Unobserved components with stochastic volatility: Likelihood-based estimation and signal extraction Supplementary Appendix

Mengheng Li*

UTS Business School, University of Technology Sydney, Australia

Siem Jan Koopman†

Department of Econometrics, VU University Amsterdam, The Netherlands,
CREATES, Aarhus University, Denmark,
Tinbergen Institute Amsterdam, The Netherlands

This version: August 11, 2019

Abstract

This supplementary appendix provides a detailed account of the econometric methodology of simulated maximum likelihood using importance sampling in the main text, Li and Koopman (2019). Some more empirical results pertaining to the application of US quarterly inflation are given in this appendix as well. Notations are the same as in the main text, unless stated otherwise.

1 Details of ML estimation via multivariate NAIS

In this section we detail an efficient method for computing the Monte Carlo estimate of the log-likelihood function for state space models with SV via a multivariate modification of NAIS Koopman et al. (2015). In order to integrate out the high dimensional vector of SV series to evaluate the log-likelihood function, we maximize a simulated Monte Carlo estimate of the log-likelihood function based on the multivariate NAIS algorithm. Given the model parameter vector θ the algorithm can be summarized by: 1) construct an importance density g; 2) draw Monte Carlo samples of SV from g; 3) conditional on each draw, evaluate the likelihood p using Kalman filter; 4) average the likelihood with weights p/g.

1.1 Likelihood function via importance sampling

In general, UCSV models can be cast into the following state space form:

$$\begin{aligned} y_t &= Z\alpha_t + \epsilon_t, \quad \epsilon_t \sim N(0, \Omega_t) \\ \alpha_{t+1} &= T\alpha_t + \eta_t, \quad \eta_t \sim N(0, Q_t), \end{aligned} \tag{1}$$

for $t = 1, ..., T$ with proper initialization of α_1. Suppose in (1), one has unobserved linear states $\alpha_t \in \mathbb{R}^m$ and the SV $h_t \in \mathbb{R}^{p+m}$ going into Ω_t and Q_t. This means $(2m + p) \times T$ integrations

*Email: mengheng.li@uts.edu.au
†Email: s.j.koopman@vu.nl

have to be carried out in order to evaluate the data likelihood function. Naively doing so is impossible as such high-dimensional integration makes function evaluations intractable.

The likelihood function of model (1) can be written as follows,

$$\begin{aligned} L(Y_T;\theta) &= \int_{H_T} \frac{p(Y_T,H_T;\theta)}{g(Y_T,H_T;\theta)} g(Y_T,H_T;\theta)dH_T \\ &= g(Y_T;\theta)\int_{H_T} \frac{p(Y_T,H_T;\theta)}{g(Y_T,H_T;\theta)} g(H_T|Y_T;\theta)dH_T \\ &= g(Y_T;\theta)\int_{H_T} \omega(Y_T,H_T;\theta) g(H_T|Y_T;\theta)dH_T, \end{aligned} \tag{2}$$

where $g(Y_T;\theta)$ is the likelihood function of a Gaussian importance model to be determined and where the importance weight function is given by

$$\omega(Y_T,H_T;\theta) = \frac{p(Y_T,H_T;\theta)}{g(Y_T,H_T;\theta)} = \frac{p(Y_T|H_T;\theta)}{g(Y_T|H_T;\theta)}.^{1} \tag{3}$$

The last equality holds valid as the joint importance density $g(Y_T,H_T;\theta)$ is Gaussian in UCSV models,indicating $p(H_T;\theta) = g(H_T;\theta)$. Notice that there is no need to integrate out $\{\alpha_1,...,\alpha_T\}$. Given a realized sequence of H_T, Kalman filter can efficiently evaluate $p(Y_T|H_T;\theta)$ via prediction error decomposition, which is a *Rao-Blackwellisation* procedure.

Direct maximization of (2) using numerical integration is near impossible, we estimate state space models with SV via Monte Carlo simulated maximum likelihood. Under regularity conditions specified in Geweke (1989) which we discuss in Section 4, one has the following unbiased and consistent estimator for the likelihood

$$\hat{L}(Y_T;\theta) = g(Y_T;\theta)\bar{\omega}, \quad \bar{\omega} = \frac{1}{M}\sum_{j=1}^{M}\omega^{(j)}, \quad \omega^{(j)} = \omega(Y_T,H_T^{(j)};\theta), \tag{4}$$

where $\omega^{(j)}$ is the realized importance weight function in (2) for a draw of $H_T^{(j)}$ from the importance density $g(H_T|Y_T;\theta)$. Numerical optimizers such as L-BFGS can be used to maximize the above Monte Carlo estimate of the likelihood function with respect to the parameter vector θ, keeping a fixed random seed. In practice however, the log-likelihood function is maximized. Because the logarithm of the estimate (4) does not converge in probability to the expected log-likelihood function, we correct the bias in $\log\hat{L}(Y_T;\theta)$ based on a second order Taylor expansion. Similar to Durbin and Koopman (1997), we therefore maximize the bias-corrected Monte Carlo estimate of the log-likelihood function

$$\begin{aligned} \hat{l}(Y_T;\theta) &= \log\hat{L}(Y_T;\theta) + \frac{1}{2M}\bar{\omega}^{-2}s_\omega^2 \\ &= \log g(Y_T;\theta) + \log\bar{\omega} + \frac{1}{2M}\bar{\omega}^{-2}s_\omega^2, \end{aligned} \tag{5}$$

where $s_\omega^2 = (M-1)^{-1}\sum_{j=1}^{M}(\omega^{(j)}-\bar{\omega})^2$ is the sample variance of importance weights.

1.2 Conditional likelihood function via Kalman filter

To evaluate the realized importance weight function $\omega^{(j)}$ in (3), we have to calculate the likelihood $p(Y_T|H_T^{(j)};\theta)$. Given a draw of the SV $H_T^{(j)}$ from the importance density $g(H_T|Y_T;\theta)$,

[1] Importance sampling boils down to evaluate an expectation under a change of probability measure. Loosely speaking, likelihood $L(y)$ is equal to $\mathbb{E}_{p(h)}(L(y,h)) = \mathbb{E}_{g(h)}(\omega L(y,h))$, and the importance weight ω is analogous to the *Radon-Nikodym derivative*.

model (1) reduces to a linear Gaussian state space model. Suppressing the dependence on $H_T^{(j)}$, we have the log-likelihood

$$\log p(Y_T|H_T^{(j)};\theta) = -\frac{1}{2}\left(np\log 2\pi + \sum_{t=1}^{n}\log(|F_t|) + \sum_{t=1}^{n} v_t' F_t^{-1} v_t\right), \tag{6}$$

where $v_t \in \mathbb{R}^p$ is the prediction error with its covariance matrix F_t. Both v_t and F_t come from the Kalman filter recursion

$$\begin{aligned} v_t &= y_t - Za_t - d, & F_t &= ZP_tZ' + H_t \\ a_{t+1} &= Ta_t + K_t v_t + c, & P_{t+1} &= TP_t(T - K_tZ)' + Q_t', \end{aligned} \tag{7}$$

for $t = 1, ..., n$, and $K_t = TP_tZ'F_t^{-1}$ is the Kalman gain. a_1 and P_1 are the mean vector and covariance matrix of the initial state vector α_1 which are initialised according to Section 2.1. The above recursion comes from standard results of conditional expectation and covariance of Gaussian random variables with

$$a_t = \mathbb{E}(\alpha_t|y_{1:t-1}), \qquad P_t = \mathrm{Var}(\alpha_t|y_{1:t-1}).$$

We use a_t, P_t, v_t and F_t for forecasting and smoothing of the unobserved state. In the next section, a linear Gaussian importance density $g(H_T^{(j)}|Y_T;\theta)$ is constructed for drawing $H_T^{(j)}$, based on simulation smoothing (Durbin and Koopman, 2012) where those objects are also used.

1.3 Importance density via multivariate NAIS

To evaluate the Monte Carlo estimate of the log-likelihood function (5), we need to calculate the conditional likelihood $g(Y_T|H_T^{(ij)};\theta)$ to obtain the importance weights, and marginal log-likelihood $\log g(Y_T;\theta)$. Moreover, we need an efficient way to draw samples $H_T^{(j)}$, $j = 1, 2, ..., M$ from $g(H_T|Y_T;\theta)$.

Richard and Zhang (2007) developed an efficient importance sampling (EIS) procedure for evaluating high-dimensional numerical integrals. Numerically accelerated importance sampling (NAIS) by Koopman et al. (2015) is a refinement of EIS. It takes advantage of Gaussian samplers such as the simulation smoother developed by De Jong and Shephard (1995) and Durbin and Koopman (2002) so that the sequence $H_T^{(j)}$ can be simulated from an importance density $g(H_T|Y_T;\theta)$ as a whole block. Similar to EIS, original NAIS method assumes *conditional independence* to hold such that, in light of (3) the density of Y_T conditional on H_T can be decomposed as

$$p(Y_T|H_T;\theta) = \prod_{t=1}^{n} p(y_t|h_t;\theta).$$

However, *conditional independence* assumption does not hold in the state space model with SV. Given h_t, we cannot evaluate $p(y_t|h_t;\theta)$ because of the unobserved states α_t in (1), which complicates the direct use of NAIS. To tackle this issue, we propose to use the prediction errors $v_{1:t}$ the variance $F_{1:t}$ produced by Kalman filter. It follows from (6) that

$$p(Y_T|H_T;\theta) = \prod_{t=1}^{n} p(v_t|h_t;\theta), \quad p(v_t|h_t;\theta) \equiv N(0, F_t).$$

Then the importance weight function (3) factorizes

$$\omega(Y_T, H_T;\theta) = \prod_{t=1}^{n}\omega_t(y_t, h_t;\theta) = \prod_{t=1}^{n}\frac{p(v_t|h_t;\theta)}{g(y_t|h_t;\theta)}. \tag{8}$$

Using prediction errors to factorize $p(Y_T|H_T;\theta)$ enables us to use NAIS. NAIS represents the Gaussian importance density as

$$g(Y_T, H_T;\theta) = \prod_{t=1}^{n} g(y_t|h_t;\theta)g(h_t|h_{t-1};\theta),$$

where $g(h_t|h_{t-1};\theta)$ specifies the Markov structure of h_t and where

$$g(y_t|h_t;\theta) = \exp\left(r_t + b_t'h_t - \frac{1}{2}h_t'C_th_t\right). \tag{9}$$

r_t is an integrating constant. $b_t \in \mathbb{R}^m$ and $C_t \in \mathbb{R}^{m\times m}$ for $t = 1, 2, ..., n$ are defined (implicitly) as functions of data Y_T and parameter vector θ. So our objective is to choose the set of importance sampling parameters b_t and C_t, $t = 1, ..., T$.

Similar to Koopman et al. (2012), we can write the density function (9) as the one associated with observation equation $y_t^+ = C_t^{-1}b_t$ in the following linear Gaussian state space model,

$$y_t^+ = h_t + \epsilon_t^+, \quad \epsilon_t \sim N(0, C_t^{-1}), \quad t = 1, 2, ..., n, \tag{10}$$

where state equation for h_t. One can easily verify the equivalence of (9) with the Gaussian log-density $\log g(y_t^+|h_t;\theta)$ for y_t^+ in (10) by

$$\begin{aligned}\log g(y_t^+|h_t;\theta) &= -\frac{1}{2}\log(2\pi)^k + \frac{1}{2}\log|C_t| - \frac{1}{2}\{(C_t^{-1}b_t - h_t)'C_t(C_t^{-1}b_t - h_t)\} \\ &= a_t + b_t'h_t - \frac{1}{2}h_t'C_th_t,\end{aligned} \tag{11}$$

where the constant $a_t = -\frac{1}{2}(k\log(2\pi) - \log(|C_t|) + b_t'C_tb_t)$ collects all the terms that are not associated with h_t. It follows that $g(y_t|h_t;\theta) \equiv g(y_t^+|h_t;\theta)$ for all t. Hence, we have

$$g(Y_T, H_T;\theta) \equiv g(Y_T^+, H_T;\theta), \quad \text{with } y_t^+ = C_t^{-1}b_t.$$

SPDK method by Shephard and Pitt (1997) and Durbin and Koopman (1997) essentially chooses b_t and C_t, $t = 1, ..., T$, such that $g(H_T|Y_T^+;\theta)$ forms a second order local approximation to $p(H_T|Y_T;\theta)$.

Geweke (1989) gave conditions for a central limit theorem of importance sampling. Importantly, the importance weights should have finite second moment. Notice that minimizing the variance of log-weights instead of weights is easier as it preserves the linear structure of log-Gaussian density. Therefore we choose the importance parameters b_t and C_t for all t to minimize the variance of the logarithm of the importance weights, i.e.

$$\min_{\{b_t,C_t\}_{t=1}^T} \int_{H_T} \lambda^2(Y_T, H_T;\theta)\omega(Y_T, H_T;\theta)g(H_T|Y_T^+;\theta)dH_T, \tag{12}$$

where

$$\lambda(Y_T, H_T;\theta) = \log p(Y_T|H_T;\theta) - \log g(Y_T^+|H_T;\theta) - \text{constant}.$$

The last term is a normalizing constant such that the mean of $\lambda(Y_T, H_T;\theta)$ is set to zero.

Using the factorization (8), we can approximate the above minimization problem by considering a minimization problem for each t separately, i.e. for $t = 1, 2, ..., n$

$$\min_{b_t,C_t} \int \lambda_t^2(y_t, h_t;\theta)\omega_t(y_t, h_t;\theta)g(h_t|Y_T^+;\theta)dh_t, \tag{13}$$

where

$$\begin{aligned}\lambda_t &= \log p(v_t|h_t;\theta) - \log g(y_t^+|h_t;\theta) - \text{constant}, \\ \omega_t(y_t, h_t;\theta) &= \frac{p(v_t|h_t;\theta)}{g(y_t^+|h_t;\theta)}.\end{aligned}$$

A natural way to solve (13) is by replacing the integral with a Monte Carlo average, i.e.

$$\min_{b_t,C_t} \frac{1}{S}\sum_{j=1}^{S} \lambda_t^2(y_t, h_t^{(j)};\theta)\omega_t(y_t, h_t^{(j)};\theta), \quad h_t^{(j)} \overset{\text{i.i.d}}{\sim} g(h_t|Y_T;\theta). \tag{14}$$

Notice that in (1), the SV $h_t \in \mathbb{R}^{p+m}$ is still multidimensional, a multidimensional summation is thus needed in (14). Monte Carlo averaging here is not ideal as draws of $h_t^{(j)}$ scatter around in the multidimensional space. So efficiency depends on a large number of draws. Furthermore, simulation smoother used for drawing $h_t^{(j)}$ from importance density (11) not only increases computational burden but also introduces Monte Carlo noise.

We notice that the importance density $g(h_t|Y_T^+;\theta)$ is Gaussian, or

$$g(h_t|Y_T^+;\theta) \equiv N(\hat{h}_t, V_t) = \frac{1}{\sqrt{(2\pi)^k|V_t|}} \exp\left\{ -\frac{1}{2}(h_t-\hat{h}_t)'V_t^{-1}(h_t-\hat{h}_t)\right\}, \tag{15}$$

where $\hat{h}_t$ and V_t are the smoothed mean and variance from Kalman filter smoother applied to the linear Gaussian state space model (10). Gaussianity implies exponential functional form, based on which we can use Gauss-Hermite (GH) quadrature to accurately approximate the integral in (13). Related techniques are discussed in the main text.

To construct the importance density $g(h_t|Y_T^+;\theta)$ for $t = 1, 2, ..., n$ one starts with a given set of importance sampling parameters, i.e. $\{b_1^+, b_2^+, ..., b_n^+, C_1^+, C_2^+, ..., C_n^+\}$. $g(h_t|Y_T^+;\theta)$ then follows from (15) where we obtain the smoothed mean $\hat{h}_t$ and variance V_t applying Kalman filter smoother to the importance model (10). Consequently, the minimization problem (13) can be approximated by

$$\min_{b_t,C_t} \sum_{j=1}^{S^*} \lambda_t^2(\tilde{h}_{tj}, y_t;\theta)\omega_{tj}, \tag{16}$$

where

$$\lambda_t(\tilde{h}_{tj}, y_t;\theta) = \log p(v_t|\tilde{h}_{tj};\theta) - \log g(y_t^+|\tilde{h}_{tj};\theta) - \text{constant}, \quad \omega_{tj} = \frac{p(v_t|\tilde{h}_{tj};\theta)}{g(y_t^+|\tilde{h}_{tj};\theta)},$$

and where S^* is the total number of GH nodes combinations after pruning, and $\tilde{h}_{tj}$ is constructed from GH nodes

$$\tilde{h}_{tj} = \hat{h}_t + L_t z_j, \quad L_t L_t' = V_t, \quad \text{for } j = 1, 2, ..., S^*.$$

The above shows that matrix L_t from Cholesky decomposition of the smoothed variance V_t serves as a rotation operator to the vector of GH nodes z_j such that correlation among the SV components $\tilde{h}_{tj}$ are taken into account. It follows from (9)

$$g(\tilde{h}_{tj}|Y_T^+;\theta) = \frac{1}{\sqrt{(2\pi)^k|V_t|}} \exp\left\{ -\frac{1}{2}z_j'z_j\right\}, \quad t = 1, 2, ..., n.$$

The linear form of our log Gaussian importance density (11) simplifies the minimization problem (16) to a weighted least square (WLS) procedure which has the vector of dependent variables

$$\mathcal{Y}_t = (\log p(v_t|\tilde{h}_{t1};\theta), \log p(v_t|\tilde{h}_{t2};\theta), ..., \log p(v_t|\tilde{h}_{tS^*};\theta))'.$$

For $j = 1, 2, ..., S^*$, the j^{th} row of the matrix of regressors $\mathcal{X}_t$ is

$$(1, \tilde{h}_{tj,1}, ..., \tilde{h}_{tj,k}, -\frac{1}{2}\tilde{h}_{tj,1}^2, ..., -\frac{1}{2}\tilde{h}_{tj,k}^2, -\tilde{h}_{tj,1}\tilde{h}_{tj,2}, ..., -\tilde{h}_{tj,k-1}\tilde{h}_{tj,k}).$$

Together with a diagonal weighting matrix $\mathcal{W}_t$ whose j^{th} diagonal element equals ω_{tj}, the solution of the WLS procedure is thus given by

$$(\mathcal{X}_t'\mathcal{W}_t\mathcal{X}_t)^{-1}\mathcal{X}_t'\mathcal{W}_t\mathcal{Y}_t. \tag{17}$$

We obtain the minimum in (16) by choosing b_t to be the vector of WLS coefficients associated with linear terms $\tilde{h}_{tj,i}$ and C_t to be the matrix whose diagonal elements equal to WLS coefficients associated with quadratic terms $-\frac{1}{2}\tilde{h}_{tj,i}^2$ and off-diagonal elements equal to quadratic terms $-\tilde{h}_{tj,i}\tilde{h}_{tj,\_i}$, with $i = 1, 2, ..., S^*$ and $\_i \neq i$. The minimization of (16) takes place iteratively in order to find the "optimal" importance density which minimizes the variance of log-weights. Once the updated set of $\{b_{1:t}, C_{1:t}\}$ is obtained via the WLS, it is used as $\{b_{1:t}^+, C_{1:t}^+\}$ in the next iteration. The iterative procedure terminates after some convergence criteria are met.

A pathology in NAIS one may encounter is that C_t in some part of WLS iterations might become negative definite[2] making (9) not well-defined, but Kalman filter smoother can still be applied and a modified Kalman filter simulation smoother is available (Jungbacker and Koopman, 2007).

1.4 Algorithm of simulated ML estimation

In an algorithmic way, the Monte Carlo simulated ML estimation via multivariate NAIS carries out the following steps:

1. *Set $m = 1$. Apply Kalman filter and prediction error decomposition to model (1) without SV. Form the initial set of importance parameters b_t each of which equals the estimated constant volatility and C_t is set to identity matrix for all $t = 1, 2, ..., n$.*

2. *Construct approximating linear Gaussian model according to (10) using $\{b_t^{(m)}, C_t^{(m)}\}_{t=1}^T$. Apply Kalman filter and smoother to obtain $\hat{h}_t$ and V_t. For each t, construct a vector of auxiliary dependent variables $\mathcal{Y}_t$ and a matrix of regressors $\mathcal{X}_t$ via GH nodes and Cholesky decomposition as described in the previous section.*

3. *Together with the weighting matrix $\mathcal{W}_t$[3], calculate the WLS solution given by (17), for $t = 1, ..., T$. Collect all estimated WLS coefficients in $\{b_t^{(m+1)}, C_t^{(m+1)}\}_{t=1}^T$. Set $m = m+1$. If $\{b_t^{(m)}, C_t^{(m)}\}_{t=1}^T$ is close to $\{b_t^{(m-1)}, C_t^{(m-1)}\}_{t=1}^T$ pointwise, go to the next step; otherwise go to step 2.*

4. *Apply prediction error decomposition to the linear Gaussian importance model (10) and calculate its log-likelihood $logg(Y_T; \theta)$. Apply a simulation smoother to draw M samples from $g(H_T|Y_T; \theta)$ with a fixed random seed.*

5. *Calculate the importance weights $\omega^{(j)} = \exp(logp(Y_T|H_T^{(j)}; \theta) - logg(Y_T|H_T^{(j)}; \theta))$ for $j = 1, ..., M$, the sample mean $\bar{\omega}$ and the sample variance s_ω^2. Compute the Monte Carlo estimate of log-likelihood $\hat{l}(Y_T; \theta)$ given by (5).*

6. *Maximize $\hat{l}(Y_n; \theta)$ w.r.t. the parameter vector θ using a numerical optimizer such as L-BFGS.*

The above procedure is numerically stable and all the computational results in the main text are obtained using the object-oriented matrix programming language Ox 7.00 (see Doornik, 2009) with the state space functions in the Ox library SsfPack by Koopman et al. (1999) on a quad-core computer. Besides the use of matrix programming language, a multicore processor makes

[2]This happens rarely. In the empirical study in Section 5 of the main text, 4 out of 695 data points lead to negative definite covariance matrix, harmless to log-weights variance minimization.

[3]To increase numerical stability, Richard and Zhang (2007) proposed to set the diagonal elements of the diagonal weighting matrix $\mathcal{W}_t$ or ω_{tj} in (16) to be $\exp(\frac{1}{2}z_j'z_j)$ for $j = 1, ..., S^*$. We can also adopt this simplification.

parallel computing feasible and brings further computational gain in several steps of the proposed estimation procedure. Firstly, because in each iteration, the WLS procedure in Step 3 is done for each $t = 1, ..., T$ separately, the T least square regressions can be done in parallel. Lastly, the L-BFGS algorithm adopted by Ox 7.00 automatically calculates directional derivatives of the objective function $\hat{l}(Y_T; \theta)$ in parallel. This is in stark contrast to Bayesian approach such as Shephard (2015)'s particle MCMC method, because an MCMC chain is recursive by nature rendering parallelization difficult.

1.5 Inference via importance sampling

An importance sampling estimator of $\mathbb{E}(h(x))$ with $x \sim p(x)$[4] and $h(.) \in \mathcal{L}^1$ follows from

$$\frac{\frac{1}{M}\sum_{j=1}^{M}\frac{p(x^{(j)})}{g(x^{(j)})}h(x^{(j)})}{\frac{1}{M}\sum_{j=1}^{M}\frac{p(x^{(j)})}{g(x^{(j)})}} = \frac{\sum_{j=1}^{M}\omega(x^{(j)})h(x^{(j)})}{\sum_{j=1}^{M}\omega(x^{(j)})} \xrightarrow{p} \mathbb{E}(h(x)), \quad M \to \infty,$$

where $\omega(x^{(j)}) = p(x^{(j)})/g(x^{(j)})$, $j = 1, ..., M$ is the importance weight. Convergence holds under the *weak law of large numbers* when $x^{(j)}$, $j = 1, ..., M$ are sampled from the importance density $g(.)$ independently. In Bayesian settings as comparison, samples drawn based on MCMC approaches are dependent, as a result convergence holds under *ergodic law of large numbers*.

In the Monte Carlo study and empirical study of main text, we give smoothed mean estimate of latent processes including SV $\exp(h_t/2)$ and stochastic level or trend μ_t with confidence bands. Inference is valid if $\omega(x^{(j)})$ or $\omega^{(j)}$ has a finite variance (Geweke, 1989). Firstly, we calculate the normalised weights by $\omega^j = \frac{\omega^{(j)}}{\sum_{j=1}^{M}\omega^{(j)}}$. Let $E_Y(.)$ and $V_Y(.)$ denote the smoothed estimate of the mean and variance, respectively, e.g. $E_Y(.) = E(.|Y_T; \hat{\theta})$; and let $H_T^{(j)}$ collect the j-th draw of all SV processes from the importance density. It follows that for the SV process we have

$$E_Y[\exp(h_t/2)] = \sum_{j=1}^{M}\omega^j \exp(h_t^{(j)}/2), \quad V_Y[\exp(h_t/2)] = \sum_{j=1}^{M}\omega^j \exp(h_t^{(j)}) - \left(E_Y(\exp(h_t^{(j)}/2))\right)^2,$$

where all entities are evaluated at the simulated ML estimate $\hat{\theta}$. For a dynamic component z_t, such as a trend or seasonal component, we have

$$\begin{aligned} E_Y(z_t) &= \sum_{j=1}^{M}\omega^j E_p(z_t|H_T^{(j)}), \\ V_Y(z_t) &= \sum_{j=1}^{M}\omega^j V_p(z_t|H_T^{(j)}) + \sum_{j=1}^{M}\omega^j \left(E_p(z_t|H_T^{(j)})\right)^2 - \left(E_Y(z_t)\right)^2, \end{aligned}$$

where $E_p(z_t|H_T^{(j)})$ and $V_p(z_t|H_T^{(j)})$ are the smoothed mean and variance of z_t derived from the Kalman smoother applied to the UCSV model conditional on $H_T^{(j)}$. The last equation comes from the *law of total variance*.

1.6 Filtering via particle filter

Smoothing delivers estimates of SV and dynamic components conditional on all observations. It is however important to have estimates of SV and states at time t conditional on information up to time t, i.e. the filtered estimates, which we use to produce point and density forecast and calculate prediction errors for diagnostics.

[4] It suffices to know the density function $p(.)$ up to an integrating constant, i.e. the density kernel.

Any particle filter methods can be used to find filtered estimates of the SV h_t and the state α_t, $t = 1, ..., T$, keeping system parameters at their ML estimates. For detailed discussions on particle filter and sequential Monte Carlo method, readers can refer to Doucet et al. (2001).

In the following, we outline the bootstrap filter algorithm for the state space models with SV (1)[5]. Suppose we have a particle system at time t

$$h_t^{(j)} = [h_t^{y,1(j)}, h_t^{y,2(j)}, \cdots, h_t^{y,p(j)}, h_t^{\alpha,1(j)}, h_t^{\alpha,2(j)}, \cdots, h_t^{\alpha,m(j)}]', \qquad j = 1, 2, ..., M,$$

where M is a preset number of forward particle draws at each $t = 1, 2, ..., n$. The particle filter procedure is carried out conditional on the system parameter θ which is set to its ML estimate obtained via the proposed simulated ML method. Algorithmically, one carries out the following.

1. Set $t = 1$, draw $h_1^{(j)}$ based on the initialisation discussed in Section 1, or from an *ad hoc* chosen prior $p_0(h_1)$. Set $a_1^{(j)}$ and $P_1^{(j)}$ to their initialised values. For $j = 1, 2, ...M$, sample $h_2^{(j)} \sim p(h_2|h_1^{(j)}; \theta)$ and, compute $a_2^{(j)}$ and $P_2^{(j)}$ according to (7), and set log-weights $\omega_t^{(j)} = 0$.

2. Set $t = t + 1$. For $j = 1, ..., M$, form $H_t^{(j)}$ and $Q_t^{(j)}$ based on $h_t^{(j)}$. Calculate $v_t^{(j)}$ and $F_t^{(j)}$ according to (7). Record

$$l_t^{(j)} = -\frac{1}{2}\big(\log(|F_t^{(j)}|) + v_t^{(j)'} F_t^{(j)} v_t^{(j)}\big).$$

   Update the log-weights $\omega_{t+1}^{(j)} = \omega_t^{(j)} + l_t^{(j)}$ and compute $a_{t+1}^{(j)}$ and $P_{t+1}^{(j)}$ according to (7). Then simulate the forward particle state $h_{t+1}^{(j)} \sim p(h_{t+1}|h_t^{(j)}; \theta)$.

3. For $j = 1, 2, ...M$ compute the normalized weights $\omega_t^{*(j)} = \exp(\omega_t^{(j)}) / \sum_{j=1}^{M} \exp(\omega_t^{(j)})$. Record the particle filter estimates of the log-likelihood contribution

$$\log \hat{L}(y_t|Y_{t-1}; \theta) = \log\left(\sum_{j=1}^{M} \omega_t^{*(j)} \exp(l_t^{(j)})\right), \tag{18}$$

   and record the following particle filter estimates of $f(\beta_t)$,

$$\sum_{j=1}^{M} \omega_t^{*(j)} f(\beta_t^{(j)}), \tag{19}$$

   where f is a measurable function[6], and β_t can be the SV or the states. So depending on f, the above can give the filtered estimate of mean, variance, quantiles, forecasts, standardised prediction errors and other functions of interest. So the residual or one-step prediction error is difference between y_t and the particle filter estimate of the mean component.

4. Check efficient sample size and resample with replacement from the following set of triples,

$$\{h_t^{(1)}, a_t^{(1)}, P_t^{(1)}\}, \{h_t^{(2)}, a_t^{(2)}, P_t^{(2)}\}, ..., \{h_t^{(M)}, a_t^{(M)}, P_t^{(M)}\},$$

   with probability $\omega_t^{*(1)}, \omega_t^{*(2)}, ..., \omega_t^{*(M)}$. Set $\omega_t^{(j)} = 0$ for $j = 1, 2, ..., M$.

[5] Scharth and Kohn (2016) develop a highly efficient particle filter algorithm using the EIS importance density. We can also incorporate the NAIS importance density in the particle filter using their sequential method. It can be shown that their importance sampler is identical to ours when the state transition of SV is linear and Gaussian. But their method replies on simulation to construct the importance density, whereas ours uses high-precision Gauss-Hermite numerical integration, which is fast and stable.

[6] We need $f(.) \in \mathcal{L}^p$ to have a consistent estimator of the p-th moment.

5. Go to 2.

Finally, the particle filter estimate of the data likelihood function is given by

$$\hat{L}(Y_T;\theta) = \exp\left(\sum_{t=2}^{n} \log \hat{L}(y_t|y_{1:t-1};\theta)\right).$$

Though the above is the basic particle filtering algorithm, namely the bootstrap filter, the main text has compared the simulated maximum likelihood method to several particle filters in literature in terms of evaluating the unbiased estimate of the intractable likelihood function $\hat{L}(Y_T;\hat{\theta})$ with the parameter vector evaluated at the SML estimates (*i.e.* particle Monte Carlo Markov chain, see for example Andrieu et al., 2010). Those alternative particle filters are designed for making the basic algorithm more efficient in terms of variance of the likelihood estimates, including the auxiliary particle filter of Pitt and Shephard (1999) and the tempered particle filter of Herbst and Schorfheide (2017). In addition to the results in main text, Figure 1 shows the distribution of log-likelihood used for computing the Monte Carlo estimate.

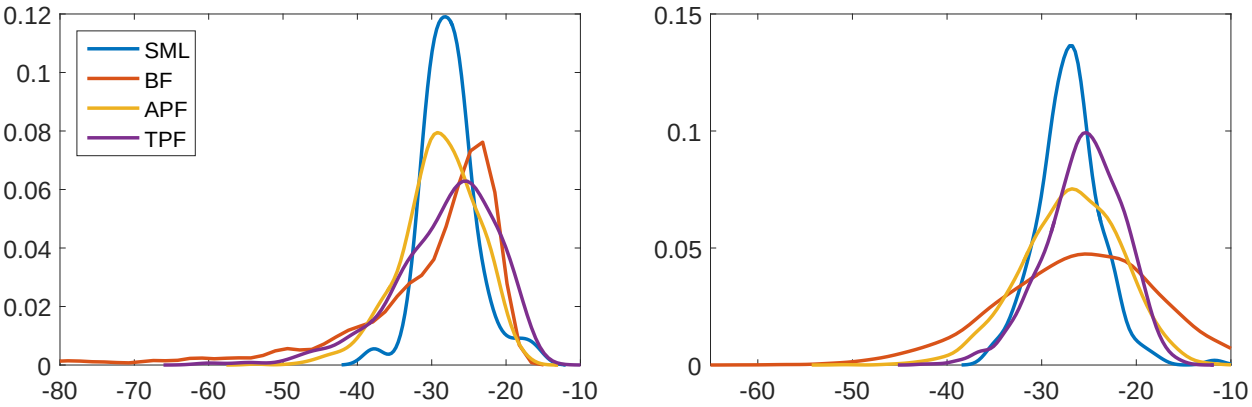

Figure 1: Distribution of Log-Likelihood Estimates. Left: Distribution of $\hat{L}(Y_T;\hat{\theta})$ estimated by SML with $M = 50$ simulations and particle methods with $M = 1000$ particles. Right: Distribution of $\hat{L}(Y_T;\hat{\theta})$ estimated by SML with $M = 200$ simulations and particle methods with $M = 5000$ particles.

2 Three multivariate models

In the application of US quarterly inflation, we have considered three multivariate unobserved components models with stochastic volatility: the factor model MUCSV-SW of Stock and Watson (2016) and a stochastic equicorrelation model MUCSV-EC and a dynamic factor model MUCSV-DF proposed by us. In these models, the dimension k of h_t is 36, rendering direct construction of importance density difficult. We outline the two-step procedure of constructing importance density for MUCSV-SW and MUCSV-EC used in the main text as an illustration.

In the first step, we run the UCSV model of Stock and Watson (2007) using our SML method series by series. Thus we obtain 18 estimated series of trends $\pi_{i,t}$ and cycles $\psi_{i,t}$, $i = 1, ..., 18$. Now we have the following sing-factor model with SV for the trends

$$\begin{pmatrix} \pi_{1,t} \\ \vdots \\ \pi_{18,t} \end{pmatrix} = \begin{pmatrix} a_1 \\ \vdots \\ a_{18} \end{pmatrix} \pi_{c,t} + \begin{pmatrix} \pi^*_{1,t} \\ \vdots \\ \pi^*_{18,t} \end{pmatrix}, \tag{20}$$

where $\pi_{c,t}$ and $\pi^*_{i,t}$, $i = 1, ..., 18$, are independent random walks with SV. Namely, denoting $z_t = (\pi_{c,t}, \pi^*_{1,t}, ..., \pi^*_{18,t})'$, we have

$$z_{j,t+1} = z_{j,t} + \exp\frac{h_{j,t}}{2}\eta_{j,t}, \qquad h_{j,t+1} = h_{j,t} + \sigma_j \zeta_{j,t},$$

where $\eta_{j,t}$ and $\zeta_{j,t}$ are i.i.d. standard normal variates, for $j = 1, ..., 18$ and all t. The second step is to write model (20) as an equivalent univariate model; that is

$$\begin{aligned}
&\pi_{i,t} = a_i z_{1,t} + z_{i+1,t}, && i = 1, ..., 18, \quad t = 1, ..., T, \\
&z_{i+1,t+1} = z_{i+1,t} + \exp\frac{h_{i+1,t}}{2}\eta_{i+1,t}, && \\
&z_{1,t} = z_{1,t}, && i = 1, ..., 17, \\
&z_{1,t+1} = z_{1,t} + \exp\frac{h_{1,t}}{2}\eta_{1,t}, && i = 18,
\end{aligned} \tag{21}$$

which is a univariate UCSV model with $18T$ observations. Building importance density for this model simply replies on a two-dimensional Gauss-Hermite grid. Similarly, we apply this univariate treatment to $\psi_{i,t}$, $i = c, 1, ..., 18$. This "scalablity" is quite general in multivariate UCSV models, and can be incorporated in our SML method straightforwardly.

For MUCSV-EC, two-step procedure is implemented as follows. Firstly, we run the standard UCSV model series by series as in MUCSV-SW, and collect filtered estimate of standardised cycle components $\psi_t^* = (\psi_{1,t}^*, ..., \psi_{18,t}^*)$ with

$$\psi_{i,t}^* = E(\psi_{i,t} e^{-\frac{h_{i,t}}{2}} | Y_t), \qquad i = 1, ..., 18.$$

This is modelled by a stochastic equicorrelation model; that is

$$\begin{aligned}
&\psi_t^* \sim N(0, R_t), && R_t = (1-\rho_t) I_{18} + \rho_t \mathbf{1}_{18}\mathbf{1}_{18}', \\
&\rho_t = \frac{1 - \exp(l_t)/17}{1 + \exp l_t}, && l_{t+1} = l_t + \sigma_l \zeta_t,
\end{aligned} \tag{22}$$

for $t = 1, ..., T$; ζ_t is standard normal variate and the link function of ρ_t is such that $\rho_t \in (-1/17, 1)$ and ensures positive definiteness of R_t. Construction of importance density for l_t (thus ρ_t) can be carried out using the least square algorithm as in the SML method. The "dependent variable" in the regression is given by

$$\log p(\psi_t^* | l_t) = -\frac{1}{2}\left(18\log 2\pi + 17\log(1-\rho_t) + \log(1+17\rho_t) + \frac{\psi_t^*\left[I_{18} - \frac{\rho_t}{1+17\rho_t}\mathbf{1}_{18}\mathbf{1}_{18}'\right]\psi_t^{*\prime}}{1-\rho_t}\right),$$

where $\rho_t = (1-\exp(l_t)/17)/(1+\exp l_t)$. The construction of importance density for the stochastic equicorrelation among trends is done similarly.

3 Data used for MUCSV models

The sectoral inflation series we use is summarised in Table 1. Estimated permanent volatility and transitory volatility for each series from MUCSV-SW, including the common ones, are illustrated in Figure 2 and 3, respectively. Estimated stochastic equicorrelation among trend components and among cycle components from MUCSV-EC are shown in Figure 4.

Table 1: Sectoral Inflation Used in MUCSV-SW And MUC-SEC

Sector	Acronym
Durable goods	
Motor vehicles and parts	MVP
Furnishings and durable household equipment	FRHE
Recreational goods and vehicles	RGV
Other durable goods	ODG
Nondurable goods	
Food and beverages purchased for off-premises consumption	FBOPC
Clothing and footwear	CF
Gsoline and other energy goods	GOEG
Other nondurable goods	ONDG
Services	
Housing and utilities	
Household consumption expenditures (for services)	HCES
Housing and utilities	HU
Health care	HC
Transportation services	TS
Recreational services	RS
Food services and accommodations	FSA
Financial services and insurance	TSI
Other services	OS
Final consumption expenditures of nonprofit institutions serving households	NPISH

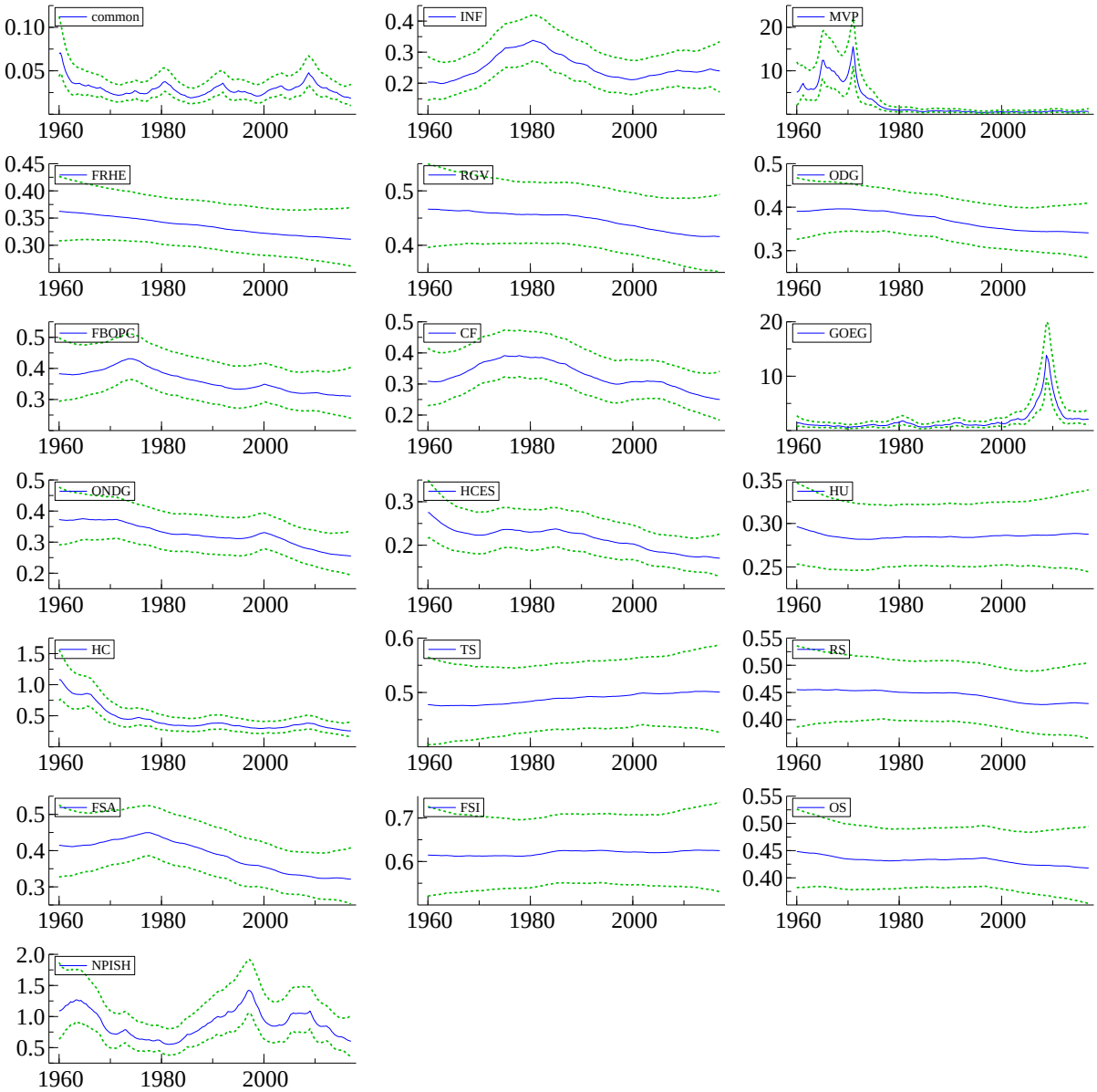

Figure 2: MUCSV-SW permanent volatility

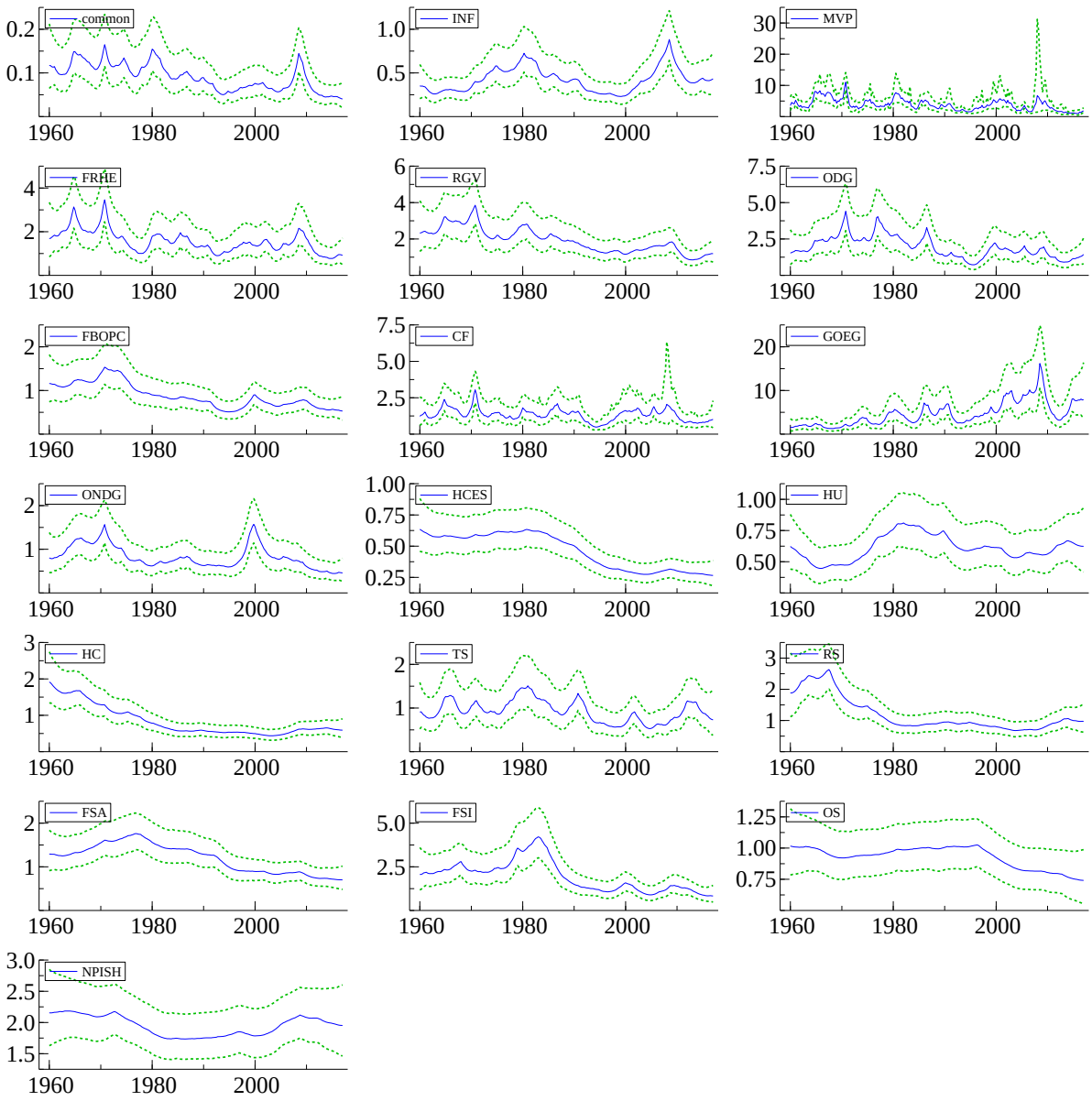

Figure 3: MUCSV-SW transitory volatility

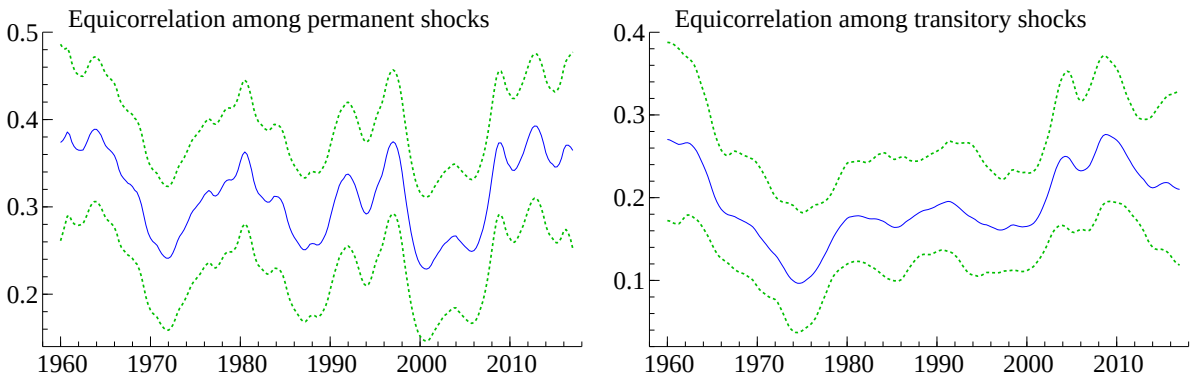

Figure 4: MUCSV-EC Stochastic equicorrelation among permanent and transitory shocks

References

Andrieu, C., Doucet, A., and Holenstein, R. (2010). Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(3):269–342.

De Jong, P. and Shephard, N. (1995). The simulation smoother for time series models. *Biometrika*, 82(2):339–350.

Doornik, J. A. (2009). An object-oriented matrix programming language Ox 6.0.

Doucet, A., Freitas, N. d., and Gordon, N., editors (2001). *Sequential Monte Carlo Methods in Practice (Statistics for Engineering and Information Science)*. Springer.

Durbin, J. and Koopman, S. J. (1997). Monte Carlo maximum likelihood estimation for non-Gaussian state space models. *Biometrika*, 84(3):669–684.

Durbin, J. and Koopman, S. J. (2002). A simple and efficient simulation smoother for state space time series analysis. *Biometrika*, 89(3):603–616.

Durbin, J. and Koopman, S. J. (2012). *Time series analysis by state space methods*. Number 38. Oxford University Press.

Geweke, J. (1989). Bayesian inference in econometric models using Monte Carlo integration. *Econometrica: Journal of the Econometric Society*, pages 1317–1339.

Herbst, E. and Schorfheide, F. (2017). Tempered particle filtering. *National Bureau of Economic Research Working Paper*, 23448.

Jungbacker, B. and Koopman, S. J. (2007). Monte Carlo estimation for nonlinear non-Gaussian state space models. *Biometrika*, 94(4):827–839.

Koopman, S. J., Lit, R., and Nguyen, T. M. (2012). Modified efficient importance sampling using state space methods. *Tinbergen Insititue Discussion Paper*, 008(4).

Koopman, S. J., Lucas, A., and Scharth, M. (2015). Numerically accelerated importance sampling for nonlinear non-Gaussian state-space models. *Journal of Business & Economic Statistics*, 33(1):114–127.

Koopman, S. J., Shephard, N., and Doornik, J. A. (1999). Statistical algorithms for models in state space using SsfPack 2.2. *The Econometrics Journal*, 2(1):107–160.

Li, M. and Koopman, S. J. (2019). Unobserved components with stochastic volatility: Likelihood-based estimation and signal extraction. *Unpublished Manuscript, VU University Amsterdam*.

Pitt, M. K. and Shephard, N. (1999). Filtering via simulation: Auxiliary particle filters. *Journal of the American statistical association*, 94(446):590–599.

Richard, J.-F. and Zhang, W. (2007). Efficient high-dimensional importance sampling. *Journal of Econometrics*, 141(2):1385–1411.

Scharth, M. and Kohn, R. (2016). Particle efficient importance sampling. *Journal of Econometrics*, 190(1):133–147.

Shephard, N. (2015). Martingale unobserved component models. In Koopman, S. J. and Shephard, N., editors, *Unobserved Components and Time Series Econometrics*, page Chapter 10. Oxford University Press.

Shephard, N. and Pitt, M. K. (1997). Likelihood analysis of non-Gaussian measurement time series. *Biometrika*, 84(3):653–667.

Stock, J. H. and Watson, M. W. (2007). Why has US inflation become harder to forecast? *Journal of Money, Credit and banking*, 39(s1):3–33.

Stock, J. H. and Watson, M. W. (2016). Core inflation and trend inflation. *Review of Economics and Statistics*, 98(4):770–784.