ANALYSIS OF HIGH DIMENSIONAL MULTIVARIATE STOCHASTIC VOLATILITY MODELS

Siddhartha Chib
John M. Olin School of Business, Washington University, St Louis, MO 63130, USA

Federico Nardari
Department of Finance, Arizona State University, Tempe, AZ 85287, USA

Neil Shephard
Nuffield College, University of Oxford, Oxford OX1 1NF, UK

July 2001; revised November 2002

Abstract

This paper is concerned with the Bayesian estimation and comparison of flexible, high dimensional multivariate time series models with time varying correlations. The model proposed and considered here combines features of the classical factor model with that of the heavy tailed univariate stochastic volatility model. A unified analysis of the model, and its special cases, is developed that encompasses estimation, filtering and model choice. The centerpieces of the estimation algorithm (which relies on MCMC methods) are (1) a reduced blocking scheme for sampling the free elements of the loading matrix and the factors and (2) a special method for sampling the parameters of the univariate SV process. The resulting algorithm is scalable in terms of series and factors and simulation-efficient. Methods for estimating the log-likelihood function and the filtered values of the time-varying volatilities and correlations are also provided. The performance and effectiveness of the inferential methods are extensively tested using simulated data. In sum, our procedures lead to the first practical inferential approach for truly high dimensional models of stochastic volatility.

Keywords: Bayesian inference; Markov Chain Monte Carlo; Marginal likelihood; Metropolis-Hastings algorithm; Particle filter; Simulation; State space model; Stochastic jumps; Student-t distribution; Volatility.

1 INTRODUCTION

Two classes of models, ARCH and stochastic volatility (SV), have emerged as the dominant approaches for modeling financial volatility (Bollerslev, Engle, and Nelson (1994) and Ghysels, Harvey, and Renault (1996)). For the most part, the literature has dealt with univariate processes despite the need for multivariate models in areas such as asset pricing, portfolio analysis, and risk management. Although some multivariate models of volatility have been proposed, inference is restricted to specifications involving only a few variables, largely because of the proliferation of parameters in

high-dimensions. A major aim of this paper is to overcome this problem and demonstrate a unified Bayesian fitting and inference framework for truly high dimensional multivariate SV models.

In previous work within the ARCH tradition, multivariate models of volatility have been discussed by Bollerslev, Engle, and Wooldridge (1988), Diebold and Nerlove (1989), Engle, Ng, and Rothschild (1990) and King, Sentana, and Wadhwani (1994). Unfortunately, these generalizations are parameter rich and difficult to estimate due to complicated constraints on the parameter space. More tractable versions of multivariate ARCH models (Bollerslev, Engle, and Nelson (1994, pp. 3002-10)) are not generally capable of modeling the complexities of the data (e.g. Bollerslev (1990) assumes that the conditional correlations amongst the series are constant over time). Engle and Sheppard (2001) have tried to overcome this problem but only two parameters index the time-varying multivariate correlation matrix. On the other hand, in the stochastic volatility context, multivariate models are discussed by Harvey, Ruiz, and Shephard (1994), Jacquier, Polson, and Rossi (1995), Kim, Shephard, and Chib (1998), Pitt and Shephard (1999b), and Aguilar and West (2000) but the models in these papers are rather special and the estimation approaches are not scalable in the dimension of the model.

In this paper we specify and estimate a new and flexible multivariate SV model that permits both series-specific jumps at each time, and student-t innovations with unknown degrees of freedom. Let $y_t = (y_{1t}, ..., y_{pt})'$ denote the p observations at time t ($t \leq n$) and suppose that conditioned on k unobserved factors $f_t = (f_{1t}, ..., f_{kt})'$ and p independent Bernoulli "jump" random variables q_t, we have

$$y_t = Bf_t + K_t q_t + u_t \,, \tag{1.1}$$

where B is a matrix of unknown parameters (subject to the identifying restrictions $b_{ij} = 0$ for $j > i$ and $b_{ii} = 1$ for $i \leq k$), $K_t = \text{diag}\{k_{1t}, ..., k_{pt}\}$ are the jump sizes, and u_t is a vector of innovations. Assume that each element q_{jt} of q_t takes the value one with probability κ_j and the value zero with probability $1 - \kappa_j$, and that each element u_{jt} of u_t follows an independent student-t distribution with degrees of freedom $\nu_j > 2$, which we express in hierarchical form as

$$u_{jt} = \lambda_{jt}^{-1/2} \varepsilon_{jt}, \quad \lambda_{jt} \overset{i.i.d.}{\sim} \text{gamma}\left(\frac{\nu_j}{2}, \frac{\nu_j}{2}\right), \; t = 1, 2, ..., n, \tag{1.2}$$

where

$$\begin{pmatrix} \varepsilon_t \\ f_t \end{pmatrix} | V_t, D_t, K_t, q_t \sim \mathrm{N}_{p+k} \left\{ 0, \begin{pmatrix} V_t & 0 \\ 0 & D_t \end{pmatrix} \right\}$$

are conditionally independent Gaussian random vectors. The time-varying variance matrices V_t and D_t are taken to depend upon unobserved random variables (log-volatilities) $h_t = (h_{1t}, ..., h_{p+k,t})$ in the form

$$\begin{aligned} V_t = V_t(h_t) &= \mathrm{diag}\left\{\exp(h_{1t}), ..., \exp(h_{pt})\right\} : p \times p \\ D_t = D_t(h_t) &= \mathrm{diag}\left\{\exp(h_{p+1,t}), ..., \exp(h_{p+k,t})\right\} : k \times k \,, \end{aligned} \tag{1.3}$$

where each h_{jt} follows an independent three-parameter $(\mu_j, \phi_j, \sigma_j)$ stochastic volatility process

$$h_{jt} - \mu_j = \phi_j(h_{jt-1} - \mu_j) + \sigma_j \eta_{jt} \,,\ \eta_{jt} \overset{i.i.d.}{\sim} \mathrm{N}(0,1) \,. \tag{1.4}$$

Our model specification is completed by assuming that the variables $\zeta_{jt} = \ln(1 + k_{jt})$, $j \leq p$, are distributed as $\mathrm{N}(-0.5\delta_j^2, \delta_j^2)$, where $\delta = (\delta_1, ..., \delta_p)$ are unknown parameters. This assumption is similar to that made by Andersen, Benzoni, and Lund (2002) in a different context and models the belief that the expected value of k_{jt} is zero.

To understand the size of this model in terms of parameters and latent variables, let β denote the elements of B after imposing the identifying restrictions. Then there are $pk-(k^2+k)/2$ elements in β, $3(p+k)$ parameters $\theta_j = (\phi_j, \mu_j, \sigma_j^2)$, $j \leq p$, in the autoregressive process of $\{h_{jt}\}$, p degrees of freedom $\nu = (\nu_1, ..., \nu_p)$, p jump intensities $\kappa = (\kappa_1, ..., \kappa_p)$, and p jump variances $\delta = (\delta_1, ..., \delta_p)$. If we let $\psi = (\beta, \theta_1, ..., \theta_{p+k}, \nu, \delta, \kappa)$ denote the entire list of parameters, then the dimension of ψ is 688 when $p = 50$ and $k = 8$, as in one of our models below. Furthermore, the model contains $n(p+k)$ latent volatilities $\{h_t\}$ that appear non-linearly in the specification of V_t and D_t, $2np$ latent variables $\{q_t\}$ and $\{k_t\}$ associated with the jump component, and np scaling variables $\{\lambda_t\}$.

In the sequel, we refer to our model as the multivariate stochastic volatility jump model with student-t errors, or *MSVJt* for short. We use the acronyms *MSVt* to denote the model without jumps, *MSVJ* to denote the model with jumps and Gaussian errors, and *MSV* to denote the model with no jumps and Gaussian errors. We compare and contrast all four models in our empirical exercises.

The rest of the paper is organized as follows. In Section 2 we discuss the Bayesian estimation approach for the *MSVJ-t* model. Because of the rather complicated form of the likelihood function, we estimate the model by Markov chain Monte Carlo methods. The problem of model comparisons is taken up in Section 3 where we develop an approach for estimating the marginal likelihood and Bayes factors for competing models. In this context, we include a simulation-based sequential procedure for computing the filtered values of the unknown volatilities. In Section 4 we provide a detailed simulation study of the performance of our estimation and model choice procedures. We conclude with some brief remarks in Section 5.

2 ESTIMATION OF THE *MSVJt* MODEL

2.1 Preliminaries

If we let $\mathcal{F}_{t-1}$ denote the history of the $\{y_t\}$ process up to time $t-1$, and $p(h_t, \lambda_t, K_t, q_t|\mathcal{F}_{t-1}, \psi)$ the density of the latent variables $(h_t, \lambda_t, K_t, q_t)$ conditioned on $(\mathcal{F}_{t-1}, \psi)$, then the likelihood function of ψ given the data $y = (y_1, ..., y_n)$ is

$$\begin{aligned} p(y|\psi) &= \prod_{t=1}^{n} \int p(y_t|h_t, \lambda_t, K_t, q_t, B) p(h_t, \lambda_t, K_t, q_t|\mathcal{F}_{t-1}, \psi)\, dh_t\, d\lambda_t\, dK_t\, dq_t \\ &= \prod_{t=1}^{n} \int \mathrm{N}_p(y_t|K_t q_t, \Omega_t) p(h_t, \lambda_t, K_t, q_t|\mathcal{F}_{t-1}, \psi)\, dh_t\, d\lambda_t\, dK_t\, dq_t\ , \end{aligned} \tag{2.5}$$

where $\mathrm{N}_p(\cdot|\cdot, \cdot)$ is the multivariate normal density function, $K_t q_t$ is the mean of y_t marginalized over f_t,

$$\Omega_t = V_t^* + BD_t(h_t)B' \quad \text{and} \quad V_t^* = V_t(h_t) \odot \mathrm{diag}(\lambda_{1t}^{-1}, ..., \lambda_{pt}^{-1})$$

is the marginalized variance depending on the latent volatilities h_t, λ_t and the loading matrix B, and the symbol $\odot$ denotes element-by element multiplication. It is not difficult to see that neither $p(h_t, \lambda_t, K_t, q_t|\mathcal{F}_{t-1}, \psi)$ nor the integral of $\mathrm{N}_p(y_t|K_t q_t, \Omega_t)$ over $(h_t, \lambda_t, K_t, q_t)$ are available in closed form.

We utilize Markov chain Monte Carlo (MCMC) methods to develop a practical Bayesian estimation approach for this model; Chib (2001) provides an extensive review of these methods. In the MCMC approach, the posterior distribution is sampled by simulation methods and the draws generated from the simulation are used to summarize the posterior distribution. The simulation is

conducted by devising, and simulating, the transition density of an irreducible, aperiodic Markov chain whose invariant distribution is the target posterior distribution. In order to implement this approach, one basic idea is to avoid the direct use of the likelihood function (which is, of course, rather complicated and difficult to compute) and to focus on the posterior distribution of the parameters and the latent variables

$$\pi\left(\beta, \{f_t\}, \{\theta_j\}, \{h_{j.}\}, \{\nu_j\}, \{\lambda_{j.}\}, \{\delta_j\}, \{\kappa_j\}, \{\zeta_{j.}\}, \{q_{j.}\}|y\right), \tag{2.6}$$

where the notation $z_{j.}$ is used to denote the collection $(z_{j1}, ..., z_{jn})$. This distribution is quite high-dimensional but as we show in the rest of the paper it can be sampled efficiently by MCMC methods provided the Markov chain is carefully constructed. Efficiency in this context refers to the serial correlations in the sampled output and is measured, for each parameter in turn, by the inefficiency factor which is, intuitively speaking, one plus the sum of all the serial correlations.

One issue of particular importance is the type of "blocking" that is used. A single block MCMC algorithm proceeds by moving ψ in one simultaneous move from the current point to the next point in the chain. Since this is infeasible given the high-dimension of ψ, the Markov chain is constructed by a divide and conquer strategy wherein blocks of parameters are updated conditioned on the values of the remaining blocks; a single, complete transition of the Markov chain occurs when all the blocks have been thus revised. By carefully managing the blocking structure we show that efficiency of the simulation scheme can be increased by orders of magnitude; without use of our refinements, the inefficiency factors often exceed 1000, and with our refinement they range between 20 to 50, even in the our highest dimensional models. The practical ramifications of this are significant. If in one case one needed perhaps quarter-million samples from the posterior distribution, in the other 10000 would suffice. For models as large as we are interested in fitting, the value of this improvement cannot be overstated.

2.2 Proposed MCMC algorithm

One key step in our algorithm is the sampling of β and the factors $\{f_t\}$ in one block, conditioned on $\left(y, \{h_{j.}\}, \{\lambda_{j.}\}, \{\zeta_{j.}\}, \{q_{j.}\}\right)$. Because β and f_t appear in product form, the obvious approach of sampling β conditioned on $\{f_t\}$ and then sampling $\{f_t\}$ conditioned on β is less effective, which we demonstrate in Section 4.

Consider then the sampling of β from the density

$$\begin{aligned}\pi\left(\beta|y,\{h_{j.}\},\{\zeta_{j.}\},\{q_{j.}\},\{\lambda_{j.}\}\right) &\propto p(\beta)\prod_{t=1}^{n}p(y_t|B,h_t,\zeta_t,q_t,\lambda_t)\\ &\propto p(\beta)\prod_{t=1}^{n}\mathrm{N}_p(y_t|K_tq_t,V_t^*+BD_tB')\end{aligned}$$

where $p(\beta)$ is the normal prior density defined above. To sample this density, which is typically quite high-dimensional, we use the Metropolis-Hastings (M-H) algorithm (Chib and Greenberg (1995)). We follow Chib and Greenberg (1994) and take the proposal density to be multivariate-t, $\mathrm{T}(\beta|m,\Sigma,\nu)$, where m is the approximate mode of $l=\log\{\prod_{t=1}^{n}\mathrm{N}_p(y_t|K_tq_t,\Omega_t)\}$, and Σ is minus the inverse of the second derivative matrix of l; the degrees of freedom ν is set arbitrarily at fifteen. If we let the ij^{th} free element of B be denoted by b_{ij} and define $\widetilde{y}_t=y_t-K_tq_t$, we have that

$$l=\sum_{t=1}^{n}\log\phi_p(y_t|K_tq_t,\Omega_t)=const-\frac{1}{2}\sum_{t=1}^{n}\log|\Omega_t|-\frac{1}{2}\sum_{t=1}^{n}(y_t-K_tq_t)'\Omega_t^{-1}(y_t-K_tq_t),$$

and

$$\begin{aligned}\frac{\partial l}{\partial b_{ij}}&=\frac{1}{2}\sum_{t=1}^{n}\left\{\widetilde{y}_t'\Omega_t^{-1}\frac{\partial\Omega_t}{\partial b_{ij}}\Omega_t^{-1}\widetilde{y}_t-tr\left(\Omega_t^{-1}\frac{\partial\Omega_t}{\partial b_{ij}}\right)\right\}\\&=\sum_{t=1}^{n}\left\{s_t'\frac{\partial B}{\partial b_{ij}}D_tB's_t-tr\left(E_t\frac{\partial B}{\partial b_{ij}}'\right)\right\},\end{aligned}$$

where $s_t=\Omega_t^{-1}\widetilde{y}_t$, $E_t=\Omega_t^{-1}BD_t$, and $\Omega_t^{-1}=(V_t^*)^{-1}-(V_t^*)^{-1}B\left\{D_t^{-1}+B'(V_t^*)^{-1}B\right\}^{-1}B'(V_t^*)^{-1}$. With these derivatives, (m,Σ) can be found by a sequence of Newton-Raphson iterations. Then the M-H step for sampling β is implemented by drawing a value β^* from the multivariate-t distribution, namely $\mathrm{T}(m,\Sigma,\nu)$, and accepting the proposal value with probability $\alpha(\beta,\beta^*|\widetilde{y},\{h_{j.}\},\{\lambda_{j.}\})$, where β is the current value, and

$$\alpha(\beta,\beta^*|\widetilde{y},\{h_{j.}\},\{\lambda_{j.}\})=\min\left\{1,\frac{p(\beta^*)\prod_{t=1}^{n}\mathrm{N}_p(\widetilde{y}_t|0,V_t^*+B^*D_tB^{*\prime})}{p(\beta)\prod_{t=1}^{n}\mathrm{N}_p(\widetilde{y}_t|0,V_t^*+BD_tB')}\frac{\mathrm{T}(\beta|m,\Sigma,\nu)}{\mathrm{T}(\beta^*|m,\Sigma,\nu)}\right\}\tag{2.7}$$

is the probability of move. If the proposal value is rejected, the next item of the chain is taken to be the current value β.

The joint sampling of β and the factors is completed by sampling $\{f_t\}$ from the distribution

$\{f_t\}|\widetilde{y}, B, h, \lambda$. This step is simple because the latter distribution breaks up into the product of the distributions $f_t|\widetilde{y_t}, h_t, \lambda_t, B$. By standard Bayesian calculations, one can derive that the latter distribution is Gaussian with mean $\hat{f}_t = F_t B'(V_t^*)^{-1}\widetilde{y}_t$ and variance $F_t = \left(B'(V_t^*)^{-1}B + D_t^{-1}\right)^{-1}$.

The next step of the algorithm is particularly interesting because given $\left(y, B, \{f_t\}, \{\lambda_{j.}\}, \{\zeta_{j.}\}, \{q_{j.}\}\right)$, and the conditional independence of the errors in (1.3), the model can be devolved into $(p+k)$ conditionally Gaussian state space models. Namely, let $\alpha_t = Bf_t$, a p vector with components α_{jt}, and let

$$z_{jt} = \begin{cases} \ln(y_{jt} - \alpha_{jt} - (\exp(\zeta_{jt}) - 1)q_{jt} + c)^2 + \ln(\lambda_{jt}) & j \leq p \\ \ln(f_{j-p,t}^2) & j \geq p+1 \end{cases} . \tag{2.8}$$

where c is an "offset" constant that is set to 10^{-6}. Then from Kim, Shephard, and Chib (1998) it follows that the $p+k$ state space models can be subjected to an independent analysis for sampling the $\{\theta_j\}$ and $\{h_{j.}\}$. In particular, the distribution of z_{jt}, which is h_{jt} plus a log chi-squared random variable with one degree of freedom, may be approximated closely by a seven component mixture of normal distributions, allowing us to express the *MSVJ-t* model as

$$\begin{aligned} z_{jt}|s_{jt}, h_{jt} &\sim \mathrm{N}(h_{jt} + m_{s_{tj}}, v_{s_{tj}}^2) \\ h_{jt} - \mu_j &= \phi_j(h_{jt-1} - \mu_j) + \sigma_j \eta_{jt} \ , \ j \leq p+k \end{aligned} \tag{2.9}$$

where s_{jt} is a discrete component indicator variable with mass function $\Pr(s_{jt} = i) = q_i, \ i \leq 7$, $t \leq n$, and $m_{s_{tj}}$, $v_{s_{tj}}^2$ and q_i are parameters that are reported in Chib, Nardari, and Shephard (2002). Thus, under this representation, conditioned on the transformed observations we have that

$$p(\{s_{j.}\}, \theta, \{h_{j.}\}|z) = \prod_{j=1}^{k+p} p(s_{j.}, \theta_j, h_{j.}|z_{j.}),$$

which implies that the mixture indicators, log-volatilities and series specific parameters can be sampled series by series. This representation of the model is one reason that our approach is scalable in both p and k.

Now, for each j, one can sample $(s_{j.}, \theta_j, h_{j.})$ by the univariate SV algorithm given by Chib, Nardari, and Shephard (2002). Briefly, $s_{j.}$ is sampled straightforwardly from

$$p(s_{j.}|z_{j.}, h_{j.}) = \prod_{t=1}^{n} p(s_{jt}|z_{jt}, h_{jt}),$$

where $p(s_{jt}|z_{jt}, h_{jt}) \propto p(s_{jt})\phi(z_{jt}|h_{jt} + m_{s_{tj}}, v^2_{s_{tj}})$ is a mass function with seven points of support. Next, θ_j is sampled by the M-H algorithm from the density $\pi(\theta_j|z_{j.}, s_{j.}) \propto p(\theta_j)p(z_{j.}|s_{j.}, \theta_j)$ where

$$p(z_{j.}|s_{j.}, \theta_j) = p(z_{j1}|s_{j.}, \theta_j) \prod_{t=2}^{T} p(z_{jt}|\mathcal{F}^*_{jt-1}, s_{j.}, \theta_j), \tag{2.10}$$

and $p(z_{jt}|\mathcal{F}^*_{j,t-1}, s_{j.}, \theta_j)$ is a normal density whose parameters are obtained by the Kalman filter recursions, adapted to the differing components, as indicated by the component vector $s_{j.}$ Finally, $h_{j.}$ is sampled from $[h_{j.}|z_{j.}, s_{j.}, \theta_j]$ by the simulation smoother algorithm of de Jong and Shephard (1995).

In the remaining steps, the degrees of freedom parameters, jump parameters and associated latent variables are sampled independently for each time series. Significant improvements in simulation efficiency are achieved by sampling ν_j marginalized over $\lambda_{j.}$ from the multinomial distribution

$$\Pr(\nu_j|y_{j.}, h_{j.}, B, f, q_{j.}, \zeta_{j.}) \propto \Pr(\nu_j) \prod_{t=1}^{n} \mathrm{T}\left(y_{jt}|\alpha_{jt} + \left\{\exp(\zeta_{jt}) - 1\right\} q_{jt}, \exp(h_{jt}), \nu_j\right). \tag{2.11}$$

Next, the jump indicators $\{q_{j.}\}$ are sampled from the two-point discrete distribution

$$\begin{aligned}
\Pr(q_{jt} = 1|\ y_{j.}, h_{j.}, B, f, \nu_j, \zeta_{j.}, \kappa_j) &\propto \kappa_j \mathrm{T}\left(y_{jt}|\alpha_{jt} + \left\{\exp(\zeta_{jt}) - 1\right\}, \exp(h_{jt}), \nu_j\right) \\
\Pr(q_{jt} = 0|\ y_{j.}, h_{j.}, B, f, \nu_j, \zeta_{j.}, \kappa_j) &\propto (1 - \kappa_j)\mathrm{T}\left(y_{jt}|\alpha_{jt}, \exp(h_{jt}), \nu_j\right),
\end{aligned}$$

followed by the components of the vector $\lambda_{j.}$ from the density

$$\lambda_{jt}|y_{jt}, h_{jt}, B, f, \nu_j, q_{jt}, \psi_{jt} \sim \text{gamma}\left(\frac{\nu_j + 1}{2}, \frac{\nu_j + (y_{jt} - \alpha_{jt} - (\exp(\zeta_{jt} - 1)q_{jt})^2}{2\exp(h_{jt})}\right).$$

Another key step concerns the sampling of the parameters δ_j and $\zeta_{j.}$. For simulation efficiency reasons, these two parameters must also be sampled in one block. To see how this is possible, note that if k_{jt} is small, as is true in financial applications with high frequency returns that are measured in decimals, $\exp(\zeta_{jt})$ may be closely approximated by $1 + \zeta_{jt}$, implying that $k_{jt}q_{jt}$ equals $\zeta_{jt}q_{jt}$ and ζ_{jt} can be marginalized out. This permits the sampling of δ_j from the density

$$\pi(\delta_j) \prod_{t=1}^{n} \mathrm{N}\left(\alpha_{jt} - 0.5\delta_j^2 q_{jt}, \delta_j^2 q_{jt}^2 + \exp(h_{jt})\lambda_{jt}^{-1}\right) \tag{2.12}$$

by the M-H algorithm. Once δ_j is sampled, the vectors $\zeta_{j.}$ are sampled, bearing in mind that their posterior distribution is updated only when q_{jt} is one. Therefore, when q_{jt} is zero, we sample ζ_{jt} from $\mathrm{N}(-0.5\delta_j^2, \delta_j^2)$, otherwise we sample from the distribution $\mathrm{N}\left(\Psi_{jt}(-0.5 + \exp(-h_{jt})\lambda_{jt}y_{jt}), \Psi_{jt}\right)$, where $\Psi_{jt} = \left(\delta_j^{-2} + \exp(-h_{jt})\lambda_{jt}\right)^{-1}$. The algorithm is completed by sampling the components of the vector κ independently from $\kappa_j|q_{j.} \sim \mathrm{beta}(u_{0j}+n_{1j}, u_{1j}+n_{0j})$, where n_{0j} is the count of $q_{jt} = 0$ and $n_{1j} = n - n_{0j}$ is the count of $q_{jt} = 1$.

A complete cycle through these various distributions completes one transition of our Markov chain. These steps are then repeated G times, where G is a large number, and the values beyond a suitable burn-in of say a 1000 cycles, are used for the purpose of summarizing the posterior distribution.

3 MODEL COMPARISON

In this section we show how the *MSVJt* model can be compared with alternative multivariate and univariate specifications. We do this comparison based on the marginal likelihood of each model and the associated Bayes factors (ratios of marginal likelihoods). Because of the dimensions involved, computation of the marginal likelihood presents several challenges. Nonetheless, our study of the problem has revealed that the method of Chib (1995) and Chib and Jeliazkov (2001) is feasible in this context, and quite effective in picking the true model, as we demonstrate in our simulation exercises.

The starting point of the Chib method is the basic marginal likelihood identity under which the log of the Bayes factor for comparing non-nested models $\mathcal{M}_1$ to $\mathcal{M}_2$ can be written as

$$\begin{aligned}\log p(y|\mathcal{M}_1) - \log p(y|\mathcal{M}_2) = \log p(y|\mathcal{M}_1, \psi_1^*) + \log p(\psi_1^*|\mathcal{M}_1) - \log \pi(\psi_1^*|\mathcal{M}_1, y) \\ - \{\log p(y|\mathcal{M}_2, \psi_2^*) + \log p(\psi_2^*|\mathcal{M}_2) - \log \pi(\psi_2^*|\mathcal{M}_2, y)\}\end{aligned} \tag{3.13}$$

where $p(y|\mathcal{M}_j, \psi_j^*)$ is the likelihood function under $\mathcal{M}_j$, $p(\psi_j^*|\mathcal{M}_j)$ and $\pi(\psi_j^*|\mathcal{M}_j, y)$ are the corresponding prior and posterior densities, evaluated at some specified point ψ_j^*, say the posterior mean. The next step is to estimate the likelihood and posterior ordinates by some efficient method.

3.1 Posterior Ordinate

To estimate the posterior ordinate we use a marginal/conditional decomposition and the output of the original and subsequent "reduced MCMC runs." To explain this technique, let

$$\begin{aligned}\pi(\psi^*|\mathcal{M},y) &= \pi(\beta^*,\nu^*,\theta^*,\delta^*,\kappa^*|\mathcal{M},y)\\ &= \pi(\beta^*|\mathcal{M},y)\pi(\nu^*|\mathcal{M},y,\beta^*)\pi(\theta^*|\mathcal{M},y,\beta^*,\nu^*)\\ &\times \pi(\delta^*|\mathcal{M},y,\beta^*,\theta^*,\nu^*)\pi(\kappa^*|\mathcal{M},y,\beta^*,\theta^*,\nu^*,\delta^*)\end{aligned}$$

and consider the estimation of each of the terms starting in the second line of this decomposition. It turns out that for the sample sizes in our applications, the marginal posterior densities of the factor loadings are very concentrated around the mean and close to normal. For simplicity, therefore, we approximate $\pi(\beta^*|\mathcal{M},y)$ by the ordinate of a normal density with mean vector and covariance matrix obtained from the full MCMC run.

Second, to estimate the p-dimensional conditional ordinate $\pi(\nu^*|\mathcal{M},y,\beta^*)$ we fix β at β^* and continue our MCMC simulation for another G iterations. Within this run, the ordinates of the conditional mass function $\Pr(\nu_j|y_{j.},h_{j.},B^*,f,q_{j.},\zeta_{j.})$ are averaged and the resulting modal probability is taken as the estimate of $\pi(\nu^*|\mathcal{M},y,\beta^*)$.

Third, to estimate the conditional ordinate $\pi(\theta_1^*,...,\theta_{p+k}^*|y,\mathcal{M},\beta^*,\nu^*)$ we group the θ_j's in groups of two (each of dimension six) and produce output from the appropriate reduced MCMC runs to estimate the resulting ordinates. Specifically, to estimate $\pi(\theta_1^*,\theta_2^*|y,\mathcal{M},\beta^*,\nu^*)$ we fix β at β^*, ν at ν^* and run the MCMC algorithm given above. The desired ordinate is then estimated by the kernel smoothing method applied to the output on θ_1 and θ_2 from this run. The process is repeated in sequence, in each case with additional parameters held fixed.

Next, we estimate the ordinate $\pi(\delta^*|\mathcal{M},y,\beta^*,\nu^*,\theta^*)$ by applying a result given in Chib and Jeliazkov (2001). Specifically, it can be shown that the ordinate

$$\begin{aligned}\pi(\delta^*|\mathcal{M},y,\beta^*,\nu^*,\theta^*) &= \int\prod_{j=1}^{p}\pi(\delta_j^*|\mathcal{M},y_{j.},\beta^*,\nu^*,\theta^*,h_{j.},f,q_{j.},\lambda_{j.})\\ &\quad d\pi(\{h_{j.}\},f,\{q_{j.}\},\{\lambda_{j.}\}|\mathcal{M},y,\beta^*,\nu^*,\theta^*),\end{aligned}$$

where π denotes generically the distribution of the enclosed random vectors, can be expressed as

$$\frac{\mathrm{E}_1 \prod_{j=1}^{p} \alpha(\delta_j, \delta_j^* | \mathcal{M}, y_{j.}, \beta^*, \nu^*, \theta^*, h_{j.}, f, q_{j.}, \lambda_{j.}) q(\delta_j^* | \mathcal{M}, y_{j.}, \beta^*, \nu^*, \theta^*, h_{j.}, f, q_{j.}, \lambda_{j.})}{\mathrm{E}_2 \prod_{j=1}^{p} \alpha(\delta_j^*, \delta_j | \mathcal{M}, y_{j.}, \beta^*, \nu^*, \theta^*, h_{j.}, f, q_{j.}, \lambda_{j.})} \tag{3.14}$$

where α is the probability of move in the M-H step for δ_j, q is the Student-t proposal density in that step, E_1 is the expectation with respect to $\pi(\{h_{j.}\}, f, \{q_{j.}\}, \{\lambda_{j.}\} | \mathcal{M}, y, \beta^*, \nu^*, \theta^*)$ and E_2 is the expectation with respect to

$$\pi(\{h_{j.}\}, f, \{q_{j.}\}, \{\lambda_{j.}\} | \mathcal{M}, y, \beta^*, \nu^*, \theta^*, \delta^*) \prod_{j=1}^{p} q(\delta_j | \mathcal{M}, y, \beta^*, \nu^*, \theta^*, \delta^*, h_{j.}, f, q_{j.}, \lambda_{j.}).$$

The first of these expectations can be computed from the output of a reduced MCMC run in which β, ν, and θ are fixed at their starred values. The second expectation can be computed from the output of an additional reduced run in which δ is also fixed; for each draw of $\{h_{j.}\}, f, \{q_{j.}\}, \{\lambda_{j.}\}$ in this reduced run, δ_j is drawn from the proposal density and these combined draws are used to average the probability of move in the denominator of (3.14).

Finally, to estimate the κ^* conditional ordinate, the parameters $(\beta, \nu, \theta, \delta)$ are fixed and the quantities $\{q_t, \kappa\}$ are drawn in a reduced MCMC run. The required ordinate then follows by averaging the beta density of κ.

3.2 Filtering and Likelihood Evaluation

We now discuss a simulation-based approach, called the auxiliary particle filtering method (see Pitt and Shephard (1999a) and the book length review of Doucet, de Freitas, and Gordon (2001)), to estimate the likelihood ordinate $\log f(y_1, ..., y_n | \mathcal{M}, \psi^*) = \sum_{t=1}^{n} \log f(y_t | \mathcal{M}, \mathcal{F}_{t-1}, \psi^*)$, where

$$f(y_t | \mathcal{M}, \mathcal{F}_{t-1}, \psi^*) = \int \mathrm{N}_p(y_t | K_t q_t, \Omega_t) p(\lambda_t, K_t, q_t | \mathcal{M}, \psi^*) p(h_t | \mathcal{M}, \mathcal{F}_{t-1}, \psi^*) \; dh_t \; d\lambda_t \; dK_t \; dq_t$$

is the one-step-ahead predictive density of y_t,

$$p(h_t | \mathcal{M}, \mathcal{F}_{t-1}, \psi^*) = \int p(h_t | \mathcal{M}, h_{t-1}, \psi^*) p(h_{t-1} | \mathcal{M}, \mathcal{F}_{t-1}, \psi^*) \; dh_{t-1},$$

is the one-step-ahead predictive density of h_t, $p(h_t|\mathcal{M}, h_{t-1}, \psi^*) = \prod_{j=1}^{p+k} \mathrm{N}(h_{tj}|\mu_j^* + \phi_j^*(h_{j,t-1} - \mu_j^*), \sigma^2)$ is the product of the Markov transition densities and $p(h_{t-1}|\mathcal{M}, \mathcal{F}_{t-1}, \psi^*)$ is the posterior distribution of h_{t-1} given $\mathcal{F}_{t-1}$ (the filtered distribution).

We now use a sequential Monte Carlo filtering procedure to efficiently estimate the one-step ahead predictive density of y_t given above. In this procedure, samples (particles) from the preceding filtered distribution (e.g., $p(h_{t-1}|\mathcal{M}, \mathcal{F}_{t-1}, \psi^*)$) are propogated forward to produce samples from the subsequent filtered distribution (namely, $p(h_{t-1}|\mathcal{M}, \mathcal{F}_{t-1}, \psi^*)$). Suppose then that we have a sample $h_{t-1}^{(g)}$ $(g \leq M)$ from the filtered distribution $h_{t-1}|\mathcal{M}, \mathcal{F}_{t-1}, \psi^*$. Based on this sample, we can approximate the one-step-ahead predictive density of h_t as

$$p(h_t|\mathcal{M}, \mathcal{F}_{t-1}, \psi^*) \simeq \frac{1}{M} \sum_{g=1}^{M} p(h_t|\mathcal{M}, h_{t-1}^{(g)}, \psi^*)$$

Under this approximation, the posterior density of the latent variables at time t is available as

$$\begin{aligned} p(\lambda_t, K_t, q_t, h_t|\mathcal{M}, \mathcal{F}_t, \psi^*) &\propto \mathrm{N}_p(y_t|K_t q_t, \Omega_t) p(\lambda_t, K_t, q_t|\mathcal{M}, \psi^*) p(h_t|\mathcal{M}, \mathcal{F}_{t-1}, \psi^*) \\ &\dot{\propto} \mathrm{N}_p(y_t|K_t q_t, \Omega_t) p(\lambda_t, K_t, q_t|\psi^*) \frac{1}{M} \sum_{g=1}^{M} f(h_t|\mathcal{M}, h_{t-1}^{(g)}, \psi^*) \quad (3.15) \end{aligned}$$

and the objective is to sample this density. This sampling is carried out as follows. In the first stage, proposal values $h_t^{*(1)}, ..., h_t^{*(R)}$ are created. These values are then resampled to produce the draws $\{h_t^{(1)}, ..., h_t^{(M)}\}$ that correspond to draws from (3.15). We have found that R should be five or ten times larger than M to ensure efficient propagation of the particles. We summarize the steps in the following algorithm.

Auxiliary particle filter for multivariate SV model

1. Given values $\{h_{t-1}^{(1)}, ..., h_{t-1}^{(M)}\}$ from $(h_{t-1}|\mathcal{M}, \mathcal{F}_{t-1}, \psi^*)$ calculate $\hat{h}_t^{*(g)} = E\left(h_t^{(g)}|h_{t-1}^{(g)}\right)$ and

$$w_g = \mathrm{N}_p(y_t|0, \Omega_t(\hat{h}_t^{*(g)}, 1, B^*), \psi^*) \ , \ g = 1, ..., M \, ,$$

and sample R times the integers $1, 2, ..., M$ with probability $\overline{w}_t^g = w_g / \sum_{j=1}^{M} w_j$. Let the sampled indexes be $k_1, ..., k_R$ and associate these with $\hat{h}_t^{*(k_1)}, ..., \hat{h}_t^{*(k_R)}$.

2. For each value of k_g from Step 1, simulate the values $\{h_t^{*(1)}, ..., h_t^{*(R)}\}$ from

$$h_{j,t}^{*(g)} = \mu_j^* + \phi_j^*(h_{j,t-1}^{(k_g)} - \mu_j^*) + \sigma_j^* \eta_{j,t}^{(g)}\ ,\ g = 1, ..., R,$$

where $\eta_{j,t}^{(g)} \sim N(0,1)$. Likewise draw $\lambda_t^{(g)}$, $K_t^{(g)}$, $q_t^{(g)}$ from their prior $p(\lambda_t, K_t, q_t|\psi^*)$, where $K_t^{(g)} = \text{diag}\left\{k_{1t}^{(g)}, ..., k_{pt}^{(g)}\right\}$ and $\zeta_{jt}^{(g)} = \ln(1 + k_{jt}^{(g)})$ is drawn from $\text{N}(-0.5\delta_j^{*2}, \delta_j^{*2})$.

3. Resample the values $\{h_t^{*(1)}, ..., h_t^{*(R)}\}$ M times with replacement using probabilities proportional to

$$w_g^* = \frac{\text{N}_p(y_t|K_t^{(g)} q_t^{(g)}, \Omega_t(h_t^{*(g)}, \lambda_t^{(g)}, B^*))}{\text{N}_p\left(y_t|0, \Omega_t(\hat{h}_t^{*(k_g)}, 1, B^*)\right)},\ g = 1, ..., R\,,$$

to produce the desired filtered sample $\{h_t^{(1)}, ..., h_t^{(M)}\}$ from $(h_t|\mathcal{M}, \mathcal{F}_t, \psi^*)$.

As discussed by Pitt (2001), the weights produced in the above algorithm provide a simulation-consistent estimate of the likelihood contribution. In particular,

$$\widehat{f}(y_t|\mathcal{M}, \mathcal{F}_{t-1}, \psi^*) = \left(\frac{1}{M}\sum_{g=1}^{M} w_g\right)\left(\frac{1}{R}\sum_{g=1}^{R} w_g^*\right)$$

which can be shown to converge to $f(y_t|\mathcal{M}, \mathcal{F}_{t-1}, \psi^*)$ in probability as M and R go to infinity. These estimates are obtained for each t and combined to produce our estimate of the likelihood ordinate $\log f(y_1, ..., y_n|\mathcal{M}, \psi^*)$.

4 SIMULATION STUDY

We now provide evidence, with the help of several simulated data sets, of the efficacy of the methods proposed in this paper. We examine the simulation efficiency of the fitting method, its estimation accuracy, robustness to changes in the prior, and of the reliability of the model selection method.

4.1 Prior distribution

In the experiments we assume that the parameters are mutually independent with distributions specified as follows. Free elements of $B : b_{ij} \sim \text{N}(1, 9)$; $\phi : \phi_j^* \sim \text{beta}(a, b)$, where $\phi_j = 2\phi_j^* - 1$, so that the prior mean of ϕ_j is 0.86 and standard deviation is 0.11; $\sigma : \sigma_j \sim \text{IG}(c/2, d/2)$ with mean of 0.25 and standard deviation of 0.4; $\nu : \nu_j$ is discrete uniform over the grid (5, 8, 11, 14, 17, 20,

30, 60); $\kappa : \kappa_j \sim \text{beta}(2, 100)$ implying jumps about 50 observations apart; and $\log(\delta) : \log(\delta_j) \sim \text{N}(-3.07, 0.148)$ implying a mean of 0.05 and standard deviation of 0.02 on δ_j.

4.2 Simulation Efficiency

A key feature of our estimation method is the sampling of B marginalized over the factors. Whereas it is simpler to condition on the factors, as done by Geweke and Zhou (1996), Pitt and Shephard (1999b), Aguilar and West (2000) and Jacquier, Polson, and Rossi (1995) in the context of static and dynamic factor models, the sampled output is far less well behaved. To show this, we generate eight datasets, labeled D1-D8, from different models and with different number of assets, factors and time series observations, and evaluate the alternative samplers in terms of the realized inefficiency factors. The inefficiency factor is the inverse of the numerical efficiency measure in Geweke (1992) and is computed from the MCMC output as the square of the numerical standard error divided by the variance of the posterior estimate under (hypothetical) i.i.d. sampling.

In generating the data, we draw the parameters of the models from the following distributions: the free elements of b_{ij} are from $\text{N}(0.9, 1)$; μ_j from $\text{N}(-9, 1)$, ϕ_j from a scaled beta with mean 0.95 and variance 0.03, σ_j from $\text{IG}(2.5, 0.5)$; ν_j from its prior; $\log \delta_j$ from $\text{N}(-3.07, 0.148)$; and κ_j from a $\text{beta}(2, 100)$ distribution. The specifics of each dataset are shown in Table 1. It should be noted that the models are quite high-dimensional; the smallest has 142 parameters and the largest has 688.

Dataset	Model	p	k	n	Parms	Dataset	Model	p	k	n	Parms
D1	*MSV*	20	4	2,000	142	D2	*MSV*	50	4	2,000	352
D3	*MSV*	20	4	1,000	142	D4	*MSV*	50	4	1,000	352
D5	*MSV*	20	4	5,000	142	D6	*MSV*	50	4	5,000	352
D7	*MSV*	40	8	2,000	428	D8	*MSVJt*	50	8	2,000	688

Table 1: *Features of simulated datasets. Parms denotes the number of parameters.*

For each dataset, we employ the marginalized sampling procedure and two other methods where the elements of B are sampled either by column or by row, conditioned on the factors. For the algorithm proposed in this paper we run the MCMC sampler for 11000 iterations, collecting the last 10000 for inferential purposes. For the other two methods, expecting a drop in simulation efficiency, we collect 50000 draws after discarding the first 5000. We compare the three methods, as they relate to the sampling of B, in terms of the relative inefficiency factors (the ratio of inefficiency

factors). As can be seen from Table 2, in models with four factors (D1 through D6) our procedure is between 20 and 40 times more efficient than the other two methods. In models with eight factors (D7 and D8), our method is about 80 times more efficient. Furthermore, the efficiency of our method does not erode as the dimensionality and complexity of the model is increased whereas the other methods become even less efficient. The performance gains from sampling B in the way we

Sampling B	Mean	S.D	Low	Upp	Max	Min	Mean	S.D	Low	Upp	Max	Min
			D1						D2			
Row/Marg	34.5	17.3	24.6	46.0	71.0	1.8	29.7	17.0	17.9	39.4	91.4	6.2
Col/Marg	37.9	21.1	22.1	50.3	83.9	1.3	33.4	19.5	18.8	45.4	106	7.4
Col/Row	0.8	0.6	0.5	1.7	2.3	0.2	1.2	0.6	0.8	1.5	3.8	0.5
			D3						D4			
Row/Marg	36.4	29.5	14.1	46.0	113	4.0	41.7	23.7	26.8	51.2	132	2.9
Col/Marg	27.5	15.1	16.5	41.4	59.3	3.6	15.9	9.8	7.6	20.4	45.7	2.0
Col/Row	0.9	0.3	0.7	1.2	1.7	0.5	0.4	0.2	0.3	0.6	1.1	0.1
			D5						D6			
Row/Marg	24.0	27.3	4.6	31.8	167	2.3	88.0	49.3	42.7	131	185	11.8
Col/Marg	14.8	16.2	4.0	18.3	101	1.9	62.3	45.5	26.4	91.0	231	2.3
Col/Row	0.7	0.2	0.5	0.9	1.8	0.3	0.9	1.1	0.4	1.0	9.0	0.1
			D7						D8			
Row/Marg	62.3	36.0	33.4	87.1	161	9.6	76.9	54.7	25.9	119	279	3.3
Col/Marg	89.7	54.9	44.4	126	238	6.1	84.6	56.2	29.2	128	294	5.1
Col/Row	1.5	0.8	1.0	1.9	6.5	0.3	1.3	0.4	1.0	1.7	2.4	0.3

Table 2: *Summary output for inefficiency factors. The Table summarizes the distribution of relative inefficiency factors for the estimated factor loadings. Row denotes sampling by row, Col sampling by column and Marg sampling marginalized over the factors. Results are reported for different simulated datasets and for alternative sampling schemes for the factor loading matrix B. Low denotes the 25th. percentile, Upp denotes the 75th. percentile.*

suggest are worth the computational burden because substantially smaller Monte Carlo samples are needed to achieve a given level of numerical accuracy. On average, our procedure is $5-6$ times slower in terms of CPU time per MCMC iteration than the alternative non-marginalized methods. For a model with 30 series and 4 factors fit to 2000 observations, our MCMC algorithm coded in C and running on Linux 2.5 megahertz Pentium 4 computer consumes about 20 hours of CPU time to generate 10,000 MCMC draws.

We next consider the specifics of our MCMC scheme as they relate to the sampling of ν and δ. We generate an additional data set, D9, from the *MSVJt model* with 50 series, 4 factors and 2000 observations per series and we employ our method along with several alternatives where one or more of the reduced blocking steps in the generation of B, ν and δ are switched off. Efficiency

factors from these runs are reported in Table 3. Two patterns are noticeable. First, the reduced blocking scheme leads to much better mixing for both ν and δ. On aveerage, our proposed method is $40-50$ times more efficient than the alternatives. Second, these performance gains are realized even when B is sampled conditioned on the factors.

Sampling	Mean	S.D	Low	Upp	Mean	S.D	Low	Upp	Mean	S.D	Low	Upp
		B				δ				ν		
s1/s2	108.4	49.6	66.5	147.8	1.0	0.3	0.9	1.2	1.1	0.3	0.9	1.3
s3/s1	1.0	0.2	1.0	1.0	54.8	27.2	37.1	65.0	41.1	13.2	24.1	49.9
s3/s2	106.4	49.6	63.3	139.1	54.2	25.2	34.5	64.4	43.1	13.0	26.7	51.3

Table 3: *Summary output for inefficiency factors. The Table summarizes the distribution of relative inefficiency factors for the estimated factor loadings (B), degrees of freedom parameters (ν) and jump intensiy parameters (δ). Results are reported for a dataset of* 50 *series and* 2000 *observations per series and for alternative sampling schemes for B, ν and δ. Specifically, s1: B non marginalized, ν marginalized, δ marginalized. s2: all marginalized. s3: all non marginalized. Low denotes the 25th. percentile, Upp denotes the 75th. percentile.*

4.3 Parameter Estimates and Factor Extraction

In this section we first show the ability of the proposed algorithm to correctly estimate the large number of parameters and latent variables in the model. Second, we assess the robustness of the algorithm to changes in the prior. We contrast the results from our proposed method with those where B is sampled by columns, conditioned on the factors.

In these experiments, the artificial datasets are generated from the *MSVJt* model with forty series and eight factors. Each simulated series has 1250 observations, equivalent to about five years of daily data. We use the same mechanism described in the previous section to generate one set of true parameters. From these parameter values we then generate a total of 40 data sets and we fit the 8 factor *MSVJt* model to each of them. Due to the differences in the simulation efficiency, the preferred MCMC algorithm is run for 10000 iterations while the non-marginalized MCMC algorithm is run for 100000 iterations. We initially use the same priors reported in section 4.1, defined collectively as Prior1. Subsequently we repeat the estimation with a more diffuse independent $\mathrm{N}(0, 1000)$ prior on b_{ij}. This prior is labeled as Prior2.

Table 4 contains correlations between the true values and the parameter estimates for the alternative procedures and priors. The estimates are obtained as the grand averages of the posterior means across simulated samples.

Sampling	B	μ_a	ϕ_a	σ_a	δ	κ	μ_f	ϕ_f	σ_f	ν
Bmarg Prior1	.97	.99	.92	.92	.95	.92	.98	.91	.86	.82
Bbycol Prior1	.84	.99	.90	.88	.95	.92	.88	.77	.38	.82
Bmarg Prior2	.95	.99	.90	.88	.98	.92	.95	.94	.84	.82
Bbycol Prior2	.84	.99	.90	.87	.98	.92	.88	.77	.39	.81

Table 4: *Summary output for simulated data. Entries are the correlation coefficients between the true parameter values and MCMC estimates. The latter are the average of posterior means across 40 samples with $n = 1250$. Bmarg denotes the sampling of B marginalized over the latent factors, Bbycol denotes the sampling of B conditioning on the factors and done by column. Prior1 and Prior2 are defined in the main text.*

Consider first the estimates for the factor loading matrix, which in this case has 284 free parameters. The correlation between the true values and the grand averages across samples is substantially higher for the more efficient procedure: 97.28% vs. 83.88%. The bar graph in Figure 1 shows that the proposed approach yields accurate estimates of the B matrix (elements for only four factors are plotted). Second, the estimates of the volatility parameters for the factors (not reported) are noticeably more accurate for the preferred algorithm. Third, the estimates of the parameters in the volatility evolution equations are also less accurate from the non-reduced blocking scheme. The log-volatility levels, denoted by μ_a, are closely identified by both procedures; somewhat larger deviations are recorded for the ϕ's and the σ's: however, the correlations of the estimates with the true values are quite high, of the order of 90%. Next, consider the jump parameters, δ and κ. Without providing a graph we mention that the average of the posterior means across the different data sets are slightly closer to the true values for δ (correlation = 95%) than κ (correlation of 92%). In both cases the standard deviations across samples are quite small compared to their respective means. For the jump parameters we don't find meaningful differences across sampling schemes. The performance of both algorithms is relatively less satisfactory for the degrees of freedom parameters of the Student-t distributions. The correlation with the true values is only 82%. This could be due to the large overall dimension of the parameter space combined with a relatively limited sample size used in the estimation.

Next, consider the effect of Prior2 on the posterior estimates, which is reported in the last two rows of Table 4. Both procedures appear to be robust to this change in the prior as the correlations between the true and simulated values are almost unaltered. It is still true, however, that the marginalized sampling scheme does a better job in estimating the factor loadings and the factor

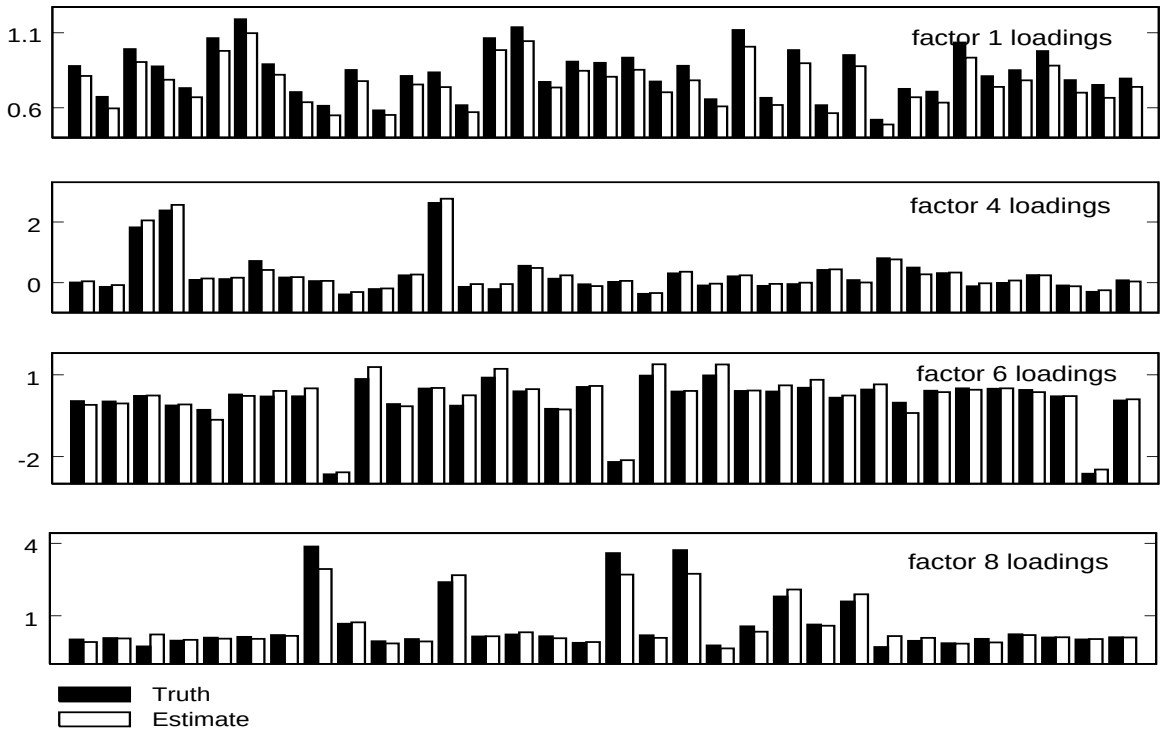

Figure 1: True Values vs. posterior estimates for the factor loadings. Each panel displays the loadings on a different factor (only factor 1, 4, 6 and 8 are reported). The posterior quantities are the average of posterior means across 40 samples with $n = 1250$.

volatility parameters.

Finally, consider the relationship between the true and estimated factors. Figure 2 displays the correlations across samples for the common factors: the estimates for these latent variables are obtained by averaging across the MCMC draws for each sample. We report the summaries for factor 1, 2, 5 and 8. In all cases the latent series are estimated well with correlations with the true values ranging between 70 and 95%. The precision is high for the first factor, decreasing somewhat for the other factors. These experiments show that the suggested estimation procedure yields reliable inferences for both the model parameters and the latent dynamic factors. Relying on the non-marginalized schemes to update the factor loadings leads to significant biases. These biases arise not only in the estimates of the loading parameters but also in those of the factor volatilities.

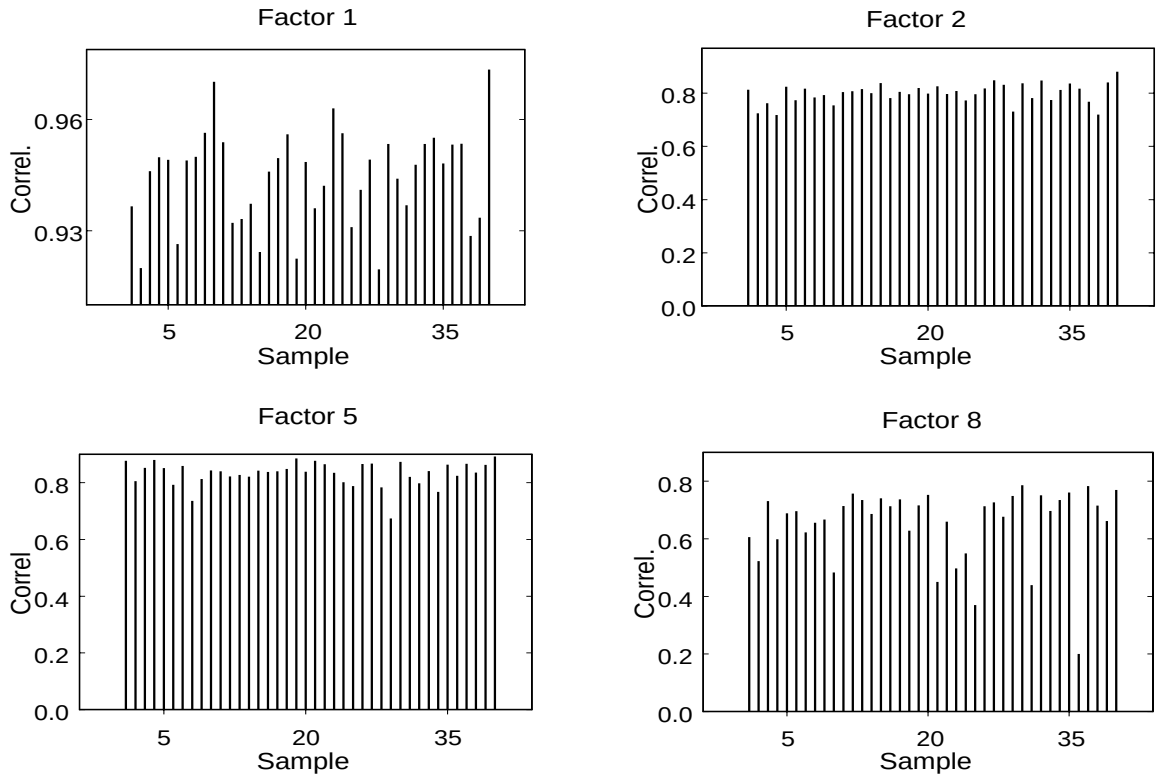

Figure 2: Correlations between true and estimated factors across simulated samples. For each dataset the estimates are obtained by averaging the draws of the MCMC sampler. The results are based on 40 simulated datasets of size 1250 each.

4.4 Performance and Stability of the Marginal Likelihood Method

In this section we utilize simulated data to assess the performance of the marginal likelihood and Bayes factor criterion in identifying the correct model across model types and, within a given model class, the correct number of factors. In the simulation design, datasets are generated from the *MSVt* model with three factors. Each simulated dataset contains thirty series of 2000 observations each. The model parameters in the true model are randomly generated as in section 4.2. We generate a total of 50 data sets from the true model. The *MSVJ*, *MSVt* and *MSVJt* models are then fitted to these data sets, each with 2, 3 and 4 factors. Thus, nine models are each estimated fifty times under the prior distributions and hyperparameters reported in section 4.1. The marginal likelihood of each model in each simulated data set is calculated from $G = 10000$ MCMC iterations (beyond a burn-in of 1000 iterations) followed by reduced runs of 10000 iterations. Finally, the two parameters of the particle filter algorithm, namely M and R, are set to 20000 and 200000, respectively.

4.4.1 Stability

First, we investigate the stability of the posterior ordinate estimate. We randomly pick 5 of our 50 simulated datasets and compute estimates of the posterior ordinate for various values of G, the

Data	Simulation Size G			
	$5,000$	$10,000$	$25,000$	$50,000$
D2	329.74	329.73	329.80	329.97
D10	325.40	327.18	327.94	327.99
D30	319.11	323.35	323.87	323.71
D40	318.19	320.86	320.14	320.19
D50	346.97	348.87	348.40	348.92

Table 5: *Natural log-posterior ordinate estimates for different simulation sizes. G denotes the number of reduced MCMC draws. Results are based on* 5 *simulated datasets.*

number of reduced-run iterations. In particular, we let G take the values 5000, 10000, 20000 and 50000. The posterior ordinates from each of the five data sets are then averaged. Although the data are generated from the *MSVt* model, we do this calculation with the *MSVJt* model which is a larger model. The estimated values are shown in Table 5.

The table values indicate that the estimates converge when the number of reduced runs is at least 10000.

4.4.2 Model Comparison

We conclude our experiments by examining the performance of the marginal likelihood criterion in selecting the true model. This is done via a sampling experiment in which we count the frequency with which each possible K-factor model ($K = 2, 3, 4$) is picked over the other models, based on the estimated marginal likelihoods. Table 6 reports the relevant results: the true model, *MSVt* with 3 factors, is compared with every other specification we estimate.

According to the Jeffreys' scale, the evidence in favor of true model is always decisive versus the basic *MSV* model as well as versus *MSVt 2f* and it is at least substantial against *MSVt 4f* in 84% of the cases. When compared to the more highly parametrized *MSVJt* model, *MSVt 3f* is still selected as the best model 100% of the times against *MSVJt 2f*, 98% of the times against *MSVJt 3f* and 88% of the times against *MSVJt 4f*. In all these cases the support in favor of the true model is strong or decisive. In summary, the simulation evidence provides a convincing validation of the

	True Model: $MSVt3f$				
	$1-3.2$	$3.2-10$	$10-100$	>100	$Total>10$
$MSVt3f/MSV2f$	0	0	0	100	100
$MSVt3f/MSV3f$	0	0	0	100	100
$MSVt3f/MSV4f$	0	0	2	98	100
$MSVt3f/MSVt2f$	0	0	0	100	100
$MSVt3f/MSVt4f$	0	4	20	60	84
$MSVt4f/MSVt2f$	0	0	4	96	100
$MSVt3f/MSVJt2f$	0	0	0	100	100
$MSVt3f/MSVJt3f$	0	0	2	94	96
$MSVt3f/MSVJt4f$	0	0	10	78	88

Table 6: *Frequency distribution (percentage) of Bayes factors across 50 simulated replications. The ranges for Bayes factor values correspond to the Jeffreys' scale.*

Bayes factor criterion along two dimensions: the identification of the correct number of common factors and in the selection of the appropriate model specification.

5 CONCLUSION

In this paper we have a proposed and analyzed a new multivariate model with time varying correlations. The model contains several features (for example fat tails and jump components) that are particularly relevant in the modeling of financial time series. Our fitting approach, which relies on tuned MCMC methods, was shown to be scalable in terms of both the multivariate dimension and the number of factors. This leads us to believe that this is first viable estimation approach for high-dimensional stochastic volatility models. In the paper we also provide a method for finding the marginal likelihood of the model. This criterion is useful in comparing the general model with various special cases, say defined by the presence or absence of jumps and fat-tails, and in identifying the correct number of pervasive factors. A detailed simulation study shows that our estimate of the marginal likelihood is both accurate and reliable.

6 ACKNOWLEDGMENTS

We thank the journal's two reviewers for their comments on previous drafts. We also thank CINECA and Brick Network for providing computing facilities.

References

Aguilar, O. and M. West (2000). Bayesian dynamic factor models and variance matrix discounting for portfolio allocation. *Journal of Business and Economic Statistics 18*, 338–357.

Andersen, T. G., L. Benzoni, and J. Lund (2002). An empirical investigation of continuous-time equity return models. *Journal of Finance 57*, 1239–1284.

Bollerslev, T. (1990). Modelling the coherence in short-run nominal exchange rates: a multivariate generalized ARCH approach. *Review of Economics and Statistics 72*, 498–505.

Bollerslev, T., R. F. Engle, and D. B. Nelson (1994). ARCH models. In R. F. Engle and D. McFadden (Eds.), *The Handbook of Econometrics, Volume 4*, pp. 2959–3038. Amsterdam: North-Holland.

Bollerslev, T., R. F. Engle, and J. M. Wooldridge (1988). A capital asset pricing model with time varying covariances. *Journal of Political Economy 96*, 116–131.

Chib, S. (1995). Marginal likelihood from the Gibbs output. *Journal of the American Statistical Association 90*, 1313–21.

Chib, S. (2001). Markov chain Monte Carlo methods: computation and inference. In J. J. Heckman and E. Leamer (Eds.), *Handbook of Econometrics*, Volume 5, pp. 3569–3649. Amsterdam: North-Holland.

Chib, S. and E. Greenberg (1994). Bayes inference for regression models with ARMA(p, q) errors. *Journal of Econometrics 64*, 183–206.

Chib, S. and E. Greenberg (1995). Understanding the Metropolis-Hastings algorithm. *The American Statistican 49*, 327–35.

Chib, S. and I. Jeliazkov (2001). Marginal likelihood from the Metropolis-Hastings output. *Journal of the American Statistical Association 96*, 270–281.

Chib, S., F. Nardari, and N. Shephard (2002). Markov chain Monte Carlo methods for generalized stochastic volatility models. *Journal of Econometrics 108*, 281–316.

de Jong, P. and N. Shephard (1995). The simulation smoother for time series models. *Biometrika 82*, 339–50.

Diebold, F. X. and M. Nerlove (1989). The dynamics of exchange rate volatility: a multivariate latent factor ARCH model. *Journal of Applied Econometrics 4*, 1–21.

Doucet, A., N. de Freitas, and N. Gordon (2001). *Sequential Monte Carlo Methods in Practice*. New York: Springer-Verlag.

Engle, R. F., V. K. Ng, and M. Rothschild (1990). Asset pricing with a factor ARCH covariance structure: empirical estimates for treasury bills. *Journal of Econometrics 45*, 213–238.

Engle, R. F. and K. Sheppard (2001). Theoretical and empirical properties of dynamic conditional correlation multivariate GARCH. Unpublished paper: UCSD.

Geweke, J. (1992). Efficient simulation from the multivariate Normal and Student-t distributions subject to linear constraints. *Computing Science and Statistics: Proceedings of the Twenty-third Symposium*, 571–578.

Geweke, J. F. and G. Zhou (1996). Measuring the pricing error of the arbitrage pricing theory. *Review of Financial Studies 9*, 557–87.

Ghysels, E., A. C. Harvey, and E. Renault (1996). Stochastic volatility. In C. R. Rao and G. S. Maddala (Eds.), *Statistical Methods in Finance*, pp. 119–191. Amsterdam: North-Holland.

Harvey, A. C., E. Ruiz, and N. Shephard (1994). Multivariate stochastic variance models. *Review of Economic Studies 61*, 247–64.

Jacquier, E., N. G. Polson, and P. E. Rossi (1995). Models and prior distributions for multivariate stochastic volatility. Unpublished paper: GSB, University of Chicago.

Kim, S., N. Shephard, and S. Chib (1998). Stochastic volatility: likelihood inference and comparison with ARCH models. *Review of Economic Studies 65*, 361–393.

King, M., E. Sentana, and S. Wadhwani (1994). Volatility and links between national stock markets. *Econometrica 62*, 901–933.

Pitt, M. K. (2001). Smooth particle filters for likelihood maximisation. Unpublished paper: Department of Economics, Warwick University.

Pitt, M. K. and N. Shephard (1999a). Filtering via simulation: auxiliary particle filter. *Journal of the American Statistical Association 94*, 590–599.

Pitt, M. K. and N. Shephard (1999b). Time varying covariances: a factor stochastic volatility approach (with discussion). In J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith (Eds.), *Bayesian Statistics 6*, pp. 547–570. Oxford: Oxford University Press.