

TI 2014-118/III  
Tinbergen Institute Discussion Paper



# Joint Bayesian Analysis of Parameters and States in Nonlinear, Non-Gaussian State Space Models

*István Barra\**

*Lennart Hoogerheide*

*Siem Jan Koopman*

*André Lucas*

*\* Duisenberg School of Finance, the Netherlands.*

*Faculty of Economics and Business Administration, VU University Amsterdam, and Tinbergen Institute, the Netherlands.*

Tinbergen Institute is the graduate school and research institute in economics of Erasmus University Rotterdam, the University of Amsterdam and VU University Amsterdam.

More TI discussion papers can be downloaded at <http://www.tinbergen.nl>

Tinbergen Institute has two locations:

Tinbergen Institute Amsterdam  
Gustav Mahlerplein 117  
1082 MS Amsterdam  
The Netherlands  
Tel.: +31(0)20 525 1600

Tinbergen Institute Rotterdam  
Burg. Oudlaan 50  
3062 PA Rotterdam  
The Netherlands  
Tel.: +31(0)10 408 8900  
Fax: +31(0)10 408 9031

Duisenberg school of finance is a collaboration of the Dutch financial sector and universities, with the ambition to support innovative research and offer top quality academic education in core areas of finance.

DSF research papers can be downloaded at: <http://www.dsf.nl/>

Duisenberg school of finance  
Gustav Mahlerplein 117  
1082 MS Amsterdam  
The Netherlands  
Tel.: +31(0)20 525 8579

# Joint Bayesian analysis of parameters and states in nonlinear non-Gaussian state space models \*

*István Barra,<sup>(a)</sup> Lennart Hoogerheide,<sup>(a)</sup>  
Siem Jan Koopman,<sup>(a,b)</sup> André Lucas<sup>(a)</sup>*

<sup>(a)</sup> Vrije Universiteit Amsterdam and Tinbergen Institute, The Netherlands

<sup>(b)</sup> CREATES, Aarhus University, Denmark

March 2016

## Abstract

We propose a new methodology for designing flexible proposal densities for the joint posterior density of parameters and states in a nonlinear, non-Gaussian state space model. We show that a highly efficient Bayesian procedure emerges when these proposal densities are used in an independent Metropolis-Hastings algorithm or in importance sampling. Our method provides a computationally more efficient alternative to several recently proposed algorithms. We present extensive simulation evidence for stochastic intensity and stochastic volatility models based on Ornstein–Uhlenbeck processes. For our empirical study, we analyse the performance of our methods for corporate default panel data and stock index returns.

*JEL classification:* C11, C15, C22, C32, C58.

*Keywords:* Bayesian inference, importance sampling, Monte Carlo estimation, Metropolis-Hastings algorithm, mixture of Student's  $t$ -distributions.

---

\*Corresponding author: S.J. Koopman, Department of Econometrics, Vrije Universiteit Amsterdam, De Boelelaan 1105, 1081 HV Amsterdam, The Netherlands. E-mail [s.j.koopman@vu.nl](mailto:s.j.koopman@vu.nl); Tel. +31205986019; Fax +31205986020. Preliminary versions of this article were presented at ESOBE 2012 in Vienna, ERCIM 2012 in Oviedo, and the 3rd Humboldt-Copenhagen Conference on Financial Econometrics in Berlin in 2013. We are indebted to several participants at these meetings for constructive comments. István Barra and André Lucas thank the Dutch National Science Foundation (NWO) for financial support under VICI grant 453-09-005. Lennart Hoogerheide was supported by VENI grant 451-09-028 of the Dutch National Science Foundation (NWO). Koopman acknowledges support from CREATES, Center for Research in Econometric Analysis of Time Series (DNR78) at Aarhus University, Denmark, and funded by the Danish National Research Foundation. Barra, Koopman, and Lucas thank the European Union 7th Framework Programme (FP7-SSH/2007-2013, grant agreement 320270 - SYRTO) for financial support.

# 1 Introduction

Empirical studies based on nonlinear non-Gaussian state space models are widespread in economics and finance. It is well known that Bayesian parameter estimation in this type of models can be difficult. One of the complications is that the joint posterior density of the parameter and state vectors is typically high-dimensional which makes it cumbersome to develop successful proposal distributions and Monte Carlo algorithms. It is standard practice to overcome this difficulty by disentangling the target density into lower dimensional densities and develop proposal densities for each of them. However, this approach leads to other problems. Although the curse of dimensionality may be resolved to some extent, it is rather demanding to design a proposal density on a case by case basis for each lower dimensional target density. Furthermore, these separately defined proposal densities may not adequately characterize the properties of the joint posterior density, possibly resulting in unsatisfactory computational performance of the method and possibly leading to biased estimates of posterior moments and marginal likelihoods.

The aim of our study is to develop flexible proposal distributions for the joint posterior distribution of the parameters and states in nonlinear non-Gaussian state space models. Our proposed Extended Mixture of  $t$  by Importance Sampling weighted Expectation Maximization (EMitISEM) method extends the *Mixture of  $t$  by Importance Sampling weighted Expectation Maximization* (MitISEM) method of Hoogerheide et al. (2012) for the case when the likelihood is not available in closed form. We differentiate two categories of nonlinear non-Gaussian state space models based on the transition density of the states. In case of a linear Gaussian state equation we propose a joint candidate for the posterior of the parameters and states. The proposal density in our *EMitISEM* method in this case consists of two components: (i) a mixture of Student's  $t$ -densities that targets the marginal posterior density of the parameters, and (ii) a different approximating density that targets the density of the states given the observations and the parameters. The mixture of Student's  $t$ -densities is constructed by means of an extension of the MitISEM method; see Hoogerheide et al. (2012). The proposal density for the states is then based on a given set of parameters. We can take any reasonable approximating density for the states including those developed by Shephard and Pitt (1997), Durbin and Koopman (1997), Richard and Zhang (2007), Koopman et al. (2015) and McCausland (2012). We can use these proposal densities in an independent MH algorithm or in an importance sampling procedure to estimate the marginal likelihood and parameters. The resulting procedure can be fully automated without requiring user intervention. If the transition density is not Gaussian, we propose to replace the likelihood used in the MitISEM method by an unbiased estimator of the likelihood in which the states are integrated out using a particle filter.

We argue and show that our approach is computationally efficient and robust and can

be regarded as an effective alternative to existing Markov chain Monte Carlo (MCMC) methods. Our method provides at least two advantages. First, the methodology can be fully automated. There is no need for case by case fine tuning of the algorithm whenever a different model specification with a possibly different observation density is considered. Second, the necessary computations can be implemented in a parallel manner. This implies that we can use state-of-the-art computer technology based on graphics cards to further reduce the computing time of our method. In addition we also show that in case of the nonlinear non-Gaussian state space model with non-Gaussian transition equation, the performance of our method is comparable to the performance of some of the recently developed sequential importance sampling procedures.

Our work relates to two strands in the literature. First, we contribute to the recent literature on Bayesian estimation for nonlinear non-Gaussian state space models by jointly sampling parameters and state paths. McCausland (2012) suggests a proposal density based on a higher order approximation of the states given the parameter vector. Although this sampler appears to be efficient, it relies on the assumption that the state vector is univariate. Chan and Strachan (2012) propose a method that overcomes this restriction. Their proposal density for the state vectors, however, is derived from a local approximation of the smoothed density, which can lead to reduced performance in higher dimensional problems.

Second, our results relate to the literature on the Bayesian estimation of nonlinear non-Gaussian state space models using particle filters. Andrieu et al. (2010) develop a collection of Particle Markov Chain Monte Carlo (PMCMC) methods for parameter estimation. As argued by Flury and Shephard (2011), the key idea of several PMCMC methods is that the unknown true likelihood can be replaced by an unbiased estimator of the likelihood within a Metropolis-Hastings iteration. Although PMCMC methods provide a general solution to parameter and state estimation in nonlinear non-Gaussian state space models, they require the application of a particle filter for each iteration, see for example Doucet et al. (2015). To overcome this computational burden, Lindsten and Schön (2012) propose a modified version of the particle Gibbs sampler. For the same motivation, Pitt et al. (2012) develop an adaptive version of the particle independent Metropolis-Hastings algorithm with partially adapted auxiliary particle filters. In an extensive simulation study we show that our EMitISEM method is a viable alternative to other recently developed methods including the Adaptive Independent Metropolis-Hastings method of Pitt et al. (2012) and the particle filter MCMC methods of Andrieu et al. (2010). We compare the methods in detail for two cases: the stochastic intensity model and the Gamma Ornstein–Uhlenbeck (OU) stochastic volatility model. For these cases, we provide evidence that our method provides posterior draws and estimates of posterior moments in a computationally more efficient manner than the state-of-the-art alternatives that we consider. We conclude that

our proposed method may be a more efficient and more robust approach than particle filters in empirically relevant cases.

The remainder of this article is organized as follows. In Section 2 we introduce the new methodology. In Section 3 we demonstrate the performance of the methodology against state-of-the-art alternatives in a Monte Carlo study designed for the parameter and state estimation in a stochastic intensity model and a Gamma OU stochastic volatility model. In Section 4, we empirically illustrate the methods by considering a large panel data set of U.S. corporate defaults and estimating a Gamma OU stochastic volatility model on major stock indices. Section 5 concludes.

## 2 The EMitISEM method for state space models

### 2.1 EMitISEM in case of linear Gaussian state equations

For a time series of observations  $y_1, \dots, y_T$ , we define the nonlinear non-Gaussian state space model by the observation density and the state equation

$$y_t \sim p_y(y_t|x_t; \theta), \quad x_t = c_t + Z_t\alpha_t, \quad (1)$$

$$\alpha_{t+1} = d_t + T_t\alpha_t + \eta_t, \quad \eta_t \sim \text{NID}(0, Q_t), \quad (2)$$

where  $p_y$  is the observation density,  $x_t$  is the latent dynamic signal,  $\theta$  is the parameter vector,  $\alpha_t$  is the state vector, for  $t = 1, \dots, T$ . Moreover the  $\eta_t$  disturbance vectors are normally distributed and independent for  $t = 1, \dots, T$ . We discuss generalizations in Section 2.3. The signal  $x_t$  is a linear function of the state vector  $\alpha_t$ , with scalar intercept  $c_t$  and loading vector  $Z_t$  both possibly depending in a deterministic way on time and on the parameter vector  $\theta$ , i.e.  $c_t = c(t; \theta)$  and  $Z_t = Z(t; \theta)$ . The state vector  $\alpha_t$  evolves as a linear Gaussian dynamic process given by (2) where the intercept vector  $d_t = d(t; \theta)$ , transition matrix  $T_t = T(t; \theta)$  and variance matrix  $Q_t = Q(t; \theta)$  are deterministic functions of  $t$  and  $\theta$ . We assume that all vectors and matrices have appropriate dimensions. Bayesian inference for the model given by equations (1) and (2) involves the estimation of the properties of interest of the posterior density  $p(\theta|y)$  of the parameter vector  $\theta$  and the smoothed density of the signal  $p(x|y)$ , where  $y = (y_1, \dots, y_T)'$  and  $x = (x_1, \dots, x_T)'$ .

In the original MitISEM procedure, Hoogerheide et al. (2012) propose to approximate the posterior density  $p(\theta|y)$  (of which only a kernel is required) by considering the Student's  $t$  mixture  $q_{\zeta'}(\theta|y)$  where  $\zeta'$  includes mode vectors, scale matrices, degrees of freedom and mixing weights for the Student's  $t$ -distributions in the candidate mixture  $q_{\zeta'}(\theta|y)$ , and minimizing the Kullback and Leibler (1951) divergence

$$\int p(\theta|y) \log p(\theta|y) d\theta - \int p(\theta|y) \log q_{\zeta'}(\theta|y) d\theta. \quad (3)$$

Since the first term does not depend on the proposal, an approximation of the Kullback-Leibler divergence can be minimized by maximizing

$$\frac{1}{N} \sum_{j=1}^N \frac{p(\theta^{(j)}|y)}{q_{\zeta}(\theta^{(j)}|y)} \log q_{\zeta'}(\theta^{(j)}|y), \quad (4)$$

where  $\theta^{(j)} \sim q_{\zeta}(\theta|y)$  is a sequence of independent and identically distributed (i.i.d) draws from a previous proposal density, for  $j = 1, \dots, N$ . Unfortunately, in nonlinear non-Gaussian state space models we do not know (a kernel of) the posterior density  $p(\theta^{(j)}|y)$  in closed form, so that we have to modify the original MitISEM method.

The Kullback-Leibler divergence between our target density  $p(x, \theta|y)$  and the joint proposal density  $q_{\zeta'}(x, \theta|y)$  is given by

$$\int p(x, \theta|y) \log p(x, \theta|y) dx d\theta - \int p(x, \theta|y) \log q_{\zeta'}(x, \theta|y) dx d\theta. \quad (5)$$

Minimizing the KL divergence is therefore equivalent to maximizing

$$\int p(x, \theta|y) \log q_{\zeta'}(x, \theta|y) dx d\theta = \int \frac{p(x, \theta|y)}{q_{\zeta}(x, \theta|y)} q_{\zeta}(x, \theta|y) \log q_{\zeta'}(x, \theta|y) dx d\theta, \quad (6)$$

where  $q_{\zeta}(x, \theta|y)$  is a previous proposal density. As our proposal density, we use  $q_{\zeta'}(x, \theta|y) = q(x|\theta, y)q_{\zeta'}(\theta|y)$ , where we obtain  $q(x|\theta, y)$  from an approximation to the smoothed state density. For example, the approximation from the NAIS method proposed by Koopman et al. (2015) appears to be sufficiently accurate. Substituting  $q_{\zeta'}(x, \theta|y) = q(x|\theta, y)q_{\zeta'}(\theta|y)$  into the right-hand side of (6) we get

$$\int \frac{p(x, \theta|y)}{q_{\zeta}(x, \theta|y)} q_{\zeta}(x, \theta|y) \log q(x|\theta, y) dx d\theta + \int \frac{p(x, \theta|y)}{q_{\zeta}(x, \theta|y)} q_{\zeta}(x, \theta|y) \log q_{\zeta'}(\theta|y) dx d\theta. \quad (7)$$

The first term in (7) does not depend on  $\zeta'$ . Hence we maximize the second term in (7), with respect to  $\zeta'$  using the approximation

$$\int \frac{p(x, \theta|y)}{q_{\zeta}(x, \theta|y)} q_{\zeta}(x, \theta|y) \log q_{\zeta'}(\theta|y) dx d\theta \approx \frac{1}{N} \sum_{j=1}^N \frac{p(x^{(j)}, \theta^{(j)}|y)}{q_{\zeta}(x^{(j)}, \theta^{(j)}|y)} \log q_{\zeta'}(\theta^{(j)}|y), \quad (8)$$

where  $(x^{(j)}, \theta^{(j)}) \sim q_{\zeta}(x, \theta|y)$  is an i.i.d. sequence for  $j = 1, \dots, N$ .

If we compare (4) with (8), we see that the difference with the original MitISEM approach is that we replace the weight  $p(\theta^{(j)}|y)/q_{\zeta}(\theta^{(j)}|y)$  by  $p(x^{(j)}, \theta^{(j)}|y) / q_{\zeta}(x^{(j)}, \theta^{(j)}|y)$ . This novel result implies that we can use the MitISEM algorithm as described in Appendix A, with only a slight modification. We note that the new weights

$$w^{(j)} = \frac{p(x^{(j)}, \theta^{(j)}|y)}{q_{\zeta}(x^{(j)}, \theta^{(j)}|y)} \propto \frac{p(y|x^{(j)}, \theta^{(j)})p(x^{(j)}|\theta^{(j)})p(\theta^{(j)})}{q(x^{(j)}|\theta^{(j)}, y)q_{\zeta}(\theta^{(j)}|y)} \quad (9)$$

can be replaced by

$$w^{(j)} \propto q(y|\theta^{(j)}) \frac{p(y|x^{(j)}, \theta^{(j)})p(\theta^{(j)})}{q(y|x^{(j)}, \theta^{(j)})q_{\zeta}(\theta^{(j)}|y)}, \quad (10)$$

where we used the relations

$$q(x^{(j)}|\theta^{(j)}, y) = \frac{q(y|x^{(j)}, \theta^{(j)})q(x^{(j)}|\theta^{(j)})}{q(y|\theta^{(j)})}, \quad (11)$$

and  $p(x^{(j)}|\theta^{(j)}) = q(x^{(j)}|\theta^{(j)})$ , which follow from the fact that we use the same linear Gaussian state equation for the true model and the approximating linear Gaussian state space model upon which our proposal density for the signal is based; see Appendix B for the details on this approximating model and the proposal density for the signal. The formulation in (10) is more convenient than (9), as we do not have to evaluate the density  $q(x^{(j)}|\theta^{(j)}, y)$ . The algorithmic details of the EMitISEM method are presented in Appendix C.

There is a trade-off between the quality of the proposal and the speed of the estimation procedure. When more draws are used, the approximation generally becomes better, but at the cost of an increased computation time. Fortunately, the draws and corresponding weights can be recycled such that we do not require the sampling of new draws when going through the iterations to obtain the mixture components. To be able to recycle previous draws, we need to implement a slight modification when computing the coefficient of variation of the importance weights that correspond to the latest candidate. Given the draws  $(x^{(2,1)}, \theta^{(2,1)}), \dots, (x^{(2,N)}, \theta^{(2,N)})$  from the proposal  $q_{\zeta(2)}(x, \theta|y) = q(x|\theta, y)q_{\zeta(2)}(\theta|y)$  with only one Student's  $t$  component, we can evaluate the coefficient of variation of the weights in iteration  $i > 2$  based on the new proposal  $q_{\zeta(i)}(x, \theta|y)$  by estimating the mean of the weights as

$$\int \frac{p(x, \theta|y)}{q_{\zeta(i)}(x, \theta|y)} q_{\zeta(i)}(x, \theta|y) d\theta dx = \int \frac{p(x, \theta|y)}{q_{\zeta(2)}(x, \theta|y)} q_{\zeta(2)}(x, \theta|y) d\theta dx, \quad (12)$$

and the variance of the weights as

$$\begin{aligned} \int \left( \frac{p(x, \theta|y)}{q_{\zeta(i)}(x, \theta|y)} \right)^2 q_{\zeta(i)}(x, \theta|y) d\theta dx &= \int \frac{p(x, \theta|y)^2}{q_{\zeta(i)}(x, \theta|y)} d\theta dx \\ &= \int \frac{p(x, \theta|y)}{q_{\zeta(i)}(x, \theta|y)} \frac{p(x, \theta|y)}{q_{\zeta(2)}(x, \theta|y)} q_{\zeta(2)}(x, \theta|y) d\theta dx. \end{aligned} \quad (13)$$

Based on these results the mean and variance of the importance weights corresponding to the importance density at iteration  $i > 2$  can then be estimated via

$$\text{mean}^{(i)} = \frac{1}{N} \sum_{j=1}^N \frac{p(x^{(j)}, \theta^{(j)}|y)}{q_{\zeta(2)}(x^{(j)}, \theta^{(j)}|y)} \quad (14)$$

and

$$\text{var}^{(i)} = \frac{1}{N} \sum_{j=1}^N \left[ \frac{p(x^{(j)}, \theta^{(j)}|y)}{q_{\zeta(i)}(x^{(j)}, \theta^{(j)}|y)} \frac{p(x^{(j)}, \theta^{(j)}|y)}{q_{\zeta(2)}(x^{(j)}, \theta^{(j)}|y)} \right] - \left[ \text{mean}^{(i)} \right]^2, \quad (15)$$

respectively, where  $(x^{(j)}, \theta^{(j)}) \sim q_{\zeta(2)}(x, \theta|y)$  is an i.i.d. sequence for  $j = 1, \dots, N$ . We emphasize that  $q_{\zeta(i)}(x^{(j)}, \theta^{(j)}|y)$  in (15) can be easily evaluated via  $q(x|\theta, y)q_{\zeta(i)}(\theta|y)$  as

$q(x|\theta, y)$  is the same as in  $q_{\zeta(2)}(x, \theta|y)$ . This modification of the procedure leads to our modified algorithm, from which we realize a substantial gain in speed. The algorithmic details of our modified algorithm are presented in Appendix C.

## 2.2 EMitISEM for nonlinear or non-Gaussian state equations

In the previous section we have shown how we can use our approach to estimate the parameters and states of a nonlinear non-Gaussian state space model with linear Gaussian state equation. This assumption can be somewhat restrictive. Here we present a brief sketch of how we can estimate the model parameters for models with a nonlinear non-Gaussian state equation, that is

$$y_t \sim p_y(y_t|\alpha_t; \theta), \quad (16)$$

$$\alpha_t \sim p_\alpha(\alpha_t|\alpha_{t-1}; \theta). \quad (17)$$

To obtain a Gaussian approximation of the state can be problematic because the smoothed state density can be very different from the Gaussian density. When the transition density (17) differs from the normal, the approximation is less accurate. However we can still use the fact that using an unbiased estimator of the likelihood  $\hat{p}(y|\theta)$  instead of the true likelihood  $p(y|\theta)$  in (4) yields a consistent estimator of the Kullback-Leibler divergence between  $p(\theta|y)$  and  $q_{\zeta'}(\theta|y)$ . Hence as long as we can compute  $\hat{p}(y|\theta)$ , we can use the original MitISEM approach but with the replacement of the likelihood by its unbiased estimator in the calculation of the weight.

$$\hat{w}^{(j)} = \frac{\hat{p}(\theta^{(j)}|y)}{q_{\zeta}(\theta^{(j)}|y)}. \quad (18)$$

It is fortunate that an unbiased estimator of the likelihood  $\hat{p}(y|\theta)$  can be calculated by a particle filter applied to the model given by equations (16) and (17). The algorithmic details are presented in Appendix C.

## 2.3 Discussion and relation to other methods

In this section we discuss the possible computational gains from our procedure and relate it to previously proposed methods.

### 2.3.1 Gaussian transition density

The proposal density constructed via EMitISEM can be used in importance sampling or in an independent Metropolis-Hastings procedure. Using the EMitISEM proposal in an independent Metropolis-Hastings iteration and the Adaptive Independent Metropolis-Hastings method proposed by Pitt et al. (2012) are similar in spirit. The independent Metropolis-Hastings algorithm is the core of both algorithms. However, there are several

clear differences between the two estimation routines. First, our method is not adaptive, which means that it is easier to parallelize as the proposal density is constant throughout the MCMC phase. Candidate draws are fully independent both within each training phase and within the MCMC phase. Second, in the Gaussian transition density case we sample one state path at each iteration; we do not need to integrate out the state. We also emphasize that our method in this case directly provides the smoothed state estimates; we do not require additional algorithms for this task. Finally, we use a mixture of Student's  $t$ -distributions instead of a mixture of normals to approximate the posterior distribution of the parameters.

Since the most intensive part of the computations in our algorithm is the generation of the signal paths  $x$  conditional on the parameters and observations, it is interesting to compare the number of signal draws that are required in our algorithm and in the PMCMC based methods. Our modified EMitISEM procedure requires  $3 \times N + I$  draws where  $N$  is the size of the training sample and  $I$  is the number of iterations in the MCMC phase. In contrast, the PMCMC methods require  $I \times S$  signal draws where  $S$  is the number of draws used to estimate the likelihood. In the two examples in our simulation and empirical studies, we use  $I = 25,000$ ,  $N = 10,000$  and  $S = 50$ . It implies that PMCMC requires around 20 times more signal paths than our modified EMitISEM method, whereas our modified EMitISEM method yields a better accuracy in less computation time in these two examples.

The choice of the mixture of Student's  $t$ -densities as a proposal for the posterior of the parameters has several theoretical and practical advantages over other choices. First, under certain regularity conditions any density can be approximated by a mixture of Student's  $t$ -densities if we use a sufficient number of mixture components as shown by Zeevi and Meir (1997). Second, sampling from a mixture of Student's  $t$ -densities is fast. Third, the fat tails make the Student's  $t$ -distribution (with small enough degrees of freedom) a robust importance sampler. We are less prone to importance weights with an infinite variance, so posterior estimates are more reliable and more efficient. Finally, the construction of Student's  $t$ -distributions in the mixture and the mixing weights can be carried out efficiently by means of the extension of the MitISEM procedure.

The exposition above concentrated on the case of a univariate signal. However, the method can be extended when the observation and signal are vectors by using the approach of Koopman et al. (2015). This extended version of the NAIS method is able to treat the signal vector via the use of quasi-random numbers for the numerical evaluation of the variance of the log weights and subsequently for its minimization.

A promising feature of EMitISEM is that the evaluation of marginal likelihoods can take place in a straightforward manner via importance sampling. Given the proposal

density obtained in the training phase we can approximate the marginal likelihood by

$$p(y) = \int \frac{p(y|x, \theta)p(x|\theta)p(\theta)}{q_{\zeta}(x, \theta|y)} q_{\zeta}(x, \theta|y) d\theta \approx \frac{1}{N} \sum_{j=1}^N w^{(j)}, \quad (19)$$

where  $w^{(j)}$  is defined as the right-hand side of (10), and  $N$  is the number of draws used to evaluate the marginal likelihood.

We are faced with a possible limitation of our method when the time dimension  $T$  increases. In this case the variance of the importance weights also increases and the Monte Carlo approximation (8) may become less reliable. However, we point out that our simulated and empirical data sets have a considerable size, consisting of 1,260 trading days (for the estimation of a stochastic volatility model) or 40 years of defaults (for the estimation of a stochastic intensity model). For both datasets the EMitISEM works smoothly and without particular difficulties.

### 2.3.2 Non-Gaussian transition density

Our EMitISEM proposal density can be used as a proposal in an independent particle Metropolis-Hastings iteration. This method is conceptually close to the Density-Tempered Marginalized Sequential Monte Carlo (DTMSMC) sampler proposed by Duan and Fülöp (2015). In a similar set-up, they suggest to integrate out the states using a particle filter. Their move step utilizes an independent particle Metropolis-Hastings step. However, they approximate the posterior density of the parameters by a Sequential Monte Carlo sampler, which forms a bridge between the prior and the posterior density. Our method instead takes advantage of the fact that we can estimate the likelihood, such that we have a reasonable initial candidate for the posterior of the parameters early on in the procedure.

## 3 Simulation study

### 3.1 EMitISEM proposal for Independent Metropolis-Hastings

We base our analysis on an independent Metropolis-Hastings sampler; see Metropolis et al. (1953) and Hastings (1970) for the original contributions. We draw from the joint posterior density of the parameters and states  $p(x, \theta|y)$ . Our procedure consists of two phases: the training phase and the Markov chain Monte Carlo (MCMC) phase.

In the training phase we use EMitISEM for the construction of a proposal density that approximates the joint posterior  $p(x, \theta|y)$ . We construct the approximation from proposal densities  $q(x|\theta, y)$  and  $q_{\zeta'}(\theta|y)$ , where  $q(x|\theta, y)$  is the conditional proposal density of  $x$  given  $\theta$ , and where  $q_{\zeta'}(\theta|y)$  is the marginal proposal density for  $\theta$  which is obtained using EMitISEM. We take  $q_{\zeta'}(\theta|y)$  as a mixture of Student's  $t$ -densities and use it as an approximation of  $p(\theta|y)$ , We take  $q(x|\theta, y)$  as a Gaussian density from the numerically

accelerated importance sampling (NAIS) method of Koopman et al. (2015). NAIS is numerically more efficient than alternative approximations such as the ones proposed by Richard and Zhang (2007), Shephard and Pitt (1997), or Durbin and Koopman (1997).

In the Markov chain Monte Carlo phase we use the candidate as the proposal density in an independent Metropolis-Hastings algorithm to draw from the joint posterior density  $p(x, \theta|y)$ . We sample the joint candidate draws  $(\theta^{(j)}, x^{(j)})$  by first sampling

$$\theta^{(j)} \sim q_{\zeta'}(\theta|y), \quad (20)$$

and then, conditioning on  $\theta^{(j)}$ , sampling

$$x^{(j)} \sim q(x|\theta^{(j)}, y). \quad (21)$$

Let  $(\theta^{(i-1)}, x^{(i-1)})$  and  $(\theta^+, x^+)$  denote the previous accepted draw of the Markov chain and the new candidate draw, respectively. We set  $(\theta^{(i)}, x^{(i)}) = (\theta^+, x^+)$  with probability

$$\alpha = \min \left\{ \frac{p(\theta^+, x^+|y) q(x^{(i-1)}|\theta^{(i-1)}, y) q_{\zeta'}(\theta^{(i-1)}|y)}{p(\theta^{(i-1)}, x^{(i-1)}|y) q(x^+|\theta^+, y) q_{\zeta'}(\theta^+|y)}, 1 \right\}, \quad (22)$$

and  $(\theta^{(i)}, x^{(i)}) = (\theta^{(i-1)}, x^{(i-1)})$  otherwise.

We carry out a detailed simulation experiment to demonstrate the performance of our estimation procedure against two alternative procedures. We estimate parameters for a stochastic intensity model using simulated data sets. This model illustrates a new and interesting application of a nonlinear non-Gaussian state space model for which the full conditional density of the parameter vector is not known in closed form. Stochastic intensity models are particularly useful in portfolio credit risk modeling, see Koopman et al. (2008), Duffie et al. (2009) and Azizpour et al. (2010) for interesting illustrations of the problem. The details of the stochastic intensity model are given in Section 3.2.

We compare the performance of our proposed EMitISEM method with state-of-the-art alternatives rather than with some feeble benchmark procedures. In particular, we compare the performance of parameter estimation using our proposed EMitISEM method versus using two competing methods of Pitt et al. (2012). The first competing method is the adaptive random walk Metropolis-Hastings (ARWMH) algorithm and is an extension of the method of Roberts and Rosenthal (2009). Our second benchmark method is the adaptive independent Metropolis-Hastings (AIMH) algorithm, where the proposal is a mixture of normals. It is an extension of the method of Giordani and Kohn (2010). These two recently developed and advanced methods provide fast and efficient solutions to parameter estimation for nonlinear non-Gaussian state space models by taking advantage of the powerful framework provided in Andrieu et al. (2010). To further enhance the numerical efficiency of the benchmark methodologies, we use a modified version of these methods by introducing the numerically accelerated importance sampling (NAIS) method of Koopman et al. (2015) for integrating out the signal vector. For the purpose of likelihood

estimation, the NAIS method is used as an alternative to the partially adapted auxiliary particle filter that was used by Pitt et al. (2012).

We use NAIS as a state sampler for the following three reasons: (i) it provides an approximation to the state smoothing density that minimizes the variance of the log importance weights; (ii) the approximating linear Gaussian state space model can be constructed in a computationally efficient way by taking advantage of standard Kalman filter methods and deterministic integration methods for one-dimensional integrals; (iii) the simulated signal paths can be efficiently computed via the simulation smoothers of de Jong and Shephard (1995) or Durbin and Koopman (2002). In an illustrative stochastic volatility model with normally distributed errors we have found that NAIS yields estimates of the likelihood with lower variance in less computing time than methods based on the particle filter; these findings are discussed in Appendix F. In Appendix B we provide the details of NAIS. For its use in the ARWMH and AIMH methods, we use 50 simulated paths of the signal for likelihood estimation. We notice that EMitISEM requires one simulated signal path from NAIS only at each iteration. Further implementation details of the competitive benchmark methods are discussed in Appendices D and E.

We estimate the parameters for 56 data sets on an 8-core computer. The data sets are generated with parameter values that are close to those estimated from the empirical data sets of Section 4. For each simulated data set, we re-estimate parameters by using the EMitISEM method, its modified version (which we denote by EMitISEM mod.), the ARWMH method and the AIMH method. For the modified EMitISEM method, the candidate draws are recycled after the first MitISEM update in the training phase. After 5,000 burn-in draws we perform 20,000 iterations of the algorithms. We calculate medians and interquartile ranges (over the 56 simulated data sets) of the parameter estimates, acceptance rates, and inefficiencies. In order to assess the quality of the simulation methods we compute the inefficiency factor ( $IF$ ), which is defined as the variance of the parameter estimate divided by the variance in case the sampling scheme would generate independent posterior draws. The  $IF$  statistic is discussed, amongst others, by Pitt et al. (2012). In our case, we define the inefficiency factor as

$$IF = 1 + 2 \sum_{j=1}^{\max(L,1000)} r_j, \quad (23)$$

where  $r_j$  is the  $j$ -th order sample serial correlation of the 20,000 parameter draws, and where  $L$  is the lowest order  $j$  for which  $r_j$  is not significant.

### 3.2 Stochastic intensity model

For our first simulation experiment, we consider a stylized version of the point processes model with stochastic intensity as used in our second empirical application in Section 4.1.

Koopman et al. (2008) and Duffie et al. (2009) consider the stochastic intensity model for studying the systematic dynamics of U.S. corporate defaults and credit rating migrations.

In this section we only provide the necessary details of the model needed for the simulations. The full model set-up is deferred till Section 4.2. For the simulation, we use the following version of the model. We consider a pool of  $K$  firms and a jump process  $y_k(t)$  for each firm  $k = 1, \dots, K$  with common jump intensity  $\lambda(t)$  as given by

$$\lambda(t) = \exp [\omega + \beta'c(t) + \gamma x(t)], \quad (24)$$

where  $\omega$  is the base log-intensity,  $\beta$  is a vector of regression parameters,  $c(t)$  is a vector of covariates and  $\gamma$  is a scale factor for the unobserved signal  $x(t)$ . The cumulative jump process over all firms is given by

$$y(t) = \sum_{k=1}^K y_k(t). \quad (25)$$

The signal  $x(t)$  is often referred to as an unobserved frailty factor. We follow standard practice and model it as a zero mean Ornstein-Uhlenbeck (OU) process, standardized to have unit variance at  $t = 1$ ,

$$dx(t) = -\rho x(t) dt + \sqrt{2\rho} dW(t), \quad (26)$$

where  $\rho > 0$  is a persistence parameter and  $W(t)$  is a standard Brownian motion. The set of covariates we use in the simulation is the same as in the empirical section, namely the (i) one year difference of the S&P500 index, (ii) term spread between the 10-year and 1-year Treasury Bond (with constant maturity rates), (iii) secondary market rate on 3 month Treasury Bills, and (iv) year-to-year percentage change of US industrial production (final output), all at the monthly frequency over the period from January 1, 1970 to March 4, 2010; compare Duffie et al. (2007), Lando and Nielsen (2010) and Azizpour et al. (2010). The covariates are obtained from the FRED and CRSP databases. We set the parameters to  $\omega = -4.75$ ,  $\beta = (-0.85, 0.01, -0.055, -5.1)$ ,  $\gamma = 1.15$  and  $\rho = 0.12$ , which are close to the empirical estimates from Section 4.

We simulate data for  $K = 3,000$  firms over the period January 1, 1970 to March 4, 2010. As  $K$  is kept fixed, a firm can jump repeatedly over the sample. If a jump is interpreted as default, this implies that the firm is directly re-started after default at the same pre-default intensity. In the empirical application, we depart from this construction and allow for an absorbing default state as well as for firms that enter the sample or leave the sample for other reasons than default.

The simulations are conditional on the four covariates and are sampled by using a discretization of the continuous time processes  $y(t)$ ,  $x(t)$  and  $\lambda(t)$ , where the discretization takes steps of  $1/32$  part of a day, i.e., 45 minutes. Over each 45 minutes slot, we use a Bernoulli approximation to generate defaults. We generate 56 data sets in this way. For

the estimation, we use (weakly informative) uniform priors on relatively wide intervals:  $[-8, -2]$  for  $\omega$ ,  $[0.01, 3]$  for  $\gamma$ ,  $[0.01, 1]$  for  $\rho$ , and  $[-20, 20]$  for each of the elements of  $\beta$ .

We consider the  $i$ th event time  $t_i$  and define the indicator variable  $D_{ki}$  to be one,  $D_{ki} = 1$ , if firm  $k$  jumps to default at the  $i$ th event time  $t_i$ , and zero otherwise. The number of jumps at event time  $t_i$  over all firms is given by  $D_i = \sum_{k=1}^K D_{ki}$ . The discrete time approximation of the jump process  $y(t)$  leads to the following dynamic model in event time,

$$\begin{aligned} p(y_i|x_i, \theta) &= \exp [D_i \log \lambda_i - \lambda_i K \Delta_i], \\ x_i &= e^{-\rho \Delta_i} x_{i-1} + \eta_i, \quad \eta_i \sim N(0, 1 - e^{-2\rho \Delta_i}), \end{aligned} \tag{27}$$

where  $p(y_i|x_i, \theta)$  is the density of  $y_i = y(t_i)$  conditional on signal  $x_i = x(t_i)$  and parameter vector  $\theta$ , with  $\lambda_i = \lambda(t_i)$  and  $\Delta_i = t_i - t_{i-1}$ . Further details of the model are presented at the empirical application in Section 4.2.

Table 1 presents the means and interquartile ranges of the parameter estimates for the 56 simulated data sets. The different simulation methods provide similar results, which suggests that none of the methods provides biased estimates. Table 2 presents the means and interquartile ranges of acceptance rates and inefficiency factors. We find that the ARWMH algorithm is clearly outperformed by the other methods. Moreover, the AIMH algorithm performs generally less favourable compared to EMitISEM. Further, the AIMH method appears to be less robust for certain simulated data sets. The median inefficiencies are higher and also the interquartile ranges of the inefficiencies are larger compared to EMitISEM. A possible explanation is that AIMH uses a mixture of normal distributions, where the thin tails of the normal distribution imply that the AIMH algorithm can sometimes get stuck for a considerable time when a rare candidate draw is simulated from the tails and accepted, after which many consecutive candidate draws are rejected and the same accepted draw is repeated many times. In contrast, the EMitISEM method uses a mixture of Student's  $t$ -densities to approximate the posterior distribution of the parameters. The fat tails of the Student's  $t$ -density prevent that the MH method repeats a draw from one of the tails for a long sequence of iterations.

The performance of the modified versions of EMitISEM are again comparable to the standard version. According to the results in Table 2, the size of the training sample can be reduced to obtain higher efficiency gains.

Table 2 shows that the acceptance rate and inefficiencies of the modified version of EMitISEM are only somewhat worse than those for the standard version; the relative gain in computing time is larger. The same observation can be made when comparing the acceptance rates, inefficiencies and computing times of the modified EMitISEM methods with 10000 and 5000 parameter draws in the training sample.

The computation times can be further split out (not shown) in the time to compute the proposal density and the time for generating the actual samples. Typically, the time to

construct the proposal density is higher for the EMitISEM. After that, however, the Effective Sample Size (ESS) or put differently the number of hypothetical independent draws that would yield estimation results with the same precision starts to increase relatively slowly for the AIMH and ARWMH methods, whereas the increase is steep for EMitISEM. The difference would be even larger if we would take full advantage of the parallelization possibilities for EMitISEM.

We conclude from the results presented for the stochastic intensity model that using the EMitISEM algorithm to construct a proposal density for independent Metropolis-Hastings iterations offers an efficient alternative to PMCMC. Substantial improvements can be obtained with respect to computational speed, accuracy, numerical efficiency, and robustness.

### 3.3 EMitISEM proposal for Importance Sampling

In this section we present a simulation experiment which assesses the performance of the EMitISEM proposal on a state space model with non-Gaussian transition density. In this second simulation study we have chosen to use the EMitISEM proposal within an importance sampling procedure. We compare its efficiency to competing sequential importance sampling procedures, namely the SMC<sup>2</sup> by Chopin et al. (2013) and the density tempered marginalized sequential Monte Carlo sampler of Duan and Fülöp (2015).

In the experiment we consider a Gamma Ornstein–Uhlenbeck (OU) stochastic volatility model for a simulated data set. The Gamma OU model has been proposed by Barndorff-Nielsen and Shephard (2001) and the inference for these models has been first developed by Barndorff-Nielsen and Shephard (2002). The model can be formulated in state space form with a non-Gaussian state equation. The analysis and estimation are challenging as the transition density is intricate; see Griffin and Steel (2006) and Chopin et al. (2013). The state space representation of the model is given as follows

$$y_n = \mu + \beta \sigma_n^2 + \sigma_n \varepsilon_n,$$

$$k \sim \text{Poi}(\lambda \xi^2 / \omega^2), \quad c_i \sim U(n, n+1), \quad e_i \sim \text{Exp}(\xi / \omega^2), \quad i = 1, \dots, k,$$

$$z_{n+1} = e^{-\lambda} z_n + \sum_{j=1}^k e^{-\lambda(n+1-c_j)} e_j, \quad \sigma_n^2 = \frac{1}{\lambda} \left[ z_n - z_{n+1} + \sum_{j=1}^k e_j \right],$$

where  $\text{Poi}(a)$  is the Poisson distribution with intensity  $a$ ,  $U(a, b)$  is the uniform distribution with support on interval  $(a, b)$  and  $\text{Exp}(a)$  is the exponential distribution with mean  $a^{-1}$ .

We simulate 1000 observations from the Gamma OU model. We use the parameter values  $\mu = 0$ ,  $\beta = 0$ ,  $\xi = 0.5$ ,  $\omega^2 = 0.065$  and  $\lambda = 0.01$ . Subsequently we estimate the parameters 10 times by using the SMC<sup>2</sup> method, the density tempered marginalized sequential Monte Carlo sampler (DTMSMC), and our EMitISEM procedure. In the prior

specification we follow Chopin et al. (2013) and we assume standard normal priors for  $\mu$  and  $\beta$ ,  $\text{Exp}(1/5)$  for  $\xi$  and  $\omega^2$ , and  $\text{Exp}(1)$  for  $\lambda$ .

The EMitISEM procedure uses an auxiliary model to construct the initial proposal. In the auxiliary model we replace the non-Gaussian state equation and model the log variance as a standard AR(1) process which has a normal unconditional distribution. We estimate the parameters by using the method of maximum likelihood. We transform these estimates to estimates of the parameters from the Gamma OU model and obtain their corresponding standard errors using the delta method that requires the inverse negative Hessian matrix, evaluated at the likelihood estimate. We obtain the mean and variance of our initial distribution from these estimates. In the DTMSMC procedure, we use a random walk proposal in the move step with variance proportional to the estimated covariance matrix based on the current particle approximation as described in Duan and Fülöp (2015). Moreover, we adaptively set the tempering using the method described in Moral et al. (2012) and we set the decay rate to 0.9. In the move step of the SMC<sup>2</sup> algorithm we use an independent normal proposal as suggested by Chopin et al. (2013) and we adaptively set the number of state particles. All methods use 2000 particles and 200 state particles in the particle filter runs except the SMC<sup>2</sup> procedure where we start with 100 particles. The SMC<sup>2</sup> does not exceed 200 particles in any run.

Using the same simulated data set, we run every method 10 times and we record the median of the posterior distribution of the parameters and the total time of the estimation. Table 3 summarizes the results of the experiment. The results show that the SMC<sup>2</sup> procedure takes about 55 minutes to finish on average while the estimation takes about 125 and 164 minutes with EMitISEM and DTMSMC, respectively. We compute the standard deviations of the 10 estimates of the posterior median for each parameter. This shows that EMitISEM yields estimates of which the standard deviations are between 4 and 11 times smaller than DTMSMC (in less computation time). EMitISEM gives estimates of which the standard deviations are between 1.2 and 3 times smaller than SMC<sup>2</sup> (in roughly twice the computation time), where the inefficiency factors are roughly comparable across these methods. In this setting, the efficiency of EMitISEM and SMC<sup>2</sup> may be considered comparable whereas both methods are more efficient than DTMSMC.

## 4 Two empirical studies

In this section we illustrate the performance of the EMitISEM proposal in a Metropolis-Hastings procedure as part of two empirical studies. First, we estimate the parameters in a stochastic intensity model using a large panel of U.S. corporate defaults. Second, we consider the Gamma OU stochastic volatility model for time series of daily returns on five well-known stock indices.

## 4.1 Stochastic intensity model

In our first empirical illustration we consider the stochastic intensity model for a large panel data set of U.S. corporate defaults obtained from Moody's. The core of the model is the same as in Section 3.2, with a slight change to account for sample extension and for attrition due to other reasons than default. The dummy variable  $D_{ki}$  is defined as before, with  $D_{ki} = 1$  if firm  $k$  jumps into default at time  $t_i$ , and  $D_{ki} = 0$  otherwise. We introduce the new dummy variables  $R_{ki}$ , with  $R_{ki} = 1$  if firm  $k$  is at risk of defaulting at time  $t_i - \varepsilon$ , for  $\varepsilon > 0$  arbitrarily small, and  $R_{ki} = 0$  otherwise. An event time  $t_i$  occurs when one of the control variates changes its value (e.g., at the end of the month or quarter), when a firm is added to the sample, or when a firm leaves the sample, either due to default or due to other reasons. We denote the default intensity of firm  $k$  at time  $t_i$  as  $\lambda_{ki} = \lambda_k(t_i)$ .

The conditional density of the observations given the complete paths of the covariates  $c_i = c(t_i), i = 1, \dots, T$  and the complete path of the unobserved process  $x_i = x(t_i), i = 1, \dots, T$  is given by

$$p(y|x, \theta) = \prod_{i=1}^T p(y_i|x_i, \theta) = \prod_{i=1}^T \prod_{k=1}^K \exp [D_{ki} \log \lambda_{ki} - R_{ki} \lambda_{ki} \Delta_i], \quad (28)$$

where  $\Delta_i = t_i - t_{i-1}$ , for  $y = (y_1, \dots, y_T)'$  and  $x = (x'_1, \dots, x'_T)'$ . Firm  $k$  only contributes to the likelihood function when it is at risk of defaulting, that is when  $R_{ki} = R_k(t_i) = 1$ . The state equation remains as in Section 3.2, for details, see Koopman et al. (2008).

Our data set contains 1,627 defaults from 12,881 U.S. firms observed daily over the period January 1, 1970 to March 4, 2010. The number of firms in the portfolio increases over time from about 1,000 firms at the beginning of the sample to around 5,000 firms around 2010. Defaults originating from parent-subsidiary relationships are excluded: if there are multiple defaults with the same parental ID, we only keep the oldest firm as this is likely to be the parent firm. Event times and durations are measured in business days. As in Koopman et al. (2008), we winsorize the number of defaults per day to one to account for outliers and other data irregularities.

The top panel of Figure 1 shows the number of defaults per day before winsorizing. The concentration of the vertical lines clearly presents evidence of default clustering over time. In particular, we find high levels of defaults during 1989-1991 at the end of the savings and loan crisis and during the subsequent recession, in 2001 after the burst of the dot-com bubble, and after the 2008 financial crisis and during the subsequent recession. The increasing numbers of defaults in the last two decades do not immediately imply an increase in the frailty process  $x(t)$ , because also the number of firms increased substantially in the later part of the sample. Furthermore, the covariates may also partly explain the movements in the data. The covariates are S&P500 returns, Treasury Bond spreads, 3 month Treasury Bill yields, and yearly changes in U.S. industrial production, see also Section 3.2.

The parameter estimates and the inefficiency factors for the chains of the EMitISEM draws are presented in Table 4. The 90% credible intervals of the coefficients  $\beta_2$  (for the term spread) and  $\beta_3$  (for the U.S. Treasury Bill rate) include zero, which indicates that these parameters are not significantly different from zero. The signs of the parameters are consistent with what we expect. Both lower returns on the S&P500 index and lower percentage changes in industrial production imply a higher default intensity. The mean reversion parameter  $\rho$  is estimated as 0.12. At the yearly frequency, this implies an autoregressive coefficient  $e^{-\rho} \approx 0.9$ , such that the frailty process has a high persistence. The inefficiency factors have values around 3 and the acceptance rate is 71.04%. The bottom panel of Figure 1 displays the smoothed estimate of the frailty process together with the 90% credible interval. We emphasize that the credible interval includes all uncertainties due to the observation noise, the randomness of the frailty process, and the uncertainty about the parameter vector  $\theta$ . This contrasts with the confidence bands around the estimated frailty process which are shown in most of the literature as part of a classical analysis where the parameter uncertainty is typically ignored. The estimated frailty process represents the credit cycle dynamics in excess of the dynamics caused by the observable controls in  $c(t_i)$ . We clearly recognize the local peaks of the 1991 recession, the burst of the dot-com bubble, and the aftermath of the financial crisis of 2008. Apparently, the control variables do not explain all of the clustering behaviour of the defaults.

## 4.2 Gamma OU stochastic volatility model

For our second empirical application we estimate Gamma OU stochastic volatility models on daily returns on major stock indices. We downloaded the daily returns of S&P500 (SPX), Dow Jones Industrial Average (DJIA), FTSE 100 (FTSE), Deutsche Boerse AG German Stock Index (DAX) and Amsterdam Exchange Index (AEX) indices from 2nd January 2012 till 5th March 2015 from the Oxford-Man Institute Realized Library. Table 5 presents the descriptive statistics of the data. All five series exhibit similar features, with modest excess kurtosis and negative skewness.

We estimate the Gamma OU models using the EMitISEM procedure with 2000 parameter draws and 2000 state particles in each particle filter run. The estimation results are given in Table 6. The estimates of  $\beta$ , the loading on the variance in the observation equation, are negative for all indices, which suggests negative skewness of the return distribution. In three cases out of five, however, zero is also in the 95% credible interval. The persistence parameter  $\lambda$  is estimated with large uncertainty, but the estimation results are indicative of high persistence especially for DAX and AEX.

## 5 Conclusion

We have introduced a new Extended Mixture of  $t$  by Importance Sampling weighted Expectation Maximization (EMitISEM) algorithm to construct a flexible proposal for the joint posterior density of the parameters and the states in nonlinear non-Gaussian state space models. We conclude that using the EMitISEM proposal in an independent Metropolis-Hastings procedure is a computationally efficient alternative to competing MCMC methods such as the adaptive particle independent Metropolis-Hastings method. In a Monte Carlo study, we have shown that our method outperforms competing methods in terms of efficiency and computation time. Moreover we have showed that using the EMitISEM proposal in an importance sampling procedure leads to better performance compared to the density tempered marginalized sequential Monte Carlo sampler in our experiment, and that the efficiency is comparable with the SMC<sup>2</sup> procedure. An interesting extension for future research is to explore the sampling of state paths with the backward smoothing algorithm described in Lindsten and Schön (2012).

## References

- Andrieu, C., Doucet, A., Holenstein, R., 2010. Particle Markov chain Monte Carlo methods. *Journal of Royal Statistical Society Series B* 72, 269–342.
- Azizpour, S., Giesecke, K., Schwenkler, G., 2010. Exploring the sources of default clustering, Working Paper, Stanford University.
- Barndorff-Nielsen, O. E., Shephard, N., 2001. Non-Gaussian Ornstein-Uhlenbeck-based models and some of their uses in financial economics. *Journal of Royal Statistical Society Series B* 63, 167–241.
- Barndorff-Nielsen, O. E., Shephard, N., 2002. Econometrics analysis of realized volatility and its use in estimating stochastic volatility models. *Journal of Royal Statistical Society Series B* 64, 253–280.
- Chan, J., Strachan, R., 2012. Estimation in non-linear non-Gaussian state space models with precision-based methods, CAMA Working Paper.
- Chopin, N., Jacob, P. E., Papaspiliopoulos, O., 2013. SMC2: an efficient algorithm for sequential analysis of state space models. *Journal of the Royal Statistical Society: Series B* 75, 397–426.
- de Jong, P., Shephard, N., 1995. The simulation smoother for time series models. *Biometrika* 82, 339–350.

- Doucet, A., de Freitas, N., Gordon, N. (Eds.), 2001. *Sequential Monte Carlo Methods in Practice*. Springer Verlag, New York.
- Doucet, A., Pitt, M., Deligiannidis, G., Kohn, R., 2015. Efficient implementation of Markov chain Monte Carlo when using an unbiased likelihood estimator. *Biometrika* 102, 295–313, Working Paper.
- Duan, J.-C., Fülöp, A., 2015. Density-Tempered Marginalized Sequential Monte Carlo Sampler. *Journal of Business and Economic Statistics* 33, 192–202, Research Paper.
- Duffie, D., Eckner, A., Horel, G., Saita, L., 2009. Frailty correlated default. *Journal of Finance* 64, 2089–2123.
- Duffie, D., Saita, L., Wang, K., 2007. Multi-period corporate default prediction with stochastic covariates. *Journal of Financial Economics* 83 (3), 635–665.
- Durbin, J., Koopman, S. J., 1997. Monte Carlo maximum likelihood estimation for non-Gaussian state space models. *Biometrika* 84, 669–684.
- Durbin, J., Koopman, S. J., 2002. A simple and efficient simulation smoother for state space time series models. *Biometrika* 89, 603–616.
- Flury, T., Shephard, N., 2011. Bayesian inference based only on simulated likelihood. *Econometric Theory* 27, 933–956.
- Giordani, P., Kohn, R., 2010. Adaptive independent Metropolis-Hastings by fast estimation of mixtures of normals. *Journal of Computational and Graphical Statistics* 19, 243–259.
- Gordon, N., Salmond, D. J., Smith, A. F. M., 1993. A novel approach to non-linear and non-Gaussian Bayesian state estimation. *IEE-Proceedings F* 140, 107–113.
- Griffin, J., Steel, M., 2006. Inference with non-Gaussian Ornstein-Uhlenbeck processes for stochastic volatility. *Journal of Econometrics* 134, 605–644.
- Hastings, W. K., 1970. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* 57, 97–109.
- Hoogerheide, L. F., Opschoor, A., van Dijk, H. K., 2012. A class of adaptive importance sampling weighted EM algorithms for efficient and robust posterior and predictive simulation. *Journal of Econometrics* 171 (2), 101–120.
- Koopman, S. J., Lucas, A., Monteiro, A., 2008. The multi-state latent factor intensity model for credit rating transition. *Journal of Econometrics* 142, 399–424.

- Koopman, S. J., Lucas, A., Scharth, M., 2015. Numerically accelerated importance sampling for nonlinear non-Gaussian state space models. *Journal of Business & Economic Statistics* 33, 114–127.
- Kullback, S., Leibler, R. A., 1951. On information and sufficiency. *The Annals of Mathematical Statistics* 22, 79–86.
- Lando, D., Nielsen, M. S., 2010. Correlation in corporate defaults: contagion or conditional independence? *Journal of Financial Intermediation* 19, 355–372.
- Lindsten, F., Schön, T. B., 2012. On the use of backward simulation in particle Markov chain Monte Carlo methods, Research Paper.
- McCausland, W. J., 2012. The HESSIAN method: Highly efficient simulation smoothing, in a nutshell. *Journal of Econometrics* 168 (2), 189–206.
- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., Teller, E., 1953. Equations of state calculations by fast computing machines. *Journal of Chemical Physics* 21, 1087–1092.
- Moral, P. D., Doucet, A., Jasra, A., 2012. An adaptive sequential monte carlo method for approximate bayesian computation. *Statistics and Computing* 22, 1009–1020.
- Omori, Y., Chib, S., Shephard, N., Nakajima, J., 2007. Stochastic volatility with leverage: Fast and efficient likelihood inference. *Journal of Econometrics* 140, 425–449.
- Pitt, M., 2002. Smoothed particle filters for likelihood evaluation and maximization, Warwick Economic Research Papers.
- Pitt, M. K., Shephard, N., 1999. Filtering via simulation: auxiliary particle filter. *Journal of the American Statistical Association* 94, 590–599.
- Pitt, M. K., Silva, R. S., Giordani, P., Kohn, R., 2012. On some properties of Markov chain Monte Carlo simulation methods based on the particle filter. *Journal of Econometrics* 171 (2), 134–151.
- Richard, J., Zhang, W., 2007. Efficient high-dimensional importance sampling. *Journal of Econometrics* 141, 1385–1411.
- Roberts, G. O., Rosenthal, J. S., 2009. Examples of adaptive MCMC. *Journal of Computational and Graphical Statistics* 18, 349–367.
- Shephard, N., Pitt, M. K., 1997. Likelihood analysis of non-Gaussian measurement time series. *Biometrika* 84, 653–667.
- Zeevi, A. J., Meir, R., 1997. Density estimation through convex combination of densities; approximation and estimation bounds. *Neural Networks* 10, 99–106.

**Table 1:** Stochastic intensity model: parameter estimates for different Metropolis-Hastings algorithms. The table presents the medians and interquartile ranges (within parentheses) over 56 simulated data sets. The estimates are based on 20,000 draws after a burn-in sample of 5,000 observations. We use 50 simulated draws to evaluate the likelihood in the ARWMH and AIMH algorithms.

	Estimates						
	$\omega$	$\beta_1$	$\beta_2$	$\beta_3$	$\beta_4$	$\gamma$	$\rho$
True values	-4.75	-0.85	0.01	-0.055	-5.1	1.15	0.12
Algorithm:							
ARWMH	-4.559 (1.212)	-0.909 (0.498)	0.021 (0.202)	-0.044 (0.107)	-4.611 (2.280)	1.307 (0.400)	0.154 (0.111)
AIMH	-4.621 (1.247)	-0.895 (0.501)	0.026 (0.206)	-0.043 (0.103)	-4.546 (2.330)	1.256 (0.459)	0.149 (0.111)
EMitISEM 10k	-4.624 (1.183)	-0.896 (0.483)	0.025 (0.209)	-0.044 (0.108)	-4.591 (2.182)	1.286 (0.382)	0.151 (0.106)
EMitISEM mod.10k	-4.580 (1.220)	-0.899 (0.482)	0.024 (0.211)	-0.044 (0.105)	-4.562 (2.219)	1.283 (0.389)	0.150 (0.107)
EMitISEM mod. 5k	-4.599 (1.189)	-0.897 (0.477)	0.023 (0.206)	-0.044 (0.106)	-4.559 (2.115)	1.302 (0.394)	0.145 (0.106)

**Table 2:** Stochastic intensity model: We compare the Adaptive Random Walk Metropolis-Hastings method (ARWMH), the Adaptive Independent Metropolis-Hastings method (AIMH), the EMitISEM with 10,000 parameter draws in the training sample (EMitISEM 10k), the modified EMitISEM with 10,000 parameter draws in the training sample (EMitISEM mod. 10k) and the modified EMitISEM with 5,000 parameter draws in the training sample (EMitISEM mod. 5k) in terms of time (in seconds), acceptance rates (AR) and inefficiency factors (23). The table presents the medians and interquartile ranges (within parentheses) over 56 simulated data sets. The estimates are based on 20,000 draws after a burn-in sample of 5,000 observations. We use 50 simulated draws to evaluate the likelihood in the ARWMH and AIMH algorithms.

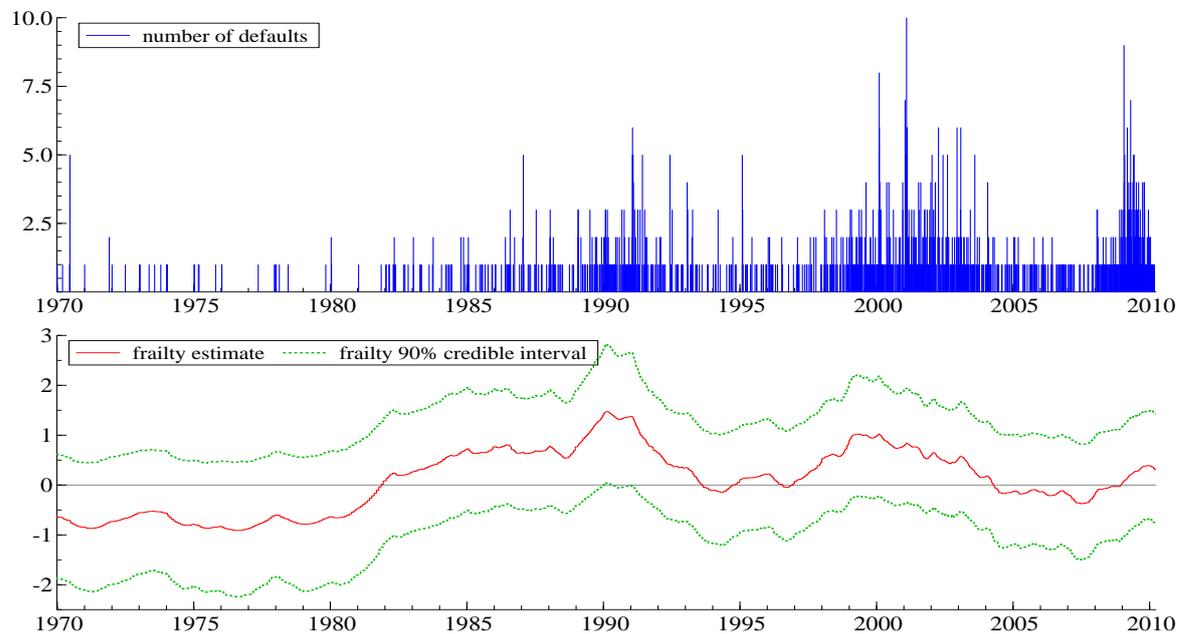
Algorithm	Time (in sec)	Acc. Rate	Inefficiency						
			$\omega$	$\beta_1$	$\beta_2$	$\beta_3$	$\beta_4$	$\gamma$	$\rho$
ARWMH	10053 (7380)	0.247 (0.015)	41.833 (13.621)	31.735 (6.883)	29.212 (4.619)	29.428 (5.650)	32.037 (7.931)	39.873 (21.481)	33.523 (1.469)
AIMH	8953 (6406)	0.411 (0.150)	25.256 (181.847)	8.308 (22.293)	10.812 (12.817)	9.439 (13.771)	8.592 (15.356)	26.513 (135.075)	10.777 (39.141)
EMitISEM 10k	5858 (3684)	0.658 (0.031)	3.431 (0.769)	2.586 (0.374)	2.575 (0.380)	2.624 (0.327)	2.642 (0.559)	3.406 (0.879)	2.893 (0.666)
EMitISEM mod. 10k	3124 (2386)	0.633 (0.041)	4.863 (3.950)	2.900 (0.735)	2.884 (0.804)	2.899 (0.946)	3.024 (0.728)	4.934 (5.605)	3.355 (1.210)
EMitISEM mod. 5k	2089 (1616)	0.618 (0.037)	5.877 (4.823)	3.265 (0.851)	3.169 (1.012)	3.279 (0.985)	3.303 (1.062)	5.926 (7.377)	4.117 (2.038)

**Table 3:** Gamma OU stochastic volatility model: We simulate 1000 observations from the Gamma OU model with parameters  $\mu = 0$ ,  $\beta = 0$ ,  $\xi = 0.5$ ,  $\omega^2 = 0.065$  and  $\lambda = 0.01$ , and we estimate the parameters 10 times with SMC<sup>2</sup>, density tempered marginalized sequential Monte Carlo sampler (DTMSMC) and our EMitISEM procedure. We present the mean of the posterior median estimates over the 10 runs. The standard deviation of the 10 estimates of the posterior median is within parentheses. The proposal construction time row shows the mean of the time needed to construct the EMitISEM proposal in hours while the total time row presents the mean total estimation time in hours.

	True value	SMC <sup>2</sup>	DTMSMC	EMitISEM
$\mu$	0	0.009 (0.006)	0.023 (0.036)	0.013 (0.004)
$\beta$	0	-0.065 (0.012)	-0.089 (0.07)	-0.07 (0.006)
$\xi$	0.5	0.558 (0.016)	0.57 (0.06)	0.541 (0.008)
$\omega^2$	0.065	0.071 (0.006)	0.088 (0.021)	0.068 (0.005)
$\lambda$	0.01	0.008 (0.001)	0.008 (0.002)	0.007 (0.0003)
proposal construction time				1.876 (0.592)
total time		0.914 (0.216)	2.737 (0.113)	2.086 (0.596)

**Table 4:** Stochastic intensity model: parameter estimates for U.S. corporate defaults from January 1, 1970 to March 4, 2010. The 90% credible intervals are within parentheses. The inefficiency factor is calculated as (23).

	Parameter	Estimate	Inefficiency
$\omega$	Constant	-4.750 (-6.357 , -3.287)	2.755
$\beta_1$	S&P500 1 year return	-0.862 (-1.346 , -0.375)	2.097
$\beta_2$	Term spread	0.013 (-0.164, 0.191)	3.495
$\beta_3$	3M TBill	-0.054 (-0.167 , 0.058)	2.489
$\beta_4$	Change in indust. prod.	-5.115 (-7.798 , -2.292)	2.450
$\gamma$	Loading on frailty	1.148 (0.548 , 2.294)	2.809
$\rho$	Mean reversion	0.120 (0.014 , 0.351)	2.688



**Figure 1:** Default data from 1st January 1970 to 4th March 2010 (top panel) and the smoothed estimate of the frailty process with its 90% credible interval (bottom panel).

**Table 5:** Descriptive statistics of the time series of daily (percentage) returns on S&P 500, DJIA, FTSE 100, DAX and AEX indices from 2nd January 2012 till 5th March 2015.

	SPX	DJIA	FTSE	DAX	AEX
mean	0.055	0.052	-0.011	0.024	0.017
st. dev	0.710	0.689	0.594	0.898	0.763
skewness	-0.224	-0.133	-0.123	-0.269	-0.194
kurtosis	4.142	3.980	4.534	4.153	4.649
min	-2.480	-2.324	-2.562	-3.173	-3.538
max	2.502	2.338	2.409	3.564	3.104
#	796	796	802	803	809

**Table 6:** Gamma OU stochastic volatility model: estimation results for the S&P 500, DJIA, FTSE 100, DAX and AEX indices from 2nd January 2012 till 5th March 2015. We used the EMitISEM procedure with 2000 parameter particles and 2000 state particles in the bootstrap filter. The estimation of the model was carried out using 12 cores in parallel. The 95% credible intervals are within parentheses. The time row shows the total computing time in hours.

	SPX	DJIA	FTSE	DAX	AEX
$\mu$	0.159 (0.087 , 0.239)	0.148 (0.073 , 0.232)	0.063 (-0.006 , 0.128)	0.142 (0.045 , 0.253)	0.1 (0.022 , 0.181)
$\beta$	-0.213 (-0.402 , -0.046)	-0.204 (-0.412 , -0.011)	-0.216 (-0.455 , 0.013)	-0.152 (-0.315 , 0.01)	-0.147 (-0.319 , 0.011)
$\xi$	0.492 (0.407 , 0.603)	0.468 (0.382 , 0.573)	0.344 (0.268 , 0.439)	0.8 (0.601 , 1.072)	0.571 (0.449 , 0.738)
$\omega^2$	0.126 (0.073 , 0.219)	0.099 (0.056 , 0.18)	0.057 (0.031 , 0.11)	0.269 (0.142 , 0.589)	0.147 (0.078 , 0.291)
$\lambda$	0.219 (0.132 , 0.387)	0.171 (0.108 , 0.294)	0.102 (0.046 , 0.182)	0.067 (0.03 , 0.171)	0.091 (0.041 , 0.166)
time	1.5	1.07	0.96	1.82	1.16

## APPENDIX

### *Joint Bayesian analysis of parameters and states in nonlinear non-Gaussian state space models*

István Barra, Lennart Hoogerheide, Siem Jan Koopman and André Lucas

#### A Expectation and Maximization steps in MitISEM

We want to maximize the weighted log-density

$$\frac{1}{N} \sum_{j=1}^N w^{(j)} \log q_{\zeta'}(\theta^{(j)}|y), \quad (\text{A.1})$$

where the weight  $w^{(j)}$  is the ratio of the target density kernel and the candidate density from which the  $d$ -dimensional vector draws  $\theta^{(j)}$  have been simulated, with weight  $w^{(j)}$  and where  $q_{\zeta'}(\theta^{(j)}|y)$  is a mixture of  $H$  Student's  $t$ -densities. The target density kernel is either the marginal posterior density kernel of  $\theta$  or the joint posterior density kernel of  $\theta$  and the signal  $x$ ; in the latter case the candidate density is the joint candidate density for  $\theta^{(j)}$  and  $x^{(j)}$  where  $x^{(j)}$  has been simulated conditionally on  $\theta^{(j)}$ . We can write the mixture of Student's  $t$ -densities using a latent variable representation where  $z^{(j)}$  is a latent  $H$  dimensional vector consisting of  $H - 1$  zeros and one element  $z_h^j = 1$  that indicates that the draw  $\theta^{(j)}$  belongs to the  $h$ -th Student's  $t$ -distribution. The mixing weight is  $Pr[z_h^j = 1] = \eta_h$ , and

$$\theta^{(j)} \sim N(\mu, \Sigma), \quad \mu = \sum_{i=1}^H z_h^i \mu_h, \quad \Sigma = \sum_{i=1}^H z_h^i \kappa_h^i \Sigma_h, \quad (\text{A.2})$$

where  $\mu_h$  and  $\Sigma_h$  are the mode vector and scale matrix of the  $h$ -th Student's  $t$ -distribution, and where the random variable  $\kappa_h^j$  has an Inverse-Gamma distribution

$$\kappa_h^j \sim IG(\nu_h/2, \nu_h/2),$$

where  $\nu_h$  is the degrees of freedom parameter of the  $h$ -th Student's  $t$ -distribution.

The Expectation-Maximization (EM) algorithm proceeds with iterations  $L = 1, 2, \dots$ , which consist of an expectation and maximization step, until it has converged to a (local) optimum. In the expectation step of iteration  $L$  the conditional expectations of the expressions involving the latent variables  $z^j$  and  $\kappa^j$  that occur in the log-density, given

the draws  $\theta^{(j)}$  and  $\zeta = \zeta^{(L-1)} = \{\mu_h, \Sigma_h, \nu_h, \eta_h; h = 1, \dots, H\}$ , the parameters from the previous EM iteration ( $L - 1$ ), are the following:

$$\tilde{z}_h^j \equiv \mathbb{E} \left[ z_h^j \mid \theta^{(j)}, \zeta = \zeta^{(L-1)} \right] = \frac{t(\theta^{(j)} \mid \mu_h, \Sigma_h, \nu_h) \eta_h}{\sum_{i=1}^H t(\theta^{(j)} \mid \mu_i, \Sigma_i, \nu_i) \eta_i}, \quad (\text{A.3})$$

where  $t(\cdot \mid \mu, \Sigma, \nu)$  is a Student's  $t$ -density with mode  $\mu$ , scale matrix  $\Sigma$  and degree of freedom  $\nu$ ,

$$\widetilde{z/\kappa}_h^j \equiv \mathbb{E} \left[ \frac{z_h^j}{\kappa_h^j} \mid \theta^{(j)}, \zeta = \zeta^{(L-1)} \right] = \tilde{z}_h^j \frac{d + \nu_h}{\rho_h^j + \nu_h}, \quad (\text{A.4})$$

$$\begin{aligned} \xi_h^i &\equiv \mathbb{E} \left[ \log \kappa_h^j \mid \theta^{(j)}, \zeta = \zeta^{(L-1)} \right] = \\ &= \left[ \log \left( \frac{\rho_h^j + \nu_h}{2} \right) - \psi \left( \frac{d + \nu_h}{2} \right) \right] \tilde{z}_h^j + \left[ \log \left( \frac{\nu_h}{2} \right) - \psi \left( \frac{\nu_h}{2} \right) \right] (1 - \tilde{z}_h^j), \end{aligned} \quad (\text{A.5})$$

$$\delta_h^i \equiv \mathbb{E} \left[ \frac{1}{\kappa_h^j} \mid \theta^{(j)}, \zeta \right] = \widetilde{z/\kappa}_h^j + (1 - \tilde{z}_h^j), \quad (\text{A.6})$$

with  $\rho_h^j \equiv (\theta^{(j)} - \mu_h)' \Sigma_h^{-1} (\theta^{(j)} - \mu_h)$ , and  $\psi(\cdot)$  is the digamma function.

The maximization step of iteration  $L$  consists of the following updates

$$\mu_h^{(L)} = \left[ \sum_{j=1}^N w^{(j)} \widetilde{z/\kappa}_h^j \right]^{-1} \left[ \sum_{j=1}^N w^{(j)} \theta^{(j)} \widetilde{z/\kappa}_h^j \right], \quad (\text{A.7})$$

$$\Sigma_h^{(L)} = \frac{\sum_{j=1}^N w^{(j)} (\theta^{(j)} - \mu_h^{(L)}) (\theta^{(j)} - \mu_h^{(L)})' \widetilde{z/\kappa}_h^j}{\sum_{j=1}^N w^{(j)} \tilde{z}_h^j}, \quad (\text{A.8})$$

$$\eta_h^{(L)} = \frac{\sum_{j=1}^N w^{(j)} \tilde{z}_h^j}{\sum_{j=1}^N w^{(j)}}. \quad (\text{A.9})$$

Finally,  $\nu_h^{(L)}$  is obtained by solving the first-order condition

$$-\psi(\nu_h/2) + \log(\nu_h/2) + 1 - \frac{\sum_{j=1}^N w^{(j)} \xi_h^j}{\sum_{j=1}^N w^{(j)}} - \frac{\sum_{j=1}^N w^{(j)} \delta_h^j}{\sum_{j=1}^N w^{(j)}} = 0 \quad (\text{A.10})$$

for  $\nu_h$ . For more details we refer to Hoogerheide et al. (2012).

## B The NAIS method

We can express the likelihood of the state space model given by (1) and (2) as

$$L(y|\theta) = \int \frac{p(x, y|\theta)}{q(y|x, \theta)} q(y|x, \theta) dx = q(y|\theta) \int \omega(x, y|\theta) q(y|x, \theta) dx, \quad (\text{B.1})$$

where  $x = (x_1, \dots, x_T)'$ , with  $x_t = c_t + Z_t \alpha_t$  being the signal at time  $t$  for  $t = 1, \dots, T$ , and where

$$\omega(x, y|\theta) \equiv p(y|x, \theta)/q(y|x, \theta). \quad (\text{B.2})$$

The Gaussian importance or proposal density can be written as

$$q(y_t|x_t, \theta) = \exp \left\{ a_t + b_t'x_t - \frac{1}{2}x_t' C_t x_t \right\} \quad (\text{B.3})$$

where  $a_t$ ,  $b_t$  and  $C_t$  depend on the observations  $y$  and the parameters in  $\theta$  for  $t = 1, \dots, T$ . The importance density at time  $t$  is effectively determined by  $b_t$  and  $C_t$  as the constant  $a_t$  is chosen such that the density integrates to one. We can represent the Gaussian importance density as the smoothed density in the linear Gaussian state space model with its observation equation given by

$$y_t^* = x_t + \varepsilon_t, \quad \varepsilon \sim \text{N}(0, C_t^{-1}), \quad t = 1, \dots, T, \quad (\text{B.4})$$

where  $y_t^* = C_t^{-1}b_t$  for  $t = 1, \dots, T$  and the transition density given in equation (2).

To formulate an effective importance density we choose its parameters, as collected in  $\chi = \{b_1, \dots, b_T, C_1, \dots, C_T\}$ , by minimizing a conveniently chosen metric associated with the importance sample variation, that is

$$\min_{\chi_t} \int \lambda^2(x_t, y_t|\theta) \omega(x_t, y_t|\theta) q(x_t|y_t^*, \theta) dx_t, \quad (\text{B.5})$$

for every  $t$ , where

$$\lambda(x_t, y_t|\theta) = \log p(y_t|x_t, \theta) - \log q(y_t^*|x_t, \theta) - \lambda_{0t}. \quad (\text{B.6})$$

We can rewrite the minimization as

$$\min_{\chi_t} \sum_{j=1}^M \lambda^2(\tilde{x}_{tj}, y_t|\theta) \omega_{tj}, \quad \omega_{tj} = q(\tilde{x}_{tj}|y_t^*, \theta) \omega(\tilde{x}_{tj}, y_t|\theta) h(z_j) e^{z_j^2}, \quad (\text{B.7})$$

with  $\tilde{x}_{tj} = \hat{x}_t + V_t^{1/2} z_j$ , for  $j = 1, \dots, M$ , and

$$q(\tilde{x}_{tj}|y_t^*, \theta) = \exp \left\{ -\frac{1}{2} z_j^2 \right\} / \sqrt{2\pi}, \quad t = 1, \dots, T, \quad (\text{B.8})$$

where  $\hat{x}_t$  is the smoothed signal,  $V_t$  is smoothed signal variance and  $z_j$  are the abscissa designated by the Gauss-Hermite quadrature. The minimization is carried out iteratively. First, for a given  $\chi$  we obtain  $\hat{x}_t$  and  $V_t$  for  $t = 1, \dots, T$  from the linear Gaussian state space model in (B.4). Second, we obtain the optimal  $\chi_t = \{C_t, b_t\}$  for  $t = 1, \dots, T$  by a weighted least squares computation with “dependent” variable  $\log p(y_t|\tilde{x}_{tj}, \theta)$  and “explanatory variables”  $\tilde{x}_{tj}$  and  $\tilde{x}_{tj}^2$ . We iterate these steps until convergence of  $\chi$ . For a more detailed discussion, we refer to Koopman et al. (2015).

## C The algorithms

### EMitISEM algorithm

1. **Initialization:** Simulate a series of  $N$  parameter vector draws  $\theta^{(0,1)}, \dots, \theta^{(0,N)}$  from the initial Student's  $t$  candidate density  $q_{\zeta(0)}(\theta|y)$  with parameters denoted by  $\zeta(0)$ . The parameters of the initial candidate density  $q_{\zeta(0)}(\theta|y)$  consist of the mode equal to the simulated maximum likelihood estimate of  $\theta$ , the scale parameter equal to minus the inverse Hessian of the log likelihood evaluated at the current parameter estimates and the degree of freedom parameter equal to 5. The evaluation of the likelihood is given in (B.1). Conditionally on the draws  $\theta^{(0,1)}, \dots, \theta^{(0,N)}$  we simulate a corresponding series of  $N$  signal paths for  $x$  from  $q(x|\theta^{(j)}, y)$  and denote these by  $x^{(0,1)}, \dots, x^{(0,N)}$ . Finally, we evaluate the joint importance sampling (IS) weights  $w^{(0,1)}, \dots, w^{(0,N)}$  given by (10).
2. **Adaptation:** Estimate the mean and variance of the target distribution via IS using the draws  $\theta^{(0,1)}, \dots, \theta^{(0,N)}$  from  $q_{\zeta(0)}(\theta|y)$ . We adopt the estimated mean and variance as the mode and scale of the new proposal distribution, which is denoted by  $q_{\zeta(1)}$ . We simulate draws  $\theta^{(1,1)}, \dots, \theta^{(1,N)}$  from  $q_{\zeta(1)}$ . Finally, we simulate signal paths  $x^{(1,1)}, \dots, x^{(1,N)}$  conditionally on the parameter draws, and evaluate the joint IS weights  $w^{(1,1)}, \dots, w^{(1,N)}$  given by (10) and using  $q_{\zeta} = q_{\zeta(1)}$ .
3. **IS weighted EM algorithm:** We obtain the updated proposal  $q_{\zeta(2)}$  from the IS weighted EM algorithm of MitISEM, using the latest draws  $\theta^{(1,1)}, \dots, \theta^{(1,N)}$  and corresponding IS weights  $w^{(1,1)}, \dots, w^{(1,N)}$ . Appendix A provides further details about the MitISEM algorithm. We simulate draws  $\theta^{(2,1)}, \dots, \theta^{(2,N)}$  from the updated proposal  $q_{\zeta(2)}$ , and signal paths  $x^{(2,1)}, \dots, x^{(2,N)}$  conditionally on these parameter draws. We compute the corresponding IS weights  $w^{(2,1)}, \dots, w^{(2,N)}$  in (10) using  $q_{\zeta} = q_{\zeta(2)}$ . Calculate the coefficient of variation of the weights  $w^{(2,1)}, \dots, w^{(2,N)}$  (i.e., the standard deviation divided by the mean)  $CoV^{(2)}$ , where we use the notation

$$CoV^{(k)} = \frac{\sqrt{\frac{1}{N} \sum_{j=1}^N (w^{(k,j)})^2 - \left( \frac{1}{N} \sum_{j=1}^N w^{(k,j)} \right)^2}}{\frac{1}{N} \sum_{j=1}^N w^{(k,j)}} \quad (\text{C.1})$$

for the coefficient of variation of the weights at iteration  $k$ . Set  $i = 3$ .

4. **Iterate on the number of mixture components:**

We consider 10% of the simulated draws from the last iteration  $\theta^{(i-1,1)}, \dots, \theta^{(i-1,N)}$  that correspond to the highest IS weights, based on the current candidate mixture density  $q_{\zeta(i-1)}$ . We use this smaller set of draws and weights to compute a new

mode and scale matrix. The new mode and scale are used as starting parameters for the additional Student's  $t$  component in the mixture. We adopt this choice because the new Student's  $t$  component should cover a part of the parameter space that is insufficiently covered by the previous candidate, when compared to the target density as discussed in Hoogerheide et al. (2012). The starting values for the mixture probability and degrees of freedom parameter for the new Student's  $t$  component are set to 0.1 and 5, respectively. The starting values of the mixture probabilities for the older Student's  $t$  components are obtained by multiplying the latest values by 0.9. Given the last set of  $N$  simulated draws  $\theta^{(i-1,1)}, \dots, \theta^{(i-1,N)}$  and the corresponding importance weights  $w^{(i-1,1)}, \dots, w^{(i-1,N)}$ , we apply the IS weighted EM algorithm to update the new mixture distribution. We simulate draws  $\theta^{(i,1)}, \dots, \theta^{(i,N)}$  from the updated proposal  $q_{\zeta(i)}$ , and signal paths  $x^{(i,1)}, \dots, x^{(i,N)}$  conditionally on these parameter draws. We compute the corresponding IS weights  $w^{(i,1)}, \dots, w^{(i,N)}$  in (10) using  $q_{\zeta} = q_{\zeta(i)}$ .

5. **Evaluate the IS weights:** We calculate the coefficient of variation  $CoV^{(i)}$  of the IS weights from the current iteration  $w^{(i,1)}, \dots, w^{(i,N)}$ . We terminate the iterations when the coefficient of variation changes by less than 5% compared to the coefficient of variation in the last iteration  $CoV^{(i-1)}$ ; otherwise we go to Step 4.

### EMitISEM modified algorithm

1. **Initialization:** Same as Step 1 of the EMitISEM algorithm.
2. **Adaptation:** Same as Step 2 of the EMitISEM algorithm.
3. **IS weighted EM algorithm:** Same as Step 3 of the EMitISEM algorithm.
4. **Iterate on the number of mixture components:** We now consider 10% of the simulated draws  $\theta^{(2,1)}, \dots, \theta^{(2,N)}$  that correspond to the highest IS weights, based on the current candidate mixture density  $q_{\zeta(i-1)}$ . We use this smaller set of draws and weights to compute a new mode and scale matrix. The new mode and scale are used as starting parameters for the additional Student's  $t$  component in the mixture. The starting values for the mixture probability and degrees of freedom parameter for the new Student's  $t$  component are set to 0.1 and 5, respectively. The starting values of the mixture probabilities for the older Student's  $t$  components are obtained by multiplying the latest values by 0.9. Given the last set of  $N$  simulated draws  $\theta^{(2,1)}, \dots, \theta^{(2,N)}$  and the corresponding importance weights  $w^{(2,1)}, \dots, w^{(2,N)}$ , we apply the IS weighted EM algorithm to update the new mixture distribution to obtain  $q_{\zeta(i)}$ .

5. **Evaluate the IS weights:** We estimate the coefficient of variation  $CoV^{(i)}$  using formulas (14) and (15). We terminate the iterations when the coefficient of variation changes by less than 5% compared to the coefficient of variation in the last iteration  $CoV^{(i-1)}$ ; otherwise we go to Step 4.

### EMitISEM algorithm for models with a non-Gaussian state

1. **Initialization:** Simulate a series of  $N$  parameter vector draws  $\theta^{(0,1)}, \dots, \theta^{(0,N)}$  from the initial Student's  $t$  candidate density  $q_{\zeta(0)}(\theta|y)$  with its mode equal to the likelihood estimates of  $\theta$  using the smooth particle filter to evaluate the likelihood by Pitt (2002) and with its scale equal to minus the inverse Hessian of the log likelihood evaluated at the current parameter estimates. Conditionally on the draws  $\theta^{(0,1)}, \dots, \theta^{(0,N)}$  we evaluate the likelihood by running  $N$  particle filters in parallel. Finally, we evaluate the estimates of the importance sampling weights  $\hat{w}^{(0,1)}, \dots, \hat{w}^{(0,N)}$  given by (18).
2. **Adaptation:** Same as Step 2 of the EMitISEM algorithm, but with the estimated weights  $\hat{w}^{(1,1)}, \dots, \hat{w}^{(1,N)}$  from (18) replacing the weights from (10).
3. **IS weighted EM algorithm:** Same as Step 3 of the EMitISEM algorithm, but with the estimated weights  $\hat{w}^{(1,1)}, \dots, \hat{w}^{(1,N)}$  from (18) replacing the weights from (10).
4. **Iterate on the number of mixture components:** Same as Step 4 of the EMitISEM algorithm, but with the estimated weights  $\hat{w}^{(1,1)}, \dots, \hat{w}^{(1,N)}$  from (18) replacing the weights from (10).
5. **Evaluate the IS weights:** Same as Step 5 of the EMitISEM algorithm, but with the estimated weights  $\hat{w}^{(1,1)}, \dots, \hat{w}^{(1,N)}$  from (18) replacing the weights from (10).

## D Adaptive random-walk Metropolis-Hastings

Roberts and Rosenthal (2009) propose an adaptive random walk Metropolis-Hastings algorithm, with a proposal of the following form

$$q_n(\theta; \theta_{n-1}) = \omega_{1n} \phi_d(\theta; \theta_{n-1}, \kappa_1 \Sigma_1) + \omega_{2n} \phi_d(\theta; \theta_{n-1}, \kappa_2 \Sigma_{2n}), \quad (\text{D.1})$$

where  $\phi_d(\theta; \hat{\theta}, \Sigma)$  is a  $d$  dimensional multivariate normal density with mean  $\hat{\theta}$  and covariance matrix  $\Sigma$ . We set  $\omega_{1n} = 1$  until  $n > n_0$ , and  $\omega_{1n} = 0.05$  afterwards. The scalars  $\kappa_1 = 0.1^2/d$  and  $\kappa_2 = 2.38^2/d$  and  $\Sigma_1 = I_d$  are constant throughout the procedure, while  $\Sigma_{2n}$  covariance matrix is estimated using the first  $n - 1$  iterates.

## E Adaptive mixture of normals

Giordani and Kohn (2010) and Pitt et al. (2012) suggest an adaptive mixture of normals proposal, which has the form

$$q_n(\theta) = \sum_{k=1}^4 \omega_{kn} q_{kn}(\theta) \quad \omega_{kn} \leq 0, \quad \text{for } k = 1, \dots, 4 \quad \text{and} \quad \sum_{k=1}^4 \omega_{kn} = 1, \quad (\text{E.1})$$

at iteration  $n$ . The adaptation has two stages. We start the first stage with setting  $\omega_{1n} = 0.8$ ,  $\omega_{2n} = 0.2$  and we use a Gaussian density for  $q_{1n}$  with mean equal to the simulated maximum likelihood estimates and variance equal to minus the inverse Hessian at the mean. Moreover we set  $q_{2n}$  as a heavy tailed version of  $q_{1n}$  by setting the covariance matrix 15 times the covariance matrix of  $q_{1n}$ . After  $5d$  accepted draws (where  $d$  is equal to the dimension of  $\theta$ ) we set  $q_{3n}$  and  $q_{4n}$  and we change the component weights in (E.1).  $q_{3n}$  is obtained as a mixture of normals using k-means clustering on the previous draws.  $q_{4n}$  is the fat tailed version of  $q_{3n}$ , it has the same means and mixture probabilities as  $q_{3n}$  but the covariance matrices are multiplied by 20. The new weights are the following  $\omega_{1n} = 0.15$ ,  $\omega_{2n} = 0.05$ ,  $\omega_{3n} = 0.7$ ,  $\omega_{4n} = 0.1$ . In the rest of the first stage we update  $q_{3n}$  at predetermined updating times or after rejecting 10 candidate draws in a row. We always set  $q_{4n}$  to be the fat tailed version of  $q_{3n}$ . The first stage ends if the minimal acceptance rate (i.e., the conditional acceptance probability in the MH algorithm) in the last 1,000 draws is above 0.02. After the first stage we set  $q_{1n} = q_{3n}$ , i.e., the last version of the mixture of normals, and  $q_{2n}$  is again the fat tailed version of the new  $q_{1n}$ . In the second stage we only update at predetermined updating times.

## F Likelihood estimation: NAIS versus particle filters

We have argued that ARWMH and AIMH methods can be implemented using both PMCMC and NAIS algorithms for drawing the signal vectors. To assess the difference between the two implementations, we use the particle filter and NAIS methods to evaluate the likelihood function for a stochastic volatility model with normally distributed errors. Note that this is an illustrative model that differs from the stochastic volatility model in the main exposition. A review of different particle filtering methods is provided in for instance Doucet et al. (2001). For this illustrative stochastic volatility model, we obtain more efficient likelihood estimates when using NAIS in comparison to using particle filters. The NAIS importance sampling estimates of the likelihood function have lower variance and need less computing time than the particle filter likelihood estimates.

We simulate 56 data sets using the same data generating process for the stochastic

volatility model

$$y_t = \exp(x_t/2)\varepsilon_t, \quad \varepsilon_t \sim \text{NID}(0, 1), \quad (\text{F.1})$$

$$x_t - \delta = \phi(x_{t-1} - \delta) + \eta_t, \quad \eta_t \sim \text{NID}(0, \sigma_\eta^2), \quad (\text{F.2})$$

where  $x_t$  is the unobserved log-volatility process, which follows an autoregressive process of order 1,  $\varepsilon_t$  is a standardized error term,  $\delta$  is the overall mean of  $x_t$ ,  $0 < \phi < 1$  is the persistence parameter, and  $\sigma_\eta^2 > 0$  is the innovation variance of the log-volatility process. The 56 data sets are generated from the basic SV model (F.1) and (F.2) with parameters set equal to  $\delta = 0.48$ ,  $\phi = 0.97$ , and  $\sigma_\eta^2 = 0.049$ . We simulate time series of length 1,250 and use the following prior specifications (see Omori et al. (2007)) for the parameters

$$\delta \sim \text{N}(0, 1), \quad \frac{\phi + 1}{2} \sim \text{Beta}(20, 1.5), \quad \frac{1}{\sigma_\eta^2} \sim \text{Gamma}\left(\frac{5}{2}, \frac{0.05}{2}\right).$$

We estimate the likelihood value at the “true” parameter values 100 times for each simulated data set using the bootstrap filter of Gordon et al. (1993), the auxiliary particle filter of Pitt and Shephard (1999) and the NAIS method of Koopman et al. (2015). We compute 100 likelihood estimates for each data set and we calculate the variance of the estimates together with the mean computing time for each data set. We report the median variance and computing times over the 56 data sets. Table F.1 presents the results. For all considered time series lengths, the median variance of the NAIS estimate is lower than the median variance of the particle filter estimates for all numbers of particles considered. Moreover, the estimation using NAIS takes much less time than the estimation using particle filters. We therefore use NAIS in all algorithms to facilitate fair comparisons.

**Table F.1:** Illustrative stochastic volatility model with normally distributed errors: Comparison of the variance of the particle filter and importance sampling estimate of the likelihood based on 56 simulated data sets. We compare the likelihood estimates from the bootstrap filter (BF), the auxiliary particle filter (APF) and the numerically accelerated importance sampling (NAIS).  $M$  denotes the number of particles/number of draws.  $T$  is the length of the simulated data set. For different  $M$  and  $T$  we report the median of the variances over the 56 data sets, where these variances are calculated from 100 runs per data set.

Method	M	Variance		Time	
		T=1000	T=2000	T=1000	T=2000
BF	250	1.079	1.505	5.970	37.901
	500	0.743	1.071	18.435	88.828
	1000	0.541	0.735	44.018	173.432
APF	250	0.988	1.420	6.362	37.434
	500	0.674	0.963	21.025	88.764
	1000	0.490	0.687	42.663	195.456
NAIS	50	0.140	0.233	0.283	0.527