MCMC Maximum Likelihood For Latent State Models

Eric Jacquier[a*], Michael Johannes[b], Nicholas Polson[c]

[a] *CIRANO, CIREQ, HEC Montréal, 3000 Cote Sainte-Catherine, Montréal QC H3T 2A7*

[b] *Graduate School of Business, Columbia University, 3022 Broadway, NY, NY, 10027*

[c] *The University of Chicago Graduate School of Business, 5807 South Woodlawn, Chicago, IL 60637.*

First draft: June 2003

This draft: September 2005

Forthcoming Journal of Econometrics

Abstract

This paper develops a pure simulation-based approach for computing maximum likelihood estimates in latent state variable models using Markov Chain Monte Carlo methods (MCMC). Our MCMC algorithm simultaneously evaluates and optimizes the likelihood function without resorting to gradient methods. The approach relies on data augmentation, with insights similar to simulated annealing and evolutionary Monte Carlo algorithms. We prove a limit theorem in the degree of data augmentation and use this to provide standard errors and convergence diagnostics. The resulting estimator inherits the sampling asymptotic properties of maximum likelihood. We demonstrate the approach on two latent state models central to financial econometrics: a stochastic volatility and a multivariate jump-diffusion models. We find that convergence to the MLE is fast, requiring only a small degree of augmentation.

JEL Classification: C1, C11, C15, G1

Key words: MCMC, Maximum Likelihood, Optimization, Simulated Annealing, Evolutionary Monte-Carlo, Stochastic volatility, Jumps, Diffusion, Financial Econometrics.

This is a preprint version of the article. The final version may be found at < http://dx.doi.org/10.1016/j.jeconom.2005.11.017 >.

*Corresponding author. Email: eric.jacquier@hec.ca

1 Introduction

Computing maximum likelihood estimates (MLE) in latent variable models is notoriously difficult for three reasons. First, the likelihood for the parameters is not known in closed form. Computing it typically requires Monte Carlo methods to draw from the latent state distribution and then approximate the integral that appears in the likelihood. Second, a nonlinear search algorithm must be used to optimize this approximated likelihood over the parameters. Finally, asymptotic standard errors depend on numerical second order derivatives of this simulated function, which introduces further computational difficulties.

In this paper, we provide a Markov Chain Monte Carlo (MCMC) algorithm that simultaneously performs the evaluation and the optimization of the likelihood in latent state models.[1] Our methodology provides parameter estimates and standard errors, as well as the smoothing distribution of the latent state variables.

Our approach combines the insights of simulated annealing (SA) and evolutionary MCMC algorithms.[2] Like SA, the goal of our approach is simulation-based optimization without resorting to gradient methods. However, unlike SA, we do not require that the objective function, in our setting the likelihood $\mathcal{L}(\theta)$, can be directly evaluated. Simulated annealing generates samples from a sequence of densities, $\pi_{J^{(g)}}(\theta) \propto \mathcal{L}(\theta)^{J^{(g)}}$, where g indexes the length of the Markov Chain. As g increases, the control $J^{(g)}$ must also be increased so that $\pi_{J^{(g)}}(\theta)$ concentrates around its maximum, the MLE. In contrast, evolutionary MCMC generates J copies (or draws) of the parameter θ and a Markov chain over these copies. It often has better convergence properties than SA despite this increased dimensionality. However, neither simulated annealing nor standard evolutionary MCMC algorithms can be applied when the likelihood, $\mathcal{L}(\theta)$, is an integral over the latent variables and cannot be directly computed.

We solve this problem by using data augmentation of the latent state variables. That is,

[1] Alternative approaches developed for computing MLE's include the expectation-maximization algorithm of Dempster, Laird and Rubin (1982), Geyer's (1991) Monte Carlo maximum likelihood approach, and Besag (1974) and Doucet et al. (2002) for maximum a posteriori estimation. Most existing simulation-based MLE methods use some form of importance sampling, and require optimization, see our discussion in section 3.

[2] See, for example, Kirkpatrick et al. (1983) and Van Laarhoven and Aarts (1987) for simulated annealing, Liang, and Wong (2001) and Mueller (2000) for evolutionary MCMC.

we generate J independent copies of the latent state variables for a given parameter draw. In contrast to standard evolutionary MCMC, we do not need to generate copies of the parameter itself. Specifically, we define a joint distribution $\pi_J(\theta, \widetilde{X}^J)$ on the space of the parameters θ and the J copies of the latent state variables $\widetilde{X}^J$. Standard MCMC methods provide samples from $\pi_J(\theta, \widetilde{X}^J)$ by, for example, iteratively sampling from the conditional distributions, $\pi(\theta|\widetilde{X}^J)$ and $\pi(\widetilde{X}^J|\theta)$. As g increases for a given J, the parameter samples, $\theta^{J,(g)}$ converge to draws from the marginal $\pi_J(\theta)$. The augmented joint distribution $\pi_J(\theta, \widetilde{X}^J)$ has the special property that the marginal distribution of the parameters $\pi_J(\theta)$ is proportional to $\mathcal{L}(\theta)^J$, the likelihood function raised to the power J. Therefore, as in simulated annealing, $\pi_J(\theta)$ collapses to the maximum of $\mathcal{L}(\theta)$, thus the draws $\theta^{J,(g)}$ converge to the MLE, as J increases.

To help choose J and compute the MLE standard errors, we provide the asymptotic distribution, as J increases, of the marginal density $\pi_J(\theta)$. We show that it is approximately normal, centered at the MLE, and that its asymptotic variance-covariance matrix is the observed MLE information matrix appropriately scaled by J. Hence, we can compute MLE standard errors by simply scaling the MCMC draws. Moreover, we can diagnose convergence of the chain and determine the degree of augmentation J required, without knowing the true MLE by checking the normality of the scaled MCMC draws. As convergence to the point estimate likely occurs before convergence in distribution, this constitutes a tough test of convergence. Normality of the scaled draws is checked informally with normality plots, and can be tested formally with, for example, a Jarque-Bera test.

Our approach has several practical and theoretical advantages. First, unlike the other simulated maximum likelihood approaches for state estimation, that substitute the final parameter estimates into an approximate filter, our algorithm also provides the optimal smoothing distribution of the latent variable, that is, its distribution at time t conditional on observing the entire sample from time. This is especially important in non-linear or non-normal latent variable models for which the Kalman filter is misspecified, see for example Carlin, Polson and Stoffer (1992) or Jacquier, Polson and Rossi (1994). Second, the algorithm has the advantages of MCMC without the disadvantages sometimes perceived. For example, unlike a Bayesian approach, we do not *per se* require prior distributions over the parameters. However, we do

need conditional and joint distributions to be integrable. Third, we compute the MLE estimate without resorting to numerical search algorithms, such as inefficient gradient based methods, which often get locked into local maxima. Our approach also handles models with nuisance parameters and latent variables as well as constrained parameters or parameters on boundaries.[3] Finally, our estimator inherits the asymptotic properties, in sample size, of the MLE.

It is important to compare and contrast our MCMC approach with the quasi-Bayes MCMC procedure proposed by Chernozhukov and Hong (2003). Their methodology applies to a wide class of criterion function. Instead of finding the maximum of the criterion, they advocate estimating its mean or quantiles. They show that their estimators have good asymptotic properties as the sample size increases. This is because they exploit the asymptotic properties of Bayes estimators, albeit with flat priors in their case (see, for example, Dawid, 1970, Heyde and Johnstone, 1978, or Schervish, 1995). It will become clear that this approach corresponds to $J = 1$ in our framework. In contrast, we show how to compute the maximum of the criterion function, the likelihood in this paper, by increasing J for a fixed sample size. As Chernozhukov and Hong note in their appendix, their method, applied to the power of the criterion essentially turns it into a form of simulated annealing algorithm. Unfortunately, SA, unlike the algorithm presented here, requires that the criterion function be analytically known and can not handle likelihood functions generated by models with latent state variables.

To illustrate our approach, we analyze two benchmark models in financial econometrics. The first is the standard log-stochastic volatility model (SV) of Taylor (1986), initially analyzed with Bayesian MCMC by Jacquier, Polson and Rossi (1994). The second is a multivariate version of Merton's (1976) jump-diffusion model. This model is of special interest in asset pricing because it delivers closed-form option prices. It is however difficult to estimate with standard methods given the well-known degeneracies of the likelihood, see, for example, Kiefer (1978). In another implementation, Boivin and Giannoni (2005) use our approach on a high dimensional macro-model for which computing the MLE by standard methods is impractical. For both models implemented here, the approach is computationally fast, of the same order

[3]Pastorello, Patilea and Renault (2003) address a case where the latent variables are a deterministic function of the observables when conditioned on the parameters.

of CPU time as popular Bayesian MCMC methods, as CPU time is linear in J and G, and convergence to the MLE occurs with low values of J.

Our approach also applies to other problems in economics and finance that require joint integration and optimization. Standard expected utility problems are an excellent example of this, as the agent first integrates out the uncertainty to compute expected utility and then maximizes it. In Jacquier, Johannes and Polson (2005) we extend the existing approach to maximum expected utility portfolio problems.

The rest of the paper proceeds as follows. Section 2 provides the general methodology together with the convergence proofs and the details of the convergence properties of the algorithm. Sections 3 and 4 provide simulation based evidence for two commonly used latent state variable models in econometrics, the log-stochastic volatility model and a multivariate jump-diffusion model. Finally, Section 5 concludes with directions for future research.

2 Simulation-based Likelihood Inference

Latent state variable models abound in finance and economics. In finance, latent state variables are used for example to model time-varying equity premium or volatility, jumps, and regime-switching. In economics, models using latent state variables include random utility discrete-choice models, censored and truncated regressions, and panel data models with missing data.

Formally, let $Y = (Y_1, ..., Y_T)$ denote the observed data, $X = (X_1, ..., X_T)$ the latent state vector, and θ a parameter vector. The marginal likelihood of θ is defined as

$$\mathcal{L}(\theta) = \int p(Y|X,\theta)p(X|\theta)dX, \tag{1}$$

where $p(Y|X,\theta)$ is the full-information or augmented likelihood function, and $p(X|\theta)$ is the distribution of the latent state variables.

Directly maximizing $\mathcal{L}(\theta)$ is difficult for several reasons. First, $\mathcal{L}(\theta)$ is rarely known in closed form. To evaluate it, one must first generate samples from $p(X|\theta)$ and then approximate

the integral with Monte Carlo methods. Even though it is sometimes possible to draw directly from $p(X|\theta)$, the resulting sampling errors are so large that a prohibitively high number of draws are often required. Second, iterating between approximating and optimizing the likelihood is typically extremely computationally burdensome. Third, in some latent variable models, the MLE may not exist. For example, in a time-discretization of Merton's (1976) jump-diffusion model the likelihood is unbounded for certain parameter values. Finally, the computation of the MLE standard errors, based on second order derivatives, at the optimum presents a final computational challenge. Our approach offers a simple solution for all of these issues.

To understand the approach, consider J independent copies (draws) of X, denoted $\widetilde{X}^J = (X^1, ..., X^J)$, where $X^j = (X_1^j, \ldots, X_T^j)'$. We will construct a Markov chain on the joint density $\pi_J(\theta, \widetilde{X}^J|Y)$ of the parameter and the J copies of the state variables. In contrast, standard MCMC-based Bayesian inference for latent state models defines a Markov Chain over $\pi(\theta, X)$. We will use insights from evolutionary Monte Carlo to show that increasing the dimension of the state space of the chain by a factor of J has important advantages. Namely, it helps us make draws that converge to the MLE.

The joint distribution of the parameters and augmented state matrix given the data Y is given by:

$$\pi_J(\theta, \widetilde{X}^J) \propto \prod_{j=1}^{J} p(Y|\theta, X^j)\, p(X^j|\theta), \tag{2}$$

since the J-copies are independent. The density $\pi_J(\theta, \widetilde{X}^J)$ may sometimes not integrate, even for large J. It is then useful to introduce a dominating measure $\mu(d\theta)$ with density $\mu(\theta)$ and consider the joint distribution $\pi_J^\mu(\theta, \widetilde{X}^J)$ defined by

$$\pi_J^\mu\left(\theta, \widetilde{X}^J\right) \propto \prod_{j=1}^{J} p(Y|\theta, X^j)\, p(X^j|\theta)\mu\left(\theta\right). \tag{3}$$

For example, the dominating measure could induce integrability without affecting the MLE by dampening the tails or bounding parameters away from zero. We discuss the choice of the dominating measure later. It can often be taken to be proportional to a constant.

MCMC algorithms sample from $\pi_J^\mu(\theta, \widetilde{X}^J)$ by drawing iteratively from the conditionals

$\pi_J^\mu(\theta|\widetilde{X}^J)$ and $\pi_J^\mu(\widetilde{X}^J|\theta)$. For formal results, see the discussions of the Clifford-Hammersley theorem in Robert and Casella (1999), Johannes and Polson (2004), or others. Specifically, given the g^{th} draws of $\widetilde{X}^J$ and θ, denoted respectively $\widetilde{X}^{J,(g)}$ and $\theta^{(g)}$, one draws

$$\theta^{(g+1)}|\widetilde{X}^{J,(g)} \sim \pi_J^\mu\left(\theta|\widetilde{X}^{J,(g)}\right) \tag{4}$$

$$\widetilde{X}^{J,(g+1)}|\theta^{(g+1)} \sim \pi_J^\mu\left(\widetilde{X}^J|\theta^{(g+1)}\right). \tag{5}$$

The J copies of the latent states in (5) are typically J independent draws from $\pi(X|\theta^{(g+1)})$, that is

$$\pi_J^\mu\left(\widetilde{X}^J|\theta^{(g+1)}\right) = \prod_{j=1}^{J}\pi\left(X^j|\theta^{(g+1)}\right).$$

Alternatively, the algorithm drawing J copies from $\pi(\widetilde{X}^J|\theta)$ can have a genetic or evolutionary component. That is, the Metropolis kernel updating X^j can be made to depend on $(X^1, ..., X^{j-1}, X^{j+1}, ..., X^J)$. This can improve convergence for difficult problems. This may seem counterintuitive as the state space has a high dimension. However, if the algorithm is genetic, it is harder for an element of X to get trapped in a region of the state space, as this can only happen if all J copies get stuck in that region, which is far more unlikely.

The joint distribution $\pi_J^\mu(\theta, \widetilde{X}^J)$ has the property that the marginal distribution of θ has the same form as the objective function used in simulated annealing. Hence, as J increases, it concentrates around the maximum likelihood estimate. The marginal distribution is

$$\pi_J^\mu(\theta) = \int \pi_J^\mu\left(\theta, \widetilde{X}^J\right) d\widetilde{X}^J.$$

Substituting for $\pi_J^\mu(\theta, \widetilde{X}^J)$ in (3), we obtain:

$$\pi_J^\mu(\theta) \propto \left(\prod_{j=1}^{J}\int p(Y|X^j,\theta)p(X^j|\theta)dX^j\right)\mu(\theta).$$

Now recall that $\mathcal{L}(\theta) = \int p(Y|X^j,\theta)p(X^j|\theta)dX^j$. Assume that we choose $\mu(d\theta)$ so that

$\int \mathcal{L}(\theta)^J \mu(d\theta) < \infty$, then we have that

$$\pi_J^\mu(\theta) = \frac{\mathcal{L}(\theta)^J \mu(\theta)}{\int \mathcal{L}(\theta)^J \mu(\theta) d\theta}.$$

If we re-write the density as $\pi_J^\mu(\theta) \propto \mu(\theta) \exp(J \log \mathcal{L}(\theta))$, the main insight of simulated annealing implies that as we increase J, $\pi_J^\mu(\theta)$ collapses onto the maximum of $\log \mathcal{L}(\theta)$, the finite sample MLE.

In summary the approach provides the following. First, the parameter draws $\theta^{(g)}$ converge to the finite-sample MLE denoted $\widehat{\theta}$. As T is fixed throughout, our approach inherits all the classical asymptotic properties of the MLE as T increases. Second, we show below that by appropriately scaling the parameter draws and looking at $\psi^{(g)} = \sqrt{J}(\theta^{(g)} - \widehat{\theta})$, one obtains a MCMC estimate of the *observed* Fisher's information matrix. Finally, the simulated distribution of $\psi^{(g)}$ provides us with a diagnostic on how large J must be. As soon as $\psi^{(g)}$ is approximately normal the algorithm is deemed to have converged. Quantile plots and formal tests such as Jarque-Bera, can be used to assess the convergence to normality of $\psi^{(g)}$. In many cases, due to the data augmentation, our approach will result in a fast mixing chain and a low value of J will be sufficient.

2.1 The Choice of J and $\mu(\theta)$

J and $\mu(d\theta)$ have two main effects on the joint density $\pi_J^\mu(\theta, \widetilde{X}^J)$. First, J raises the marginal likelihood to the J^{th} power. Second, $\mu(d\theta)$ can be used to ensure integrability, which can be useful in some state space models, for example, the jump model. In many cases however, we can assume that $\mu(\theta)$ is proportional to a constant.

It helps to distinguish three different cases. First, when $J = 1$ and $\mu(\theta) = p(\theta)$ where $p(\theta)$ is a subjective prior distribution, π_1^p is the posterior distribution of the states and parameters given the data. Our approach collapses to Bayesian inference of the posterior distribution. Second, when $J = 1$ and $\mu(\theta) \propto 1$, there may be a danger of non-integrability of the objective function. This is exactly the situation that may arise when using diffuse uniform priors in a

non-informative Bayesian analysis. Third, for $J > 1$, the likelihood is raised to the J^{th} power and the effect of $\mu(\theta)$ disappears (as J increases) on the range of values where the likelihood assigns positive mass. However, raising the likelihood to the power of J may or may not by itself overcome the non-integrability of the likelihood.

To illustrate the role of the dominating measure, consider two simple examples. The examples are, of course, highly stylized. Since the marginal likelihood is rarely available in closed form in latent variable models, it is difficult to find examples in that class of models.

First, consider the simplest random volatility model: $y_t = \sqrt{V_t}\varepsilon_t$, $\varepsilon_t \sim N(0,1)$, and $V_t \sim \mathcal{IG}(\alpha, \beta)$, where $\mathcal{IG}$ denotes the inverse Gamma distribution and α is known. The joint distribution of the parameters and volatilities is

$$\pi(\alpha, \beta, V) \propto \prod_{t=1}^{T} \frac{1}{V_t} e^{-\frac{y_t^2}{2V_t}} \beta^{\alpha} V_t^{\alpha+1} e^{-\frac{\beta}{2V_t}}.$$

For this model, the marginal likelihood for β is given by

$$\pi(\beta) \propto \prod_{t=1}^{T} \left(\frac{\beta}{y_t^2 + \beta} \right)^{\alpha}.$$

$\pi(\beta)$ does not integrate in the right tail for any α. Hence, raising the likelihood to a power J will not change anything. In this case, a dominating measure downweighting the right tail is required to generate a well-defined likelihood. A similar degeneracy occurs with the time-discretization of Merton's (1976) jump-diffusion model. In this model, when one of the volatility parameters is driven to zero, the likelihood function increases without bound, and thus has no maximum. In this case, $\mu(\theta)$ helps by bounding this parameter away from the origin.

In contrast, the second example is a model for which raising the likelihood to a power generates integrability. Consider a two-factor volatility model, where $y_t = v_t + \sigma\epsilon_t, v_t \sim N(0, \tau_t^2)$. The joint likelihood of the parameters is

$$\pi(\tau, \sigma) \propto \left[\prod_{t=1}^{T} \frac{1}{\sqrt{\tau_t^2 + \sigma^2}} \right] e^{-\sum_{t=1}^{T} \frac{y_t^2}{(\tau_t^2 + \sigma^2)}},$$

where $\tau = (\tau_1, \ldots, \tau_T)$. Consider the conditional density $\pi(\tau_t|\sigma, \tau_{-t})$ implied by this likelihood, where τ_{-t} refers to the τ's for all periods but t. In its right tail, for fixed σ and τ_{-t}, this density behaves like τ_t^{-1}, which is not integrable. On the other hand, $\pi_J(\tau_t|\sigma, \tau_{-t})$ behaves like τ_t^{-J} in that tail and integrates without the need for a dominating measure whenever $J > 1$.

These examples show how the dominating measure and raising the likelihood to a power can help overcome integrability problems. It is difficult to make general statements regarding these issues as integrability is model dependent and, in the presence of latent variables, one can rarely integrate the likelihood analytically.

2.2 Convergence Properties of the Algorithm

This section formally describes the convergence properties of the Markov chain as a function of G, the length of the chain, and J, the augmentation parameter.

2.2.1 Convergence in G

For a fixed J, the standard MCMC convergence implies that $\{\theta^{(g)}, \widetilde{X}^{J,(g)}\}_{g=1}^{G} \rightarrow \pi_J(\theta, \widetilde{X}^J)$ as $G \rightarrow \infty$, see Casella and Robert (2002). Hence we can choose the length G of the MCMC simulation using standard convergence diagnostics such as the information content of the draws. Johannes and Polson (2004) provide a review of practical issues in implementing MCMC algorithms.

Next, consider the convergence of the distribution of the latent state variables. Since the vectors $X^j|\theta$ are independent across j, we can first fix j and consider the convergence of the marginal distribution of X^j. As $g \rightarrow \infty$, we have that

$$p(X^j) = E_{\theta^{(g)}}\left[p(X^j|\theta^{(g)})\right] \rightarrow p(X^j|\widehat{\theta}).$$

which implies that the algorithm recovers the exact smoothing distribution of the state variables. The argument underlying this is as follows. Ergodicity implies that the average of the G draws of a function with a finite mean converges to that mean as the number of draws increases. That

is,

$$\frac{1}{G}\sum_{g=1}^{G} f\left(\theta^{(g)}, X^{J,(g)}\right) \rightarrow E\left[f\left(\theta, X^{J}\right)\right].$$

Now apply this to $f(\theta, X^J) = p(X^J|\theta)$, it follows that:

$$\frac{1}{G}\sum_{g=1}^{G} p\left(X^{j,(g)}|\theta^{(g)}\right) \rightarrow E_{\theta^{(g)}}\left[p(X^j|\theta)\right] \quad \forall j.$$

Since $\theta^{(g)} \rightarrow \widehat{\theta}$, we also have that $p\left(X^j\right) = \underset{J,g\rightarrow\infty}{lim}\, p(X^j|\theta^{(g)}) = p(X^j|\widehat{\theta})$. Hence, each of the latent variable draws comes from the smoothing distribution of X_t^j conditional on $\widehat{\theta}$.

2.2.2 Convergence in J

We now discuss the limiting behaviour of $\pi_J^{\mu}(\theta)$ as J increases, for a fixed G. The key result from simulated annealing, see for example Pincus (1968) and Robert and Casella (1999), is that for sufficiently smooth $\mu(d\theta)$ and $\mathcal{L}(\theta)$,

$$\lim_{J\rightarrow\infty} \frac{\int \theta \mathcal{L}(\theta)^J \mu(d\theta)}{\int \mathcal{L}(\theta)^J \mu(d\theta)} = \widehat{\theta},$$

where we recall that $\widehat{\theta}$ is the MLE. Simulated annealing requires that J increases asymptotically together with G. For example, Van Laarhoven and Aarts (1987) show how to choose a suitable sequence $J^{(g)}$ so that $\underset{J,g\rightarrow\infty}{lim}\, \theta^{J^{(g)}} = \widehat{\theta}$. Instead, we choose J by first proving an asymptotic normality result for $\pi_J^{\mu}(\theta)$. While this requires some suitable smoothness conditions for $\mathcal{L}(\theta)$, it will provide us with a diagnostic of whether a given J is large enough. Moreover, it will allow us to find the asymptotic variance of the MLE. We find that J as small as 10 is appropriate in our applications.

The main result given now shows formally how $\theta^{(g)}$ converges to $\widehat{\theta}$ as J increases. Define $\sigma^2(\hat{\theta}) = \ell''(\widehat{\theta})^{-1}$, the inverse of the observed information matrix.

Theorem: *Suppose that the following regularity conditions hold:*

(A1) The density $\mu(\theta)$ is continuous and positive at $\hat{\theta}$;

(A2) $\mathcal{L}(\theta)$ *is almost surely twice differentiable in some neighborhood of* $\hat{\theta}$*;*

(A3) Define the neighborhood $N_{\hat{\theta}}^{(a,b)}(J) = (\hat{\theta} + \frac{a\sigma(\hat{\theta})}{\sqrt{J}}, \hat{\theta} + \frac{b\sigma(\hat{\theta})}{\sqrt{J}})$*. Also define* $R_T(\theta) = \left(\mathcal{L}''(\hat{\theta})\right)^{-1}\left(\mathcal{L}''(\hat{\theta}) - \mathcal{L}''(\theta)\right)$*. There exists a* J *and an* ϵ_J *such that* $\epsilon_J \to 0$ *as* $J \to \infty$ *and* $\sup_{N_{\hat{\theta}}^{(a,b)}(J)} |R_T(\theta)| < \epsilon_J < 1$*. Then,*

$$\psi^{(g)} = \sqrt{J^{(g)}}(\theta^{(g)} - \hat{\theta}) \Rightarrow N(0, \sigma^2(\widehat{\theta})).$$

Hence,

$$\mathrm{Var}(\psi^{(g)}) \to \sigma^2(\widehat{\theta})$$

Proof: See the Appendix.

Assumptions (1) and (2) are clearly innocuous. $R(\theta)$ quantifies the difference in curvature of the likelihood at its maximum and at any other point θ. Assumption (3) is a regularity condition on the curvature of the likelihood stating that, as θ gets closer to $\widehat{\theta}$, the curvature of the likelihood gets closer to its value at the MLE. This results means that asymptotically in J, $\theta^{(g)}$ converges to the MLE $\widehat{\theta}$, and the variance covariance matrix of $\psi^{(g)}$ converges to the variance covariance matrix. The draws $\theta^{(g)}$ only need to be multiplied by $\sqrt{J}$ to compute a MCMC estimate of the observed variance covariance matrix of the MLE. Finally, the convergence to a normal distribution is the basis for a test of convergence based on the normality of the draws, which does not require the knowledge of $\widehat{\theta}$.

2.3 Details of the MCMC algorithm

Standard MCMC techniques can be used to simulate the joint distribution $\pi_J(\theta, \widetilde{X}^J)$. MCMC algorithms typically use the Gibbs sampler, with Metropolis steps if needed. For a Gibbs sampler, as outlined in (5) and (4), at step $g+1$, we generate independent draws of each copy $j = 1, ..., J$ of the state variable vector:

$$X^{j,(g+1)}|\theta^{(g)}, Y \sim p(X^j|\theta^{(g)}, Y) \propto p(Y|\theta^{(g)}, X^j)p(X^j|\theta^{(g)}), \tag{6}$$

and a single draw of the parameter given these J copies:

$$\theta^{(g+1)}|\widetilde{X}^{J,(g)}, Y \sim \prod_{j=1}^{J} p(Y|\theta^{(g)}, X^{j,(g+1)}) p(X^{j,(g+1)}|\theta^{(g)}) \mu(\theta^{(g)}). \tag{7}$$

For complex models, it is possible that the draws in (7) and especially (6) may not be made directly but required a Metropolis step.

Metropolis algorithms provide additional flexibility in updating the J states. For example, instead of drawing each of the X^j's independently as in (6), we could propose from a transition kernel

$$Q\left(X^{(g+1)}, X^{(g)}\right) = \prod_{j=1}^{J} Q\left(X^{(j,g+1)}, X^{(g)}\right),$$

accepting with the probability

$$\alpha\left(X^{(g)}, X^{(g+1)}\right) = \min\left[1, \frac{p\left(X^{(g+1)}|\theta, Y\right) Q\left(X^{(g+1)}, X^{(g)}\right)}{p\left(X^{(g)}|\theta, Y\right) Q\left(X^{(g)}, X^{(g+1)}\right)}\right].$$

The key here is that the Metropolis kernel can now depend on the entire history of the X's. The intuition why this may help is as follows. Consider a random walk Metropolis proposal, $X^{j,(g+1)} = X^{j,(g)} + \tau\varepsilon$. It is well known that the random walk step can wander too far and the choice of τ is problematic. Using the information in the other $J-1$ samples, we can instead propose $X^{j,(g+1)} = \frac{1}{J}\sum_{i=1}^{J} X^{i,(g)} + \tau\varepsilon$ and similarly adjust the variance of the random walk error.

We now develop MCMC algorithms to find the MLE and its standard errors for two benckmark latent state models in financial econometrics; a stochastic volatility and a multivariate jump-diffusion models, precisely showing how to construct $\pi_J^{\mu}(\theta, \widetilde{X}^J)$, and documenting convergence in J.

3 Application to the Stochastic Volatility Model

We first consider the benchmark log-stochastic volatility model where returns y_t follow a latent state model of the form:

$$y_t = \sqrt{V_t}\epsilon_t$$
$$\log(V_t) = \alpha + \delta \log(V_{t-1}) + \sigma_v v_t.$$

V_t is the unobserved volatility and y_t can be a mean-adjusted continuously-compounded return. The shocks ϵ_t and v_t are uncorrelated i.i.d. normal. Let $\theta = (\alpha, \delta, \sigma_v)$ denote the parameter governing the evolution of volatility. This model has been analyzed with a number of econometric techniques, for example Bayesian MCMC by Jacquier, Polson and Rossi (1994), Method of Moments by Melino and Turnbull (1991) and Andersen and Sorensen (1996), and simulated method of moments by Gallant et al. (1997). A more realistic extended model with leverage effect and fat tails in ϵ_t could easily be implemented but this would only complicate the exposition, see Jacquier, Polson and Rossi (2004) for a Bayesian MCMC implementation.

More related to our MLE approach here are a number of approximate methods involving simulated maximum likelihood. Danielsson (1995) proposed a simulated maximum likelihood algorithm for the basic stochastic volatility model. Durham (2002) studies the term structure of interest rates using stochastic volatility models. He proposes a simulated method of moments procedure for likelihood evaluation but notes that it can be computationally burdensome. Durham and Gallant (2004) examine a variety of numerical techniques and greatly accelerate the convergence properties. Their approach applies to nonlinear diffusion models with discretely sampled data and is based on a carefully chosen importance sampling function for likelihood evaluation. Brandt and Santa-Clara (2002) also provide a simulation likelihood approach for estimating discretely sampled diffusions.

Other methods based on Monte Carlo importance sampling techniques for full likelihood function evaluation are reviewed in Fridman and Harris (1998) and Sandmann and Koopman (1998). Brandt and Kang (2004) provide an application of this methodology to a model with

time-varying stochastic expected returns and volatilities. Durbin and Koopman (1997) develop general importance sampling methods for Non-Gaussian state space models from both a Classical and Bayesian perspective and consider an application to stochastic volatility models. Lee and Koopman (2004) describe and compare two simulated maximum likelihood estimation procedures for a basic stochastic volatility model and Liesenfeld and Richard (2003) develop an efficient importance sampling procedures for stochastic volatility models with the possibility of fat-tails.

The main difficulty encountered by all these methods is that the likelihood requires the integration of the high-dimensional vector of volatilities with a non-standard distribution. Indeed, the likelihood is the integral $\int p(Y|V,\theta)p(V|\theta)dV$, where V is the vector of volatilities. Therefore the likelihood is not known in closed form and direct computation of the MLE is impossible. Maximizing an approximate likelihood is also complicated for the same reasons. Moreover, these methods do not provide estimates of the latent volatility states, other than substituting the MLE into a Kalman filter. We now describe the implementation of our MCMC maximum likelihood approach.

3.1 Algorithm

We first derive the conditional distributions required for the algorithm. The distribution $\pi_J^{\mu}(\widetilde{V}^J,\theta)$ requires J independent copies of the volatility states. Let $V^j=(V_1^j,...,V_T^j)$ be the draw j of a $1\times T$ vector of volatilities, $\widetilde{V}_t^J=(V_t^1,...,V_t^J)'$ be a vector of J copies of V_t and $\widetilde{V}^J=\left[V^1,...,V^J\right]'$ is the $J\times T$ matrix of stacked volatilities. We need to draw from $p(\theta|\widetilde{V}^J,Y)$ and $p(V^j|\theta,Y)$ for $j=1,...,J$. For the J copies of V_t, we can write:

$$\log\left(\widetilde{V}_t^J\right)=\alpha+\delta\log\left(\widetilde{V}_{t-1}^J\right)+\sigma_v v_t^J=[\mathbf{1}^J,\log\left(\widetilde{V}_{t-1}^J\right)]\begin{pmatrix}\alpha\\ \delta\end{pmatrix}+\sigma_v v_t^J, \tag{8}$$

where $\mathbf{1}_J$ is a $J \times 1$ vector of 1's. Stacking the equations (8) over t, we obtain:

$$\begin{pmatrix} \log\left(\widetilde{V}_1^J\right) \\ \vdots \\ \log\left(\widetilde{V}_T^J\right) \end{pmatrix} = \begin{pmatrix} \mathbf{1}_J & \log\left(\widetilde{V}_0^J\right) \\ \vdots & \vdots \\ \mathbf{1}_J & \log\left(\widetilde{V}_{T-1}^J\right) \end{pmatrix} \begin{pmatrix} \alpha \\ \delta \end{pmatrix} + \sigma_v \begin{pmatrix} v_1^J \\ \vdots \\ v_T^J \end{pmatrix}.$$

This is a regression of the form $\log V = X\beta + \sigma_v v$. We can write the density $\pi_J(\theta|\widetilde{V}^J, Y)$ for this regression as

$$p\left(\theta|\widetilde{V}^J, Y\right) \propto (\sigma_v^2)^{-\frac{JT}{2}} \exp\left(\frac{-1}{2\sigma_v^2}\left[\left(\beta - \widehat{\beta}\right)' X'X\left(\beta - \widehat{\beta}\right) + S\right]\right), \tag{9}$$

where $S = \left(\log V - X\widehat{\beta}\right)'\left(\log V - X\widehat{\beta}\right)$, and $\hat{\beta} = (X'X)^{-1}X' \log V$.

The algorithm for the evaluation of the maximum likelihood by MCMC is as follows: Choose an initial parameter value $\theta^{(0)}$. Then, for $g = 1, ..., G$

1. Draw from $\pi_J(\widetilde{V}^J|\theta^{(g)}, Y)$:

   Here we draw the J copies of the volatilities from $p\left(V^j|\theta^{(g)}, Y\right)$ independently for each j, using a Metropolis Hastings algorithm as in JPR (1994). Note that these are the smoothed values of the volatilities because the conditioning is on the entire vector of observables Y.

2. Draw from $\pi_J(\theta|\widetilde{V}^{J,(g+1)}, Y)$ in (9):

   This is a simple regression with normal errors, done here in two steps, first $p(\sigma_v|\widetilde{V}^J, Y)$, then $p(\alpha, \delta|\sigma_v, \widetilde{V}^J, Y)$. Note that Y is redundant given the knowledge of V. Namely, we draw $\sigma_v^{(g+1)}$ from

   $$p(\sigma_v|\alpha^{(g+1)}, \delta^{(g+1)}, \widetilde{V}^{J,(g+1)}) \sim IG\left(J, S^{(g+1)}\right).$$

   And we draw $\alpha^{(g+1)}, \delta^{(g+1)}$ from

   $$p(\alpha, \delta|\sigma_v, \widetilde{V}^J) \sim N\left(\widehat{\beta}^{(g+1)}, \left(\sigma_v^2\right)^{(g)}\left[X^{(g+1)\prime}X^{(g+1)}\right]^{-1}\right),$$

   where $\widehat{\beta}^{(g)}$ is the OLS estimate $\left(X^{(g+1)\prime}X^{(g+1)}\right)^{-1} X^{(g+1)\prime} \log V^{(g+1)}$.

One noticeable feature of the algorithm is that we can choose $\mu(\theta) = 1$ for $J \geq 2$. This is because, even for diffuse priors, the distribution for σ_v, $p(\sigma_v|\alpha, \delta, \widetilde{V}^J)$, is proper when $J \geq 2$. This contrasts with the Bayesian MCMC analysis of this problem, which requires a proper prior for σ_v, see for example Jacquier, Polson and Rossi (1994, 2004).

3.2 Performance

We demonstrate the behavior of the algorithm for the basic stochastic volatility model with parameters $\alpha = -0.363$, $\delta = 0.95$, $\sigma_v = 0.26$. These parameters are consistent with empirical estimates for daily equity return series and often used in simulation studies. We simulate one series of $T = 1000$ observations. We then run the MCMC algorithm for $G = 25000$ draws, for four different values of $J = 1, 2, 10$ and 20. For $J = 1$, the algorithm is essentially identical to that in JPR (1994). As a dominating measure is needed for $J = 1$, we use the same prior, very diffuse but proper, as JPR (1994). In this case, the algorithm converges to draws of the posterior distribution of the parameters and volatilities. As J becomes large, the algorithm produces a sequence of draws of the parameters, which average converges to the MLE.

The CPU time required to run the algorithm is comparable to a standard Bayesian MCMC algorithm, as the CPU time is linear in J and in G. A run with $J = 20$ and $G = 25000$ required about 20 minutes on a SUN workstation. However these $J = 20$ times $G = 25000$ draws are far more informative than a CPU-time equivalent run of, say, $J = 1$ times $G = 500000$ draws. This is because as J increases, the variance of the sequence of draws decreases.

The theoretical convergence results in the previous sections indicate that, as J increases, the draws will converge to the MLE. In practice, these results would not be very useful if an inordinately high value of J was required for the algorithm to approach the MLE. We now empirically show that this is not the case in our examples. Namely, the algorithm is quite effective for even moderate values of J. To study this we look at the distribution of the scaled draws of $\psi^{(g)}$. Recall that looking at whether $\theta^{(g)}$ is close to or far from the true θ is not useful in itself since $\theta^{(g)}$ for large g is only an estimate of $\widehat{\theta}$ which we know converges to the true parameter only as the sample size T gets large.

Figure 1 shows the draws of δ for the four runs of the algorithm with $J = 1, 2, 10$ and 20. Each draw of δ is conditional on a vector of volatilities of length TJ. The plots confirm that moderate increases in J quickly reduce the variance of the draws. Figure 2 shows a similar result for σ_v. One may worry that a drastic reduction in the variability of the draw might hamper the ability of the algorithm to dissipate initial conditions. We see that this is not the case. Even for J as large as 20, the algorithm dissipates initial conditions very quickly, moving promptly to the MLE. Now note the horizontal lines showing the true parameter value and the average of the last 24000 draws. The estimate of the MLE for σ_v, $J = 20$, is very close to the true value of 0.26, that for δ is around 0.93. These results, obtained for this one sample do not constitute a sampling experiment for the MLE. One expects the MLE itself to converge to the true value only for very large sample sizes T, as the finite sample MLE is in general biased in these types of models.

Figure 3 shows the rate at which the draws converge to normality in distribution. The left and right plots in Figure 3 show the normal probability plots for δ and σ_v, with $J = 1, 2$, and 10. Recall that for $J = 1$, the algorithm produces the Bayesian posterior. Panels (a) and (d), $J = 1$, show strong skewness and kurtosis, reflecting the well known non-normality of the posterior distribution for this model, see for example JPR (1994). Then the convergence to normality is all the more remarkable as J increases to 2, in panels (b) and (e), and then 10, in panels (c) and (f). With as few as $J = 10$ copies of the states, the algorithm produces samples of draws very close to normality. This is consistent with a very rapid convergence to the MLE as J increases. Note again that this visual diagnostic can be supplemented by formal tests of normality of the draws, such as Jarque-Bera or others.

While Figures 1 to 3 confirm that the algorithm effectively estimates the MLE for even small values of J, we now turn to the effect of J on the smoothed volatilities. Again, for $J = 1$, the algorithm produces the Bayes estimator of the volatility. For example, averaging the draws yields a Monte Carlo estimate of the posterior mean. Consider a specific draw, for example the last one $G = 25000$. For this G, consider computing the average of the J draws. Figure 4 plots the time series of these averages of the V_t's. Panel (a), the Bayes case of $J = 1$, shows the large noise inherent in one draw, obscuring any time series pattern in the true smoothing

distribution of the V_t's. As the averaging is done over an increasing J, the noise decreases and a time series pattern of the volatilities emerges. Figure 4 shows that this happens immediately for relatively small values of J. Namely, panels (c) and (d) - averages over $J = 10$ and 20, present remarkably similar time series although coming from two unrelated runs. This increase in precision for the state variables is at the source of a commensurate increase in precision in the draws of the parameters α, δ, σ_v, as each draw uses the information over the J copies.

The algorithm also produces smoothed estimates of the volatilities V_t, by averaging the JG draws of V_t. Figure 5 shows that these smoothed estimates of volatility are nearly identical for all values of J. Panel (a) follows from an averaging over $G = 25000$ draws while panel (c) is over $GJ = 250000$ draws. Their estimates are identical because the precision in the averaging in the Bayesian case is high enough to make any further increase in precision - via a larger J, insignificant. Effectively, the small changes in the parameter estimates result in even smaller changes in the volatility estimates. This is confirmed in Figure 6, which plots the estimated versus the true volatilities. Panel (a) and (b) represent $J = 1$ and 20. They are very similar with a cross-correlation of 0.74. So, our algorithm preserves the efficient smoother originally produced by the Bayesian MCMC algorithm. This is in sharp contrast with an approximate smoother which would for example substitute the MLE of the parameters into a Kalman filter.

4 Application to Merton's Jump-Diffusion Model

A multivariate version of Merton's jump-diffusion model specifies that a vector of asset prices, S_t, solves the stochastic differential equation:

$$dS_t = \mu S_t dt + \sigma S_t dW_t + d\left(\sum_{j=1}^{N_t} S_{\tau_{j-}}(e^{Z_j} - 1)\right),$$

where $\sigma\sigma' = \Sigma \in R^K \times R^K$ is the diffusion matrix, N_t is a Poisson process with constant intensity λ and the jump sizes, $Z_j \in R^K$ $Z_j \sim N(\mu_z, \Sigma_z)$. Solving this stochastic differential

equation, continuously compounded equity returns (Y_t) over a daily interval are

$$Y_{t+1} = \mu + \sigma\epsilon_{t+1} + \sum_{j=N_t+1}^{N_{t+1}} Z_j$$

where, $\epsilon_{t+1} = W_{t+1} - W_t$ and the drift vector is redefined to account for the variance correction.

Following Merton (1976), the univariate version of this model is commonly used for option pricing. The multivariate version maybe even more useful. For a vector of risky assets, the multivariate jump model generates fat tails and allows for a different correlation structure between 'normal' movements (σ) and large market movements (Σ_z). For example, this allows for large returns to be more highly correlated than small returns. Duffie and Pan (2001) provide closed form approximations for the value-at-risk (VAR) of a portfolio of the underlying or options on the underlying, but they do not estimate the model.

Likelihood based estimation of the model is very difficult for two reasons. First, random mixtures of normal have well-known degeneracies in one dimension. These degeneracies are likely much worse in higher dimensions. Second, even for moderate K, there are a large number of parameters and gradient based optimization of a complicated likelihood surface is rarely attempted. Our simulation based optimization is immune to these difficulties. We now describe the algorithm and provide evidence on the performance of the algorithm.

4.1 Algorithm

The time-discretization of the model which we consider implies that at most a single jump can occur over each time interval:

$$Y_{t+1} = \mu + \sigma\epsilon_{t+1} + I_{t+1}Z_{t+1}, \tag{10}$$

where ϵ_t is a unit normal shock, I_t is 1 if there is a jump and 0 otherwise, $P[I_t = 1] = \lambda \in (0, 1)$ and the jumps retain their structure. Johannes, Kumar and Polson (1999) document that, in the univariate case, the effect of time-discretization in the Poisson arrivals is minimal, as jumps

are rare events.

The parameters and state variable vector are given by $\theta = (\mu, \Sigma, \lambda, \mu_Z, \Sigma_Z)$ and $X = \{I_t, Z_t\}_{t=1}^T$. Our MCMC algorithm samples from $p(\theta, X|Y) = p(\theta, I, Z|Y)$, where I and Z are vectors containing the time series of jump times and sizes.

Our MCMC algorithm draws θ, Z and I sequentially. Each posterior conditional is a standard distribution that can easily be sampled from. Thus the algorithm is a pure Gibbs sampler. This is because the augmented likelihood function is

$$p\left(Y|\theta, I, Z\right) = \prod_{t=1}^{T} p\left(Y_t|\theta, I_t, Z_t\right),$$

where $p\left(Y_t|\theta, I_t, Z_t\right) \sim N\left(\mu + Z_t I_t, \Sigma\right)$, i.e., it is conditionally Gaussian. On the other hand, the marginal likelihood, $p\left(Y|\theta\right)$, is difficult to deal with because it is a mixture of multivariate normal distributions. In the univariate case, this observed likelihood has degeneracies, it is infinite for certain parameter values, and well-known multi-modalities. Multivariate mixtures are even more complicated and direct maximum likelihood is rarely attempted.

We assume standard Normal-Inverse Wishart conjugate prior distributions for the parameters μ, Σ, and μ_Z, Σ_Z. Namely, $\mu \sim \mathcal{N}\left(a, A\right)$, $\Sigma \sim \mathcal{W}^{-1}\left(b, B\right)$, $\mu_Z \sim \mathcal{N}\left(c, C\right)$, and $\Sigma_Z \sim \mathcal{W}^{-1}\left(d, D\right)$. For λ, we use a conjugate beta distribution, $\lambda \sim \mathcal{B}\left(e, E\right)$. The MCMC algorithm iteratively draws the parameters and the state variables:

$$\begin{aligned}
\text{Diffusive Parameters} &: p\left(\mu|\Sigma, I, Z, Y\right) \propto N\left(a^*, A^*\right) \\
&: p\left(\Sigma|\mu, I, Z, Y\right) \propto \mathcal{W}^{-1}\left(b^*, B^*\right) \\
\text{Jump Size Parameters} &: p\left(\mu_Z|\Sigma_Z, I, Z\right) \propto N\left(c^*, C^*\right) \\
&: p\left(\Sigma_Z|\mu_Z, I, Z\right) \propto \mathcal{W}^{-1}\left(d^*, D^*\right) \\
\text{Jump Time Parameters} &: p\left(\lambda|I\right) \propto \mathcal{B}\left(e^*, E^*\right) \\
\text{Jump Sizes} &: p\left(Z_t|\theta, I_t, Y_t\right) \propto N\left(m_t^*, V_t^*\right) \\
\text{Jump Times} &: p\left(I_t|\theta, Z_t, Y_t\right) \propto \mathit{Binomial}\left(\lambda_t^*\right)
\end{aligned}$$

The MCMC algorithm samples from $p\left(\theta, \widetilde{I}, \widetilde{Z}|Y\right)$ by the iteratively drawing

$$\begin{aligned}
\theta^{(g+1)} &\sim p\left(\theta|\widetilde{I}^{(g)}, \widetilde{Z}^{(g)}, Y\right) \\
\widetilde{I}^{(g+1)} &\sim p\left(\widetilde{I}|\theta^{(g+1)}, \widetilde{Z}^{(g)}, Y\right) \\
\widetilde{Z}^{(g+1)} &\sim p\left(\widetilde{Z}|\theta^{(g+1)}, \widetilde{I}^{(g+1)}, Y\right)
\end{aligned}$$

where we note that the last two draws are just J draws from the same distribution.

4.2 Performance

We analyze a three-dimensional version of Merton's model. We simulate a vector of 1000 returns using the following parameter values, scaled to daily units: $\mu = (0.2, 0.15, 0.1)$, $\mu_z = (-3, -3.5, -4)$, $\lambda = 0.10$, $\sigma_z^{11} = 3$, $\sigma_z^{22} = 4$, $\sigma_z^{33} = 5$, $\sigma^{11} = 1.5$, $\sigma^{22} = 1$, $\sigma^{33} = 0.5$, the off diagonal elements of the diffusive and jump-covariance matrix are such that the diffusive or jump correlation between any two assets is fifty percent. These parameters are typical of those that would be found in an analysis of large, volatile equity indices in the United States. The method can generate a large number of draws in little CPU time. On a Pentium 4, 3.00Ghz, 5000 draws (G) for this sample size and $J = 1$ take about 30 seconds. Recall that the CPU time is proportional to J as well as G and T.

We report results of sampling experiments similar to those in the previous subsection. Figures 7 to 10 display a summary of the MCMC output for $G = 5000$, and $J = 1$, 2, 10 and 20. The results are largely consistent with those seen for the SV model. For example, consider Figure 7, which plots the draws of the parameter λ. As J increases, the variability of the draws reduces drastically, collapsing on the true value of 0.10. As a comparison, the volatility of the draws for λ decreases from 0.00705 to 0.0016, when J increases from 1 to 20, a reduction by a factor of 4.4. This is right in line with the implications of our central limit theorem as $\sqrt{20} = 4.47$. Figure 8 provides normality plots of the draws for $J = 1$, 2, 10 and 20. As J increases, the draws converge very fast to their limiting standard normal distribution.

Figure 9 and 10 provide the trace plots for two other parameters of interest, the jump

mean μ_z^1 and the jump intensity σ_z^1. Again the plots collapse as expected, albeit with slight biases. For example, for μ_z^1 the average of the draws is roughly -2.7, for a true value of -3. Similarly, for σ_z^1, the mean of the draws for $J = 20$ is 3.15, a bit above the true value of 3. These estimates appear slightly biased. Again, there is no reason to believe that either the Bayes ($J = 1$) or the maximum likelihood estimator are unbiased in finite samples. Both estimators are only asymptotically (in T) unbiased.

5 Conclusion

In this paper, we develop MCMC algorithms for computing finite sample MLE's and their standard errors in latent state models. Computing the MLE requires a joint integration of the state variables and optimization over the parameters. Our MCMC algorithms simultaneously perform this integration and optimization without the need to resort to gradient method.

Our approach makes use of data augmentation and evolutionary MCMC. MCMC methods allow us to simulate from the high-dimensional joint distribution $\pi_J^\mu(\theta, \widetilde{X}^J)$ that arises for the parameters and J copies of the latent state variables. We show how to avoid singularities in the marginal likelihood by the choice of a suitable dominating measure for this joint density. Our asymptotic is in J, and we provide diagnostics to determine whether J is high enough. We estimate a stochastic volatility and a multivariate jump diffusion model, two complex and very different models. Our implementation shows that convergence occurs quickly for low values of J for both models.

Acknowledgements:
The paper has benefitted from comments from Ron Gallant, Lars Hansen, Eric Renault, the seminar participants at the 2003 Montreal Econometrics conference and Columbia University, and two anonymous referees. Jacquier acknowledges support from the Montreal Mathematical Finance Institute.

References

Andersen, T., and B. Sorensen, 1996, GMM Estimation of a Stochastic Volatility Model: A Monte Carlo Study, Journal of Business and Economic Statistics 14, 328-352.

Besag, J., 1974, Spatial Interaction and the statistical analysis of lattice systems (with discussion), Journal of the Royal Statistical Society, Series B 36, 192-326.

Boivin, J, and Giannoni, M.P., 2005. DSGE Models in a Data-Rich Enviroment. Working paper, Columbia university.

Brandt, M. and Kang, 2004, On the Relationship betwween the conditional mean and volatility of stock returns: a latent VAR approach. Jounal of Financial Economics, 72, 217-257.

Carlin, B.P. and N.G. Polson, 1991, Inference for Nonconjugate bayesian Models using the Gibbs sampler. Canadian Journal of Statistics 19, 399-405.

Chernozhukov, V. and H. Hong, 2003, Likelihood inference for some nonregular econometric models, Journal of Econometrics 115, 293-346.

Dawid, A. P., 1970. On the limiting normality of posterior distributions. Proceedings of the Cambridge Philosophical Society 67, 625-633.

Dempster, A.P., N.M. Laird and D. B. Rubin, 1977, Maximum likelihood from incomplete data via the EM algorithm, Journal of the Royal Statistical Society Series B, 339, 1-38.

Doucet, A., Godsill, S and C. Robert, 2002, Marginal maximum posterior estimation using Markov Chain Monte Carlo, Statistics and Computing 12, 77-84.

Duffie, D., and J. Pan, 2001, Overview of Value at Risk, in Options Markets, edited by G. Constantinides and A. G. Malliaris, London: Edward Elgar.

Durbin, J., and Koopman, S.J., 1997, Monte Carlo maximum likelihood estimation for non-Gaussian state space models, Biometrika 80, 669-684.

Durham, G. 2002, Likelihood-based specification analysis of continuous-time models of the short-term interest rate. working paper U. of Iowa Economics department.

Durham, G. B., and A.R. Gallant, 2004, Numerical techniques for simulated maximum likelihood estimation of stochastic differential equations, Journal of Business and Statistics.

Erkanli A., 1994, Laplace Approximations for Posterior Expectations When the Mode Occurs

on the Boundary of the Parameter Space, Journal of the American Statistical Association 89, 250-258.

Fridman, M., and Harris, L., 1998, A maximum likelihood approach for non-gaussian stochastic volatility models, Journal of Business and Economic Statistics, 16, 3, 284-291.

Gallant, A. Ronald, David A. Hsieh and George Tauchen, 1997, Estimation of Stochastic Volatility Models with Diagnostics, Journal of Econometrics, 81(1), 159-192.

Geyer, C.J., 1994, Markov Chain Monte Carlo Maximum Likelihood. In: Keramidas E.M.(Ed.), Journal of the Royal Statistical Society B.

Heyde, C., and I.M. Johnstone, 1979, On asymptotic posterior normality for stochastic processes, Journal of the Royal Statistical Society Series B, 41, 184-189.

Jacquier, E., Johannes, M., and N.G. Polson, 2005, MCMC Methods for Expected Utility Problems, working paper Columbia University and HEC Montreal.

Jacquier, E., Polson, N.G., and P. Rossi, 1994, Bayesian Analysis of Stochastic Volatility Models (with discussion). Journal of Business and Economic Statistics 12, 4, 371-417.

Jacquier, E., Polson, N.G., and P. Rossi, 2004, Bayesian Analysis of Stochastic Volatility Models with Fat Tails and Leverage Effects, Journal of Econometrics 122.

Johannes, M., Kumar, A., and N.G. Polson, 1999, State-Dependent Jump Models: How do US Indicies Jump? working paper, columbia university.

Johannes, M and N.G. Polson, 2004, MCMC methods for Financial Econometrics. Handbook of Financial Econometrics, Y. Ait-Sahalia and L. Hansen eds.

Johannes, M and N.G. Polson, 2003, Discussion of Pastorello S., Patilea V., and E. Renault, Iterative and Recursive Estimation in Structural Non-Adaptive Models," Journal of Business and Economic Statistics 21, No 4, 449-509.

Kiefer, N.M., 1978, Discrete parameter variation: efficient estimation of a switching regression mode, Econometrica, 46:427–434, 1978.

Kirkpatrick, S., 1984, Optimization by simulated annealing: quantitative studies, Journal of Statistical Physics 34, 975-986.

Kirkpatrick, S., C.D. Gelatt and M.P. Vecchi, 1983, Optimization by simulated annealing, Science 220, 671-680.

Lee, K.M., and S.J. Koopman, 2004, Estimating stochastic volatility Models: a comparison of two importance samplers, studies in nonlinear dymamics & Econometrics, edited by Dagum and Proietti, Vol 8, issue 2.

Liang, F., and W.H. Wong, 2001, Real Parameters Evolutionary Monte Carlo with Applications to Bayesian Mixture Models, Journal of the American Statistical Association 96, 653-666.

Liesenfeld, R., and Richard, J.F., 2003, Univariate and multivariate stochastic volatility models: estimation and diagnostics, Journal of Empirical Finance 10, 505-531

Merton, R. 1976, Option pricing when the underlying stock returns are discontinuous. Journal of Financial Economics, 3, 125-144.

Mueller, P., 2000, Simulation based optimal design, Bayesian Statistics 6, Bernardo, et al. eds., (Oxford).

Pastorello S., Patilea V., and E. Renault, 2003, Iterative and Recursive Estimation in Structural Non-Adaptive Models, Journal of Business and Economic Statistics 21, No 4, 449-509.

Pincus, M., 1968, A Closed Form Solution of Certain Programming Problems, Operation Research 18, 1225-1228.

Polson, N., 1996, Convergence of MCMC algorithms, Bayesian Statistics 5, Bernardo, et al. eds., Oxford, 297-321.

Robert, C. and G. Casella, 1999, Monte Carlo Statistical Methods, Springer-Verlag, New York.

Sandmann, G., and S.J. Koopman, 1998, Estimation of stochastic volatility via Monte Carlo maximum likelihood, Journal of Econometrics, 87, 271-301.

Schervish M.J., 1995, Theory of Statistics, Springer Verlag, New York.

Taylor, S., 1986, Modeling Financial Time Series, Wiley, New York.

Van Laarhoven, P.J., and E.H.L Aarts, 1987, Simulated Annealing: Theory and Applications, CWI Tract 51, Reidel, Amsterdam.

Appendix: Convergence in J

Consider the marginal distribution $\pi_J^{\mu}(\theta) = \mathcal{L}(\theta)^J \mu(\theta)/m_J$, where $m_J = \int \mathcal{L}(\theta)^J \mu(d\theta)$. We study the asymptotics , in J, of $\psi^{(g)} = \sqrt{J}(\theta^{(g)} - \hat{\theta})$, where $\theta^{(g)}$ are the MCMC draws from the algorithm and $\hat{\theta}$ is the MLE for the sample considered.

We first write the target marginal density as a function of $\ell(\theta)$, the logarithm of the likelihood

$$\pi_J^{\mu}(\theta) = \frac{e^{J\ell(\theta)}\mu(\theta)}{m_J}$$

Now write

$$\pi_J^{\mu}(\theta) = \pi_J^{\mu}(\hat{\theta}) e^{J(\ell(\theta) - \ell(\hat{\theta}))}.$$

By Taylor's theorem we can write $\ell(\theta) = \ell(\hat{\theta}) + \frac{1}{2}(\theta - \hat{\theta})^2 \ell''(\theta^\star)$, where $\theta^\star = \theta + \gamma(\hat{\theta} - \theta)$ with $0 < \gamma < 1$. Denote $\sigma^2(\hat{\theta})$ the inverse of $\mathcal{L}''(\hat{\theta})$, and let $R(\theta) = \sigma^2(\hat{\theta})(\ell''(\theta) - \ell''(\hat{\theta}))$. Note that R is a measure of distance of θ to $\hat{\theta}$. Combining the above two equations, and substituting for $R(\theta)$, we can write

$$\pi_J^{\mu}(\theta) = \pi_J^{\mu}\left(\hat{\theta}\right) e^{-\frac{J}{2\sigma^2}(\theta - \hat{\theta})^2 (1 - R(\theta))}$$

Recall that ψ_J is $\sqrt{J}\sigma^{-1}(\theta - \hat{\theta})$. We need to show that for any $a < b$ we have that

$$\lim_{J \to \infty} P(a < \psi_J < b) = \Phi(b) - \Phi(a)$$

where $\Phi(\cdot)$ is the standard normal cumulative distribution function. Now

$$P(a < \psi_J < b) = P\left(\hat{\theta} + \frac{a\sigma}{\sqrt{J}} < \theta < \hat{\theta} + \frac{b\sigma}{\sqrt{J}}\right) = \int_{N_{\hat{\theta}}^{(a,b)}(J)} \frac{e^{J\ell(\theta)}}{m_J} \mu(d\theta).$$

Now, for any $\epsilon > 0$, by continuity of $\mu(\theta)$ at $\hat{\theta}$, assumption (A1), we can find J so that

$$(1 - \epsilon) < \inf_{N_{\hat{\theta}}^{(a,b)}(J)} \frac{\mu(\theta)}{\mu(\hat{\theta})} < \sup_{N_{\hat{\theta}}^{(a,b)}(J)} \frac{\mu(\theta)}{\mu(\hat{\theta})} < 1 + \epsilon.$$

Hence we need only consider

$$I = \int_{N_{\hat{\theta}}^{(a,b)}(J)} \frac{e^{J\ell(\theta)}}{m_J} d\mu(\theta) = \frac{\pi_J\left(\hat{\theta}\right)}{m_J} \int_{N_{\hat{\theta}}^{(a,b)}(J)} e^{-\frac{J}{2\sigma^2}(\theta-\hat{\theta})^2(1-R(\theta))} d\mu(\theta)$$

By assumption (A3) we can also find J such that there exists an $0 < \varepsilon_J < 1$ where $\varepsilon_J \to 0$ as $J \to \infty$ and

$$\sup_{N_{\hat{\theta}}^{(a,b)}(J)} |R(\theta)| < \varepsilon_J < 1.$$

Therefore

$$\int_{N_{\hat{\theta}}^{(a,b)}(J)} e^{-\frac{J}{2\sigma^2}(\theta-\hat{\theta})^2(1-\varepsilon_J)} d\theta < I < \int_{N_{\hat{\theta}}^{(a,b)}(J)} e^{-\frac{J}{2\sigma^2}(\theta-\hat{\theta})^2(1+\varepsilon_J)} d\theta.$$

Now as $N_{\hat{\theta}}^{(a,b)}(J) = \left(\hat{\theta} + \frac{a\sigma}{\sqrt{J}}, \hat{\theta} + \frac{b\sigma}{\sqrt{J}}\right)$ we have

$$\int_{N_{\hat{\theta}}^{(a,b)}(J)} e^{-\frac{J}{2\sigma^2}(\theta-\hat{\theta})^2(1+\varepsilon_J)} d\theta = \sqrt{2\pi}\sigma(1+\varepsilon_J)^{-\frac{1}{2}} \left(\Phi\left(\sqrt{J}\sigma^{-1}(\frac{b\sigma}{\sqrt{J}})\right) - \Phi\left(\sqrt{J}\sigma^{-1}(\frac{a\sigma}{\sqrt{J}})\right)\right)$$

Taking the limit as $J \to \infty$ and noting that $\epsilon_J \to 0$ we have that $P(a < \psi_J < b) \to \Phi(b) - \Phi(a)$ as required.

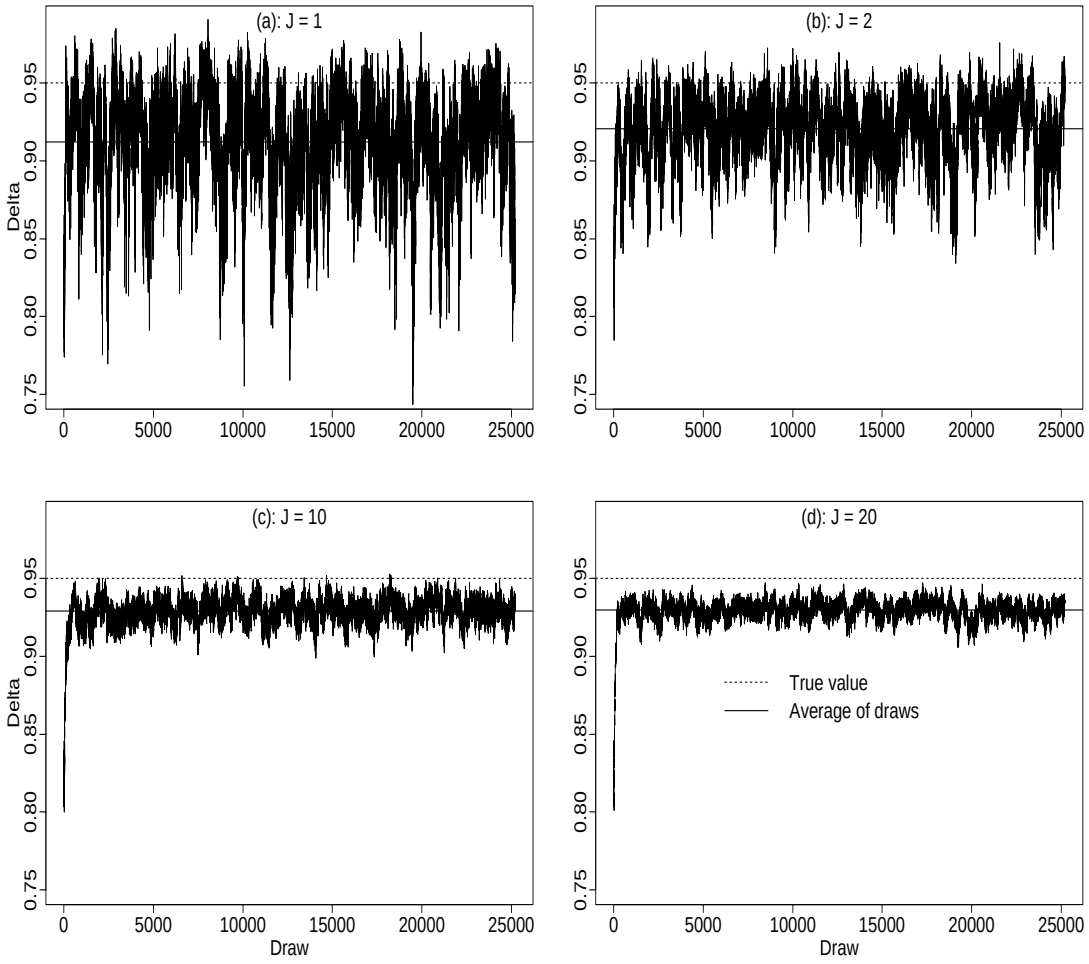

Figure 1: Draws of δ for the SV model, $\delta = 0.95, T = 1000, G = 25000$

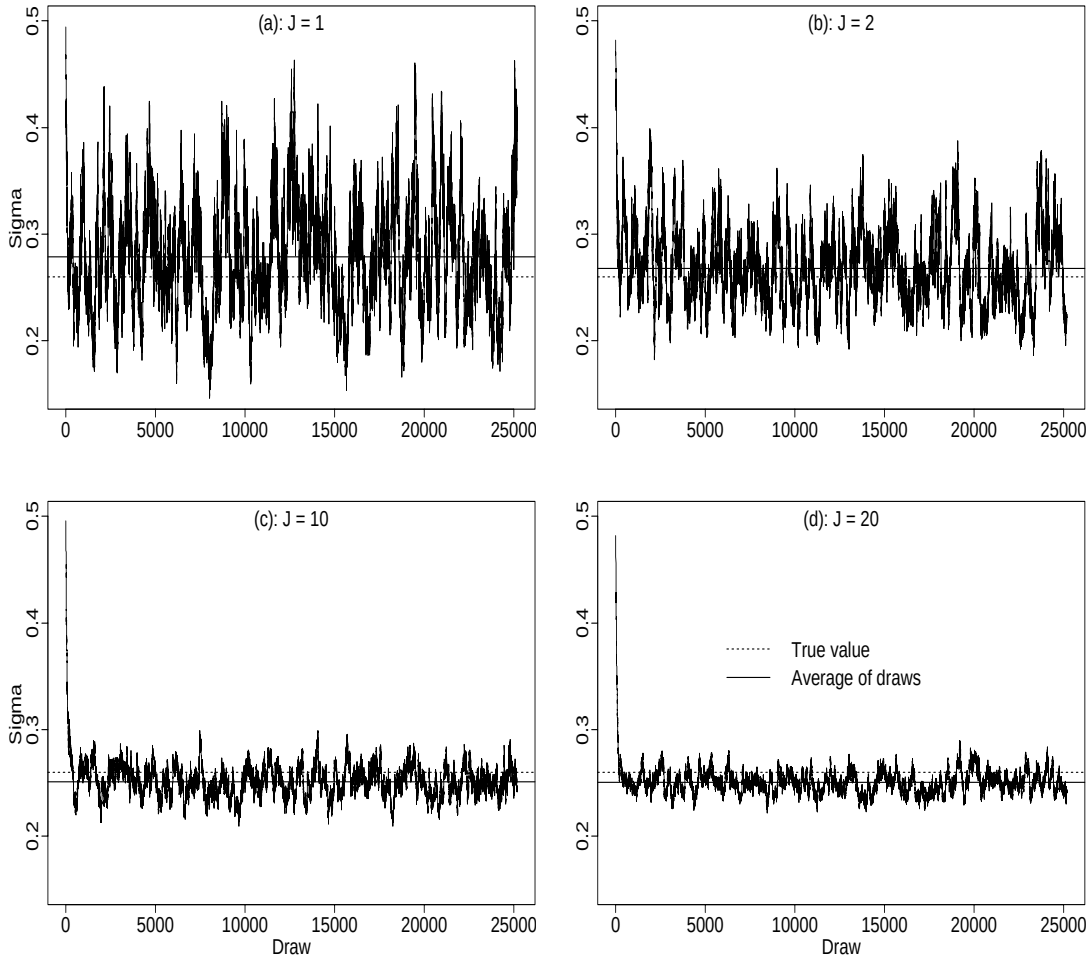

Figure 2: Draws of σ_v for the SV model, $\sigma_v = 0.26, T = 1000, G = 25000$

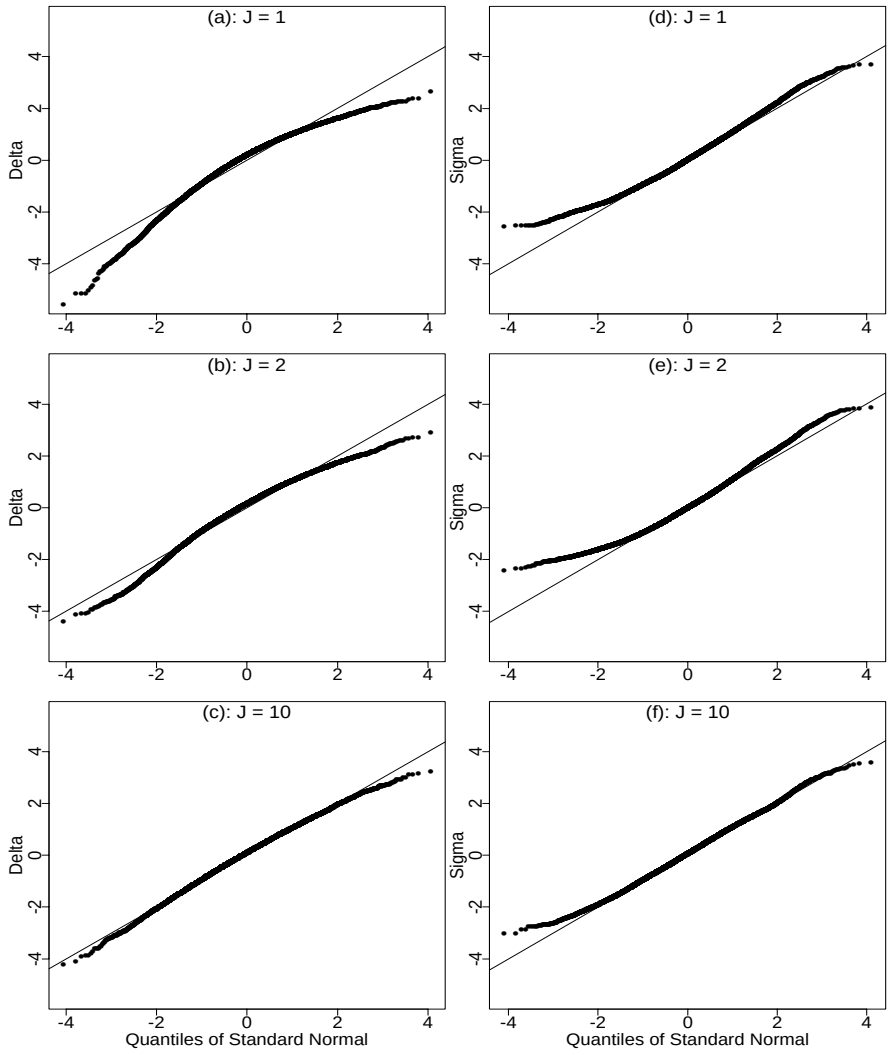

Figure 3: Normality of draws of δ and σ_v, SV model, $T = 1000, G = 25000$

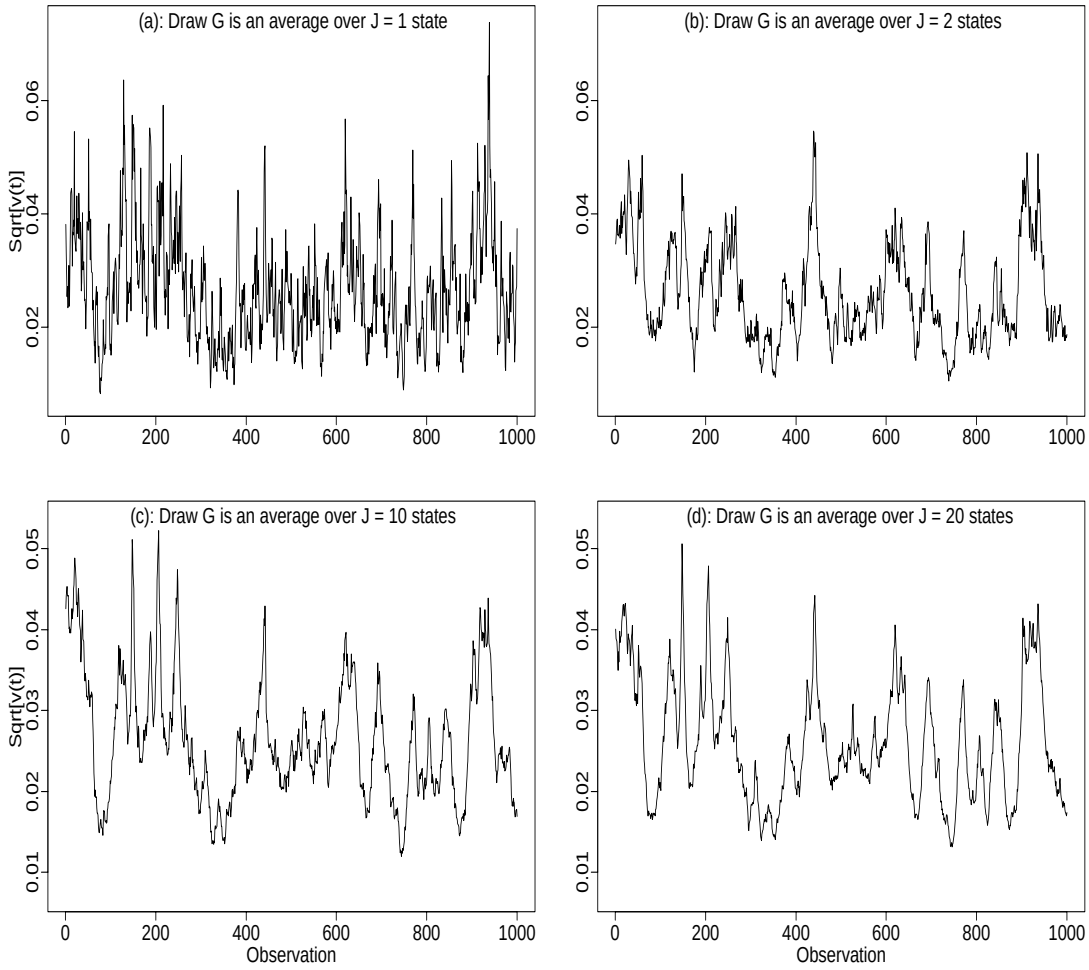

Figure 4: Last draw (G =25000) of $\sqrt{V_t}$, SV model, $\delta = 0.95, T = 1000$

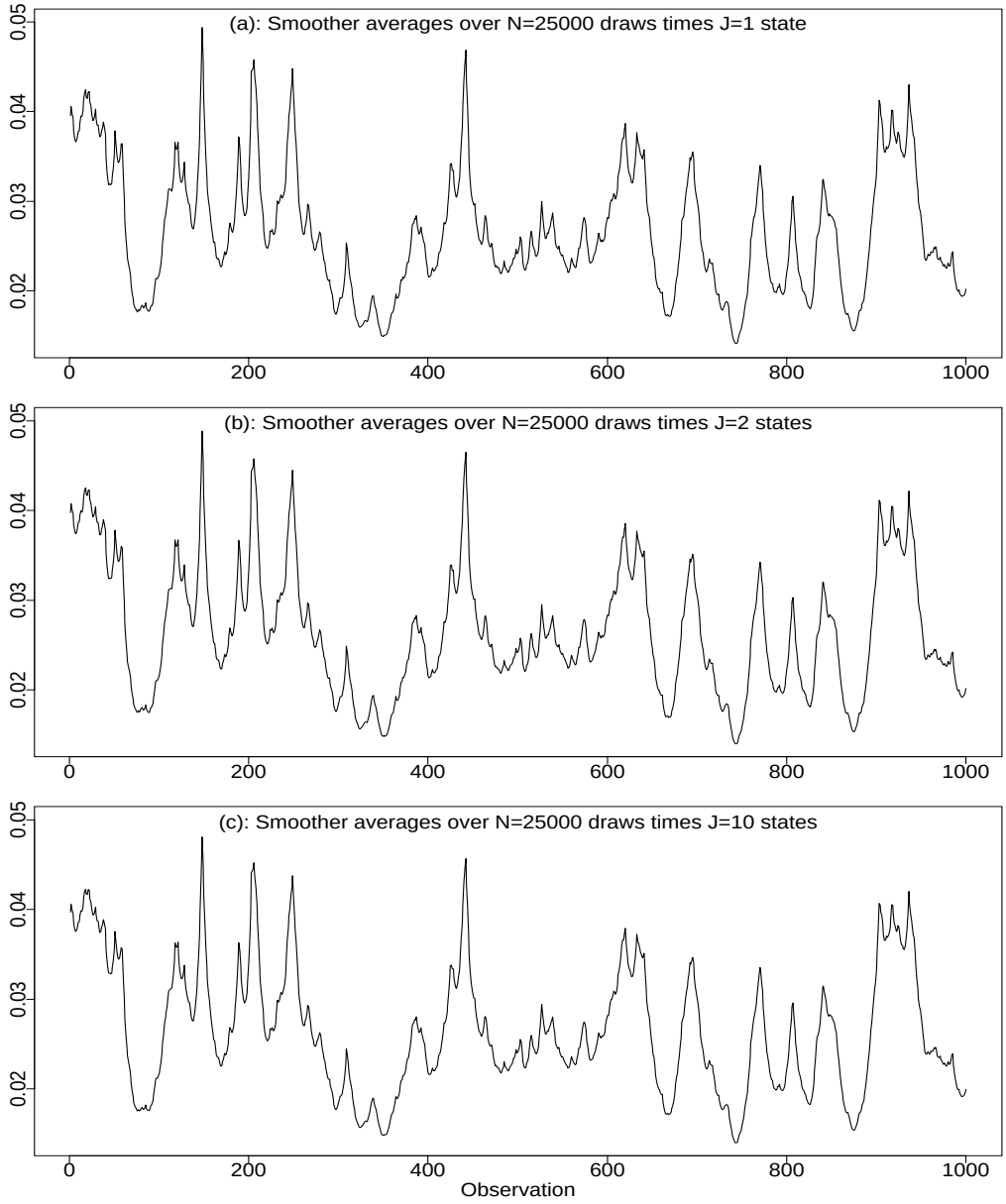

Figure 5: Smoothed estimates of $\sqrt{V_t}$, SV model, $\delta = 0.95, T = 1000, G = 25000$

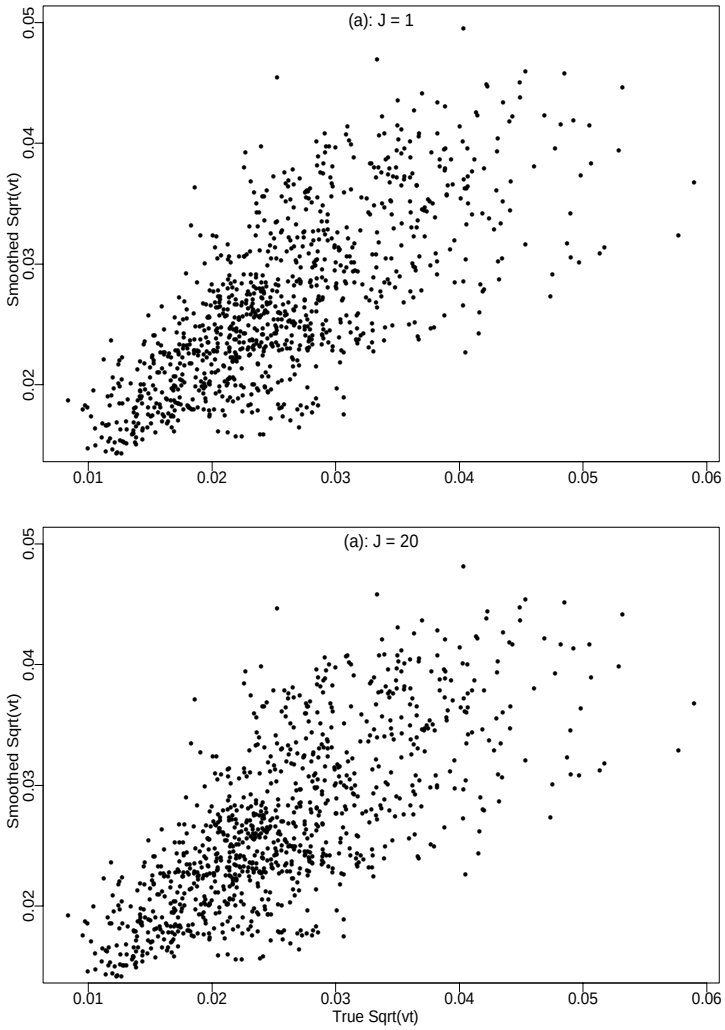

Figure 6: Smoothed estimates versus true $\sqrt{V_t}$, SV model $\delta = 0.95, T = 1000, G = 25000$

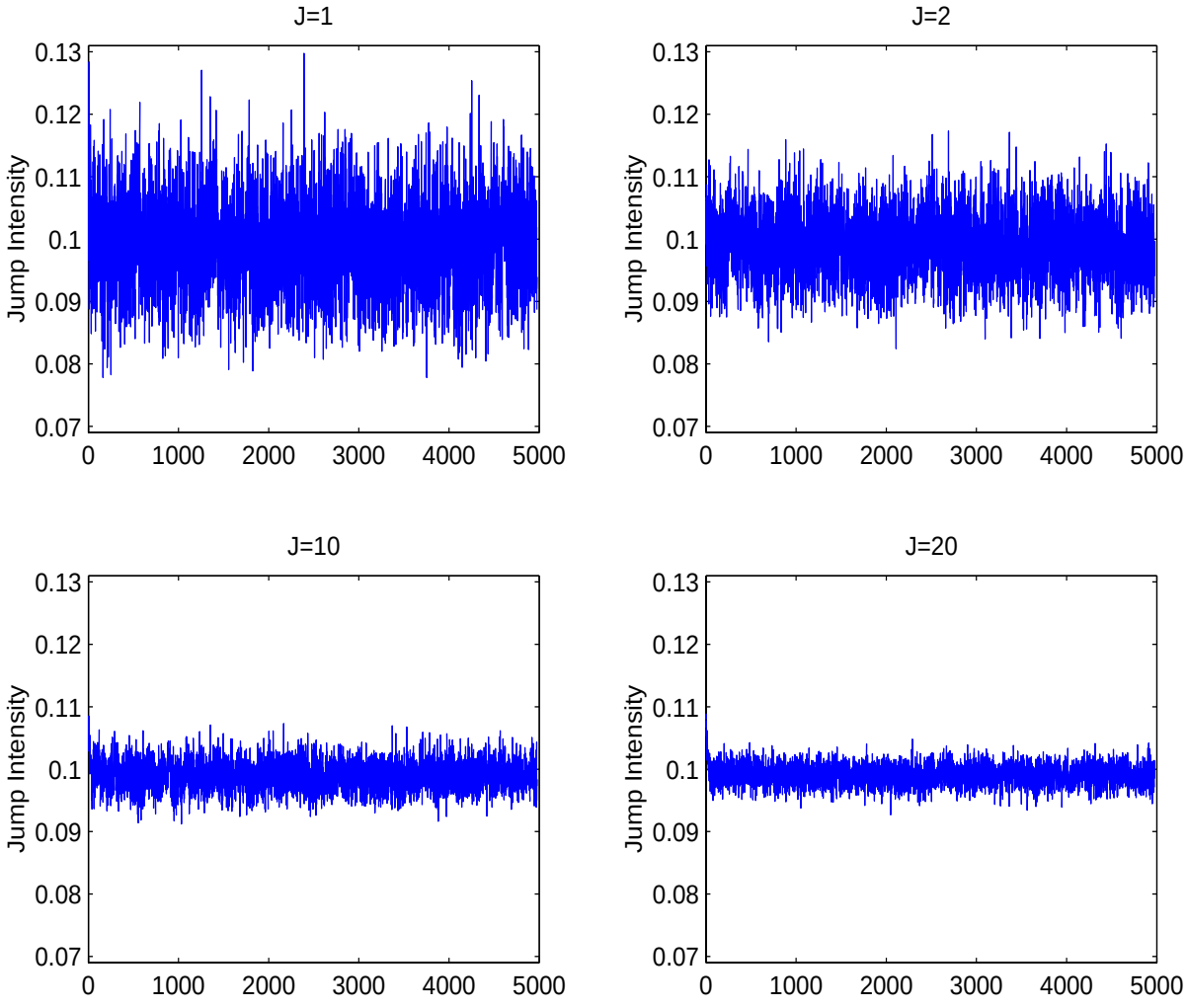

Figure 7: MCMC draws for the jump intensity, λ, three-dimensional Merton's model, $G = 5000$, $J = 1, 2, 10, 20$. The true λ is 0.10.

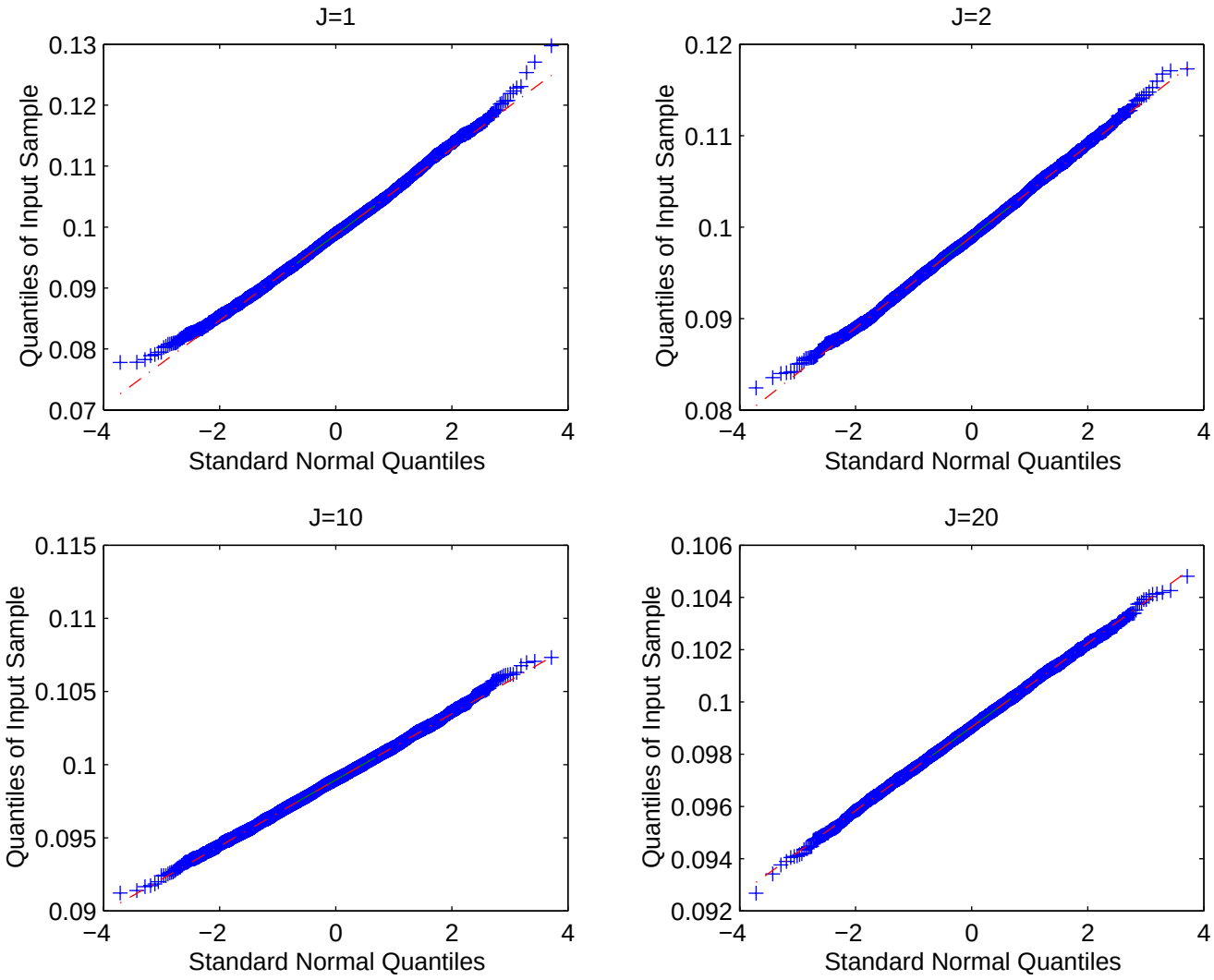

Figure 8: Normality plots for the MCMC draws for the jump intensity, λ, three-dimensional Merton's model, $G = 5000$, $J = 1, 2, 10, 20$.

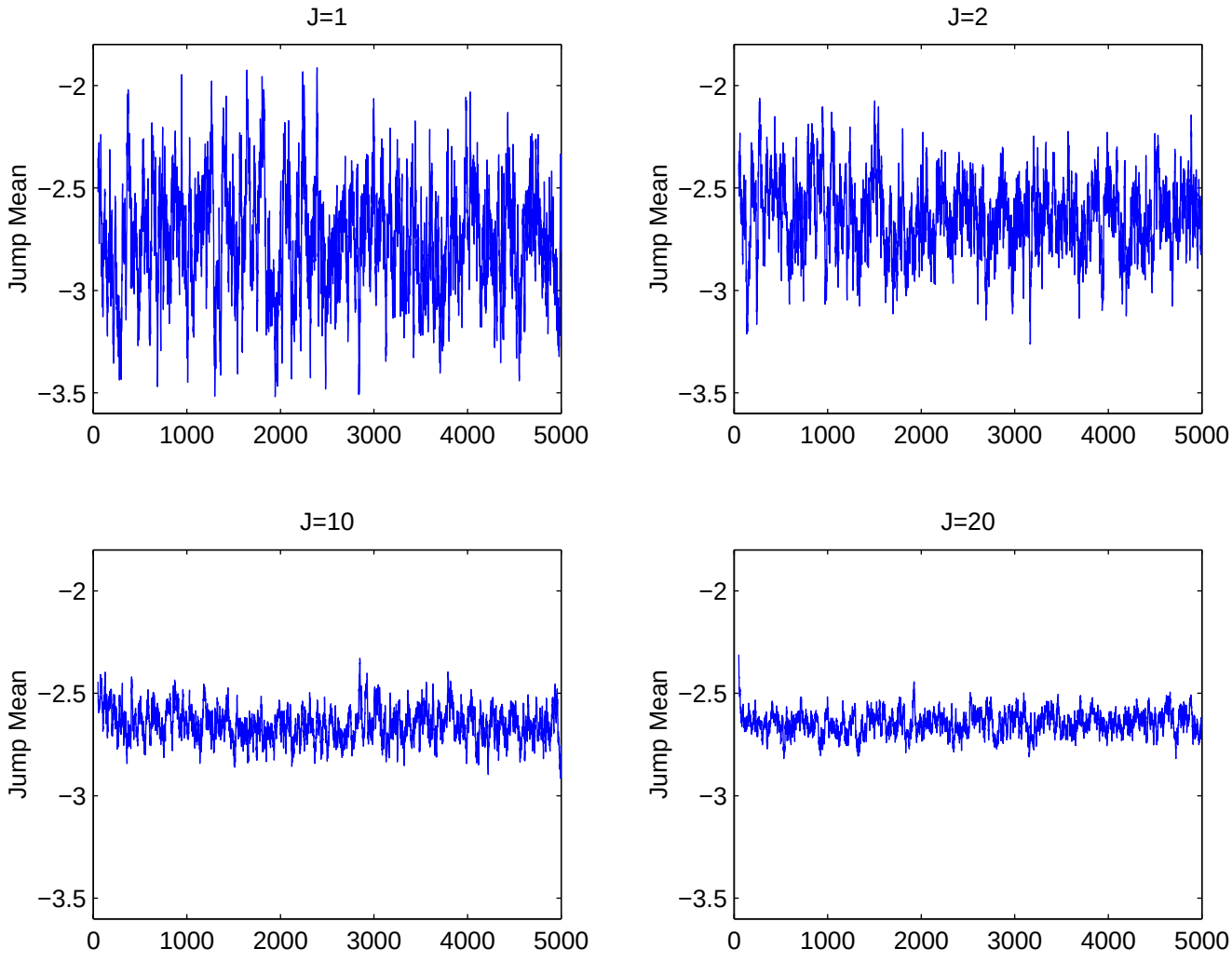

Figure 9: MCMC draws for the jump intensity, μ_z^1, three-dimensional Merton's model, $G = 5000$, $J = 1, 2, 10, 20$. The true μ_z^1 is -3.

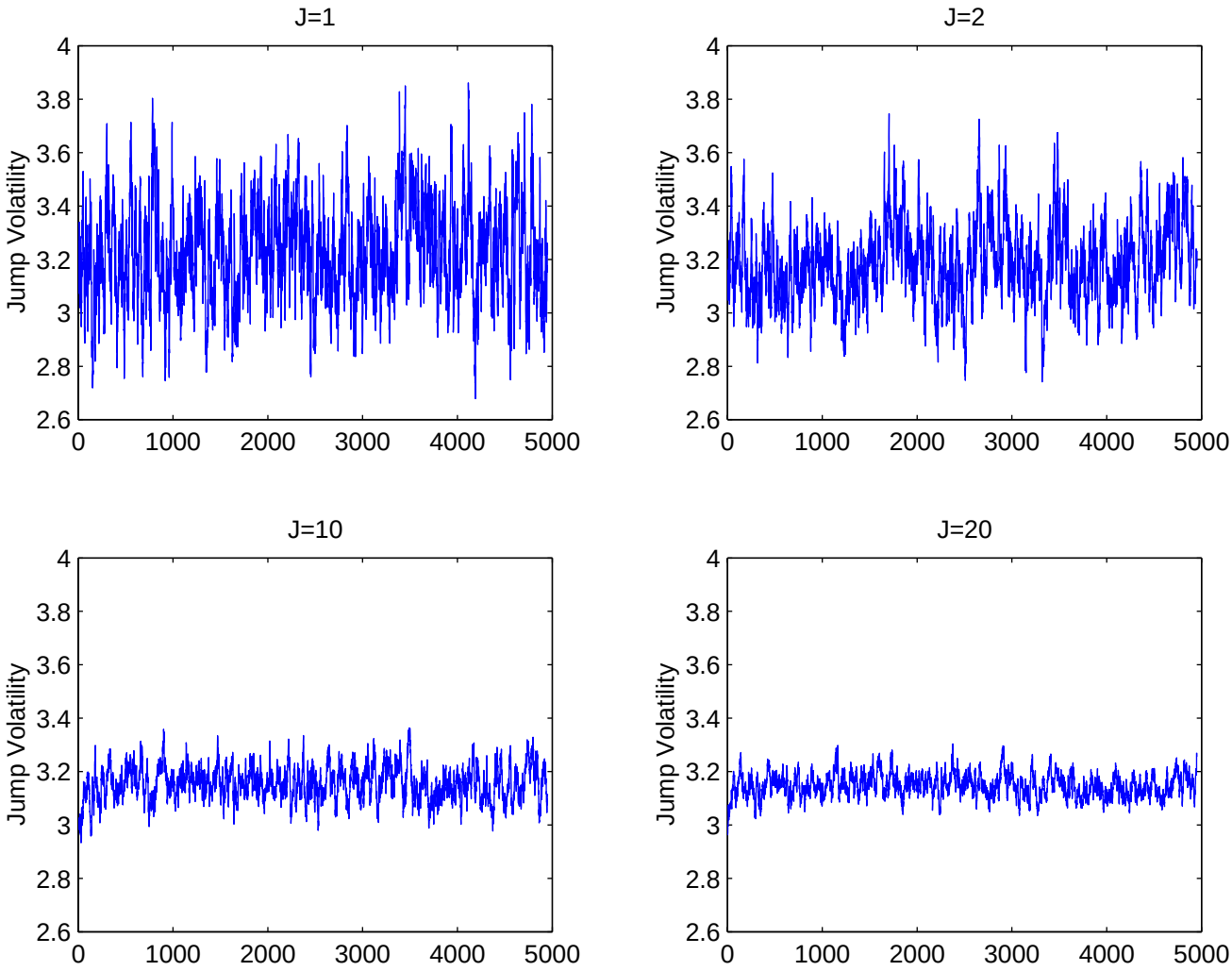

Figure 10: MCMC draws for the jump intensity, σ_z^1, from the three-dimensional Merton's model for $G = 5000$, and the cases $J = 1, 2, 10$, and 20. The true σ_z is 3.