = \xi_t^Y N_t, P(N_t = 1) = \lambda\Delta. \tag{3.7}$$

Here the unknown parameter set $\Theta = \{k, \theta, \alpha, \rho, \mu, \mu_y, \lambda, \sigma_y\}$, and $\mathbf{v}$, $\mathbf{N}$ and $\xi^{\mathbf{Y}}$ are the latent state variables to be extracted.

In the SV-NIG-0 model,

$$J_{Y,t} = \gamma G_{Y,t} + \sigma\sqrt{G_{Y,t}}\epsilon_t^{J_Y}. \tag{3.8}$$

Here the unknown parameter set $\Theta = \{k, \theta, \alpha, \rho, \mu, \gamma, \nu, \sigma\}$, and $\mathbf{v}$, $\mathbf{G_Y}$ and $\mathbf{J_Y}$ are the latent state variables to be extracted. $\epsilon^{J_Y} \sim N(0,1)$ and is independent of any other random source.

In the Hesotn-0 model, the joint posterior is

$$p(\Theta, \mathbf{v}|\mathbf{Y}) \propto p(\mathbf{Y}, \mathbf{v}|\Theta)p(\Theta) \tag{3.9}$$

$$\propto \prod_{t=0}^{T-1} \frac{1}{\alpha v_t \Delta\sqrt{1-\rho^2}} \exp\left\{-\frac{\epsilon_{Y,t+1}^2 - 2\rho\epsilon_{Y,t+1}\epsilon_{v,t+1} + \epsilon_{v,t+1}^2}{2(1-\rho^2)}\right\} p(\Theta), \tag{3.10}$$

where $p(\Theta)$ is the prior of the parameters, and

$$\epsilon_{Y,t} = \frac{Y_t - Y_{t-1} - \left(\mu - \frac{1}{2}v_{t-1}\right)\Delta}{\sqrt{v_{t-1}\Delta}}, \tag{3.11}$$

$$\epsilon_{v,t} = \frac{v_t - v_{t-1} - \kappa(\theta - v_{t-1})\Delta}{\alpha\sqrt{v_{t-1}\Delta}}. \tag{3.12}$$

The joint posteriors of the Bates-0 and SV-NIG-0 models can be derived similarly as the Heston-0 model, and the extra likelihoods of jump variables can be computed on the basis of the model assumptions in Equations 3.7 and 3.8.

In the estimation, I cannot directly generate samples from the joint posterior, because it is not of a known form. Therefore, I break the joint posterior into conditionals of parameters and latent state variables, and sample from the conditionals with a Gibbs step, or the MH and more advanced algorithms, such as the AM and FUSS algorithms, if sampling from one or some of the conditionals is impossible. Alternatively, I use the PMMH to approximate the marginal posterior of parameters or the PGAS to approximate the joint posterior.

Before the estimation, I need to choose appropriate priors for the parameters. I choose conjugate priors to make the conditional posteriors of known forms if possible. The priors should be uninformative so that the posteriors are dominated by the likelihoods.[6] The

[6]The prior is a part of the model specification. If we have strong prior information regarding the distribution of a parameter, we should use an informative prior for this parameter so that the posterior is not too dispersed and sampling from the posterior would be easier. For example, one may observe that jumps in the path of the log asset price are usually large and negative; then it would be sensible to assume a normal distribution with negative mean and large variance as the prior for the parameter controlling the jump size.

selected priors are reported as follows:

$$p(k) = N(4, 3^2), p(\theta) = N(0.05, 0.1^2), p(\mu) = N(0, 1^2).$$

The parameter pair $(\alpha, \rho)$ is re-parameterised as in Jacquier et al. (2002) to improve the efficiency and to rule out the large correlation between $\alpha$ and $\rho$: $\phi_v = \alpha\rho$ and $w_v = \alpha^2(1-\rho^2)$. The priors for $\phi_v$ and $w_v$ are conjugate: $\phi_v|w_v \sim N(0, \frac{1}{2}w_v)$ and $w_v \sim IG(2, 200)$.

When the models are estimated with the PMMH, I set the priors for $\rho$ and $\alpha$ as:

$$p(\alpha) = \frac{1}{\alpha}, p(\rho) = \text{Uniform}(-1, 1).$$

In the Bates-0 model, I assume priors for the model-specific parameters as:

$$p(\mu_y) = N(0, 1^2), p(\lambda) = \text{Uniform}(0, 1), p(\sigma_y) = \frac{1}{\sigma_y}.$$

In the SV-NIG-0 model, I assume priors for the model-specific parameters as:

$$p(\gamma) = N(0, 1^2), p(\nu) = N(6, 5^2), p(\sigma) = \frac{1}{\sigma}.$$

Except in the PMMH where the candidate draw of the parameter set $\Theta^*$ is generated by the proposal $q(\cdot|\Theta^{(i-1)})$, with the other estimation methods, the parameters are sampled from their conditional posteriors. Specifically, the conditional posteriors of the parameters $\kappa$, $\theta$, $\phi_v$, $\mu$ in all three models, $\mu_y$ in the Bates-0 model, and $\gamma$ and $\nu$ in the SV-NIG-0 model are normal distributions, the conditional posteriors of $w_v$, $\sigma_y$ in the Bates-0 model and $\sigma$ in the SV-NIG-0 model are inverse Gamma distributions, and the conditional posterior of $\lambda$ in the Bates-0 model is a Beta distribution. Thus, all the parameters are updated by a Gibbs step.

As for the extraction of the latent state variables, in the Bates-0 model, the conditional posterior of $\mathbf{N}$ is a Bernoulli distribution and that of $\xi$ is a normal distribution. In the SV-NIG-0 model, the conditional posterior of $\mathbf{J_Y}$ is a normal distribution. Consequently, these variables are updated with a Gibbs step. However, because the conditional posteriors of $\mathbf{v}$ in all three models and $\mathbf{G}$ in the SV-NIG-0 model are of complex forms, the Gibbs sampler is not applicable.

After several estimation experiments, I notice that when the jump variable $\mathbf{G}$ is updated by the random-walk MH, the acceptance rate is reasonable, ranging from 15% to 30% with different proposals and models, and the application of the AM and FUSS algorithms to the update of $\mathbf{G}$ seems to make little improvement. Therefore, considering the computational costs, I only apply advanced MCMC methods to update $\mathbf{v}$, and use the random-walk MH with the proposal $G_t^{(i)} = N\left(G_t^{(i-1)}, (1 \times 10^{-3})^2\right), \forall t$, to sample $\mathbf{G}$. Note that $\mathbf{G}$ should be positive, and to retain the positiveness, I repeatedly sample from the proposal until a non-negative value is obtained. In this way, a rejection sampler is actually implemented to sample from the proposal distribution truncated at zero. The truncated density has the same density as the un-truncated one, apart from the differences in supports and

normalising constants. When the random-walk MH[7] is applied to the update of $\mathbf{v}$, the trick of repeated sampling is also used to ensure the positiveness of $\mathbf{v}$.

When the AM is applied, I follow the practice in Haario et al. (1999b, 2001) and set $s = 2.4^2$ and $\epsilon = 1 \times 10^{-4}$. In particular, instead of adapting the proposal at each MCMC run, I adapt the proposal at every 100 runs after the initial non-adaptation period of 1,000 runs.

When the FUSS algorithms, including the FUSS-RC and FUSS-MH, are applied, I use type 4 as the pruning scheme and set $s_1 = 0.0001$, $s_M = 1$, $\Delta_{s_i} = s_{i+1} - s_i = 0.0001$, for $i = 1, ..., M-1$, $\delta = 0.01$, and $K = 3$. The use of the pruning type 4 is motivated by the findings in Martino et al. (2015) that it is the best pruning type because it selects the final set of support points according to a minimax optimality criterion, leading to better-quality estimators, a lower correlation between samples, and less computation time, especially when the target distribution is multimodal with narrow modes. In the estimation, I try all four pruning types and find that the type 4 outperforms the other pruning types.

In the PF update of the PMMH, I sample the particles $\{v_t^i\}$ from $v_t^i = v_{t-1}^{A_i} + \kappa(\theta - v_{t-1}^{A_i})\Delta + \alpha\sqrt{v_{t-1}^{A_i}\Delta}\epsilon$, where $A_i \sim q_t(A) \propto w_{t-1}^i$ and $\epsilon$ is a standard normal random number. The particles of the jump variables $\mathbf{N}$ and $\xi^{\mathbf{Y}}$ in the Bates-0 model and $\mathbf{G}$ in the SV-NIG-0 model are drawn according to model assumptions, and the particles of $\mathbf{J_Y}$ are updated by $J_{Y,t} = N_t \xi_t^Y, \forall t$, in the Bates-0 model and by $J_{Y,t} = \gamma G_t + \sqrt{\sigma G_t}\epsilon^J, \forall t$, in the SV-NIG-0 model, where $\epsilon^J$ is a standard normal random number.

The APF in the PMMH is implemented like the PF in the PMMH. The only difference is that I first sample $\hat{v}_t^i = v_{t-1}^i + \kappa(\theta - v_{t-1}^i)\Delta$ without the diffusion term and the sampling of index $A_i$. Next, I sample the index $k^i$ for $v_{t-1}^{k^i}$ according to the likelihood of $Y_t$ conditional on $\hat{v}_t^i$ and the weights computed at the previous time step, and then resample $v_t^{k^i}$ according to its tradition density. It is worth mentioning that when the variance variable $\mathbf{v}$ is resampled, the jump variables are associated with $k^i$, but they are not resampled because of the independence of the jumps.

The implementation of the PF in the PGAS: After the particles at time $t$ have been generated, a key step in the PGAS is to sample the ancestor step for the reference trajectory of $\mathbf{v}$. Specifically, the weights for the ancestor step are computed by:

$$\hat{w}_{t-1|T}^i \propto w_{t-1}^i \frac{1}{\alpha\sqrt{v_{t-1}^i}} \exp\left(-\frac{(v_t^N - v_t^i)^2}{2\alpha^2 v_{t-1}^i \Delta}\right) \tag{3.13}$$

Concerning the update of the parameter set in the PMMH, while reading the attached discussion in Andrieu et al. (2010), I noticed an interesting suggestion by Professors Michael Johannes, Nick Polson, and Seung Min Yae that the update step in the PMMH algorithm can be modified as follows:

Step One: Generate a full vector of latent state variables, $X_{1:T}$, by an SMC update and accept or reject these draws by a Metropolis update;

[7]The proposal is a normal distribution with variance $1 \times 10^{-4}$. It is possible to increase the acceptance rate of $v$ by lowering the variance of the proposal. However, there seems to be a trade-off between the acceptance rate and the moving speed of the chain. Considering the reasonable value interval of $v$, $1 \times 10^{-4}$ is quite small as the variance of the proposal, and further reduction of the variance may worsen the mixing speed of the chain.

Step Two: Update the parameters according to $p(\Theta|X_{1:T}, Y_{1:T})$.

I seek to study the performance of this modified PMMH algorithm combined with the PF as the SMC update in this thesis. In particular, the modified PMMH-PF is labeled PMMH-PF2, and the original version is labeled PMMH-PF1.

Finally, I need to decide the length of the Markov chain as well as the length of the burn-in period and the number of particles in the PMCMC methods. Suppose $M$ is the length of the Markov chain, $M_0$ is the length of the burn-in period, and $N$ is the number of particles. After a number of estimation experiments, I observe that when estimating different models, the same algorithm may produce different mixing performance. Therefore, the algorithmic settings, including $M$, $M_0$, and $N$, may differ with different models. The details of the algorithmic settings are reported in Tables 3.1 – 3.3. Basically, according to the choice of $M$ and $N$, the simulation experiments are divided into two groups: to illustrate the estimation performance with a small number of MCMC iterations and a small particle set, and to obtain a *stabilised* performance.[8]

## 3.7 Results

First of all, it is important to point out that assessing the numerical accuracy of MCMC algorithms is very difficult. We use MCMC algorithms because we cannot directly sample from the target distribution $p(\Theta, X|Y)$; instead, we use MCMC algorithms to approximate the target distribution and examine the performance of algorithms on the basis of the approximate distribution $\hat{p}(\Theta, X|Y)$. For example, in Subsection 3.7.1, I compute the estimation error as the mean squared error of the approximate $\hat{E}(v|Y)$ and the simulated variance $v$, and in Subsection 3.7.4, I compare the performance of algorithms in estimating parameters on the basis of the distance between the approximate $\hat{E}(\Theta|Y)$ and the preset parameters $\Theta$.

However, ideally, the comparison should be based on the distance between the *true distribution* and the approximate distribution, that is, $\hat{E}(v|Y)$ and $E(v|Y)$, and $\hat{E}(\Theta|Y)$ and $E(\Theta|Y)$. However, we do not know the true distribution; therefore, in this thesis, I base the comparison on the approximate distribution, the simulated variance, and preset parameters, following practices in the literature.

### 3.7.1 Performance of Algorithms with Different Settings: Acceptance Rate, Estimation Error, Log-Likelihood, and Computation Time

Tables 3.1 – 3.3 report the estimation results of different algorithms with the Heston-0, Bates-0, and SV-NIG-0 models. The results comprise the acceptance rate (AR) of the

[8] The equilibrium distribution of the Markov chain generated by the MCMC algorithms converges to the target distribution as the chain length goes to infinity. In this sense, the larger the number of MCMC iterations ($M$), the better the performance. However, the improvement brought by increasing $M$ may become negligible when $M$ is sufficiently large. Similarly, for the PMCMC methods, increasing $N$ significantly increases the acceptance rate and reduce the estimation error; however, in the simulation tests, I observe that there seems to be an optimal $N$, after which a further increase of $N$ provides a negligible improvement. In the estimation, I seek to find out the sufficiently large $M$ and $N$, and refer to the results as "stabilised results".

latent state variable $\mathbf{v}$, the mean squared error of $\mathbf{v}$ ($\text{MSE}_v$), the log-likelihood (LL) of the parameters and state variables, and the computation time.[9]

It is worth mentioning that the results in Tables 3.1 – 3.3 are estimated using the simulated paths of one-year log asset prices, as shown in Figures 3.1, 3.3, and 3.5, and computed on the basis of 100 experiments, except that the computationally costliest estimation experiments for each algorithm are conducted only 10 times.[10]

First, I compare the performance of the algorithms in the Heston-0, Bates-0, and SV-NIG-0 models, respectively.

In estimatingthe Heston-0 model, all the algorithms achieve comparably good performance in terms of $\text{MSE}_v$. In particular, with the efficient adaptive proposal in the AM, the required length of the chain to reach a stabilised performance, that is, $M$, is reduced from 100,000 in the MH to 20,000, thanks to the significantly higher AR. However, $M$ becomes even smaller in more efficient algorithms, including the FUSS algorithms with the MH or RC step and the PMCMC algorithms. An important observation about the FUSS algorithms is that, unlike the others, they can construct a proposal very close to the target distribution and significantly improve the mixing speed of the chain and, thus, $M$ can be very small. In fact, the performance of the FUSS algorithms with $M = 1,000$ is almost as remarkable as that with $M = 5,000$. Moreover, the length of the burn-in period, that is, $M_0$, is very small, indicating that the FUSS algorithms are very efficient in that most of the computation work is useful. Among the PMCMC methods employed to estimate the Heston-0 model, the three versions of PMMH show comparable performance, with the PMMH-PF2 slightly better than the PMMH-PF1/APF in terms of a lower $\text{MSE}_v$ and a higher LL, but generally the PMMH algorithms underperform the PGAS. Surprisingly, with only 500 MCMC iterations and 500 particles, the PGAS performs better than the PMMH algorithms with 5,000 iterations and 1,000 particles in terms of $\text{MSE}_v$. This suggests that the mixing of the PGAS kernel is very fast even with a small number of particles.

In estimating the Bates-0 model,[11] the performance of different algorithms varies. When the MH and AM obtain similar $\text{MSE}_v$, the difference in $\text{MSE}_v$ between {AM, MH} and {FUSS, PMMH-PF1} is significant. This observation shows that when the model specification becomes complex, the MH and AM are less capable of extracting the variance. In the experiments, I notice that the performance of the MH and AM are sensitive to the initial value of the variance, and prior information about the variance may remarkably improve the estimation performance.[12] Further, the FUSS algorithms achieve a stabilised performance with 5,000 MCMC iterations, as with the Heston-0 model estimation, but the required number of particles in the PMMH-PF1 significantly increases from 1,000 to 5,000 for a reasonable AR, a small $\text{MSE}_v$, and a high LL comparable to the FUSS algorithms. Surprisingly, the PGAS, which performs best in estimating the Heston-0 model, performs worst among all algorithms in estimating the Bates-0 model.

[9] The computation time is the running time for the whole algorithm, normalised with respect to the time required by the MH algorithm with $M = 100,000$.

[10] The main reason for reducing the number of experiments is the much longer computation time. The reduction is justified by the observation that when $M$ and $N$ are sufficiently large, no significant differences are produced in $\text{MSE}_{\mathbf{v}}$, AR, and LL, using the same algorithm.

[11] When estimating the Bates-0 and SV-NIG-0 models, I notice that the PMMH-PF2 underperforms the PMMH-PF1, and the PMMH-APF fails to achieve reasonable performance; therefore, the results of the PMMH-APF are omitted, and selected results of the PMMH-PF2 are reported.

[12] In the estimation, I use $v_t = 0.02$, $\forall t$, as the initial value of $\mathbf{v}$. However, if the initial value is replaced with a path positively correlated with the true path, the MH and AM achieve much better estimation performance.

**Table 3.1:** Estimation results of the Heston-0 model with different numbers of MCMC iterations (M) and particles (N) in the PMCMC methods. This table shows the acceptance rate (AR), the mean squared error of spot variance $\mathbf{v}$ ($\text{MSE}_v$), the log-likelihood (LL) of the parameters and latent state variables, and the computation time. The computation time is the running time for the whole algorithm, normalised with respect to the time required by the MH algorithm with $M = 100,000$. The results presented in this table are computed on the basis of 100 experiments, except that the estimation experiments computationally the costliest for each algorithm are conducted 10 times.

| | $M_0$ | $M$ | $N$ | AR (%) | $\text{MSE}_v$ | LL | Time |
|---|---|---|---|---|---|---|---|
| MH | 4000 | 5000 | | 4.38 | 3.42E-05 | 888.3 | 0.04 |
| | 15000 | 20000 | | 4.48 | 2.56E-05 | 900.2 | 0.17 |
| | 80000 | 100000 | | 3.93 | 1.30E-05 | 1111.5 | 1 |
| AM | 4000 | 5000 | | 45.53 | 1.33E-05 | 1037.1 | 0.05 |
| | 15000 | 20000 | | 54.60 | 1.27E-05 | 1082.3 | 0.21 |
| FUSS-RC-P4 | 150 | 200 | | 100.00 | 1.99E-05 | 1010.8 | 0.25 |
| | 300 | 500 | | 100.00 | 1.41E-05 | 1090.2 | 0.71 |
| | 300 | 1000 | | 100.00 | 1.24E-05 | 1204.5 | 1.93 |
| | 1000 | 5000 | | 100.00 | 1.20E-05 | 1232.0 | 19.79 |
| FUSS-MH-P4 | 100 | 200 | | 100.00 | 1.68E-05 | 1061.9 | 0.28 |
| | 300 | 500 | | 99.98 | 1.28E-05 | 1144.7 | 0.69 |
| | 300 | 1000 | | 99.97 | 1.26E-05 | 1150.6 | 1.83 |
| | 1000 | 5000 | | 99.93 | 1.24E-05 | 1166.1 | 17.53 |
| PMMH-PF1 | 400 | 500 | 500 | 29.60 | 1.66E-05 | 1107.6 | 0.31 |
| | 400 | 500 | 1000 | 33.96 | 1.64E-05 | 1104.7 | 0.38 |
| | 4000 | 5000 | 500 | 24.58 | 1.33E-05 | 1102.4 | 3.24 |
| | 4000 | 5000 | 1000 | 34.86 | 1.24E-05 | 1108.4 | 3.53 |
| | 4000 | 20000 | 1000 | 33.79 | 1.24E-05 | 1113.0 | 16.14 |
| PMMH-PF2 | 400 | 500 | 500 | 19.95 | 1.49E-05 | 1107.6 | 0.40 |
| | 400 | 500 | 1000 | 24.73 | 1.43E-05 | 1152.4 | 0.43 |
| | 4000 | 5000 | 500 | 16.13 | 1.30E-05 | 1152.5 | 4.19 |
| | 4000 | 5000 | 1000 | 23.93 | 1.25E-05 | 1157.4 | 4.61 |
| | 4000 | 20000 | 1000 | 24.55 | 1.21E-05 | 1177.4 | 18.36 |
| PMMH-APF | 400 | 500 | 500 | 33.09 | 1.58E-05 | 858.2 | 0.43 |
| | 400 | 500 | 1000 | 19.33 | 1.45E-05 | 885.2 | 0.47 |
| | 4000 | 5000 | 500 | 8.74 | 1.69E-05 | 903.3 | 4.42 |
| | 4000 | 5000 | 1000 | 16.88 | 1.26E-05 | 909.6 | 4.88 |
| | 4000 | 20000 | 1000 | 16.21 | 1.26E-05 | 910.4 | 19.81 |
| PGAS | 400 | 500 | 500 | 100.00 | 1.23E-05 | 1046.7 | 0.07 |
| | 400 | 500 | 1000 | 100.00 | 1.16E-05 | 1059.5 | 0.12 |
| | 4000 | 5000 | 500 | 100.00 | 1.15E-05 | 1047.1 | 0.70 |
| | 4000 | 5000 | 1000 | 100.00 | 1.13E-05 | 1060.0 | 1.20 |
| | 4000 | 5000 | 5000 | 100.00 | 1.08E-05 | 1095.0 | 5.54 |
| | 10000 | 50000 | 5000 | 100.00 | 1.07E-05 | 1092.9 | 45.13 |

**Table 3.2:** Estimation results of the Bates-0 model with different numbers of MCMC iterations (M) and particles (N) in the PMCMC methods. This table shows the acceptance rate (AR), the mean squared error of spot variance $\mathbf{v}$ ($\text{MSE}_v$), the log-likelihood (LL) of the parameters and latent state variables, and the computation time. The computation time is the running time for the whole algorithm, normalised with respect to the time required by the MH algorithm with $M = 100,000$. The results presented in this table are computed on the basis of 100 experiments, except that the estimation experiments computationally the costliest for each algorithm are conducted 10 times.

| | $M_0$ | $M$ | $N$ | AR(%) | $\text{MSE}_v$ | LL | Time |
|---|---|---|---|---|---|---|---|
| MH | 4000 | 5000 | | 11.62 | 4.24E-05 | 975.7 | 0.05 |
| | 15000 | 20000 | | 7.72 | 4.04E-05 | 1033.9 | 0.22 |
| | 80000 | 100000 | | 8.54 | 2.59E-05 | 1055.9 | 1 |
| AM | 4000 | 5000 | | 66.60 | 8.91E-05 | 1050.2 | 0.07 |
| | 15000 | 20000 | | 64.16 | 2.61E-05 | 1107.5 | 0.30 |
| FUSS-RC-P4 | 150 | 200 | | 99.88 | 7.39E-05 | 1015.1 | 0.21 |
| | 300 | 500 | | 99.95 | 4.68E-05 | 1066.5 | 0.40 |
| | 300 | 1000 | | 99.98 | 2.46E-05 | 1170.7 | 0.98 |
| | 1000 | 5000 | | 99.98 | 1.62E-05 | 1192.2 | 10.24 |
| FUSS-MH-P4 | 150 | 200 | | 100.00 | 4.32E-05 | 1015.0 | 0.17 |
| | 300 | 500 | | 99.96 | 4.42E-05 | 1062.6 | 0.45 |
| | 300 | 1000 | | 99.80 | 1.96E-05 | 1113.1 | 1.18 |
| | 1000 | 5000 | | 100.00 | 1.87E-05 | 1123.4 | 10.13 |
| PMMH-PF1 | 400 | 500 | 500 | 5.27 | 5.14E-05 | 956.3 | 0.24 |
| | 400 | 500 | 1000 | 5.47 | 4.29E-05 | 1042.5 | 0.31 |
| | 400 | 500 | 5000 | 15.30 | 2.50E-05 | 1064.2 | 0.55 |
| | 4000 | 5000 | 500 | 1.27 | 5.35E-05 | 1015.6 | 0.43 |
| | 4000 | 5000 | 1000 | 1.56 | 2.39E-05 | 1031.8 | 0.65 |
| | 4000 | 5000 | 5000 | 21.24 | 1.89E-05 | 1152.3 | 2.40 |
| PMMH-PF2 | 4000 | 5000 | 1000 | 1.10 | 2.70E-05 | 962.8 | 0.67 |
| PGAS | 4000 | 5000 | 500 | 97.28 | 7.43E-05 | 1067.8 | 0.55 |
| | 4000 | 5000 | 1000 | 98.16 | 6.19E-05 | 1078.5 | 0.92 |
| | 4000 | 5000 | 5000 | 99.34 | 3.43E-05 | 1088.5 | 3.73 |
| | 4000 | 20000 | 5000 | 98.29 | 3.46E-05 | 1091.4 | 14.00 |

**Table 3.3:** Estimation results of the SV-NIG-0 model with different numbers of MCMC iterations (M) and particles (N) in the PMCMC methods. This table shows the acceptance rate (AR), the mean squared error of spot variance $\mathbf{v}$ ($\text{MSE}_v$), the log-likelihood (LL) of the parameters and latent state variables, and the computation time. The computation time is the running time for the whole algorithm, normalised with respect to the time required by the MH algorithm with $M = 100,000$. The results presented in this table are computed on the basis of 100 experiments, except that the estimation experiments computationally the costliest for each algorithm are conducted 10 times.

| | M0 | M | N | AR (%) | $\text{MSE}_v$ | LL | Time |
|---|---|---|---|---|---|---|---|
| MH | 4000 | 5000 | | 12.82 | 2.12E-05 | 927.02 | 0.05 |
| | 15000 | 20000 | | 11.82 | 1.87E-05 | 942.82 | 0.27 |
| | 80000 | 100000 | | 37.91 | 1.77E-05 | 1017.70 | 1 |
| AM | 4000 | 5000 | | 33.22 | 2.19E-05 | 922.88 | 0.14 |
| | 15000 | 20000 | | 56.07 | 1.58E-05 | 953.88 | 0.69 |
| | 40000 | 50000 | | 56.94 | 1.55E-05 | 966.95 | 2.71 |
| FUSS-RC-P4 | 150 | 200 | | 100.00 | 2.61E-05 | 898.86 | 0.12 |
| | 300 | 500 | | 100.00 | 2.17E-05 | 918.43 | 0.32 |
| | 300 | 1000 | | 100.00 | 2.20E-05 | 926.36 | 0.89 |
| | 1000 | 5000 | | 100.00 | 1.15E-05 | 955.64 | 10.35 |
| | 3000 | 10000 | | 100.00 | 8.31E-06 | 1022.5 | 33.79 |
| FUSS-MH-P4 | 150 | 200 | | 100.00 | 2.46E-05 | 974.94 | 0.12 |
| | 300 | 500 | | 100.00 | 2.14E-05 | 979.08 | 0.34 |
| | 300 | 1000 | | 100.00 | 1.63E-05 | 938.97 | 0.87 |
| | 1000 | 5000 | | 100.00 | 1.46E-05 | 983.45 | 10.24 |
| | 3000 | 10000 | | 100.00 | 1.31E-05 | 991.81 | 33.71 |
| PMMH-PF1 | 400 | 500 | 500 | 35.30 | 1.97E-05 | 963.23 | 0.36 |
| | 400 | 500 | 1000 | 49.86 | 1.47E-05 | 1020.15 | 0.39 |
| | 4000 | 5000 | 500 | 36.01 | 1.14E-05 | 1006.39 | 3.56 |
| | 4000 | 5000 | 1000 | 45.37 | 9.00E-06 | 1019.26 | 3.66 |
| | 10000 | 20000 | 1000 | 50.20 | 7.63E-06 | 1026.49 | 11.77 |
| PMMH-PF2 | 4000 | 5000 | 1000 | 46.42 | 1.21E-05 | 1002.80 | 3.00 |
| | 10000 | 20000 | 1000 | 58.60 | 1.19E-05 | 1001.20 | 13.83 |
| PGAS | 400 | 500 | 500 | 98.67 | 2.25E-05 | 998.84 | 0.06 |
| | 400 | 500 | 1000 | 98.84 | 1.31E-05 | 1008.69 | 0.09 |
| | 4000 | 5000 | 500 | 99.50 | 2.15E-05 | 997.45 | 0.50 |
| | 4000 | 5000 | 1000 | 99.71 | 6.53E-06 | 1013.50 | 0.74 |
| | 4000 | 5000 | 5000 | 99.86 | 1.01E-05 | 998.49 | 3.09 |
| | 10000 | 20000 | 1000 | 99.80 | 6.24E-06 | 1027.51 | 1.67 |

According to the estimation results of the SV-NIG-0 model, the $\text{MSE}_v$ of the MH is the largest; however, the LL obtained by the MH ranks the fourth highest, implying an estimation result more consistent with the data than the AM with 20,000 MCMC iterations. To investigate, I increase $M$ for the AM to 50,000; however, its performance is comparable to that with $M = 20,000$. The FUSS algorithms significantly outperform the MH and AM, but with a more complex model specification, the optimal $M$ for the FUSS algorithms significantly increases. Furthermore, even with the optimal $M$, the FUSS algorithms still slightly underperform the PMMH-PF1 and PGAS, the latter being the best algorithm again in estimating the SV-NIG-0 model, as with the Heston-0 model estimation.

Moreover, the computation times of different algorithms in estimating the Heston-0, Bates-0, and SV-NIG-0 models are reported in Tables 3.1 – 3.3. In each table, the computation time is normalised with respect to the time required by the MH with $M = 100,000$. Generally, the computation time increases proportionally to the increase of $M$, whereas it changes more slowly than the increase of $N$ in the PMCMC methods. As expected, with a fixed $M$, the MH is the fastest, the AM ranks the second fastest, followed by the PMCMC methods, and the FUSS algorithms are the slowest. In particular, the FUSS-RC is slightly slower than the FUSS-MH owing to the RS test in the FUSS-RC, and the computation time of the PGAS is only about 30% of that of the PMMH algorithms with the same $M$ and $N$. However, if we focus on achieving comparable estimation results, in estimating the Heston-0 and Bates-0 models, the FUSS algorithms are more efficient than the others; however, the advantage of the FUSS algorithms over the AM and MH becomes smaller when estimating the complicated SV-NIG-0 model. Furthermore, although its performance is poor in estimating the Bates-0 model, the PMMH-PF1 is very competitive in estimating the SV-NIG-0 model. Specifically, the PMMH-PF1 takes about 40% of the computation time of the MH to outperform the MH and the AM, and about 30% of the computation time of the FUSS-RC to beat the FUSS-RC. More importantly, the PGAS is extremely efficient in estimating the SV-NIG-0 model, because it only takes 74% of the computation time of the MH to beat all the other algorithms.

Next, I analyse how different jump structures affect the estimation performance.

First, I discuss how jump structures change the AR. The ARs of the MH in estimating the Heston-0 and Bates-0 models are very low; however, the AR becomes reasonable in estimating the SV-NIG-0 model. The ARs of the AM, FUSS, and PGAS algorithms are stable with different models. For the Bates-0 model, the AR of the PMMH-PF1 is very low.[13] A possible reason for the low AR of the PMMH-PF1 in estimating the Bates-0 model is that, although the simulated compound Poisson jumps are rare events, their jump sizes are significant; therefore, in the estimation, they play an important role in computing likelihoods and in deciding the weights of the particles. However, since I have no prior information of the jumps, and since the jump times and sizes are independent of each other, the particles generated for the jumps are highly random. Therefore, only particles with "correct" jump times and sizes are assigned reasonable weights and most of the generated particles are actually useless, which largely affects the performance of the SMC update. On the other side, in the SV-NIG-0 model with infinite-activity jumps, where the jumps are on average less significant and, therefore, less distinguishable

[13] In the estimation, I use the random-walk normal distribution with a diagonal covariance matrix as the proposal. I have also tried to use other covariance matrices by running the estimation for some iterations, computing the covariance of the parameters, and setting the covariance matrix as the proposal covariance matrix. However, this step produces negligible improvement.

from diffusions, even if the particles of the jumps are unrealistic, they may be assigned reasonable weights, because "wrong jumps" can be offset by diffusions. In other words, the impact of the jumps in the SV-NIG-0 model on the weights of the particles is less significant than that of the compound Poisson jumps in the Bates-0 model.

Another observation is that when the performance of the MH, AM, FUSS-RC, and FUSS-MH is stable across different models, the performance of the PMCMC algorithms seems mixed:

1. Comparison of the PMMH-PF1 and the PMMH-PF2.

When estimating the Heston-0 model, the PMMH-PF1 yields a poorer performance than the PMMH-PF2, which is plausible because the samples of the parameters in the latter are updated on the basis of the updated variance and data. However, when estimating the Bates-0 and SV-NIG-0 models, the PMMH-PF2 underperforms the PMMH-PF1. In fact, in the simulation studies, I observe that the performance of all the PMCMC methods is very sensitive to the parameter $\alpha$, because it is important in generating particles of $\mathbf{v}$, and with a higher $\alpha$, the particles of $\mathbf{v}$ may vary significantly. Further, the particles of the jumps may be quite varied as well, especially for the Bates-0 model, and compared to the Heston-0 model without jumps, jumps in the Bates-0 model and SV-NIG-0 model play a relatively important role in the likelihoods. Therefore, fewer particles are needed to create a diverse combination of jumps and variance $\mathbf{v}$ with a higher $\alpha$. However, a higher $\alpha$ may not be consistent with the data, and the PMMH-PF2 seems to produce a less significant $\alpha$ than the PMMH-PF1, as confirmed by the estimation results of the parameters in the Heston-0 model.

2. Comparison of the PF and the APF used in the PMMH.

When estimating the Heston-0 model, the PMMH-APF obtains an estimation performance comparable to that of the PMMH-PF1/PF2; however, when estimating the other two models, its estimation performance is rather poor. The reason for the failure of the APF to estimate the Bates-0 and SV-NIG-0 models is the inclusion of independent jumps, which makes the advantage of the APF become a handicap. The key idea of the APF does not work for models with independent random sources. In the estimation with the PMMH-APF, I first predict the variance at current time $t$ on the basis of the particles of variance at $t-1$ without the random diffusion term, and simulate the particles of the jumps without any prior information, and then choose the most likely particles of variance at $t-1$ according to the likelihood of the observation at time $t$ and weights at $t-1$. Second, I re-predict the variance at $t$ with the random diffusion term, and choose the simulated jumps with the indices of the selected particles, rather than re-simulating the jumps, because the jumps are independent of the past track and re-simulating them may affect the effect of auxiliary variables. Therefore, the independence of the jumps reduces the effectiveness of the APF.

3. Comparison of the PMMH and the PGAS.

Although the PGAS is very similar in design to the PMMH-PF2, they differ in the step of updating particles of the variance and jumps. In the estimation using the PMMH-PF2, I use the PF to generate the particles, whereas in the PGAS, the update of particles uses some prior information embodied in the reference trajectory. In addition, unlike the PG sampler, after each time step in the SMC update, a new trajectory is obtained by an ancestor sampling step, which connects part of the reference trajectory, $X'_{t:T}$, with one of the generated particle trajectories, $X^i_{1:t-1}$, $i \in \{1, ..., N\}$, with probabilities given by their importance weights. As pointed out in Lindsten et al. (2014), the PGAS is

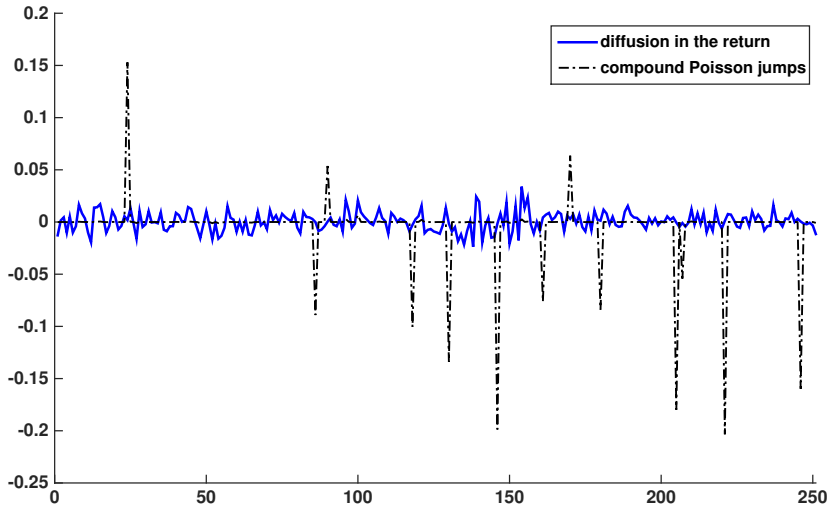


**Figure 3.7:** Randomness sources in the Bates-0 model.

particularly competitive in extracting a high-dimensional and highly autocorrelated state trajectory. This is confirmed by the superior estimation results of the PGAS in estimating the Heston-0 and SV-NIG-0 models, and its performance with only a small number of MCMC iterations and a small set of particles is better than the other algorithms with a large number of MCMC iterations and particles.

Indeed, there is a remarkable difference between the PMMH and PGAS, although both belong to the PMCMC methods. In particular, just as the PMMH algorithm can be regarded an approximation of the ideal marginal MH sampler targeting $p(\Theta|Y_{1:T})$, the PGAS can be regarded an approximation of the ideal Gibbs sampler targeting $p(\Theta, X_{1:T}|Y_{1:T})$. Therefore, as pointed out in Lindsten et al. (2014), the respective performances of the PMMH and PGAS depend on the properties of the marginal MH sampler and Gibbs sampler and, therefore, depend heavily on the properties of specific problems. For the Heston-0 and SV-NIG-0 models, the ideal Gibbs sampler seems to have better mixing properties than the marginal MH sampler.

4. Comparison of the PMCMC methods and other MCMC methods.

Generally, compared to the MH, the use of adaptive proposals in the AM can significantly raise the AR, thus reducing $M$ to achieve a stabilised performance. However, as the model specification becomes complex, the MH and AM underperform the algorithms that are able to deal with a complicated target distribution and strong dependence between parameters and state variables. In particular, when estimating all three models, the FUSS-RC outperforms the FUSS-MH, because of the RS step in the FUSS-RC.

Although the PMCMC methods achieve comparably good estimation performance as the FUSS algorithms in estimating the Heston-0, their performance deteriorates in estimating the Bates-0 model with rare large jumps included, even when the number of particles and length of the chain are quite large. However, the PMCMC algorithms beat the others in estimating the SV-NIG-0 model. The reason is the different features of the

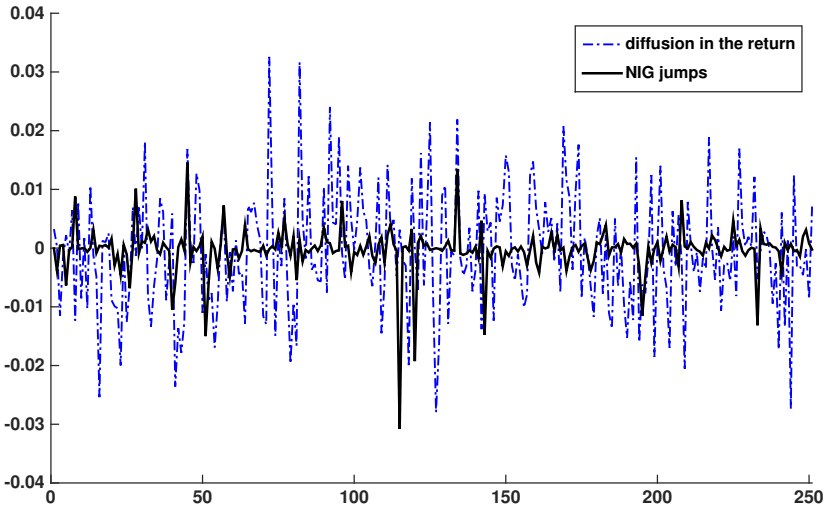


**Figure 3.8:** Randomness sources in the SV-NIG-0 model.

jumps in the Bates-0 and SV-NIG-0 models. The simulated compound Poisson jumps are discrete jumps, and their expected mean size and variance are large, with $\mu_y = -0.05$ and $\sigma_y = 0.1$; therefore, the compound Poisson jumps bring on large changes in the log prices, which are relatively easy to identify. Since there is a distinctive difference between the simulated compound Poisson jumps and simulated diffusion terms, as shown in Figure 3.7, the algorithms can easily classify the different sources of randomness created by the jumps and diffusions. However, for the model with NIG jumps, the expected mean of jumps is $\gamma/\nu = -0.02$, and as shown in Figure 3.8 of the simulated NIG jumps against the simulated diffusions, large jumps are less significant than those created by the Bates-0 model. When estimating the SV-NIG-0 model, the algorithms may mix the randomness created by the infinite-activity jumps and diffusions, especially for the small frequent jumps. Since the NIG jumps are not significant compared to the compound Poisson jumps, their role in the likelihoods and importance weights of particles is less important than that of the compound Poisson jumps. In other words, even if the jumps are not correctly identified, the extra randomness can be captured by the diffusion terms, which explains the higher AR and the better mixing of the chain in the estimation of the SV-NIG-0 model than the Bates-0 model.

Overall, the strength of the PMMH algorithms, compared to the other MCMC methods, is that they simultaneously update the parameters and state variables, targeting $p(\Theta|Y_{1:T})$, and the PGAS simultaneously update all the variables, targeting $p(\Theta, X_{1:T}|Y_{1:T})$, which is very important in the presence of a strong dependence between state variables and parameters. In the Bates-0 model, the independent rare large events reduce the efficiency of the PMCMC methods; however, in the SV-NIG-0 model, where the jumps are hard to distinguish from the diffusions and their role in deciding the weights of the particles is relatively small, most particles generated by the PMCMC methods are useful and, therefore, the PMCMC beat the other algorithms and are very competitive in this case.

### 3.7.2 Variance

In this subsection, I analyse the extraction of variance in the Heston-0, Bates-0, and SV-NIG-0 models. The figures of the extracted variance against the simulated path are listed in Appendix A.

The figures show that the MH and AM yield relatively poor results when little prior information of variance is available. Specifically, given that the initial variance is $v_t = 0.02$, $\forall t = 1, ..., 252$, the MH and AM do not achieve the true variance dynamics and are sensitive to the initial flat values, whereas the other algorithms capture the variance dynamics relatively well. Moreover, I observe that when estimating all three models, the FUSS and PMCMC algorithms tend to extract a path of variance more volatile than the true one. In particular, in the Bates-0 model, the path extracted by the PMMH-PF1 is the most volatile: it captures the concaveness that occurs between 50 and 100, which the other algorithms miss. In the SV-NIG-0 model, the differences between the paths of variance extracted by the MH, AM, and FUSS algorithms are much less significant, and clearly the PMCMC methods achieve better performance in generating more volatility and more consistent levels for the paths.

The extraction of the path of variance is relevant to values of $\kappa$ and $\alpha$. The paths of variance extracted by the FUSS and PMCMC algorithms are more volatile than those by the MH and AM algorithms, and correspondingly, $\kappa$ and $\alpha$ estimated by the FUSS and PMCMC are more significant than those by the MH and AM. In fact, in the estimation with the PMCMC methods, the update of variance is very sensitive to $\alpha$. Specifically, a more significant $\alpha$ can introduce more diversity to the particle set of variance; therefore, it may prevent the particles from accumulating in a local region and increase the efficiency of the SMC update.

### 3.7.3 Jumps

The figures of the extracted jumps against the simulated jumps are listed in Appendix B. In the Bates-0 model, all the algorithms extract the discrete rare large jumps quite well in terms of estimating both jump times and jump sizes. However, extracting infinite-activity NIG jumps is much more difficult than extracting discrete finite-activity compound Poisson jumps. Overall, the extracted NIG jumps are smaller than the true jumps, especially the downside jumps and jumps extracted by the FUSS and PMMH. Interesting, although the extraction of NIG jumps is not as accurate as that of compound Poisson jumps, there are significant similarities in the extracted NIG jumps across different algorithms, suggesting that the MCMC methods tend to reallocate the randomness created by Brownian diffusions and infinite-activity jumps, especially the frequent small jumps.

### 3.7.4 Parameter Estimation

First, I analyse the parameter estimation results of the Heston-0 model as shown in Table 3.4. The MH and AM produce similar parameter estimation results, except that $\kappa$ and $\alpha$ are more significant with the AM. Further, $\kappa$ and $\alpha$ become higher with the FUSS algorithms and are highest with the PMMH and PGAS. A possible reason for the significant $\kappa$ and $\alpha$ with some algorithms is that they tend to extract a volatile path of variance with more spiky peaks and valleys, and this may affect the values of the parameters that are key in creating the changes in the path of variance.

**Table 3.4:** Parameter estimates of the Heston-0 model using simulated one-year data of daily log asset prices. The parameter values are the mean of the posteriors as annual decimals. The standard errors are the standard deviations of the posteriors, reported in parentheses.

| | True values | MH | AM | FUSS-RC-P4 | FUSS-MH-P4 |
|---|---|---|---|---|---|
| $M_0$ | | 80000 | 15000 | 1000 | 1000 |
| $M$ | | 100000 | 20000 | 5000 | 5000 |
| $\kappa$ | 3 | 2.536 | 2.562 | 2.615 | 3.240 |
| | | (1.895) | (1.799) | (1.705) | (1.860) |
| $\theta$ | 0.02 | 0.022 | 0.023 | 0.016 | 0.018 |
| | | (0.025) | (0.025) | (0.009) | (0.008) |
| $\alpha$ | 0.3 | 0.147 | 0.156 | 0.186 | 0.171 |
| | | (0.031) | (0.013) | (0.035) | (0.025) |
| $\rho$ | -0.8 | -0.849 | -0.840 | -0.790 | -0.731 |
| | | (0.091) | (0.064) | (0.098) | (0.202) |
| $\mu$ | -0.1 | -0.151 | -0.152 | -0.142 | -0.136 |
| | | (0.104) | (0.105) | (0.096) | (0.100) |

| | True values | PMMH-PF1 | PMMH-PF2 | PMMH-APF | PGAS |
|---|---|---|---|---|---|
| $M_0$ | | 4000 | 4000 | 4000 | 10000 |
| $M$ | | 20000 | 20000 | 5000 | 50000 |
| $N$ | | 1000 | 1000 | 1000 | 5000 |
| $\kappa$ | 3 | 2.895 | 2.730 | 2.662 | 3.180 |
| | | (0.649) | (1.561) | (0.421) | (1.881) |
| $\theta$ | 0.02 | 0.021 | 0.017 | 0.016 | 0.019 |
| | | (0.005) | (0.008) | (0.001) | (0.009) |
| $\alpha$ | 0.3 | 0.251 | 0.139 | 0.469 | 0.199 |
| | | (0.067) | (0.043) | (0.033) | (0.052) |
| $\rho$ | -0.8 | -0.825 | -0.874 | -0.701 | -0.804 |
| | | (0.076) | (0.072) | (0.003) | (0.069) |
| $\mu$ | -0.1 | -0.186 | -0.161 | -0.163 | -0.152 |
| | | (0.071) | (0.092) | (0.037) | (0.095) |

Further, in the Heston-0 model, all the algorithms estimate $\mu$ more negative than its true value. This may also be explained by the extracted path of variance being flatter than the true path. As captured by the correlation parameter $\rho$, a strong negative correlation occurs between the changes in variance and log asset prices, and when the spot variance increases, the log price process tends to drop. However, when the peaks in the true path of variance are not captured by the algorithms, the extracted variance is less capable of causing a drop with "enough" magnitude in the log asset price, which has to be compensated for by a $\mu$ more negative than its true value.

According to the estimation results of the Bates-0 and SV-NIG-0 models as shown in Tables 3.5 and 3.6, the values of $\kappa$ and $\alpha$ are also correlated with the performance of algorithms in extracting variance, and overall, $\alpha$ is higher with the PMCMC methods. This suggests a potential problem of the PMMH. In the estimation with the PMMH, I use an SMC algorithm to update the latent state variable $\mathbf{v}$. In particular, I generate a set of particles of the variance $\mathbf{v}$ according to its transition kernel: $v_t = v_{t-1} + \kappa(\theta - v_{t-1})\Delta +$

**Table 3.5:** Parameter estimates of the Bates-0 model using simulated one-year data of daily log asset prices. The parameter values are the mean of the posteriors as annual decimals. The standard errors are the standard deviations of the posteriors, reported in parentheses.

| | True values | MH | AM | FUSS-RC-P4 |
|---|---|---|---|---|
| $M_0$ | | 80000 | 15000 | 1000 |
| $M$ | | 100000 | 20000 | 5000 |
| $\kappa$ | 3 | 2.482 | 3.035 | 3.305 |
| | | (1.481) | (1.652) | (1.295) |
| $\theta$ | 0.02 | 0.025 | 0.023 | 0.023 |
| | | (0.021) | (0.018) | (0.009) |
| $\alpha$ | 0.3 | 0.171 | 0.141 | 0.205 |
| | | (0.023) | (0.027) | (0.013) |
| $\rho$ | -0.8 | -0.937 | -0.867 | -0.798 |
| | | (0.031) | (0.070) | (0.028) |
| $\mu$ | -0.1 | -0.071 | -0.052 | -0.041 |
| | | (0.116) | (0.136) | (0.119) |
| $\mu_y$ | -0.05 | -0.062 | -0.060 | -0.060 |
| | | (0.028) | (0.029) | (0.026) |
| $\lambda$ | 0.05 | 0.067 | 0.067 | 0.067 |
| | | (0.017) | (0.017) | (0.018) |
| $\sigma_y$ | 0.1 | 0.108 | 0.106 | 0.110 |
| | | (0.024) | (0.021) | (0.022) |

| | True values | FUSS-MH-P4 | PMMH-PF | PGAS |
|---|---|---|---|---|
| $M_0$ | | 1000 | 4000 | 4000 |
| $M$ | | 5000 | 5000 | 20000 |
| $N$ | | | 5000 | 5000 |
| $\kappa$ | 3 | 2.977 | 3.014 | 3.008 |
| | | (0.779) | (0.046) | (0.498) |
| $\theta$ | 0.02 | 0.022 | 0.015 | 0.019 |
| | | (0.008) | (0.000) | (0.005) |
| $\alpha$ | 0.3 | 0.201 | 0.256 | 0.345 |
| | | (0.040) | (0.004) | (0.001) |
| $\rho$ | -0.8 | -0.749 | -0.758 | -0.860 |
| | | (0.113) | (0.022) | (0.104) |
| $\mu$ | -0.1 | -0.047 | -0.023 | 0.028 |
| | | (0.119) | (0.000) | (0.094) |
| $\mu_y$ | -0.05 | -0.062 | -0.045 | -0.062 |
| | | (0.028) | (0.001) | (0.029) |
| $\lambda$ | 0.05 | 0.069 | 0.068 | 0.066 |
| | | (0.018) | (0.000) | (0.017) |
| $\sigma_y$ | 0.1 | 0.110 | 0.126 | 0.111 |
| | | (0.021) | (0.003) | (0.023) |

**Table 3.6:** Parameter estimates of the SV-NIG-0 model using simulated one-year data of daily log asset prices. The parameter values are the mean of the posteriors as annual decimals. The standard errors are the standard deviations of the posteriors, reported in parentheses.

| | True values | MH | AM | FUSS-RC-P4 |
|---|---|---|---|---|
| $M_0$ | | 80000 | 40000 | 3000 |
| $M$ | | 100000 | 50000 | 10000 |
| $\kappa$ | 3 | 3.293 | 2.828 | 3.062 |
| | | (2.093) | (1.584) | (2.024) |
| $\theta$ | 0.02 | 0.024 | 0.021 | 0.022 |
| | | (0.009) | (0.005) | (0.008) |
| $\alpha$ | 0.1 | 0.088 | 0.095 | 0.080 |
| | | (0.011) | (0.020) | (0.012) |
| $\rho$ | -0.8 | -0.805 | -0.844 | -0.790 |
| | | (0.063) | (0.070) | (0.070) |
| $\mu$ | -0.1 | -0.211 | -0.139 | -0.191 |
| | | (0.037) | (0.061) | (0.055) |
| $\gamma$ | -0.1 | -0.122 | -0.088 | -0.156 |
| | | (0.072) | (0.116) | (0.167) |
| $\nu$ | 5 | 8.693 | 8.457 | 5.900 |
| | | (3.353) | (3.422) | (1.406) |
| $\sigma$ | 0.2 | 0.254 | 0.253 | 0.265 |
| | | (0.058) | (0.055) | (0.054) |

| | True values | FUSS-MH-P4 | PMMH-PF1 | PGAS |
|---|---|---|---|---|
| $M_0$ | | 3000 | 10000 | 10000 |
| $M$ | | 10000 | 20000 | 20000 |
| $N$ | | | 1000 | 1000 |
| $\kappa$ | 3 | 3.160 | 3.615 | 3.162 |
| | | (1.786) | (1.788) | (1.826) |
| $\theta$ | 0.02 | 0.020 | 0.019 | 0.027 |
| | | (0.006) | (0.004) | (0.008) |
| $\alpha$ | 0.3 | 0.088 | 0.131 | 0.095 |
| | | (0.010) | (0.048) | (0.037) |
| $\rho$ | -0.8 | -0.849 | -0.826 | -0.814 |
| | | (0.072) | (0.086) | (0.076) |
| $\mu$ | -0.1 | -0.131 | -0.132 | -0.307 |
| | | (0.038) | (0.050) | (0.127) |
| $\gamma$ | -0.1 | -0.078 | -0.098 | -0.088 |
| | | (0.071) | (0.031) | (0.195) |
| $\nu$ | 5 | 5.974 | 5.218 | 5.567 |
| | | (1.513) | (1.145) | (1.418) |
| $\sigma$ | 0.2 | 0.252 | 0.137 | 0.244 |
| | | (0.055) | (0.083) | (0.058) |

$\alpha\sqrt{v_{t-1}\Delta}\epsilon$, where $\epsilon$ is a standard normal random number. The transition kernel shows that the diversity of the particle set is determined by the parameter $\alpha$; therefore, a larger $\alpha$ is more likely to produce a particle set that better represents the target distribution. In other words, $\alpha$ has an important role in the performance of the SMC update in that it not only directly affects the likelihood as a parameter involved in the likelihood function, but also indirectly changes it by generating a particle set. Thus, it is not surprising that a larger $\alpha$ is more likely to be accepted with the PMMH-PF1, whereas with the PGAS and PMMH-PF2, $\alpha$ is updated according to its conditional posterior as in the MH, AM, and FUSS, and whether it is accepted or not is not related to the likelihood of the particle set.

In the Bates-0 model, all the algorithms succeed in estimating the jump-related parameters $\mu_y$, $\lambda$, and $\sigma_y$, owing to very accurate extraction of the jumps. However, in the SV-NIG-0 model, the jump-related parameters, including $\gamma$, $\nu$, and $\sigma$, are harder to identify from the data. Specifically, the expected mean size of the NIG jumps is $\gamma/\nu$, and the variance of the NIG jumps is $\nu\sigma^2 + \gamma^2/\nu^3$. When the estimated values are similar in the other algorithms, $\gamma$ is more negative with the FUSS-RC, and $\sigma$ is lower in the PMMH-PF1. Correspondingly, the NIG jumps extracted by the FUSS-RC show downside jumps that are more negative, and the jumps extracted by the PMMH-PF1 are less significant than those extracted by the other algorithms. Moreover, the arrival rate of the NIG jumps is:

$$\pi_{NIG}(dx) = \frac{\sigma\alpha}{\pi}\frac{e^{\beta x}K_1(\alpha|x|)}{|x|}dx, \tag{3.14}$$

where $\alpha = \nu^2/\sigma^2 + \gamma^2/\sigma^4$, $\beta = \gamma/\sigma^2$, and $K_1$ is the modified Bessel function of the third kind with index 1. Therefore, generally, a more significant $\nu$ with the MH and AM leads to a higher arrival rate, and as reflected in the figures of extracted jumps, there are more frequent small jumps extracted by the MH and AM.

Further, in the SV-NIG-0 model, the values of $\mu$ with all algorithms are more negative than its true value. Indeed, this result may come from the less significant jumps. Compared to the simulated NIG jumps, the number of the extracted jumps are fewer, the jump sizes are smaller, and, markedly, the algorithms miss several large downside jumps.

## 3.8 Discussion

In this chapter, I compare the estimation performance of advanced MCMC algorithms comprising the AM (Haario et al., 1999a, 2001), the FUSS algorithms (Martino et al., 2015), and the PMCMC methods against conventional MH algorithms (Hastings, 1970; Metropolis et al., 1953). The PMCMC methods comprise the PMMH-PF1/PF2/APF (Andrieu et al., 2010) and the PGAS (Lindsten et al., 2014). The comparison is based on simulation studies. In particular, I examine, from simulated data, how these algorithms perform in estimating the affine Heston-0, Bates-0, and SV-NIG-0 models with different jump structures and in dealing with the problems of high dimensions, complicated target distributions, and strong dependence between latent state variables and parameters. While some finance research applies the conventional MCMC or SMC algorithms to estimate stochastic volatility models with jumps (see, for example, Christoffersen et al., 2010a; Eraker et al., 2003; Kaeck and Alexander, 2013a), few attempts have been made to apply the above algorithms. This chapter seek to examine how the advanced estimation methods perform in estimating complicated financial models compared to conventional MH algorithms.

The results of the simulation studies show that different jump structures significantly affect the estimation performance of the algorithms. In particular, when all the algorithms perfectly extract the compound Poisson jumps in the Bates-0 model, they fail to distinguish the randomness created by the infinite-activity NIG jumps and Brownian diffusions in the log price process of the SV-NIG-0 model. This makes parameter estimation and variance extraction very challenging, and the extra complexity brought on by the NIG jumps requires the FUSS and PMCMC methods to increase the numbers of MCMC runs and particles, while the MH and AM are less capable of dealing with complicated model specifications. In contrast, compound Poisson jumps create rare but significant changes in the path of the log price process, which are distinctive of diffusions. Therefore, in the PMCMC methods, jumps in log asset prices play an important role in deciding the importance weights of particles. The results of the simulation studies show that in the estimation of the Bates-0 model, the acceptance rate of the PMCMC methods significantly decreases, and the efficiency drops. However, the other algorithms are less vulnerable to the specification of compound Poisson jumps and achieve consistent performance in different models.

Considering the complexity and computational costs of different algorithms, I conclude that for models with a simple specification, the MH is very competitive because of its low computational cost. However, if the target distribution is complicated or the proposal of the MH is inappropriate, the acceptance rate may be very low, and most generated draws are wasted. The AM makes use of the previous samples and dynamically tunes the proposal, and this online-tuned adaptive proposal can significantly raise the acceptance rate and speed up the convergence of the chain.

Moreover, when there is a strong correlation between state variables and the target distribution becomes complicated, numerous MCMC iterations may be required for the chain generated by the MH and AM to converge. The FUSS algorithms can tackle this problem by constructing an efficient proposal and producing virtually independent samples, as noted in Martino et al. (2015). On the other hand, with a fixed number of MCMC iterations, the FUSS algorithms are slower than the very fast MH and AM algorithms. However, if one focuses on achieving a good estimation performance, despite the relatively high computational costs, the FUSS algorithms are very efficient because

they only require a small number of MCMC iterations to achieve a good mixing.

In addition, the PMCMC methods can deal with strong dependence and complicated target distributions, and their computational costs are significantly lower than those of the FUSS algorithms, which make them very competitive. However, when other algorithms achieve stable performance across different model specifications, the performance of the PMCMC methods depends largely on the properties of specific problems, which may deteriorate when the dependence between some state variables is weak. Moreover, as pointed out in Lindsten et al. (2014), the relative performances of the PMMH and PGAS depend on whether the ideal marginal MH or the Gibbs sampler, that is, the samplers that PMMH and PGAS approximate, has the better mixing property for the specific problem. In estimating the Heston-0 and SV-NIG-0 model, the PGAS outperforms the PMMH with a small number of MCMC iterations and a small set of particles, suggesting a fast and good mixing of the PGAS kernel in estimating these two models.

Finally, it would be interesting to discuss the performance of MCMC algorithms on the basis of more extensive data sets. For example, in this study, the comparison is based on one simulated path of one-year data, and in further research it is possible to consider a number of simulated paths with more observations.

# 4 Model Estimation with Advanced MCMC Algorithms: Empirical Studies

## 4.1 Motivation

In this chapter, I apply the advanced MCMC algorithms that prove efficient in the simulation studies to estimate the SV-NIG-1 model using real market data. The algorithms comprise the AM (Haario et al., 2006), the FUSS-RC with pruning type 4 (Martino et al., 2015), the PMMH (Andrieu et al., 2010), and the PGAS (Lindsten et al., 2014), and their performance is compared to that of the conventional random-walk MH algorithm.

The model estimated is the SV-NIG-1 model, namely, the stochastic volatility model with infinite-activity NIG jumps in returns and the non-affine linear variance process. This model is selected because, as shown in Chapter 2, the SV-NIG-1 performs best in option pricing and is a good fit with the data of the S&P 500 index returns and the 30-day VIX in 1996 – 2009.

This chapter focuses on empirical studies. The SV-NIG-1 model is estimated using the joint information of the S&P 500 index returns and the 30-day VIX from January 2002 to December 2005, the same time period used in Andrieu et al. (2010).[1] Using the joint information enables me to estimate both the physical and risk-neutral dynamics of the index returns and variance. The first reason of estimating both dynamics is that the risk-neutral dynamics has a number of important real-life financial applications, one of the most important being the valuation of derivatives. Second, estimating the risk-neutral dynamics from the joint information adds to the existing difficulty of estimating the physical dynamics in the presence of high dimensions and complicated posteriors, which challenges an algorithm to efficiently fit a model to data.

The objectives of this chapter are

- To apply the advanced MCMC algorithms to estimate a complex model using the joint information of empirical index returns and VIX data;
- To compare the estimation performance of different algorithms in the presence of strong dependence between a number of unknown parameters and high-dimensional state variables.

[1] Andrieu et al. (2010) use only daily index data.

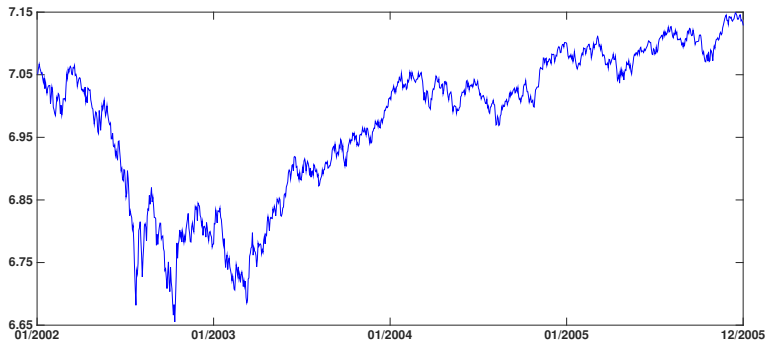


**Figure 4.1:** Daily S&P 500 index returns in 2002 – 2005.

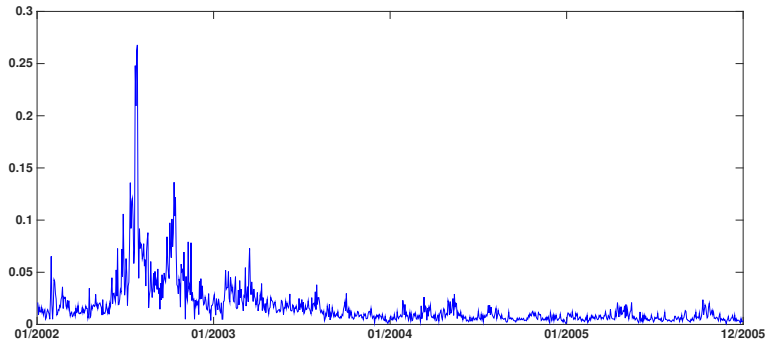


**Figure 4.2:** Annualised realised variance of the S&P 500 index in 2002 – 2005.

This chapter is organised as follows. Section 2 briefly describes the data. Section 3 presents the initial settings and implementation of algorithms. Section 4 presents the estimation results and discusses the performance of algorithms. Section 5 concludes this chapter.

## 4.2 Data

In the empirical studies, the SV-NIG-1 model is estimated using the joint information of the S&P 500 index returns and daily 30-day VIX data from January 2002 to December 2005, covering a total of four years and 1,000 trading days. The daily returns of the S&P 500 index, the annualised realised variance of the index, and the 30-day VIX index are plotted in Figures 4.1 – 4.3. The VIX index is reformulated from the volatility surface provided by OptionMetrics, and unlike the CBOE's methodology, this approach reduces the systemic biases in the VIX index provided by the CBOE, as pointed out in Kaeck and Alexander (2012).

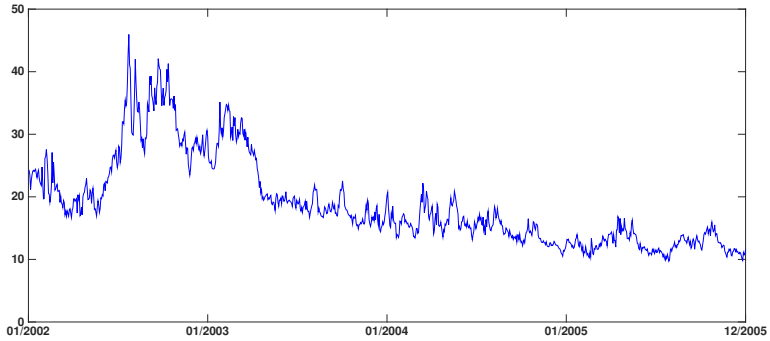


**Figure 4.3:** 30-day VIX of the S&P 500 index in 2002 – 2005.

## 4.3 Implementation of Algorithms

Since the likelihood inference of the SV-NIG-1 model has been discussed in detail in Chapter 2, in this section, I focus only on the initial settings and implementation of the algorithms employed.

Before the estimation, I need to assign appropriate priors to the parameters. I choose conjugate priors to make the conditional posterior a known form if possible, and the priors should be uninformative so that the posteriors are dominated by the likelihoods. The selected priors are as follows:

$$p(\kappa^P) = N(4, 3^2), p(\kappa^Q) = N(4, 3^2), p(\theta) = N(0.05, 0.1^2), \tag{4.1}$$

$$p(\eta_s) = N(0, 2^2), p(\rho_\varepsilon) = N(0, 1^2), p(\sigma_\varepsilon) = \frac{1}{\sigma_\varepsilon}. \tag{4.2}$$

As in the simulation studies, the parameter pair $(\alpha, \rho)$, which is expected to be strongly correlated, is re-parameterised as in Jacquier et al. (2002): $\phi_v = \alpha\rho$ and $w_v = \alpha^2(1-\rho^2)$. This improves estimation efficiency and helps to rule out the large correlation between $\alpha$ and $\rho$. The priors for $\phi_v$ and $w_v$ are conjugate: $\phi_v | w_v \sim N(0, \frac{1}{2}w_v)$ and $w_v \sim IG(2, 200)$.

When the model is estimated with the PMMH-PF1, I set the priors for $\rho$ and $\alpha$ as follows:

$$p(\alpha) = \frac{1}{\alpha}, p(\rho) = \text{Uniform}(-1, 1).$$

Further, I assume the priors for the jump-related parameters as:

$$p(\gamma) = N(0, 1^2), p(\gamma^Q) = N(0, 1^2), p(\nu^Q) = N(6, 5^2), p(\sigma) = \frac{1}{\sigma}.$$

Except for the PMMH, where the candidate draw of the parameter set $\Theta^*$ is generated by a proposal $q(\cdot|\Theta^{(i-1)})$, with the other estimation methods the parameters are sampled from their conditional posteriors. In particular, the conditional posteriors of parameters $\phi_v$, $\eta_s$, $\rho_c$, and $\gamma$ in the SV-NIG-1 model are normal distributions, and the conditional posteriors of $w_v$ and $\sigma_c$ are the inverse Gamma distributions. Consequently, these parameters are updated by a Gibbs step.

$\kappa^Q$, $\gamma^Q$, and $\nu^Q$ are updated by the random-walk MH with Gaussian proposals, and the variances[2] of the proposals for $\kappa^Q$, $\gamma^Q$, and $\nu^Q$ are 0.09, 0.0025, and 0.04, respectively. It is possible to apply the FUSS-RC or the AM to estimate these three parameters; however, after the estimation with the random-walk MH, the acceptance rates of these parameters are very high, reaching 85% for $\kappa^Q$, 91% for $\gamma^Q$, and 86% for $\nu^Q$. Sufficiently high acceptance rates obviate the computationally expensive algorithms.

The conditional posteriors of $\kappa^P$, $\theta$ and $\sigma$ can be factorised by the product of a known distribution and the likelihood of the VIX; therefore, they can be updated by the DWW method. After the estimation, the acceptance rates of $\kappa^P$, $\theta$, and $\sigma$ are 56%, 68%, and 79%, respectively, and for the same reason as mentioned above, I decide to use the fast DWW method, instead of the AM and FUSS algorithms.

In the extraction of latent state variables, I follow the same approach as in the simulation studies, updating the jump variable $\mathbf{J_Y}$ by a Gibbs step, updating $\mathbf{G}$ by the random-walk MH with a Gaussian proposal with variance $1\times10^{-6}$, and applying the random-walk MH with a Gaussian proposal with variance $4\times10^{-6}$, the AM, or the FUSS-RC to extract $\mathbf{v}$. The settings of the AM and the FUSS-RC are the same as in the simulation studies. Alternatively, I use the PMCMC methods to extract the state variables $\mathbf{J_Y}$, $\mathbf{G}$, and $\mathbf{v}$ by an SMC update.

Importantly, I choose the realised variance as the initial value for variance, and in the PGAS, it is the initial reference trajectory. The choice of the realised variance is motivated by the fact that it is an important measure of the physical dynamics of the variance process and may contain some information about the physical parameters. Moreover, the implied variance, although usually not as volatile as the realised variance and with fewer spiky peaks, is positively correlated with the realised variance. Therefore, instead of using $v_t = a, \forall t$, for some constant $a$, as in the simulation studies, in the empirical studies, I exploit the information of the realised variance. This may improve the performance of the MH, which is sensitive to initial values.

## 4.4 Results

### 4.4.1 Parameter Estimation

Let us first look at the parameter estimation results as reported in Table 4.1. The MH and AM obtain similar parameter estimates. Compared to the estimates of the MH and AM, the FUSS-RC differs mainly in a higher $\theta^P$ and higher $\kappa^Q$, because the FUSS-RC algorithm tends to extract a path of variance with a higher level than the other two algorithms. A higher $\kappa^Q$ compensates for the large $\theta^P\kappa^P$ to imply a $\theta^Q$ more consistent with the VIX data.

On the other hand, the PGAS and PMMH-PF2 achieve similar results, which is not surprising because in structure, these two algorithms are very similar, except for a reference trajectory used in the PGAS. The difference between the two lies in their estimation of jump-related parameters. Uniformly, the jump-related parameters in the PMMH-PF2 are more significant than those in the PGAS, suggesting a higher frequency of jumps and jumps of a larger mean size under the physical measure $P$, corresponding to the jumps extracted by the PMMH-PF2 as shown in Figure 4.19.

[2]The variances are proper in the sense that they consider the trade-off between a high acceptance rate and a generated chain mixing well.

**Table 4.1:** Parameter estimates, acceptance rate of $v$, and DICs of model parameters of the SV-NIG model with $\beta = 1$. The model is estimated using daily spot returns of the S&P 500 index and the reformulated 30-day VIX from January 2002 to December 2005. The parameter values are the mean of the posteriors as annual decimals. The standard errors are the standard deviations of the posteriors, reported in parentheses.

| | MH | AM | FUSS-RC-P4 | PMMH-PF1 | PMMH-PF2 | PGAS |
|---|---|---|---|---|---|---|
| $M_0$ | 60000 | 60000 | 1000 | 10000 | 10000 | 10000 |
| $M$ | 100000 | 100000 | 10000 | 50000 | 50000 | 50000 |
| $N$ | | | | | 1000 | 1000 |
| $\theta^P$ | 0.014 | 0.011 | 0.024 | 0.030 | 0.031 | 0.025 |
| | (0.005) | (0.003) | (0.010) | (0.002) | (0.010) | (0.008) |
| $\kappa^P$ | 1.946 | 2.097 | 2.178 | 2.457 | 1.896 | 1.920 |
| | (0.795) | (0.812) | (0.865) | (0.103) | (0.136) | (0.187) |
| $\kappa^Q$ | 1.322 | 1.628 | 1.957 | 1.698 | 2.502 | 2.472 |
| | (1.164) | (1.238) | (1.692) | (0.081) | (1.460) | (1.422) |
| $\alpha$ | 1.860 | 1.719 | 1.829 | 2.198 | 1.408 | 1.165 |
| | (0.096) | (0.084) | (0.116) | (0.039) | (0.009) | (0.053) |
| $\rho$ | -0.945 | -0.944 | -0.943 | -0.858 | -0.984 | -0.978 |
| | (0.006) | (0.008) | (0.009) | (0.003) | (0.001) | (0.002) |
| $\eta_s$ | -1.433 | -2.026 | -1.338 | -0.506 | -1.726 | -1.278 |
| | (1.472) | (1.630) | (1.602) | (0.059) | (0.802) | (0.805) |
| $\gamma$ | 1.312 | 1.424 | 1.310 | 0.211 | -0.430 | 0.225 |
| | (0.449) | (0.408) | (0.457) | (0.003) | (0.158) | (0.371) |
| $\gamma^Q$ | -0.254 | -0.277 | -0.232 | -0.171 | -0.036 | 0.033 |
| | (0.175) | (0.176) | (0.700) | (0.003) | (0.574) | (0.598) |
| $\nu^Q$ | 5.079 | 4.517 | 9.161 | 3.134 | 9.274 | 6.497 |
| | (1.599) | (1.388) | (1.623) | (0.060) | (1.618) | (0.946) |
| $\sigma$ | 0.176 | 0.170 | 0.177 | 0.114 | 0.105 | 0.111 |
| | (0.022) | (0.020) | (0.019) | (0.006) | (0.002) | (0.009) |
| AR of $\mathbf{v}$ (%) | 33 | 57 | 100 | $< 1$ | $< 1$ | 12 |
| LL | 7810.1 | 7853.1 | 7873.7 | 7523.4 | 7606.5 | 7844.5 |

Generally, compared to the estimates of the MH and AM, the results obtained by the PMMH-PF1 differ in a higher $\alpha$ and $\theta^P$, and a lower $\gamma$, $\nu^Q$, and $\sigma$. As mentioned above, the estimates of $\theta^P$ and $\alpha$ depend strongly on the level and shape of the extracted variance, respectively, and a more volatile extracted path with a higher level may lead to a more significant $\alpha$ and $\theta^P$. In addition, as will be discussed below, in the PMMH-PF1, the volatile path of variance mitigates the need of frequent large jumps to create the desired volatility in returns, which directly affects the estimation of jump-related parameters.

The ACFs of parameters are plotted in Figures 4.4–4.8.[3] Autocorrelation is an important measure of the independence of samples generated from the posterior distribution, and a lower autocorrelation suggests more independent samples. In practice, when the generated

[3]The ACFs of the PMMH-PF1 are not reported, because in the PMMH-PF1, the acceptance rate of the set of parameters is lower than 1% and the autocorrelation between parameters stays at a very high level.

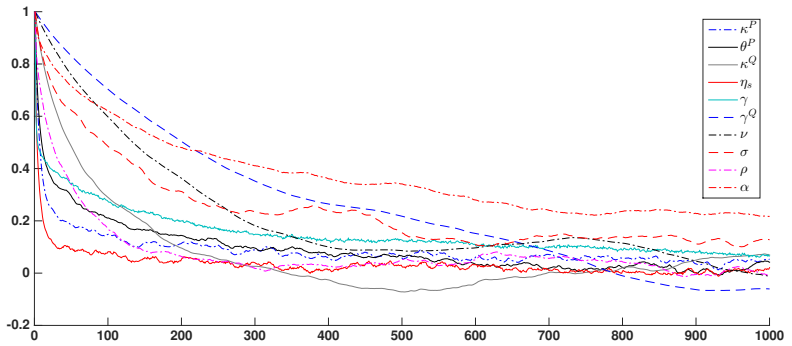


**Figure 4.4:** MH: ACFs of parameters in the SV-NIG-1 model.

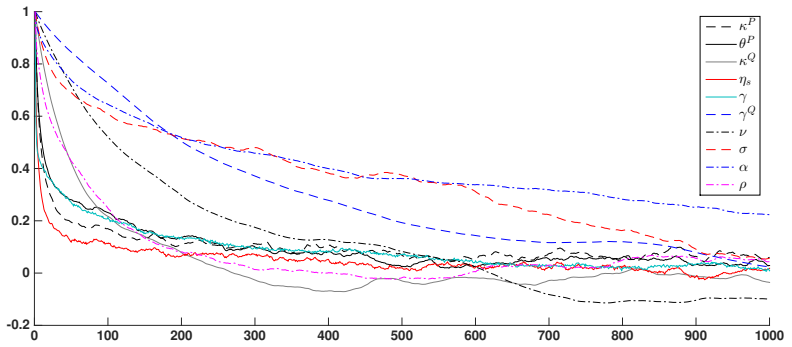


**Figure 4.5:** AM: ACFs of parameters in the SV-NIG-1 model.

samples show a high positive autocorrelation, it means that the samples within a certain length of chain are "trapped" in a local mode, and the higher the autocorrelation, the longer the length of the concerned chain. A high positive autocorrelation suggests that the generated samples are not good representatives of the target distribution, because their empirical distribution may be only a portion of the complete target distribution. Therefore, a strong autocorrelation may suggest that the samples are not very informative of the target distribution and that more MCMC runs are required to make the chain move to the target distribution.

In this regard, the PMMH-PF2 performs best, followed by the FUSS-RC and PGAS, while the performance of the MH and AM is relatively poor. Apparently, the ACFs of some parameters drop more slowly than those of other parameters. In particular, with the MH and AM, the ACF of $\alpha$ is the slowest to move towards zero, and even though I use the re-parameterisation for $\alpha$ and $\rho$ and sample them in pairs, the autocorrelation generated by $\alpha$ still remains strong.[4] Moreover, some parameters obtained with the

[4] A possible reason is that in the estimation, the algorithm is trapped in a local mode of the multimodal posterior of $\alpha$, and if so, choosing different initial values may help to solve this problem. In the estimation presented in this thesis, I use 2 as the initial value for $\alpha$. However, with different initial values, the strong

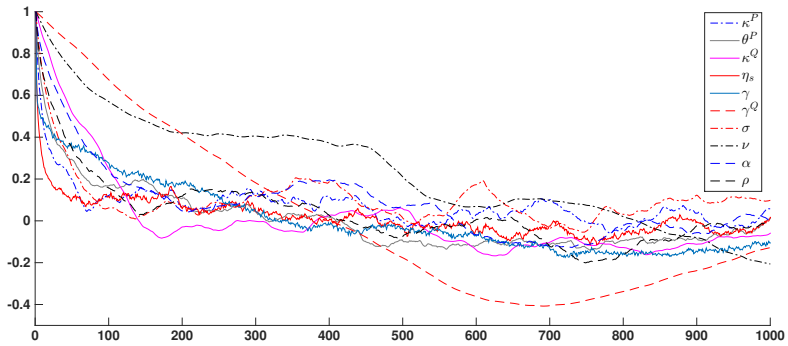


**Figure 4.6:** FUSS-RC-P4: ACFs of parameters in the SV-NIG-1 model.

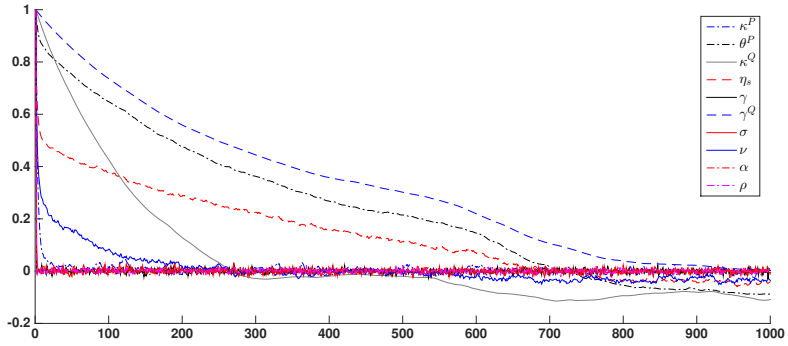


**Figure 4.7:** PMMH-PF2: ACFs of parameters in the SV-NIG-1 model.

FUSS-RC, PMMH-PF2, and PGAS even show negative autocorrelations, which means that the chain efficiently generates representatives of the target distribution and is even better than a chain with independent samples.

One comment on the low ARs of PMMH-PF1/PF2. To improve the AR, I increase the number of particles to 5,000; however, the ARs of both algorithms are still lower than 1. It reminds me of the poor performance of the PMMH-PF1/PF2 in the estimation of the Bates-0 model in the simulation studies, and suggests that for a jump model, even if the data suggests only a few large jumps, the PMMH algorithms may need a large particle set to yield a reasonable AR. Furthermore, I try various updating schemes with the PMMH-PF1, but they seem inefficient in reducing the ACFs of parameters. To achieve reasonable performance in the AR and ACFs, a considerably large $N$ for the PMMH-PF1/PF2 is required. However, for a high-dimensional model and a large-volume data set, increasing $N$ further is extremely time-consuming. Therefore, considering the expensive computational cost incurred with a large set of particles, I present only the results obtained with 1,000 particles.

autocorrelation remains.

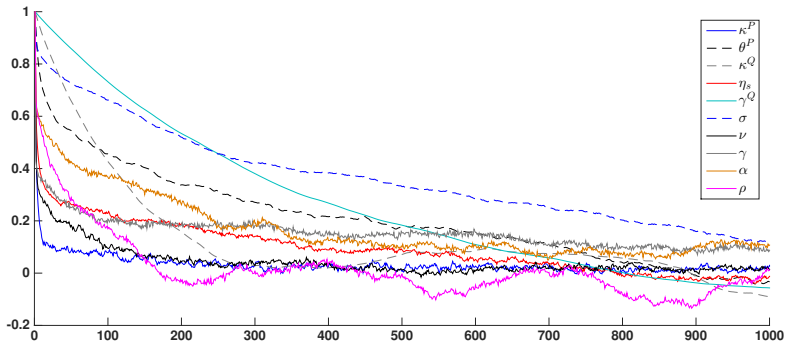


**Figure 4.8:** PGAS: ACFs of parameters in the SV-NIG-1 model.

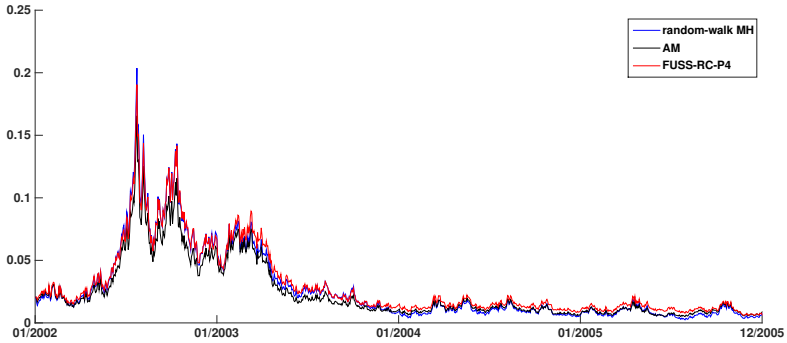


**Figure 4.9:** Extracted variance of the SV-NIG-1 model by the random-walk MH, AM, and FUSS-RC-P4.

### 4.4.2 Variance

The extracted paths of variance are presented in Figures 4.9 and 4.10, and the figures of individual paths are plotted in Figures C.1 – C.6 in Appendix C. The patterns of variance extracted by the MH and AM are similar, except that on average the level of variance extracted by the AM is lower than that extracted by the MH. This may be because in the estimation, I use the volatile realised variance with high peaks as the initial value of variance, and for the MH, it takes longer for the chain to move owing to a relatively inefficient proposal and a low acceptance rate. However, after a sufficiently large number of MCMC runs, and after the chain generated by the MH has mixed well, the variance extracted by the MH still differs from that extracted by the AM with a similar pattern but a different level and scale. This suggests that more than one combination of parameters and latent state variables may make a good goodness of fit. Among the PMCMC methods, the paths of variance extracted by the PMMH-PF2 and PGAS are similar, whereas the PMMH-PF1 tends to extract a path of variance with the same pattern as the VIX.

Next, I move to the ACFs of variance at selected time points, as shown in Figures 4.11 – 4.14. Clearly, the AM with an adaptive proposal improves the update of variance in the

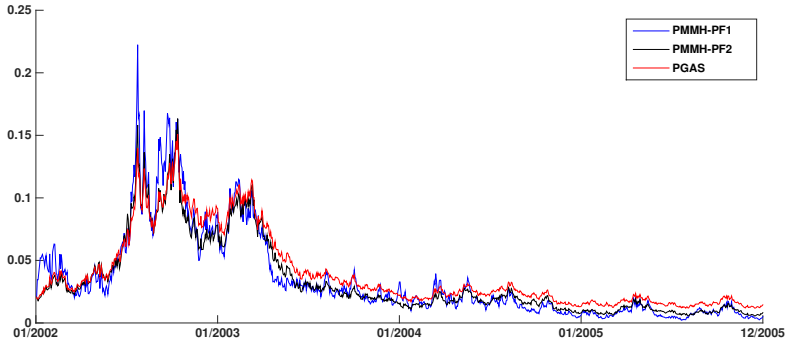


**Figure 4.10:** Extracted variance of the SV-NIG-1 model by the PMMH-PF1, PMMH-PF2, and PGAS.

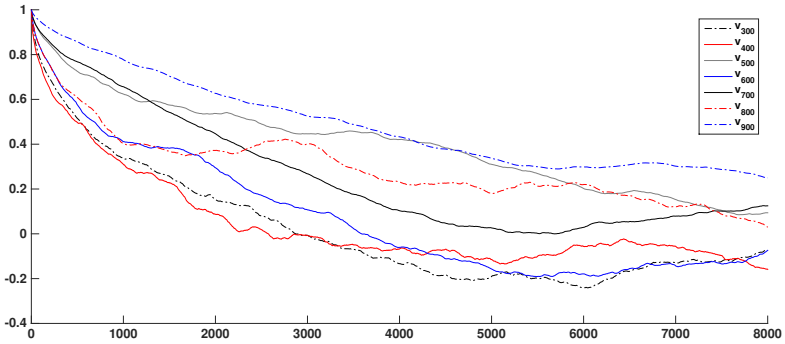


**Figure 4.11:** MH: ACFs of variance at selected time points.

MH, because the samples of variance generated by the AM show a weaker autocorrelation. The FUSS-RC further reduces the ACF, with even a negative autocorrelation. For the PGAS, $N = 1,000$ is large enough to make the ACF drop sharply and remain at the zero level.

### 4.4.3 Jumps

The extracted jumps are plotted in Figures 4.15 – 4.20. The jumps extracted by the MH, AM, and FUSS-RC are very similar, not only the large jumps, but also the frequent small jumps. On the other hand, there is a significant difference in the patterns of jumps extracted by the PMCMC methods. Compared to the jumps extracted by the MH, AM, and FUSS-RC, the mean sizes of positive jumps extracted by the PMCMC methods are much smaller. The potential explanation is that the paths of variance extracted by the PMCMC methods are more volatile with higher levels than those extracted by the other algorithms.

One remark on why the PMCMC methods choose the variance, rather than generating

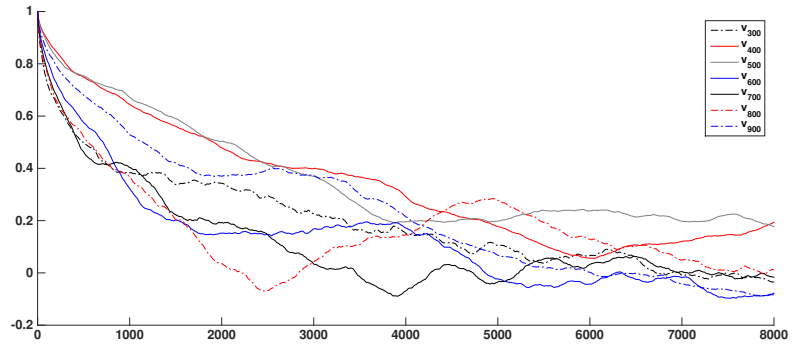


**Figure 4.12:** AM: ACFs of variance at selected time points.

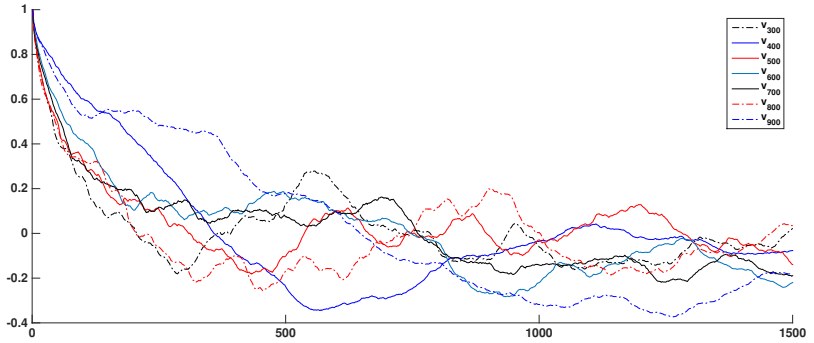


**Figure 4.13:** FUSS-RC-P4: ACFs of variance at selected time points.

large jumps, to create the desired changes in the path of returns. A possible reason is that in the estimation, the jumps are involved only in the likelihood of returns, whereas the variance is involved in the likelihoods of both returns and the VIX index. In particular, in the SV-NIG-1 model, the VIX index is a linear function of the spot variance, and roughly speaking, the changes in the VIX correlate negatively with the changes in returns, especially when the market became turbulent and there were sudden changes in the VIX and returns. Therefore, considering the above factors, it is more efficient for the PMCMC methods to use the spot variance, instead of jumps, to capture the large changes in returns and VIX index simultaneously, and particles suggesting volatile variance paths following the pattern of the VIX are likely to gain larger importance weights.

Importantly, with the MH, AM, and FUSS-RC algorithms, the jumps are updated according to their conditional posteriors, whereas with the PMCMC methods, the jumps are updated using an SMC update. In the SMC update, since the jumps are assumed to be independent of each other, the particles of jumps are simulated without any prior information; therefore, the extracted jumps are more a sequence of randomly generated NIG samples than jumps extracted from the data, as reflected in Figures 4.18–4.19. Another observation is that, in terms of jump extraction, the PMMH-PF2 and PGAS

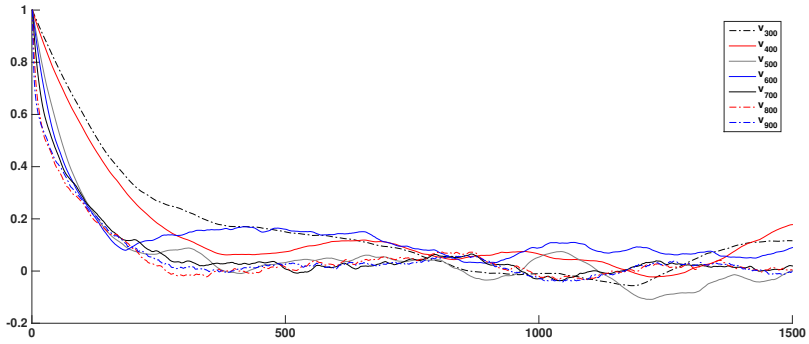


**Figure 4.14:** PGAS: ACFs of variance at selected time points.

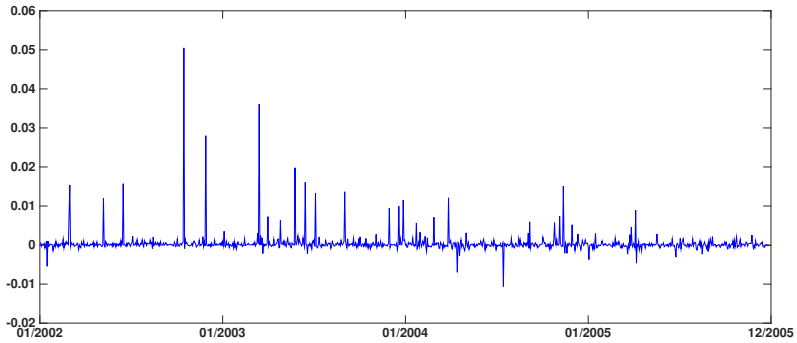


**Figure 4.15:** Random-walk MH: extracted jumps of the SV-NIG-1 model.

disagree sharply, which is surprising because the results of the simulation and empirical studies obtained so far indicate that these two algorithms usually produce similar results. To investigate this disagreement, I re-estimate the SV-NIG-1 model by the PGAS and use the jumps extracted by the FUSS-RC as the initial reference trajectory; however, the pattern of jumps extracted by the PGAS highly resembles that in Figure 4.20. In fact, the use of a reference trajectory provides little improvement in updating the particles of jumps, because the NIG jumps are assumed to be independent of each other and, therefore, the reference trajectory of jumps cannot guide simulated particles to the relevant region.

### 4.4.4 Fit to the VIX

The model-implied VIX obtained by different algorithms is plotted in Figures D.1 – D.6 in Appendix D, and the estimates of $\rho_\varepsilon$ and $\sigma_\varepsilon$ and the mean squared errors (MSE) of the model-implied VIX are presented in Table 4.2. Interestingly, although all the algorithms estimate the SV-NIG-1 model from the joint information of the index returns and VIX data, there is a clear difference in the model-implied VIX computed from estimated parameters and the extracted path of variance. In particular, the model-implied VIX with the PMMH-PF1 achieves the smallest MSE, and it almost completely coincides with

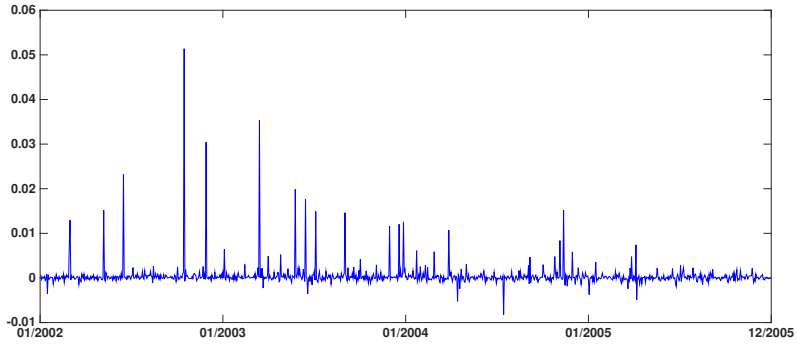


**Figure 4.16:** AM: extracted jumps of the SV-NIG-1 model.

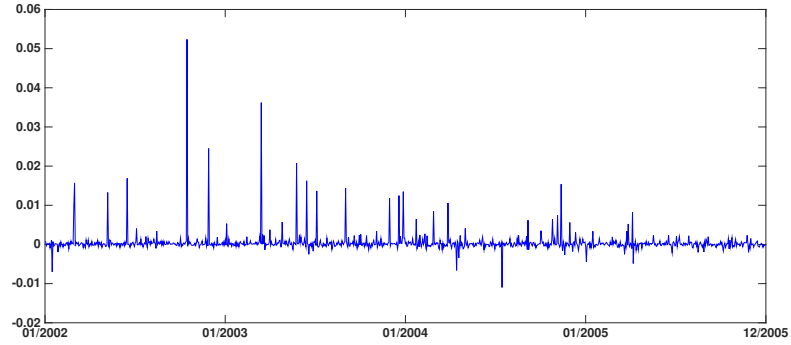


**Figure 4.17:** FUSS-RC-P4: extracted jumps of the SV-NIG-1 model.

the market VIX as shown in Figure D.4. The reason is obvious: the PMMH-PF1 is the only algorithm that updates parameters and state variables simultaneously on the basis of the likelihoods of the VIX and index returns. This ensures that the combination of parameters and state variables, if accepted, produces a very good fit to the VIX index. This suggests that the PMMH-PF1 may perform particularly well in fitting models to data that can be represented as a function of model parameters and state variables, say the VIX. However, in the other algorithms, for example, the PMMH-PF2 and the PGAS, where parameters and state variables are updated sequentially, the SMC update only samples state variables on the basis of the previous sample of the parameter set. After the SMC update of state variables, parameters are updated according to their conditional posteriors. Consequently, candidate draws of parameters, even if they are accepted, do not have to be optimal to maximise the posteriors.

Next, I discuss the estimates of $\rho_\varepsilon$ and $\sigma_\varepsilon$. An interesting observation is that $\rho_\varepsilon$ estimated by the MH and AM is very close to 1, suggesting a significant correlation between VIX pricing errors on neighbouring days, and $\rho_\varepsilon$ estimated by the PMMH-PF2 and PGAS is also close to 1, but slightly below the estimates by the MH and AM. In contrast, $\rho_\varepsilon$ estimated by the PMMH-PF1 is as low as 0.125, suggesting a weak autocorrelation

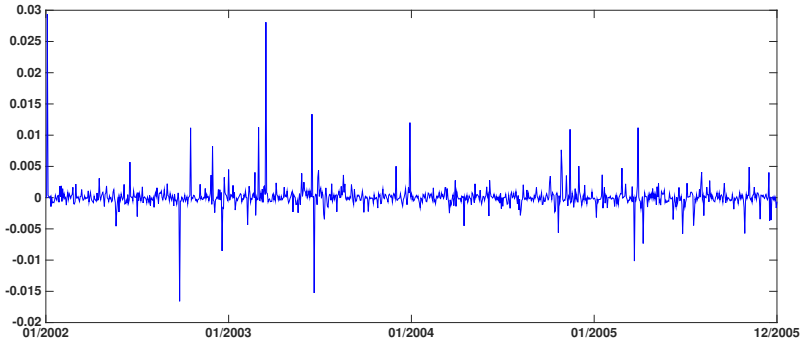


**Figure 4.18:** PMMH-PF1: extracted jumps of the SV-NIG-1 model.

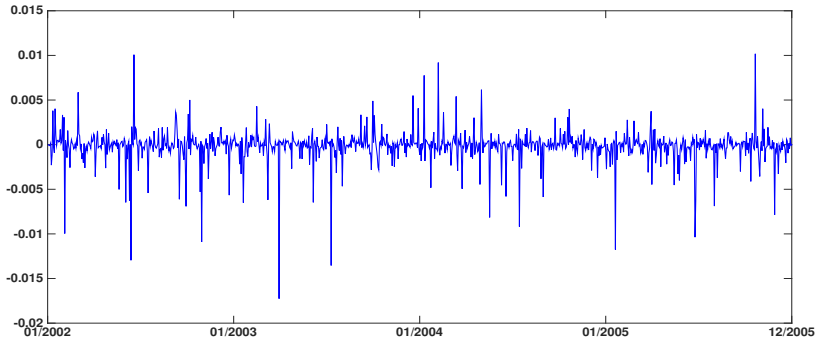


**Figure 4.19:** PMMH-PF2: extracted jumps of the SV-NIG-1 model.

between daily VIX pricing errors. One explanation is the difference in the paths of extracted variance. As mentioned above, the variance extracted by the PMMH-PF1 is more volatile than the others, and its shape highly resembles that of the VIX; therefore, the pricing errors are more random rather than caused by the flatter variance path.

### 4.4.5 Option pricing performance

To test the option pricing performance, I apply the SV-NIG-1 model estimated using the joint information of index returns and the 30-day VIX in 2002 – 2005 to price option in two time periods, 2002 – 2005 and 2006 – 2010. First, I use the Wednesday call and put options in 2002 – 2005 (Sample A) to best avoid weekend effects, and then enhance the sample by also using the option data on Thursdays in 2002 – 2005 (Sample B). To test the option pricing performance in a future period, I use the Wednesday option data in 2006 – 2010 (Sample C). Moreover, as suggested by Bakshi et al. (1997), I filter out options with prices below 3/8 dollar. Then I remove options with maturities shorter than one week or longer than one year to reduce the liquidity bias and focus on the short-run option pricing performance. Moreover, to study how the estimated model performs in predicting future option prices, I compute the option pricing errors in each year for 2006 – 2010. The

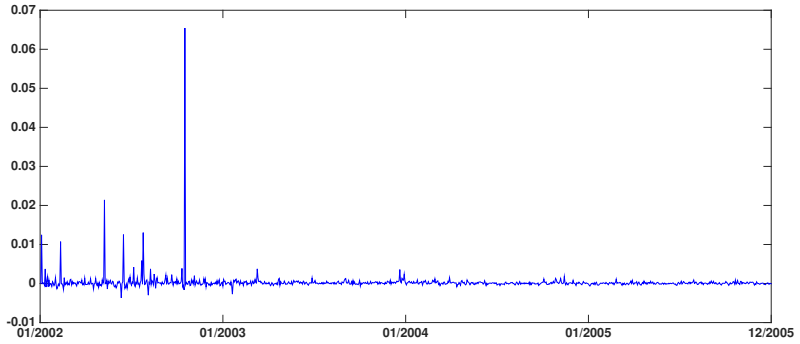


**Figure 4.20:** PGAS: extracted jumps of the SV-NIG-1 model.

**Table 4.2:** Mean squared errors of the model-implied VIX and estimates of $\rho_\varepsilon$ and $\sigma_\varepsilon$ in the SV-NIG-1 model. The mean squared errors are computed as the mean of squared differences between the model-implied VIX and market VIX in 2002 – 2005. The parameters are estimated using daily spot returns of the S&P 500 index and the reformulated 30-day VIX from January 2002 to December 2005. The parameter values are the mean of the posteriors as annual decimals. The standard errors are the standard deviations of the posteriors, reported in parentheses.

| | MH | AM | FUSS-RC-P4 | PMMH-PF1 | PMMH-PF2 | PGAS |
|---|---|---|---|---|---|---|
| $M_0$ | 60000 | 60000 | 1000 | 10000 | 10000 | 10000 |
| $M$ | 100000 | 100000 | 10000 | 50000 | 50000 | 50000 |
| $N$ | | | | 1000 | 1000 | 1000 |
| MSE | 5.51E-04 | 1.07E-03 | 9.89E-04 | 3.04E-05 | 6.84E-04 | 5.08E-04 |
| $\rho_\varepsilon$ | 0.979 | 0.989 | 0.984 | 0.125 | 0.956 | 0.912 |
| | (0.009) | (0.005) | (0.011) | (0.007) | (0.021) | (0.020) |
| $\sigma_\varepsilon$ | 0.034 | 0.035 | 0.034 | 0.026 | 0.042 | 0.042 |
| | (0.001) | (0.001) | (0.001) | (0.000) | (0.001) | (0.001) |

statistics of the option data in all samples are listed in Tables 4.3 and 4.4.

First, it is important to point out that the option pricing performance is *not* a strict measure of the performance of the algorithms. Instead, it allows us to examine the robustness of the estimated model parameters and to understand how different estimation results affect the option pricing performance, thereby affecting the application of the model to investment and risk management.

Second, regarding the option pricing performance with Sample C and its sub-samples, although it works as a robustness check, the robustness is made of model robustness and estimation robustness. As shown in Figures 4.21 – 4.23, the market remained stable and quiet in 2002 – 2007, and from 2008 on, it shifted sharply and became very volatile. Therefore, it is probable that if I re-estimated the models using the data from 2008 – 2010, the results would be very different. However, since nobody could anticipate future market

**Table 4.3:** Properties of options in Samples A, B, and C. Sample A: Wednesday call and put options in 2002−2005. Sample B: Thursday call and put options in 2002−2005. Sample C: Wednesday call and put options in 2006−2010. Options with prices below 3/8 dollar or maturities shorter than one week or longer than one year are filtered out. This table shows the number of contracts, the average price (in parentheses), and the average bid-ask spread {in braces}.

| Sample A<br>Wed, 2002-2005 | Sample B<br>Thu, 2002-2005 | Sample C<br>Wed, 2006-2010 |
|---|---|---|
| 17660<br>(21.56)<br>{1.25} | 17082<br>(21.28)<br>{1.25} | 40543<br>(26.45)<br>{1.74} |

**Table 4.4:** Properties of options in Sub-Samples of Sample C. Options with prices below 3/8 dollar or maturities shorter than one week or longer than one year are filtered out. This table shows the number of contracts, the average price (in parentheses), and the average bid-ask spread {in braces}.

| Sub-samples of Sample C | | | | |
|---|---|---|---|---|
| Wed, 2006 | Wed, 2007 | Wed, 2008 | Wed, 2009 | Wed, 2010 |
| 5872<br>(18.70)<br>{1.12} | 7198<br>(26.16)<br>{1.64} | 8889<br>(35.81)<br>{2.38} | 8808<br>(26.49)<br>{1.57} | 9776<br>(22.77)<br>{1.76} |

**Table 4.5:** Option $\mathcal{V}$RMSEs for the SV-NIG-1 model. Sample A: Wednesday options from January 2002 to December 2005. Sample B: Thursday options from January 2002 to December 2005. Sample C: Wednesday options from January 2006 to December 2010. Options with prices below 3/8 dollar or maturities shorter than one week or longer than one year are filtered out. The option prices are computed by the Monte Carlo method with the technique of antithetic variates. The number of Monte Carlo paths is $50,000$, and the model is discretised at daily frequency. In each sample, the number in bold is the lowest $\mathcal{V}$RMSE.

| | Sample A | Sample B | Sample C |
|---|---|---|---|
| MH | 3.019 | 3.150 | 4.042 |
| AM | 3.067 | 3.212 | 4.185 |
| FUSS-RC-P4 | 2.971 | 3.250 | 4.157 |
| PMMH-PF1 | **2.854** | **3.090** | **3.966** |
| PMMH-PF2 | 3.076 | 3.502 | 4.575 |
| PGAS | 3.207 | 3.627 | 4.849 |

**Table 4.6:** Option $\mathcal{V}$RMSEs for the SV-NIG-1 model. The option samples are Wednesday options from 2006 to 2010. Options with prices below 3/8 dollar or maturities shorter than one week or longer than one year are filtered out. The option prices are computed by the Monte Carlo method with the technique of antithetic variates. The number of Monte Carlo paths is 50,000, and the model is discretised at daily frequency. In each sample, the number in bold is the lowest $\mathcal{V}$RMSE.

| | 2006 | 2007 | 2008 | 2009 | 2010 |
|---|---|---|---|---|---|
| MH | 2.332 | 3.260 | 4.776 | 4.592 | 4.114 |
| AM | **2.221** | 3.271 | 4.920 | 4.844 | 4.318 |
| FUSS-RC-P4 | 2.292 | 3.382 | 4.819 | 4.748 | 4.322 |
| PMMH-PF1 | 2.244 | **3.224** | **4.708** | **4.513** | **4.005** |
| PMMH-PF2 | 2.571 | 3.848 | 5.134 | 5.177 | 4.883 |
| PGAS | 2.662 | 4.031 | 5.452 | 5.502 | 5.197 |

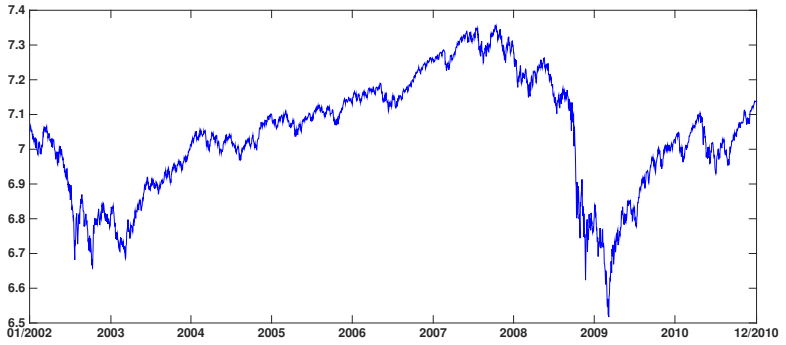


**Figure 4.21:** Daily returns of the S&P 500 index in 2002 – 2010.

changes, a model capable of explaining different market conditions and an estimation method capable of identifying a robust parameter set will win the day.

The option pricing performance in Samples A, B, and C correlates highly with the fit to the VIX. This is not surprising because the VIX index is constructed by option prices, and as mentioned above, the VIX is very informative of the risk-neutral dynamics of a model. Clearly, the PMMH-PF1, which performs best in the fit to the VIX, outperforms the others in all three Samples A, B, and C. In pricing the options in the sub-samples of Sample C, all the algorithms perform better in 2006 – 2007 than in 2008 – 2010, because market conditions in 2006 – 2007 were similar to those in 2002 – 2005, whereas in 2008 – 2010, the market shifted dramatically.

Surprisingly, despite its good fit to the VIX, the PGAS performs worst in all three samples A, B, and C. The reason is the unrealistic extracted jumps and jump-related parameters inconsistent with the data. In fact, the VIX index can be represented as a deterministic function of the spot variance and risk-neutral parameters; therefore, the importance of appropriate jump-related parameters is relatively small in computing the VIX index, as the other risk-neutral parameters may compensate for the mis-speciation of the jumps-related parameters. However, when pricing options, I need to simulate

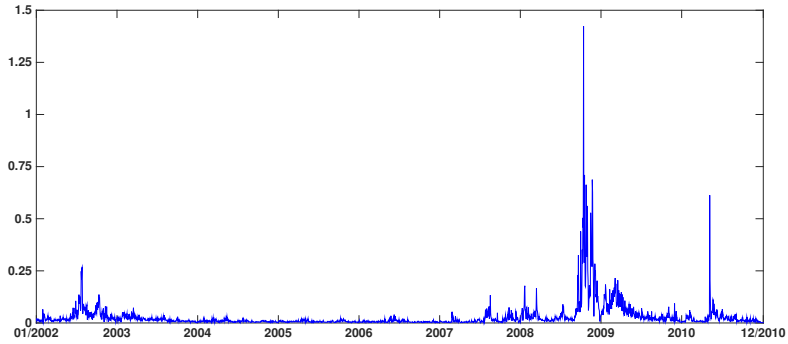


**Figure 4.22:** Annualised realised variance of the S&P 500 index in 2002 – 2010.

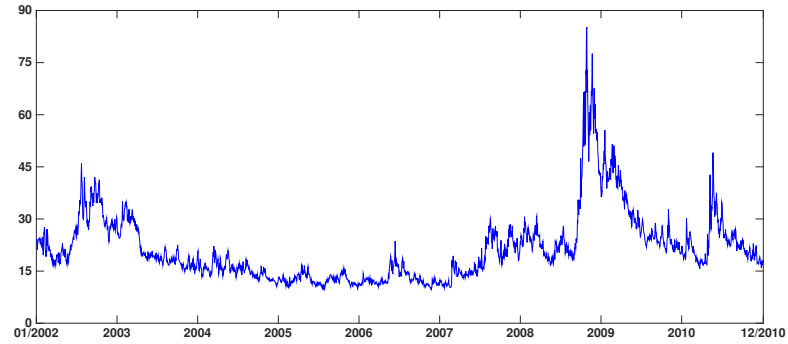


**Figure 4.23:** 30-day VIX of the S&P 500 index in 2002 – 2010.

jumps from the risk-neutral jump-related parameters, and the simulation with unrealistic jump-related parameters may miss the potential risks of jumps.

Another surprising observation is that the option pricing errors with the MH in Samples A, B, and C are relatively small. In particular, although the MH underperforms the FUSS-RC in Sample A, the MH ranks the second best in predicting future option prices (Samples C), better than the AM and FUSS-RC. Let us take a closer look at the pricing errors of the MH in individual years from 2006 – 2010. In fact, the pricing error of the MH in 2006 is larger than that of the FUSS-RC, because market conditions in 2006 were similar to those in 2002 – 2005. However, as the market became volatile, the MH demonstrates an increasing advantage over the AM and FUSS-RC. Recall that in the estimation with the MH, I use the realised variance in 2002 – 2005 as the initial variance. Since the extraction performance of the MH is sensitive to the choice of initial values, the use of realised variance leads to a volatile extracted path of variance and affects the parameter estimates, which are more consistent with the market conditions in 2006 – 2010 than those estimated with the AM and FUSS-RC. This may explain the remarkable performance of the MH in predicting future option prices, and importantly, this suggests the importance of initial values for the MH.

## 4.5 Discussion

In this chapter, I apply the advanced MCMC algorithms to estimate the SV-NIG-1 model using the joint information of the S&P 500 index returns and daily 30-day VIX data from January 2002 to December 2005. The algorithms comprise the AM, the FUSS-RC with pruning type 4, the PMMH-PF1/PF2, and PGAS, and their performance is compared to that of the conventional random-walk MH algorithm.

When extracting the strongly autocorrelated spot variance, the adaptive proposal used in the AM increases the acceptance rate of the MH and reduce the autocorrelation between samples of the spot variance. The FUSS-RC and PGAS further improve the extraction of the spot variance in that the ACFs of the spot variance extracted by the FUSS-RC and PGAS drop sharply, which is in line with the results of Martino et al. (2015) and Lindsten et al. (2014). Furthermore, the FUSS-RC even reduces the ACF with negative autocorrelation, suggesting that this algorithm is efficient in generating good representatives of the target distribution. In contrast, the acceptance rates of the PMMH-PF1 and the PMMH-PF2 are very low owing to the model specification of independent NIG return jumps.

Moreover, the choice of estimation methods may affect the option pricing performance. In particular, the model-implied VIX with the PMMH-PF1 almost completely coincides with the market VIX; therefore, since the option pricing performance correlates highly with the fit to the VIX, the PMMH-PF1 outperforms the other algorithms in the option pricing performance in all three samples. In contrast, the PGAS performs worst in all three samples owing to the poorly identified jump-related parameters under the risk-neutral measure. The MH ranks the second best in predicting option prices due to the use of realised variance as the initial variance, which makes the estimation results more consistent with the market conditions in 2006 – 2010 than those estimated with other algorithms.

To sum up, the PGAS can cope with strong dependence between parameters and state variables, and if one emphasises the fit to observations that can be represented as a function of model parameters and state variables, the PMMH-PF1 is very competitive. However, the performance of the PMMH-PF1 and the PGAS may be weakened by the inclusion of independent jumps. In contrast, the other algorithms are less vulnerable to the inclusion of independent jumps, and they achieve a stable performance in parameter estimation, extraction of state variables, fit to the VIX, and option pricing. In particular, an appropriate choice of initial values for state variables may significantly improve the performance of the MH; moreover, the FUSS-RC is very competitive because it can improve the estimation performance of the MH and AM and generate good representatives of the target distribution of state variables in the presence of high dimensions and a strong dependence structure.

# 5 Discussion and Conclusion

In this thesis, I first compare the use of infinite-activity VG/NIG jumps in the return process, or in both the return and variance processes, and the specification of the non-affine variance process, against the popular affine Heston, Bates, and double-Poisson-jump models without jumps or with finite-activity compound Poisson jumps. The models are estimated by conventional MCMC algorithms (Gibbs sampler, random-walk MH, and DWW methods) from an extensive data set of daily S&P 500 index returns and the reformulated 30-day VIX index from January 1996 to December 2009. Only a little research has examined the empirical option pricing performance of stochastic volatility models with infinite-activity Lévy jumps. Notable exceptions include Li et al. (2008), Yu et al. (2011), and Ornthanalai (2014). The difference between my research and theirs lies mainly in the use of non-affine variance dynamics and infinite-activity variance jumps.

Second, I compare the estimation performance of advanced MCMC algorithms comprising the AM (Haario et al., 1999a, 2001), the FUSS algorithms (Martino et al., 2015), and the PMCMC methods against conventional MH algorithms (Hastings, 1970; Metropolis et al., 1953). The PMCMC methods comprise the PMMH-PF1/PF2/APF (Andrieu et al., 2010) and the PGAS (Lindsten et al., 2014). The comparison is based on simulation and empirical studies. In particular, I first examine, from simulated data, how these algorithms perform in estimating the affine Heston-0, Bates-0, and SV-NIG-0 models with different jump structures and in dealing with the problems of high dimensions, complicated target distributions, and strong dependence between latent state variables and parameters. Then, the non-affine SV-NIG-1 model, which performs best in the model comparison, is estimated using the joint information of the daily S&P 500 index returns and the reformulated 30-day VIX from January 2002 to December 2005 with selected algorithms that perform well in the simulation studies. When some finance research has applied the conventional MCMC or SMC algorithms to estimate stochastic volatility models with jumps (see, for example, Christoffersen et al., 2010a; Eraker et al., 2003; Kaeck and Alexander, 2013a), only a few attempts have been made to apply the above newly proposed algorithms. These algorithms have been successfully applied in other fields of science. For example, the AM has been used in climate models (Järvinen et al., 2010; Solonen et al., 2012), and the PMCMC methods have been applied in biology models (Golightly and Wilkinson, 2011; Holenstein, 2009) and ecology models (Peters et al., 2010). Indeed, the problem of estimating stochastic volatility models with jumps is particularly difficult because of the strong dependence between parameters and state variables and the inclusion of jump components. The second part of the thesis seeks to examine how advanced estimation methods perform in estimating complicated financial models compared to conventional MH algorithms.

As for the model comparison, the empirical results in Chapter 2 clearly confirm the importance of non-affine variance process in terms of both the goodness of fit and

option pricing, as noted in the literature (see, for example, Christoffersen et al., 2010a; Kaeck and Alexander, 2012). Moreover, the non-affine variance specification can improve model robustness, whereas affine variance dynamics may lead to unstable option pricing performance across different option samples. More importantly, the improvement brought by the infinite-activity Lévy jumps, which has not been widely recognised, is critical with the non-affine variance specification. The empirical results indicate that the infinite-activity VG/NIG return jumps generate more realistic index dynamics and option prices, and capture the uncertainty of future jumps better, which is missed by the finite-activity compound Poisson jumps. Furthermore, the role of infinite-activity variance jumps is less important than that of infinite-activity return jumps. The empirical results show that the SV-NIG-1 markedly outperforms the others in pricing options and the SV-VG-NIG-1 achieves the best performance in goodness of fit. However, overall, the SV-VG-NIG-1 and SV-NIG-1 models are comparable, and considering the extra model complexity of the SV-VG-NIG-1, I conclude that the parsimonious model SV-NIG-1 is the more competitive one.

As for the algorithm comparison, the results of the simulation studies in Chapter 3 suggest that different jump structures may significantly affect the estimation performance of the algorithms. When estimating the Bates-0 model, all the algorithms are able to extract the compound Poisson jumps; however, the inclusion of rare large jumps reduces the acceptance rate of the PMCMC methods and worsens their estimation performance. Moreover, the extraction of the infinite-activity NIG jumps is difficult for all the algorithms, and the inclusion of NIG jumps makes the estimation of parameters and the extraction of variance very challenging. Considering the complexity and computational costs of different algorithms, I conclude that for models with simple specifications, the MH is very competitive because of its low computational cost. The AM further improves the performance of the MH because it can significantly raise the acceptance rate and accelerate the convergence of the chain by an online-tuned adaptive proposal. However, the MH and AM are less capable of dealing with complicated model specifications. The FUSS algorithms are particularly competitive in coping with strong dependence between state variables and in generating virtually independent samples from complicated target distributions, as noted in Martino et al. (2015). Generally, they achieve a good mixing with the smallest MCMC runs, and their performance is stable and excellent regardless of model specifications; however, they are relatively computationally expensive. In contrast, the PMCMC methods are computationally cheap and very efficient in mixing, and in the simulation studies, they outperform the FUSS algorithms in estimating the Heston-0 and SV-NIG-0 models. However, the PMCMC methods are more vulnerable to model specifications than the other algorithms; in particular, the inclusion of large independent jumps may significantly reduce the efficiency of the PMCMC methods.

The results of the empirical studies in Chapter 4 show that in extracting the strongly autocorrelated spot variance, the adaptive proposal used in the AM increases the acceptance rate of the MH and reduces the autocorrelation between samples of the spot variance. The FUSS-RC and PGAS further improve the extraction of the spot variance in that the ACFs of the spot variance extracted by the FUSS-RC and PGAS drop sharply, which is in line with the results in Martino et al. (2015) and Lindsten et al. (2014). Furthermore, the FUSS-RC even reduces the ACF with negative autocorrelation, suggesting that this algorithm is efficient in generating good representatives of the target distribution. In contrast, the acceptance rates of the PMMH algorithms are very low owing to the model specification of independent NIG jumps in the return process, and the jump-related parameters are poorly identified by the PGAS. This suggests that, compared to the MH,

AM, and FUSS-RC, the PMCMC methods are less capable of extracting independent jumps; moreover, the inclusion of independent jumps may negatively affect their estimation performance. Furthermore, the choice of estimation methods affects the option pricing performance. In particular, the model-implied VIX with the PMMH-PF1 almost completely coincides with the market VIX; therefore, since the option pricing performance correlated highly with the fit to the VIX, the PMMH-PF1 outperforms the others in the option pricing performance in all three option samples. In contract, the PGAS performs worst in all three samples owing to the poorly identified jump-related parameters under the risk-neutral measure. The MH ranks the second best in predicting option prices due to the use of the realised variance as the initial variance in the estimation, which leads to an estimation result consistent with the market conditions in 2006 – 2010.

In conclusion, according to the empirical results in the model comparison, there is clear evidence for the importance of the infinite-activity return jumps and non-affine variance dynamics in terms of the goodness of fit and option pricing performance, and the role of variance jumps is less important than that of return jumps. In particular, a relatively parsimonious model with infinite-activity NIG return jumps and non-affine variance dynamics is very competitive. The results of the simulation and empirical studies on the algorithm comparison suggest that each estimation algorithm has its own strengths and drawbacks, and no single algorithm has shown an overwhelming advantage. The PMCMC methods perform well in the presence of high dimensions and strong dependence between parameters and latent state variables, and if one emphasises the fit to data that can be represented as a function of model parameters and state variables, the PMMH-PF1 is particularly competitive; however, their performance may be weakened by the inclusion of large independent jumps. Despite the high computational cost, the FUSS algorithms are particularly competitive in generating virtually independent samples and in achieving the fastest mixing with a fixed number of MCMC runs, and their performance is stable regardless of model specifications.

Finally, in future research, it would be interesting to use a volatility term structure to identify the variance dynamics and to use a second factor to model the long-term mean of variance, as in Kaeck and Alexander (2012). Moreover, as pointed out in Andersen et al. (2014), a left tail factor is critical in forecasting the variance risk premium dynamics, and the time-varying jump risk premia improve the risk-neutral dynamics. It would be interesting to add the specification of time-varying jump risk premia to the current models, for example, by making the jump-related parameters time-varying, or by assuming another factor. As for the estimation methods, a number of extensions to the PMCMC methods, as mentioned in Andrieu et al. (2010), may further improve the estimation performance. In particular, a lot of literature has contributed to improving SMC algorithms, and the techniques discussed in the SMC literature, such as the PF scheme dealing with anomalous observations (Maiz et al., 2012), may be applied to the SMC update in the PMCMC methods. Moreover, the FUSS algorithms prove effective in this thesis; however, their high computational cost may limit their application. Therefore, it would be helpful to make the FUSS algorithms more efficient in pruning support points and constructing proposals. Further, the particle learning (Carvalho et al., 2010) has shown to be competitive to the MCMC, and this would be a promising direction for future research.

# Bibliography


Ait-Sahalia, Y. and Kimmel, R., "Maximum likelihood estimation of stochastic volatility models," *Journal of Financial Economics*, vol. 83, no. 2, pp. 413–452, 2007.

Amengual, D., "The term structure of variance risk premia," *Princeton University*, 2009.

An, S. and Schorfheide, F., "Bayesian analysis of DSGE models," *Econometric reviews*, vol. 26, no. 2-4, pp. 113–172, 2007.

Andersen, T. G., Fusari, N., and Todorov, V., "The risk premia embedded in index options," *Working paper, Northwestern University*, 2014.

Andersen, T. G. and Sørensen, B. E., "GMM estimation of a stochastic volatility model: a Monte Carlo study," *Journal of Business & Economic Statistics*, vol. 14, no. 3, pp. 328–352, 1996.

Andrieu, C., Doucet, A., and Holenstein, R., "Particle Markov Chain Monte Carlo methods," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 72, no. 3, pp. 269–342, 2010.

Bakshi, G., Cao, C., and Chen, Z., "Empirical performance of alternative option pricing models," *The Journal of Finance*, vol. 52, no. 5, pp. 2003–2049, 1997.

Barndorff-Nielsen, O. E., "Normal inverse Gaussian distributions and stochastic volatility modelling," *Scandinavian Journal of statistics*, vol. 24, no. 1, pp. 1–13, 1997.

———, "Processes of normal inverse Gaussian type," *Finance and stochastics*, vol. 2, no. 1, pp. 41–68, 1997.

Bates, D. S., "Jumps and stochastic volatility: Exchange rate processes implicit in deutsche mark options," *Review of financial studies*, vol. 9, no. 1, pp. 69–107, 1996.

———, "Post-'87 crash fears in the S&P 500 futures option market," *Journal of Econometrics*, vol. 94, no. 1, pp. 181–238, 2000.

———, "Maximum likelihood estimation of latent affine processes," *Review of Financial Studies*, vol. 19, no. 3, pp. 909–965, 2006.

———, "US stock market crash risk, 1926–2010," *Journal of Financial Economics*, vol. 105, no. 2, pp. 229–259, 2012.

Berg, A., Meyer, R., and Yu, J., "Deviance information criterion for comparing stochastic volatility models," *Journal of Business & Economic Statistics*, vol. 22, no. 1, pp. 107–120, 2004.

Berzuini, C., Best, N. G., Gilks, W. R., and Larizza, C., "Dynamic conditional independence models and Markov Chain Monte Carlo methods," *Journal of the American Statistical Association*, vol. 92, no. 440, pp. 1403–1412, 1997.

Black, F. and Scholes, M., "The pricing of options and corporate liabilities," *The journal of political economy*, pp. 637–654, 1973.

Broadie, M., Chernov, M., and Johannes, M., "Model specification and risk premia: Evidence from futures options," *The Journal of Finance*, vol. 62, no. 3, pp. 1453–1490, 2007.

Carr, P. and Madan, D. B., "Option valuation using the fast fourier transform," *Journal of computational finance*, vol. 2, no. 4, pp. 61–73, 1999.

———, "Optimal positioning in derivative securities," *Quantitative Finance*, vol. 1, pp. 19–37, 2001.

Carr, P. and Wu, L., "The finite moment log stable process and option pricing," *Journal of Finance*, vol. 58, pp. 753–777, 2003.

———, "Time-changed Lévy processes and option pricing," *Journal of Financial Economics*, vol. 71, no. 1, pp. 113–141, 2004.

Carr, P., Geman, H., and Madan, D. B., "The fine structure of asset returns: An empirical investigation," *Journal of Business*, vol. 75, pp. 305–332, 2002.

Carr, P., Geman, H., Madan, D. B., and Yor, M., "Stochastic volatility for Lévy processes," *Mathematical Finance*, vol. 13, no. 3, pp. 345–382, 2003.

Carvalho, C., Johannes, M., Lopes, H. F., and Polson, N., "Particle learning and smoothing," *Statistical Science*, vol. 25, no. 1, pp. 88–106, 2010.

Chacko, G. and Viceira, L. M., "Spectral GMM estimation of continuous-time processes," *Journal of Econometrics*, vol. 116, no. 1, pp. 259–292, 2003.

Chernov, M. and Ghysels, E., "A study towards a unified approach to the joint estimation of objective and risk neutral measures for the purpose of options valuation," *Journal of financial economics*, vol. 56, no. 3, pp. 407–458, 2000.

Chopin, N. and Singh, S. S., "On the particle Gibbs sampler," *arXiv preprint arXiv:1304.1887*, 2013.

Christoffersen, P. and Jacobs, K., "Which GARCH model for option valuation?" *Management Science*, vol. 50, no. 9, pp. 1204–1221, 2004.

Christoffersen, P., Dorion, C., Jacobs, K., and Wang, Y., "Volatility components, affine restrictions, and non-normal innovations," *Journal of Business & Economic Statistics*, vol. 28, no. 4, pp. 483–502, 2010.

Christoffersen, P., Jacobs, K., and Mimouni, K., "Volatility dynamics for the S&P500: evidence from realized volatility, daily returns, and option prices," *Review of Financial Studies*, p. hhq032, 2010.

Christoffersen, P., Jacobs, K., and Ornthanalai, C., "GARCH option valuation: Theory and evidence," *Available at SSRN 2054859*, 2013.

Creal, D., "Analysis of filtering and smoothing algorithms for Lévy-driven stochastic volatility models," *Computational Statistics & Data Analysis*, vol. 52, no. 6, pp. 2863–2876, 2008.

–––, "A survey of sequential Monte Carlo methods for economics and finance," *Econometric Reviews*, vol. 31, no. 3, pp. 245–296, 2012.

Damlen, P., Wakefield, J., and Walker, S., "Gibbs sampling for Bayesian non-conjugate and hierarchical models by using auxiliary variables," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 61, no. 2, pp. 331–344, 1999.

Djuric, P. M., Kotecha, J. H., Zhang, J., Huang, Y., Ghirmai, T., Bugallo, M. F., and Miguez, J., "Particle filtering," *Signal Processing Magazine, IEEE*, vol. 20, no. 5, pp. 19–38, 2003.

Doucet, A., De Freitas, N., and Gordon, N., *An introduction to sequential Monte Carlo methods*. Springer, 2001.

Duan, J. C. and Yeh, C. Y., "Jump and volatility risk premiums implied by VIX," *Journal of Economic Dynamics and Control*, vol. 34, no. 11, pp. 2232–2244, 2010.

–––, "Price and volatility dynamics implied by the VIX term structure," *Available at SSRN 1788252*, 2011.

Duffie, D., Pan, J., and Singleton, K., "Transform analysis and asset pricing for affine jump-diffusions," *Econometrica*, vol. 68, no. 6, pp. 1343–1376, 2000.

Duffie, D., Filipović, D., and Schachermayer, W., "Affine processes and applications in finance," *Annals of applied probability*, pp. 984–1053, 2003.

Eberlein, E. and Madan, D. B., "On correlating Lévy processes," *Robert H. Smith School Research Paper No. RHS*, pp. 06–118, 2009.

Eraker, B., "Do stock prices and volatility jump? Reconciling evidence from spot and option prices," *The Journal of Finance*, vol. 59, no. 3, pp. 1367–1404, 2004.

Eraker, B., Johannes, M., and Polson, N., "The impact of jumps in volatility and returns," *The Journal of Finance*, vol. 58, no. 3, pp. 1269–1300, 2003.

Fernández-Villaverde, J. and Rubio-Ramírez, J. F., "Estimating dynamic equilibrium economies: linear versus nonlinear likelihood," *Journal of Applied Econometrics*, vol. 20, no. 7, pp. 891–910, 2005.

Gelman, A., Roberts, G., and Gilks, W., "Efficient Metropolis jumping hules," *Bayesian statistics*, vol. 5, no. 599-608, p. 42, 1996.

Gilks, W. R., "Derivative-free Adaptive Rejection Sampling for Gibbs Sampling," *Bayesian Statistics*, vol. 4, pp. 641–649, 1992.

Gilks, W. R. and Wild, P., "Adaptive Rejection Sampling for Gibbs Sampling," *Applied Statistics*, vol. 41, no. 2, pp. 337–348, 1992.

Gilks, W. R., Best, N. G., and Tan, K. K. C., "Adaptive Rejection Metropolis Sampling within Gibbs Sampling," *Applied Statistics*, vol. 44, no. 4, pp. 455–472, 1995.

Gilks, W. R., Neal, R., Best, N. G., and Tan, K. K. C., "Corrigidum: Adaptive Rejection Metropolis Sampling within Gibbs Sampling," *Applied Statistics*, vol. 46, no. 4, pp. 541–542, 1997.

Golightly, A. and Wilkinson, D. J., "Bayesian parameter inference for stochastic biochemical network models using particle markov chain monte carlo," *Interface focus*, p. rsfs20110047, 2011.

Gordon, N. J., Salmond, D. J., and Smith, A. F., "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," in *IEE Proceedings F (Radar and Signal Processing)*, vol. 140, no. 2. IET, 1993, pp. 107–113.

Haario, H., Saksman, E., and Tamminen, J., "Adaptive proposal distribution for random walk Metropolis algorithm," *Computational Statistics*, vol. 14, pp. 375–395, 1999.

———, "Adaptive proposal distribution for random walk Metropolis algorithm," *Computational Statistics*, vol. 14, no. 3, pp. 375–396, 1999.

———, "An adaptive Metropolis algorithm," *Bernoulli*, pp. 223–242, 2001.

Haario, H., Laine, M., Mira, A., and Saksman, E., "DRAM: efficient adaptive MCMC," *Statistics and Computing*, vol. 16, no. 4, pp. 339–354, 2006.

Hammersley, J. M. and Clifford, P., "Markov fields on finite graphs and lattices," 1971.

Hastings, W. K., "Monte carlo sampling methods using markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.

Heston, S. L., "A closed-form solution for options with stochastic volatility with applications to bond and currency options," *Review of Financial Studies*, vol. 6, no. 2, pp. 327–343, 1993.

Heston, S. L. and Nandi, S., "A closed-form GARCH option valuation model," *Review of Financial Studies*, vol. 13, no. 3, pp. 585–625, 2000.

Holenstein, R., "Particle markov chain monte carlo," Ph.D. dissertation, The University Of British Columbia (Vancouver, 2009.

Ignatieva, K., Rodrigues, P., and Seeger, N., "Stochastic volatility and jumps: Exponentially affine yes or no? An empirical analysis of S&P500 dynamics," *Journal of Business and Economic Statistics*, 2015, forthcoming.

Jacquier, E., Polson, N. G., and Rossi, P. E., "Bayesian analysis of stochastic volatility models," *Journal of Business & Economic Statistics*, vol. 20, no. 1, pp. 69–87, 2002.

Jacquier, E., Johannes, M., and Polson, N., "MCMC maximum likelihood for latent state models," *Journal of Econometrics*, vol. 137, no. 2, pp. 615–640, 2007.

Järvinen, H., Räisänen, P., Laine, M., Tamminen, J., Ilin, A., Oja, E., Solonen, A., and Haario, H., "Estimation of echam5 climate model closure parameters with adaptive mcmc," *Atmospheric Chemistry and Physics*, vol. 10, no. 20, pp. 9993–10 002, 2010.

Johannes, M. and Polson, N., "MCMC methods for continuous-time financial econometrics," *Available at SSRN 480461*, 2003.

Johansen, A. M., Doucet, A., and Davy, M., "Particle methods for maximum likelihood estimation in latent variable models," *Statistics and Computing*, vol. 18, no. 1, pp. 47–57, 2008.

Jones, C. S., "The dynamics of stochastic volatility: evidence from underlying and options markets," *Journal of Econometrics*, vol. 116, no. 1, pp. 181–224, 2003.

Kaeck, A. and Alexander, C., "Volatility dynamics for the S&P500: Further evidence from non-affine, multi-factor jump diffusions," *Journal of Banking & Finance*, vol. 36, no. 11, pp. 3110–3121, 2012.

———, "Stochastic volatility jump-diffusions for European equity index dynamics," *European Financial Management*, vol. 19, no. 3, pp. 470–496, 2013.

———, "Continuous-time VIX dynamics: On the role of stochastic volatility of volatility," *International Review of Financial Analysis*, vol. 28, pp. 46–56, 2013.

Kallsen, J., "A didactic note on affine stochastic volatility models," in *From stochastic calculus to mathematical finance*. Springer, 2006, pp. 343–368.

Kanniainen, J., Lin, B., and Yang, H., "Estimating and using GARCH models with VIX data for option valuation," *Journal of Banking & Finance*, vol. 43, pp. 200–211, 2014.

Koch, K. R., "Gibbs sampler by sampling-importance-resampling," *Journal of Geodesy*, vol. 81, no. 9, pp. 581–591, 2007.

Lewis, A. L., "Option valuation under stochastic volatility," *Option Valuation under Stochastic Volatility*, 2000.

Li, H., Wells, M., and Yu, C., "A Bayesian analysis of return dynamics with Lévy jumps," *Review of Financial Studies*, vol. 21, no. 5, pp. 2345–2378, 2008.

Li, J., "Sequential Bayesian analysis of time-changed infinite activity derivatives pricing models," *Journal of Business & Economic Statistics*, vol. 29, no. 4, 2011.

Liang, F., Liu, C., and Caroll, R., *Advanced Markov Chain Monte Carlo Methods: Learning from Past Samples*. England: Wiley Series in Computational Statistics, 2010.

Lindsten, F. and Schön, T. B., "Backward simulation methods for Monte Carlo statistical inference." *Foundations and Trends in Machine Learning*, vol. 6, no. 1, pp. 1–143, 2013.

Lindsten, F., Jordan, M. I., and Schön, T. B., "Particle Gibbs with ancestor sampling," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 2145–2184, 2014.

Liu, J. S., *Monte Carlo Strategies in Scientific Computing*. Springer, 2004.

Liu, J. S. and Chen, R., "Sequential monte carlo methods for dynamic systems," *Journal of the American statistical association*, vol. 93, no. 443, pp. 1032–1044, 1998.

Madan, D., Carr, P., and Chang, E., "The Variance Gamma process and option pricing," *European finance review*, vol. 2, no. 1, pp. 79–105, 1998.

Maiz, C. S., Molanes-Lopez, E. M., Míguez, J., and Djurić, P. M., "A particle filtering scheme for processing time series corrupted by outliers," *Signal Processing, IEEE Transactions on*, vol. 60, no. 9, pp. 4611–4627, 2012.

Martino, L., Read, J., and Luengo, D., "Improved adaptive rejection Metropolis sampling algorithms," *arXiv:1205.5494*, pp. 1 – 32, Oct. 2012.

Martino, L., Casarin, R., Leisen, F., and Luengo, D., "Adaptive sticky generalized Metropolis," *arXiv:1308.3779*, pp. 1 – 44, Sep. 2013.

Martino, L., Read, J., and luengo, D., "Independent doubly adaptive rejection Metropolis sampling," in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2014, pp. 8048 – 8052.

Martino, L., Yang, H., Luengo, D., Kanniainen, J., and Corander, J., "A fast universal self-tuned sampler within Gibbs sampling," *Digital Signal Processing*, 2015, forthcoming.

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., and Teller, E., "Equations of state calculations by fast computing machines," *Journal of Chemical Physics*, vol. 21, pp. 1087 – 1091, 1953.

Meyer, R., Cai, B., and Perron, F., "Adaptive rejection Metropolis sampling using Lagrange interpolation polynomials of degree 2," *Computational Statistics and Data Analysis*, vol. 52, no. 7, pp. 3408 – 3423, March 2008.

Müller, P., "Monte Carlo integration in general dynamic models," *Contemporary Mathematics*, vol. 115, pp. 145 – 163, 1991.

Ornthanalai, C., "Lévy jump risk: Evidence from options and returns," *Journal of Financial Economics*, vol. 112, no. 1, pp. 69 – 90, 2014.

Pan, J., "The jump-risk premia implicit in options: Evidence from an integrated time-series study," *Journal of Financial Economics*, vol. 63, no. 1, pp. 3 – 50, 2002.

Peters, G. W., Hosack, G. R., and Hayes, K. R., "Ecological non-linear state space model selection via adaptive particle Markov Chain Monte Carlo (AdPMCMC)," *arXiv preprint arXiv:1005.2238*, 2010.

Pitt, M. K. and Shephard, N., "Auxiliary variable based particle filters," in *Sequential Monte Carlo methods in practice*. Springer, 2001, pp. 273 – 293.

Robert, C. P. and Casella, G., *Monte Carlo Statistical Methods*. Springer, 2004.

Sato, K., "Lévy processes and infinite divisibility," 1999.

Solonen, A., Ollinaho, P., Laine, M., Haario, H., Tamminen, J., Järvinen, H. *et al.*, "Efficient mcmc for climate model parameter estimation: Parallel adaptive chains and early rejection," *Bayesian Analysis*, vol. 7, no. 3, pp. 715 – 736, 2012.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., and Van Der Linde, A., "Bayesian measures of model complexity and fit," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 64, no. 4, pp. 583 – 639, 2002.

Storvik, G., "Particle filters for state-space models with the presence of unknown static parameters," *Signal Processing, IEEE Transactions on*, vol. 50, no. 2, pp. 281 – 289, 2002.

Tankov, P., *Financial modelling with jump processes*. CRC press, 2003.

Wu, X., *Stochastic volatility with Lévy processes: calibration and pricing.* University of Maryland, 2005.

Yu, L., Li, H., and Wells, M. T., "MCMC estimation of Lévy jump models using stock and option prices," *Mathematical Finance*, vol. 21, no. 3, pp. 383–422, 2011.

# Appendices

# A Simulation Studies: Extracted Variance

In this appendix, I report the spot variance extracted by different algorithms in the Heston-0, Bates-0, and SV-NIG-0 models. The extracted path is plotted in a blue solid line and the true path is plotted in a red dotted-dashed line.

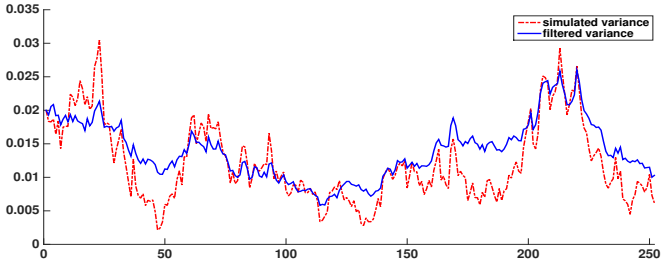


**Figure A.1:** Random-walk MH: extracted variance of the Heston-0 model.

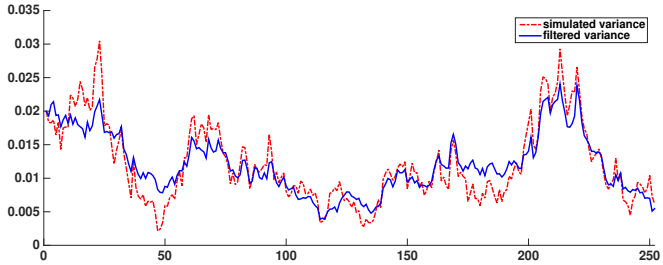


**Figure A.2:** AM: extracted variance of the Heston-0 model.

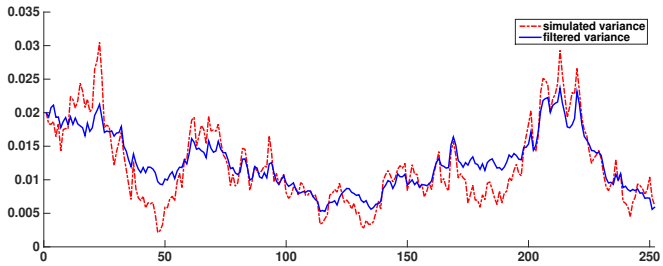


**Figure A.3:** FUSS-RC-P4: extracted variance of the Heston-0 model.

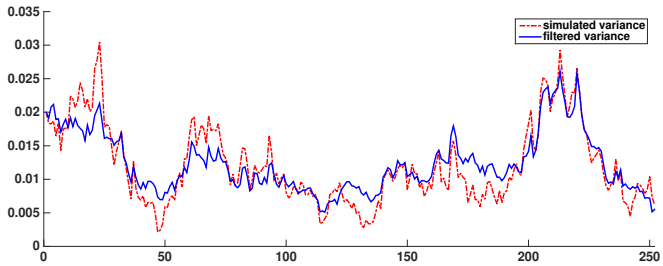


**Figure A.4:** FUSS-MH-P4: extracted variance of the Heston-0 model.

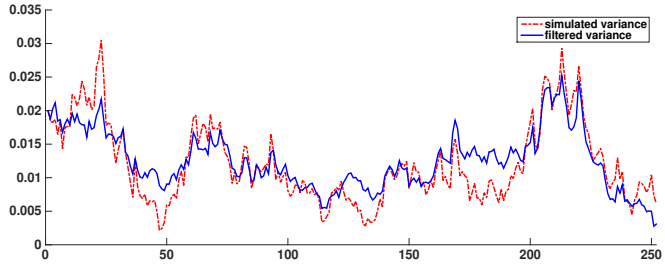


**Figure A.5:** PMMH-PF1: extracted variance of the Heston-0 model.

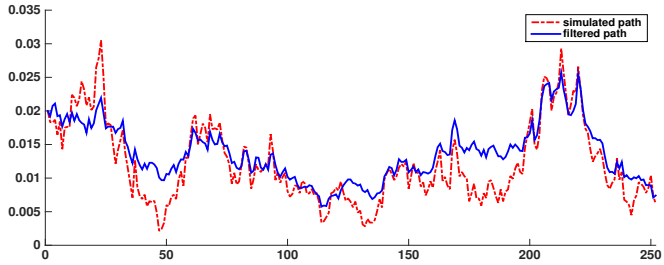


**Figure A.6:** PMMH-PF2: extracted variance of the Heston-0 model.

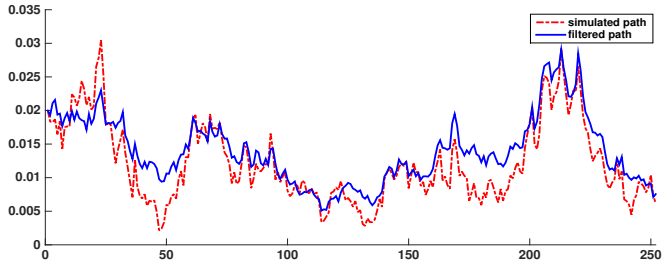


**Figure A.7:** PGAS: extracted variance of the Heston-0 model.

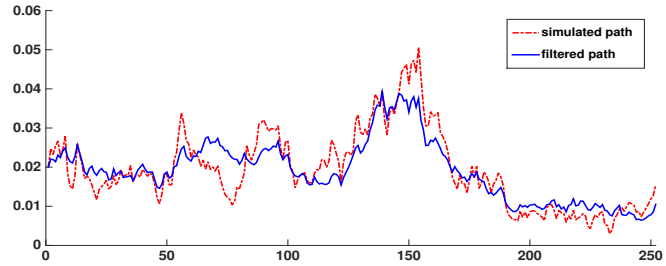


**Figure A.8:** Random-walk MH: extracted variance of the Bates-0 model.

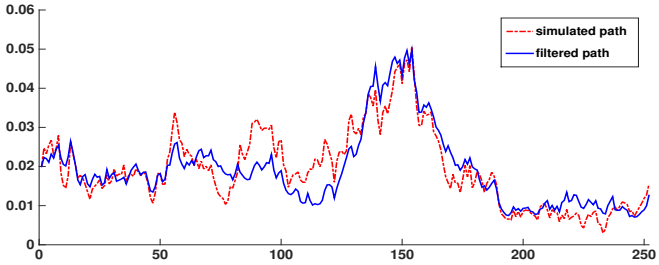


**Figure A.9:** AM: extracted variance of the Bates-0 model.

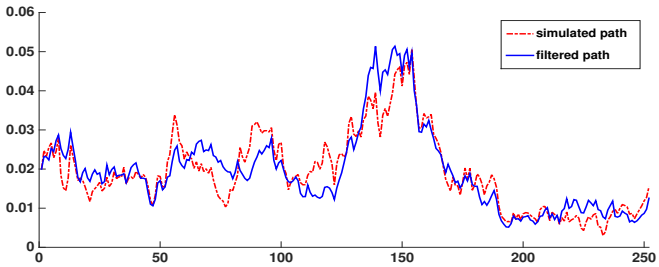


**Figure A.10:** FUSS-RC-P4: extracted variance of the Bates-0 model.

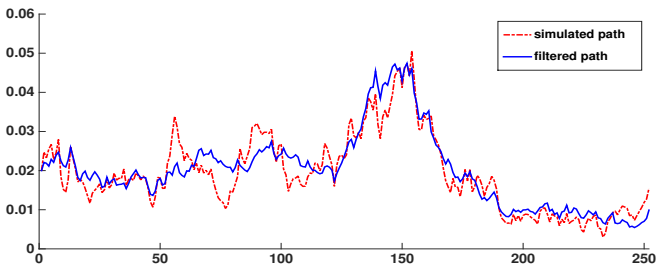


**Figure A.11:** FUSS-MH-P4: extracted variance of the Bates-0 model.

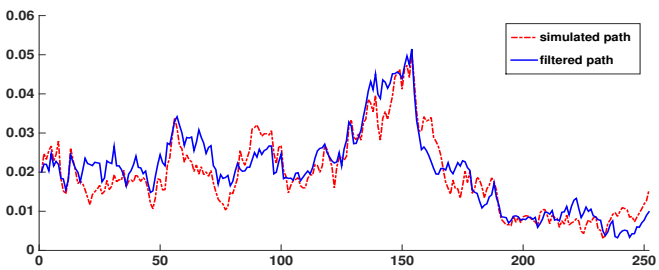


**Figure A.12:** PMMH-PF1: extracted variance of the Bates-0 model.

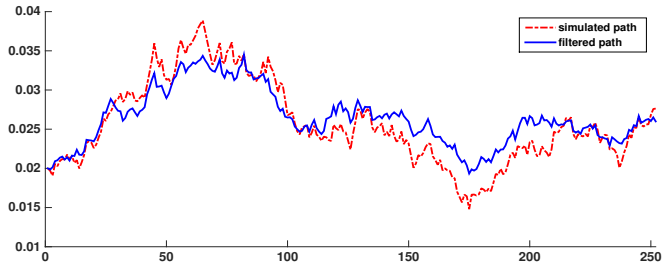


**Figure A.13:** Random-walk MH: extracted variance of the SV-NIG-0 model.

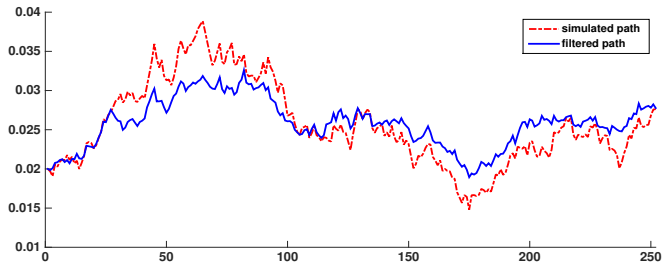


**Figure A.14:** AM: extracted variance of the SV-NIG-0 model.

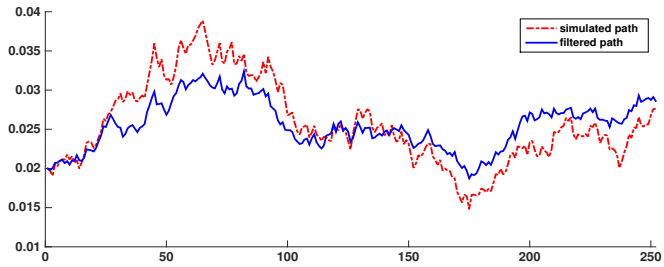


**Figure A.15:** FUSS-MH-P4: extracted variance of the SV-NIG-0 model.

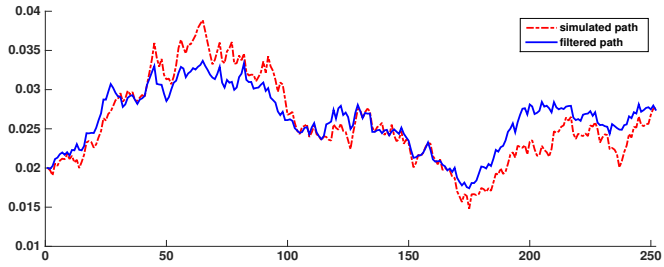


**Figure A.16:** FUSS-RC-P4: extracted variance of the SV-NIG-0 model.

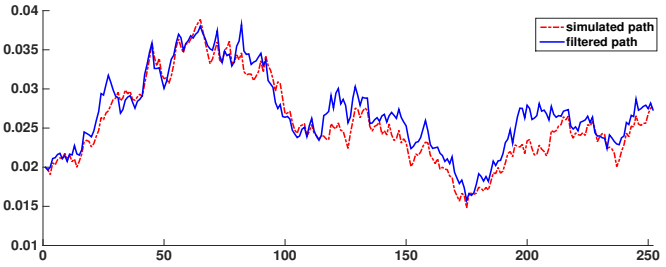


**Figure A.17:** PMMH-PF1: extracted variance of the SV-NIG-0 model.

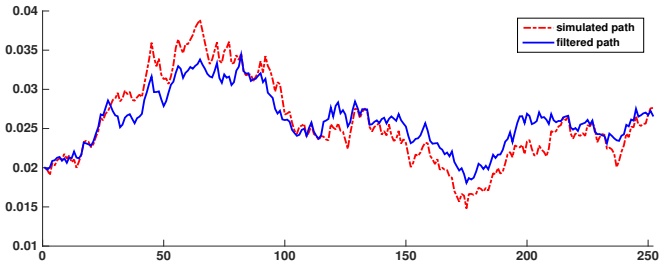


**Figure A.18:** PGAS: extracted variance of the SV-NIG-0 model.

# B Simulation Studies: Extracted Jumps

In this appendix, I report the jumps extracted by different algorithms in the Bates-0 and SV-NIG-0 models. The extracted jumps are plotted in a blue solid line and the true jumps are plotted in a red dotted-dashed line.

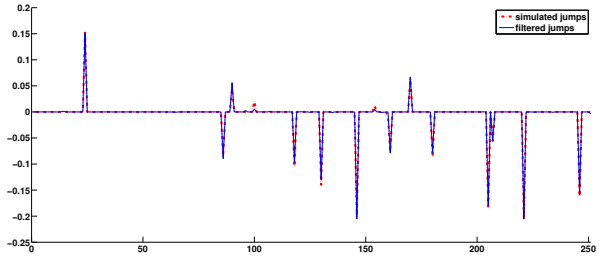


**Figure B.1:** Random-walk MH: extracted jumps of the Bates-0 model.

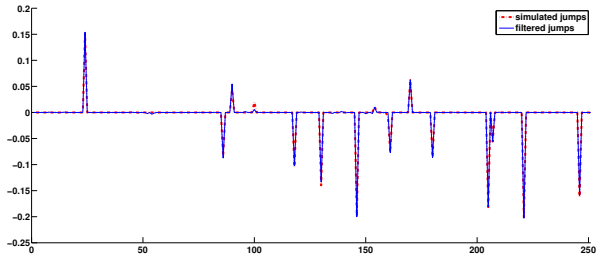


**Figure B.2:** AM: extracted jumps of the Bates-0 model.

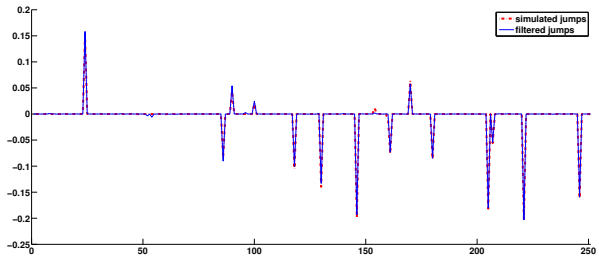


**Figure B.3:** FUSS-RC-P4: extracted jumps of the Bates-0 model.

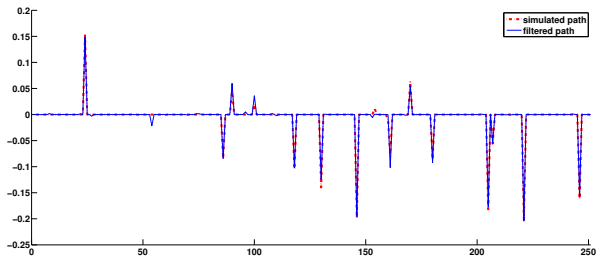


**Figure B.4:** FUSS-MH-P4: extracted jumps of the Bates-0 model.

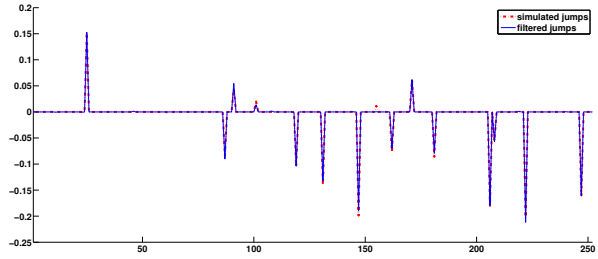


**Figure B.5:** PMMH-PF: extracted jumps of the Bates-0 model.

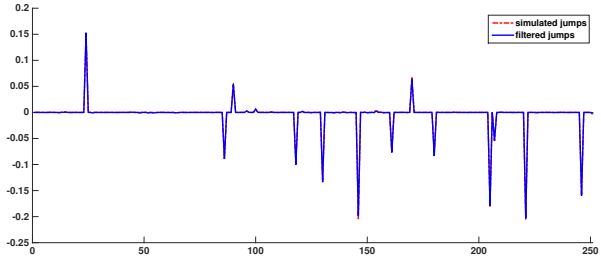


**Figure B.6:** PGAS: extracted jumps of the Bates-0 model.

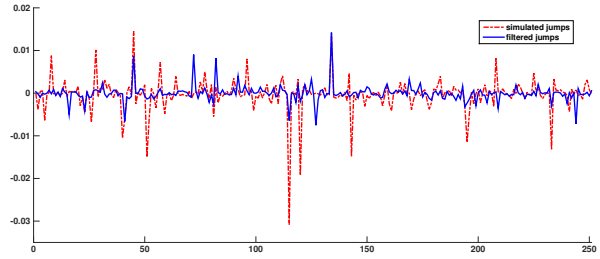


**Figure B.7:** random-walk MH: extracted jumps of the SV-NIG-0 model.

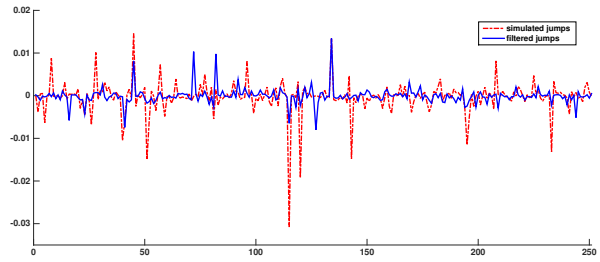


**Figure B.8:** AM: extracted jumps of the SV-NIG-0 model.

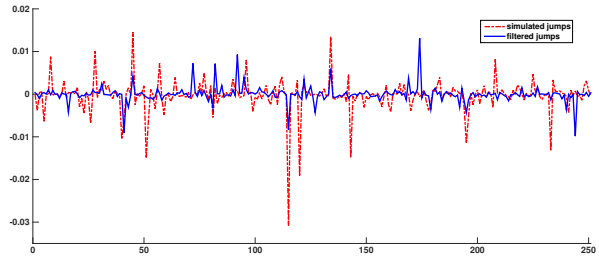


**Figure B.9:** FUSS-RC-P4: extracted jumps of the SV-NIG-0 model.

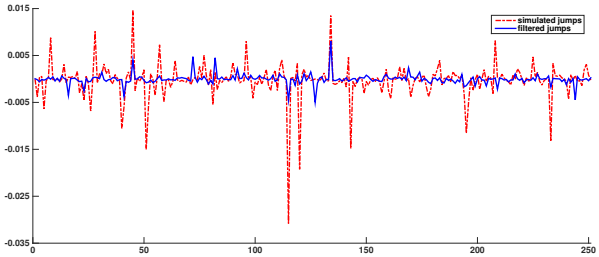


**Figure B.10:** PMMH-PF: extracted jumps of the SV-NIG-0 model.

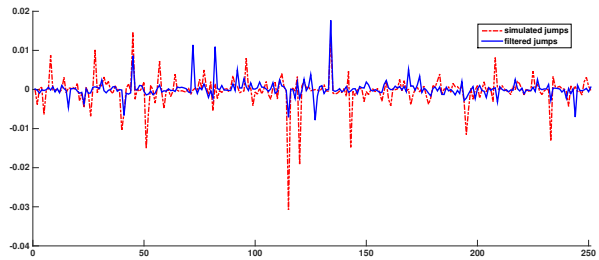


**Figure B.11:** PGAS: extracted jumps of the SV-NIG-0 model.

# C Empirical Studies: Extracted Variance of the SV-NIG-1 Model

In this appendix, I report the spot variance extracted by different algorithms in the SV-NIG-1 model.

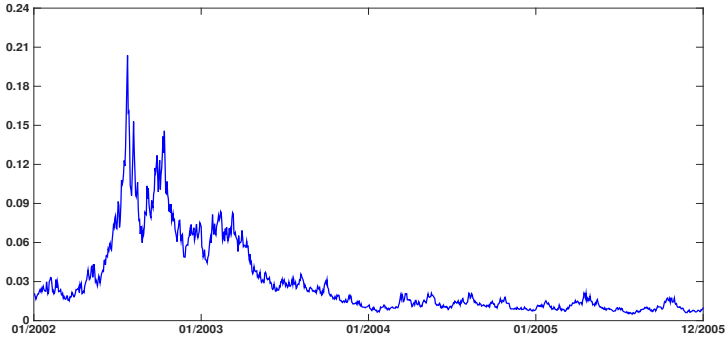


**Figure C.1:** Random-walk MH: extracted variance of the SV-NIG-1 model.

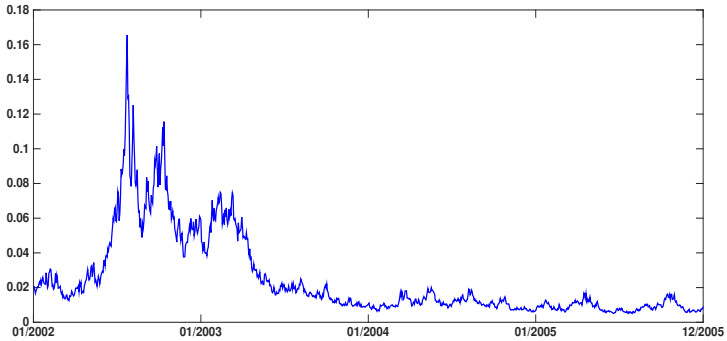


**Figure C.2:** AM: extracted variance of the SV-NIG-1 model.

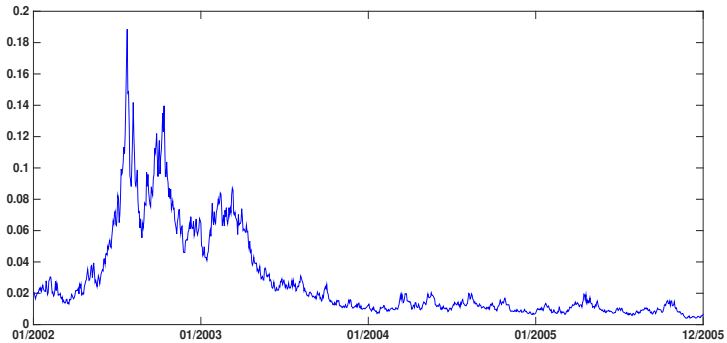


**Figure C.3:** FUSS-RC-P4: extracted variance of the SV-NIG-1 model.

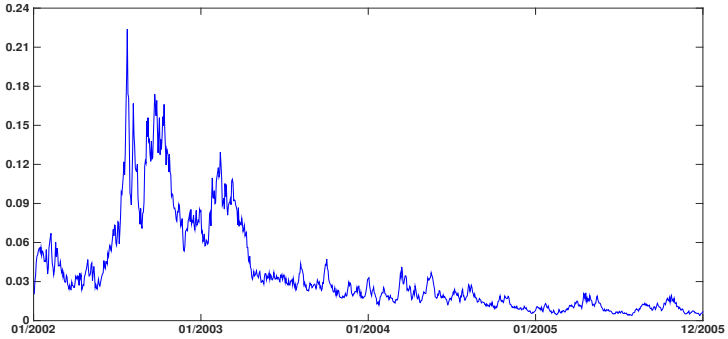


**Figure C.4:** PMMH-PF1: extracted variance of the SV-NIG-1 model.

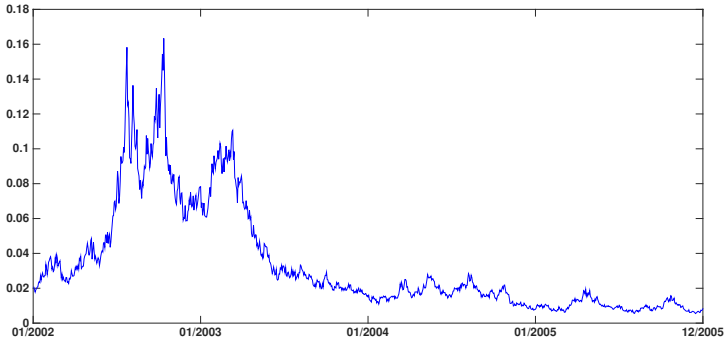


**Figure C.5:** PMMH-PF2: extracted variance of the SV-NIG-1 model.

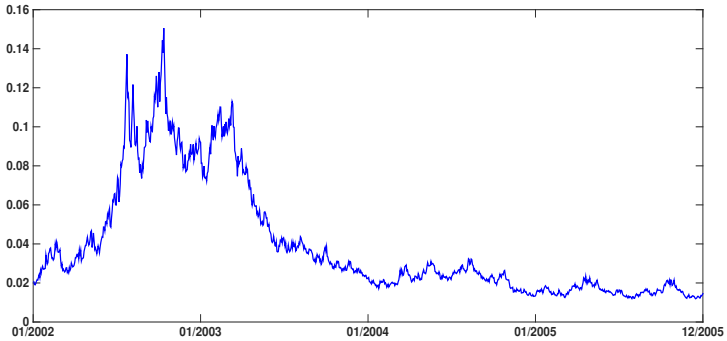


**Figure C.6:** PGAS: extracted variance of the SV-NIG-1 model.

# D Empirical Studies: Model-implied VIX of the SV-NIG-1 Model

In this appendix, I report the model-implied VIX based on parameter estimates and extracted variance of different algorithms. The model-implied VIX is plotted in a blue solid line and the market VIX is plotted in a red dotted-dashed line.

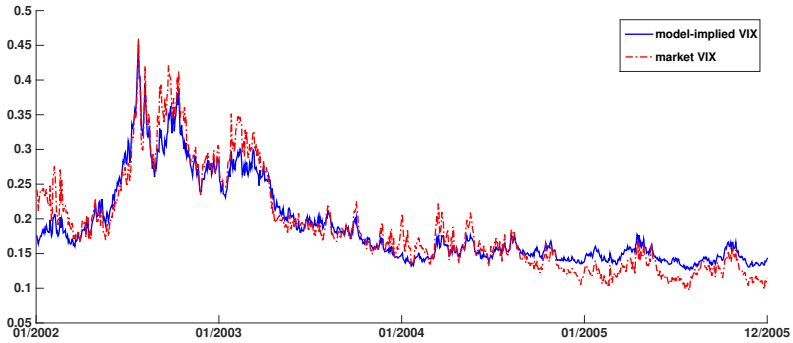


**Figure D.1:** Random-walk MH: the model-implied 30-day VIX compared against the market 30-day VIX in 2002-2005.

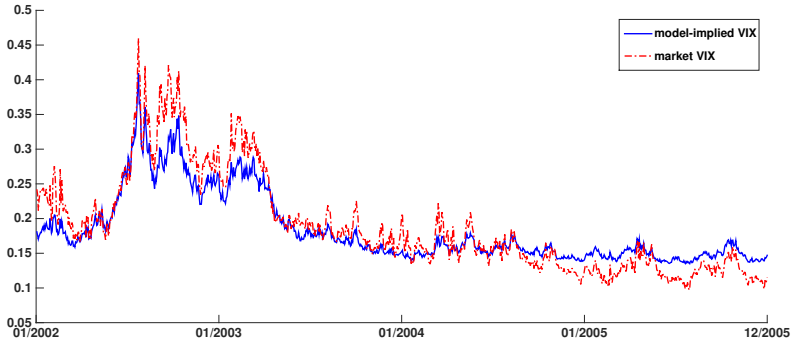


**Figure D.2:** AM: the model-implied 30-day VIX compared against the market 30-day VIX in 2002-2005.

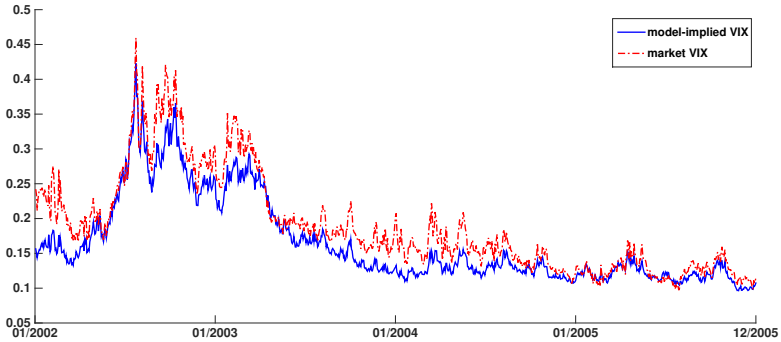


**Figure D.3:** FUSS-RC-P4: the model-implied 30-day VIX compared against the market 30-day VIX in 2002-2005.

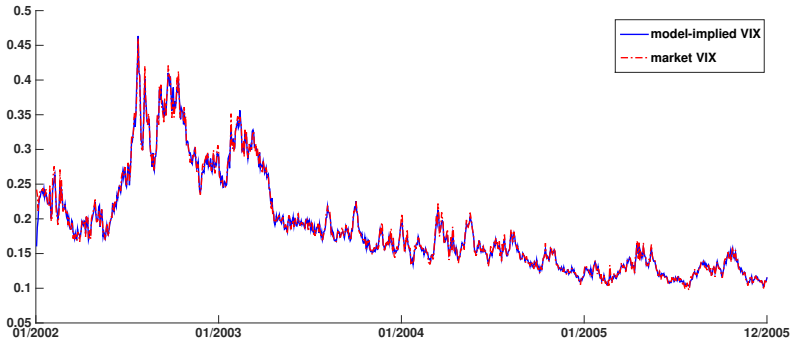


**Figure D.4:** PMMH-PF1: the model-implied 30-day VIX compared against the market 30-day VIX in 2002-2005.

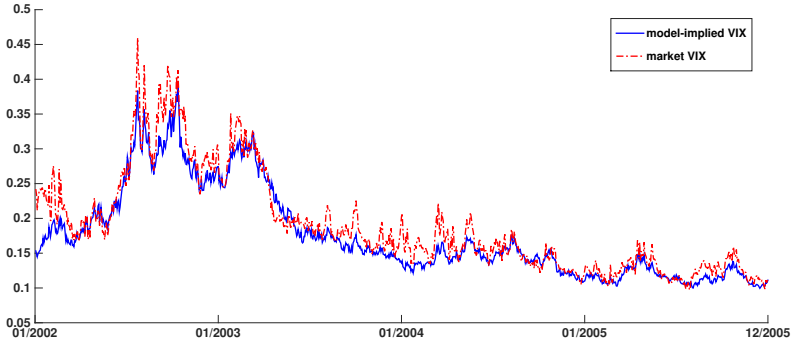


**Figure D.5:** PMMH-PF2: the model-implied 30-day VIX compared against the market 30-day VIX in 2002-2005.

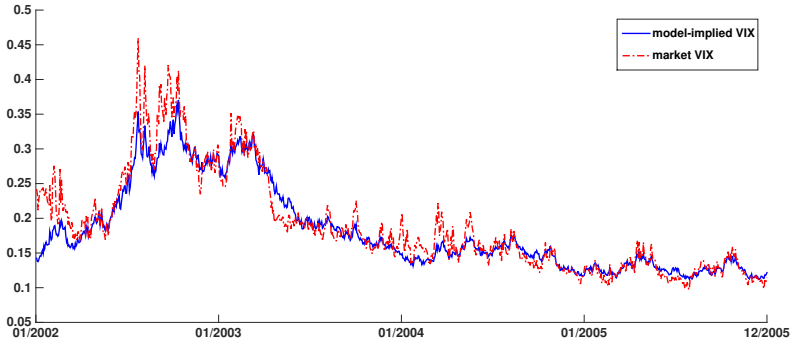


**Figure D.6:** PGAS: the model-implied 30-day VIX compared against the market 30-day VIX in 2002-2005.

Tampereen teknillinen yliopisto
PL 527
33101 Tampere

Tampere University of Technology
P.O.B. 527
FI-33101 Tampere, Finland

ISBN 978-952-15-3597-0
ISSN 1459-2045