

Online Appendix for: Large Bayesian VARMA*

Joshua C.C. Chan Eric Eisenstat
Australian National University The University of Queensland

Gary Koop
University of Strathclyde

May 2015

1 Overview

This appendix is divided into seven sections labelled A through G. Almost all details of the prior are specified in the paper itself, but the few remaining details about the prior are given in Appendix A. An outline of the MCMC algorithm was provided in the paper, but complete details and formulae are provided in Appendix B. Appendix C describes how we calculate the DIC which is one of the methods of model comparison used in our empirical section. Appendix D provides details about the transformation of the expanded form VARMA parameters to other ways of parameterizing the VARMA. Appendix E presents results on the performance of our algorithms in a variety of artificial data sets. Appendix F presents additional empirical results for our macroeconomic application.

2 Appendix A: Priors

The empirical work in this paper uses relatively noninformative priors. The hierarchical SSVS priors for the VARMA coefficients are described in Section 3 of the paper. Recall that in terms of these, we specify uniform priors on the Kronecker indices $\boldsymbol{\kappa}$. Moreover, for both the hard and soft SSVS priors, we set $\tau_{1,j,ik}^2 = 1$; for soft SSVS we set $\tau_{0,j,ik}^2 = 0.01$.

The remaining parameters are assigned the following priors:

$$\begin{aligned}\Lambda_{ii} &\sim \mathcal{IG}(\nu_{\lambda,0}, S_{\lambda,0}), \\ \Omega_{ii} &\sim \mathcal{IG}(\nu_{\omega,0}, S_{\omega,0}).\end{aligned}$$

We set $\nu_{\lambda,0} = 0$, $S_{\lambda,0} = 0.1$, which implies an improper prior on Λ_{ii} , and $\nu_{\omega,0} = 5$, $S_{\omega,0} = 0.4$, resulting in a weakly informative prior on Ω_{ii} .

*Gary Koop is a Senior Fellow at the Rimini Center for Economic Analysis. Emails: joshuacc.chan@gmail.com, eric.eisenstat@gmail.com and gary.koop@strath.ac.uk

3 Appendix B: MCMC Algorithm

We write the model as

$$\mathbf{y}_t = \mathbf{B}\mathbf{X}_t + \mathbf{\Phi}\mathbf{F}_t + \boldsymbol{\eta}_t, \quad (1)$$

where $\mathbf{B} = (\mathbf{I}_n - \mathbf{B}_0, \mathbf{B}_1, \dots, \mathbf{B}_p)$, $\mathbf{\Phi} = (\mathbf{\Phi}_0, \dots, \mathbf{\Phi}_p)$, $\mathbf{X}_t = (\mathbf{y}'_t, \dots, \mathbf{y}'_{t-p})'$ and $\mathbf{F}_t = (\mathbf{f}'_t, \dots, \mathbf{f}'_{t-p})'$. Note that this nests both the expanded and echelon form VARMA. For notational convenience, define the vector of row degrees $\mathbf{p} = (p_1, \dots, p_n)'$. The model parameters are sampled using a Gibbs sampler consisting of the following steps:

1. Sample $(\mathbf{p} | \boldsymbol{\gamma}^S, \mathbf{f}, \boldsymbol{\Lambda})$ marginal of $\mathbf{B}, \mathbf{\Phi}$ and compute $\boldsymbol{\gamma}^R$ as $\gamma_{j,ik}^{B,R} = \gamma_{j,ik}^{\Phi,R} = 1$ iff $0 < j \leq p_i$ or $j = 0, i < k$. This is done with a multi-move sampler that draws $(p_i | \mathbf{p}_{-i}, \boldsymbol{\gamma}^S, \mathbf{f}, \boldsymbol{\Lambda})$ for each $i = 1, \dots, n$. For the exact echelon algorithm set $\kappa_i = p_i$. To sample p_i , we compute the weights $\mathbb{P}(p_i = l | \cdot)$ using the conditional *collapsed* likelihood $p(\mathbf{y}_i | \mathbf{p}^l, \boldsymbol{\gamma}^S, \mathbf{f}, \Lambda_{ii})$ where $\mathbf{p}^l = (p_1, \dots, l, \dots, p_n)$.

To evaluate each conditional likelihood, observe that conditional on \mathbf{f} , the model maybe written as n independent regressions. Let $\mathbf{X} = (\mathbf{X}_1, \dots, \mathbf{X}_T)'$, $\mathbf{F} = (\mathbf{F}_1, \dots, \mathbf{F}_T)'$ and set $\mathbf{W} = (\mathbf{X}, \mathbf{F})$. Then,

$$\mathbf{y}_i = \mathbf{W}\boldsymbol{\delta}_i + \boldsymbol{\eta}_i, \quad \boldsymbol{\eta}_i \sim \mathcal{N}(\mathbf{0}, \Lambda_{ii}\mathbf{I}_T), \quad (2)$$

where $\mathbf{y}_i = (y_{i,1}, \dots, y_{i,T})'$ and $\boldsymbol{\delta}_i$ is the i -th column of $(\mathbf{B}, \mathbf{\Phi})'$.

Now, a given set of indicators $\boldsymbol{\gamma}^{R,l} = \{\gamma_{j,ik}^{B,R,l}, \gamma_{j,ik}^{\Phi,R,l}\} = \mathcal{R}(p_1, \dots, l, \dots, p_n)$ will force certain elements in $\boldsymbol{\delta}_i$ to be zero. Define $\boldsymbol{\delta}_i^*$ to be the vector containing only the *free* elements of $\boldsymbol{\delta}_i$ and \mathbf{W}_i^* the matrix \mathbf{W} with column \mathbf{W}_k removed for any $\delta_{i,k} = 0$. Clearly, $\mathbf{W}\boldsymbol{\delta} = \mathbf{W}_i^*\boldsymbol{\delta}_i^*$ and

$$(\boldsymbol{\delta}_i^* | \boldsymbol{\gamma}^S) \sim \mathcal{N}(\mathbf{0}, \mathbf{V}_{\delta_i,0}^*),$$

if “soft” SSVS priors are specified on $B_{j,ik}, \Phi_{j,ik}$. In this case, $\mathbf{V}_{\delta_i,0}^*$ is a diagonal matrix with element $V_{\delta_i,0, ll}^* = \tau_{0,j,ik}^2$ (i.e. the “small” variance) if $\delta_{i,l}^*$ corresponds to either $B_{j,ik}$ with $\gamma_{j,ik}^{B,S} = 0$ or to $\Phi_{j,ik}$ with $\gamma_{j,ik}^{\Phi,S} = 0$. Otherwise, $V_{\delta_i,0, ll}^* = \tau_{1,j,ik}^2$ (i.e. the “large” variance).

It is straightforward to show in this case that

$$\begin{aligned} (\mathbf{y}_i | \mathbf{p}^l, \boldsymbol{\gamma}^S, \mathbf{f}, \Lambda_{ii}) &\sim \mathcal{N}(\mathbf{0}, \hat{\mathbf{V}}_{y_i}), \\ \hat{\mathbf{V}}_{y_i} &= \left(\Lambda_{ii}^{-1}\mathbf{I}_T - \Lambda_{ii}^{-2}\mathbf{W}_i^*\hat{\boldsymbol{\Delta}}_i^{-1}\mathbf{W}_i^{*'} \right)^{-1}, \\ \hat{\boldsymbol{\Delta}}_i &= \mathbf{V}_{\delta_i,0}^{*-1} + \Lambda_{ii}^{-1}\mathbf{W}_i^{*'}\mathbf{W}_i^*. \end{aligned} \quad (3)$$

The quadratic term $\mathbf{y}_i'\hat{\mathbf{V}}_{y_i}^{-1}\mathbf{y}_i$ is easy to evaluate (i.e. without the need to separately compute the inverse of $\hat{\mathbf{V}}_{y_i}$), as well as the determinant $|\hat{\mathbf{V}}_{y_i}| = \Lambda_{ii}^T |\mathbf{V}_{\delta_i,0}^*| |\hat{\boldsymbol{\Delta}}_i|$. Therefore, computing the likelihood ratio in (3) entails little computation difficulty.

To evaluate the likelihood ratio under the “hard” SSVS prior, define \mathbf{W}_i° as the matrix \mathbf{W}_i^* with the l -th column removed for every $\delta_{i,l}^*$ that corresponds to either

$B_{j,ik}$ with $\gamma_{j,ik}^{B,S} = 0$ or to $\Phi_{j,ik}$ with $\gamma_{j,ik}^{\Phi,S} = 0$. Also, let $\mathbf{V}_{\delta_i,0}^\circ$ be the prior covariance for the unrestricted elements in $\boldsymbol{\delta}_i$. The conditional likelihood is now

$$\begin{aligned} (\mathbf{y}_i | \mathbf{p}^l, \boldsymbol{\gamma}^S, \mathbf{f}, \Lambda_{ii}) &\sim \mathcal{N}(\mathbf{0}, \hat{\mathbf{V}}_{y_i}), \\ \hat{\mathbf{V}}_{y_i} &= \left(\Lambda_{ii}^{-1} \mathbf{I}_T - \Lambda_{ii}^{-2} \mathbf{W}_i^\circ \hat{\boldsymbol{\Delta}}_i^{-1} \mathbf{W}_i^{\circ'} \right)^{-1}, \\ \hat{\boldsymbol{\Delta}}_i &= \mathbf{V}_{\delta_i,0}^{\circ-1} + \Lambda_{ii}^{-1} \mathbf{W}_i^{\circ'} \mathbf{W}_i^\circ, \end{aligned} \quad (4)$$

and computation is similarly straightforward.

Now, to enforce the echelon form in an exact manner, we need to take into account the prior in (7) in the main text. Practically, this means computing the indicators $\boldsymbol{\gamma}^{E,l} = \{\gamma_{j,ik}^{B,E,l}, \gamma_{j,ik}^{\Phi,E,l}\} = \mathcal{E}(p_1, \dots, l, \dots, p_n)$ and setting the weights:

$$\mathbb{P}(p_i = l | \cdot) \propto \begin{cases} 0 & \text{if } \gamma_{j,ik}^{B,E,l} = 0, \gamma_{j,ik}^{B,R,l} \neq 0, \gamma_{j,ik}^{B,S} \neq 0 \text{ for any } i, j, k \\ p(\mathbf{y}_i | \mathbf{p}^l, \boldsymbol{\gamma}^S, \mathbf{f}, \Lambda_{ii}) & \text{otherwise} \end{cases}. \quad (5)$$

For the approximate row degree algorithm, however, the above step is skipped and we simply set:

$$\mathbb{P}(p_i = l | \cdot) \propto p(\mathbf{y}_i | \mathbf{p}^l, \boldsymbol{\gamma}^S, \mathbf{f}, \Lambda_{ii}). \quad (6)$$

Observe that in this case, not only do we circumvent the need to compute echelon form indicators and check them against the row degree and SSVS indicators, but also p_i is conditionally independent of the other row degrees \mathbf{p}_{-i} . This algorithm, therefore, is both computationally simpler and more efficient (albeit at the cost of loosing the exact canonical form).

2. Sample $(\boldsymbol{\gamma}_i^S, \mathbf{B}_i, \boldsymbol{\Phi}_i | \mathbf{p}, \mathbf{f}, \Lambda_{ii}, \mathbf{y}_i)$ for each $i = 1, \dots, n$, where \mathbf{B}_i denotes the i -th row of \mathbf{B} , $\boldsymbol{\Phi}_i$ the i -th row of $\boldsymbol{\Phi}$, and $\boldsymbol{\gamma}_i^S$ is the set of all SSVS indicators pertaining to $\mathbf{B}_i, \boldsymbol{\Phi}_i$. Under the ‘‘hard’’ SSVS prior, this is done by first sampling $(\boldsymbol{\gamma}_i^S, | \mathbf{p}, \mathbf{f}, \Lambda_{ii})$ marginal of $\mathbf{B}_i, \boldsymbol{\Phi}_i$ using (2). In particular, we sample each $\gamma_{j,ik}^{S}$ for every i, j, k conditional on $\{\gamma_{l,im}^{S}\}_{l \neq j, m \neq k}$, using the approach outlined in Step 1 to compute the likelihood ratio

$$\varrho_{j,ik}^{S} = \frac{p(\mathbf{y}_i | \mathbf{p}, \gamma_{j,ik}^{S} = 1, \{\gamma_{l,im}^{S}\}_{l \neq j, m \neq k}, \mathbf{f}, \Lambda_{ii})}{p(\mathbf{y}_i | \mathbf{p}, \gamma_{j,ik}^{S} = 0, \{\gamma_{l,im}^{S}\}_{l \neq j, m \neq k}, \mathbf{f}, \Lambda_{ii})}. \quad (7)$$

Given our priors, this implies

$$\mathbb{P}\left(\gamma_{j,ik}^{\Phi,S} = 1 | \mathbf{p}, \{\gamma_{l,im}^{\Phi,S}\}_{l \neq j, m \neq k}, \mathbf{f}, \Lambda_{ii}, \mathbf{y}_i\right) = \varrho_{j,ik}^{\Phi,S} / \left(1 + \varrho_{j,ik}^{\Phi,S}\right), \quad (8)$$

for both the exact echelon form algorithm and the approximate row degrees algorithm. For the indicators on \mathbf{B}_i , however, imposing the echelon form once again requires that the relationship between $\boldsymbol{\gamma}_i^{B,S}$ and p_1, \dots, p_n established in (7) in the

main text be respected. In consequence, the correct conditional distribution is given by

$$\begin{aligned} & \mathbb{P}\left(\gamma_{j,ik}^{B,S} = 1 \mid \mathbf{p}, \{\gamma_{l,im}^{B,S}\}_{l \neq j, m \neq k}, \mathbf{f}, \Lambda_{ii}, \mathbf{y}_i\right) \\ &= \begin{cases} 0 & \text{if } \gamma_{j,ik}^{B,E} = 0, \gamma_{j,ik}^{B,R} \neq 0 \\ \varrho_{j,ik}^{B,S} / \left(1 + \varrho_{j,ik}^{B,S}\right) & \text{otherwise} \end{cases}, \end{aligned} \quad (9)$$

where $\gamma_{j,ik}^{B,E}$ is computed from $\mathcal{E}(p_1, \dots, p_n)$ using previous draws of p_1, \dots, p_n . For the approximate row degrees algorithm, however, we simply draw from

$$\mathbb{P}\left(\gamma_{j,ik}^{B,S} = 1 \mid \mathbf{p}, \{\gamma_{l,im}^{B,S}\}_{l \neq j, m \neq k}, \mathbf{f}, \Lambda_{ii}, \mathbf{y}_i\right) = \varrho_{j,ik}^{B,S} / \left(1 + \varrho_{j,ik}^{B,S}\right). \quad (10)$$

For sake of efficient computation, we note that whenever $\gamma_{j,ik}^{B,R} = 0$, we always obtain $\varrho_{j,ik}^{B,S} = 1$, and therefore, the conditional likelihoods need not be computed. Instead, $\gamma_{j,ik}^{B,S}$ in this case is sampled from the prior.

When using the “soft” SSVS prior, the indicators are sampled conditional on \mathbf{B}_i, Φ_i . For $\gamma_{j,ik}^{B,S}$, if the echelon form is imposed, then $\mathbb{P}\left(\gamma_{j,ik}^{B,S} = 1 \mid B_{j,ik}\right) = 0$ when $\gamma_{j,ik}^{B,E} = 0$ and $\gamma_{j,ik}^{B,R} \neq 0$; otherwise

$$\mathbb{P}\left(\gamma_{j,ik}^{B,S} = 1 \mid B_{j,ik}\right) = \frac{\frac{1}{\tau_{1,j,ik}} \exp\left(-\frac{B_{j,ik}^2}{2\tau_{1,j,ik}^2}\right)}{\frac{1}{\tau_{1,j,ik}} \exp\left(-\frac{B_{j,ik}^2}{2\tau_{1,j,ik}^2}\right) + \frac{1}{\tau_{0,j,ik}} \exp\left(-\frac{B_{j,ik}^2}{2\tau_{0,j,ik}^2}\right)}. \quad (11)$$

If the row degree algorithm is used, we sample $\gamma_{j,ik}^{B,S}$ using only (11).

For $\gamma_{j,ik}^{\Phi,S}$, the success probabilities are

$$\mathbb{P}\left(\gamma_{j,ik}^{\Phi,S} = 1 \mid \Phi_{j,ik}, \gamma_{j,ik}^{\Phi,R} \neq 0\right) = \frac{\frac{1}{\tau_{1,j,ik}} \exp\left(-\frac{\Phi_{j,ik}^2}{2\tau_{1,j,ik}^2}\right)}{\frac{1}{\tau_{1,j,ik}} \exp\left(-\frac{\Phi_{j,ik}^2}{2\tau_{1,j,ik}^2}\right) + \frac{1}{\tau_{0,j,ik}} \exp\left(-\frac{\Phi_{j,ik}^2}{2\tau_{0,j,ik}^2}\right)}.$$

Once again, $\gamma_{j,ik}^{\Phi,S}$ is drawn from the prior $\mathbb{P}(\gamma_{j,ik}^{\Phi,S} = 1 \mid \gamma_{j,ik}^{\Phi,R} = 0) = 0.5$ whenever $\gamma_{j,ik}^{\Phi,R} = 0$ and hence the corresponding coefficient is excluded by the row degrees.

Given a draw of γ_i^S , the coefficients \mathbf{B}_i, Φ_i are sampled jointly in standard way for both of the SSVS specifications. In particular, letting once again $\mathbf{W}_i^* = (\mathbf{X}^*, \mathbf{F}^*)$ be the reduced regressors and factors matrix corresponding to the unrestricted coefficients δ_i^* in δ_i , textbook regression analysis dictates

$$\begin{aligned} (\delta_i^* \mid \gamma_i^S, p_i, \mathbf{f}, \Lambda_{ii}, \mathbf{y}_i) &\sim \mathcal{N}(\hat{\delta}_i, \hat{\Delta}_i), \\ \hat{\delta}_i &= \hat{\Delta}_i^{-1} \Lambda_{ii}^{-1} \mathbf{W}_i^{*'} (\mathbf{y}_i - \mathbf{f}_i), \\ \hat{\Delta}_i &= \left(\mathbf{V}_{\delta_i,0}^* + \Lambda_{ii}^{-1} \mathbf{W}_i^{*'} \mathbf{W}_i^* \right)^{-1}, \end{aligned} \quad (12)$$

where $\mathbf{f}_i = (f_{i,1}, \dots, f_{i,T})'$. The remaining elements in δ_i (and therefore \mathbf{B}_i, Φ_i) are set to zero.

3. Sample

$$(\Lambda_{ii} | \mathbf{B}_i, \Phi_i, p_i, \gamma_i^S, \mathbf{f}, \mathbf{y}_i) \sim \mathcal{IG} \left(\nu_{\lambda,0} + \frac{T}{2}, S_{\lambda,0} + \frac{1}{2} \sum_{t=1}^T (y_{i,t} - \mathbf{B}_i \mathbf{X}_t - \Phi_i \mathbf{F}_t)^2 \right)$$

for each $i = 1, \dots, n$.

4. Sample $(\Omega_{ii} | \mathbf{f}_i)$ or $(h_{i,0}, \dots, h_{i,T}, \sigma_{h,i}^2 | \mathbf{f}_i)$ —depending on whether stochastic volatility is specified—for each $i = 1, \dots, n$. In either case, standard methods are used.

5. Sample $(\mathbf{f} | \mathbf{B}, \Phi, \tilde{\Omega}, \Lambda, \mathbf{p}, \gamma^S, \mathbf{y})$, where $\tilde{\Omega} = \mathbf{I}_T \otimes \Omega$ for the constant variance case and

$$\tilde{\Omega} = \text{diag}(\exp h_{1,1}, \dots, \exp h_{n,1}, \dots, \exp h_{1,T}, \dots, \exp h_{n,T})$$

for stochastic volatility. An efficient sampler for this purpose is constructed by first rewriting the working model (1) in stacked form as

$$\mathbf{y}^* = \Psi \mathbf{f} + \boldsymbol{\eta}, \quad (13)$$

where $\mathbf{y}^* = ((\mathbf{y}_1 - \mathbf{B}\mathbf{X}_1)', \dots, (\mathbf{y}_T - \mathbf{B}\mathbf{X}_T)')'$ and Ψ is a $Tn \times Tn$ lower triangular matrix with Φ_0 on the main diagonal block, Φ_1 on first lower diagonal block, Φ_2 on second lower diagonal block, and so forth. For example, for $q = 2$, we have

$$\Psi = \begin{pmatrix} \Phi_0 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \Phi_1 & \Phi_0 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \Phi_2 & \Phi_1 & \Phi_0 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \Phi_2 & \Phi_1 & \Phi_0 & \cdots & \mathbf{0} \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \Phi_2 & \Phi_1 & \Phi_0 \end{pmatrix}.$$

It is important to note that in general Ψ is a sparse $Tn \times Tn$ matrix that contains at most

$$n^2 \left((q+1)T - \frac{q(q+1)}{2} \right) < n^2(q+1)T$$

non-zero elements, which grows *linearly* in T and is substantially less than the total $(Tn)^2$ elements for typical applications where $T \gg q$.

Now, the vector of factors is sampled jointly as

$$\begin{aligned} (\mathbf{f} | \mathbf{B}, \Phi, \Omega_{(t)}, \Lambda, \mathbf{p}, \gamma^S, \mathbf{y}) &\sim \mathcal{N}(\hat{\mathbf{f}}, \hat{\mathbf{V}}_f), \\ \hat{\mathbf{f}} &= \hat{\mathbf{V}}_f (\Psi' (\mathbf{I}_T \otimes \Lambda^{-1}) \mathbf{y}^*), \\ \hat{\mathbf{V}}_f &= \left(\tilde{\Omega}^{-1} + \Psi' (\mathbf{I}_T \otimes \Lambda^{-1}) \Psi \right)^{-1}, \end{aligned} \quad (14)$$

which is once again efficiently implemented using sparse matrix routines.

4 Appendix C: Deviance Information Criterion

The Deviance Information Criterion (DIC) was introduced in Spiegelhalter, Best, Carlin, and van der Linde (2002). For latent variable models there are numerous definitions (Celeux, Forbes, Robert, and Titterton, 2006) depending on the exact notion of the likelihood. Given a likelihood function $f(\mathbf{y} | \boldsymbol{\theta})$, the DIC is defined as:

$$\text{DIC} = \overline{D(\boldsymbol{\theta})} + p_D,$$

where

$$\overline{D(\boldsymbol{\theta})} = -2\mathbb{E}_{\boldsymbol{\theta}}[\log f(\mathbf{y} | \boldsymbol{\theta}) | \mathbf{y}]$$

is the posterior mean deviance and p_D is the effective number of parameters. That is, the DIC is the sum of the posterior mean deviance, which can be used as a Bayesian measure of model fit or adequacy, and the effective number of parameters that measures model complexity. The effective number of parameters is in turn defined as

$$p_D = \overline{D(\boldsymbol{\theta})} - D(\tilde{\boldsymbol{\theta}}),$$

where $D(\boldsymbol{\theta}) = -2\log f(\mathbf{y} | \boldsymbol{\theta})$ and $\tilde{\boldsymbol{\theta}}$ is an estimate of $\boldsymbol{\theta}$, which is typically taken as the posterior mean.

Our VARMA model has a few equivalent latent variable representations. Hence, in principle we can use any of the representations and compute the DIC based on the conditional likelihood (i.e., the likelihood given the latent variables). However, as pointed out in Chan and Grant (2014), conditional DICs tend to be numerically unstable. Instead, we use the likelihood implied by the system

$$\mathbf{y}_t = \sum_{j=1}^p \mathbf{A}_j \mathbf{y}_{t-j} + \sum_{j=1}^q \boldsymbol{\Theta}_j \boldsymbol{\epsilon}_{t-j} + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}), \quad (15)$$

where all the parameters are identified and can be recovered from the main sampling algorithm.

To derive this density, we stack (15) over t and obtain:

$$\mathbf{y} = \mathbf{a} + \boldsymbol{\Theta}\boldsymbol{\epsilon},$$

where $\boldsymbol{\epsilon} = (\boldsymbol{\epsilon}'_1, \dots, \boldsymbol{\epsilon}'_T)' \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_T \otimes \boldsymbol{\Sigma})$, $\mathbf{a} = ((\sum_{j=1}^p \mathbf{A}_j \mathbf{y}_{1-j})', \dots, (\sum_{j=1}^p \mathbf{A}_j \mathbf{y}_{T-j})')'$ and $\boldsymbol{\Theta}$ is a $Tn \times Tn$ lower triangular matrix with the identity matrix \mathbf{I}_n on the main diagonal block, $\boldsymbol{\Theta}_1$ on first lower diagonal block, $\boldsymbol{\Theta}_2$ on second lower diagonal block, and so forth. Hence, we have

$$(\mathbf{y} | \mathbf{A}_1, \dots, \mathbf{A}_p, \boldsymbol{\Theta}_1, \dots, \boldsymbol{\Theta}_q, \boldsymbol{\Sigma}) \sim \mathcal{N}(\mathbf{a}, \boldsymbol{\Theta}(\mathbf{I}_T \otimes \boldsymbol{\Sigma})\boldsymbol{\Theta}').$$

Since the covariance matrix $\boldsymbol{\Theta}(\mathbf{I}_T \otimes \boldsymbol{\Sigma})\boldsymbol{\Theta}'$ is a band matrix, this Normal density can be evaluated quickly using the band matrix algorithms discussed in Chan and Grant (2014).

5 Appendix D: Recovering VARMA Parameters from the Expanded Form

In this appendix we describe how to recover the VARMA parameters $\Theta_1, \dots, \Theta_p, \Sigma$ (which appear in the semi-structural VARMA and in the echelon form) from the expanded form parameters $\Phi_0, \dots, \Phi_p, \Omega, \Lambda$. The procedure was introduced in Chan and Eisenstat (2015). We reproduce it for convenience here and refer the interested reader to the aforementioned paper for further details.

Recall that $\mathbf{B}_0, \dots, \mathbf{B}_p$ are the same in expanded and echelon forms. Consequently, the ensuing procedure assumes these coefficients are given. With this in mind, let

$$\begin{aligned} \mathbf{u}_t &\equiv \mathbf{B}_0 \mathbf{y}_t - \mathbf{B}_1 \mathbf{y}_{t-1} - \dots - \mathbf{B}_p \mathbf{y}_{t-p}, \\ \mathbf{u}_t &= \Theta_0 \boldsymbol{\epsilon}_t + \Theta_1 \boldsymbol{\epsilon}_{t-1} + \dots + \Theta_p \boldsymbol{\epsilon}_{t-p}, & \boldsymbol{\epsilon}_t &\sim \mathcal{N}(\mathbf{0}, \Sigma), & (16) \\ &= \Phi_0 \mathbf{f}_t + \Phi_1 \mathbf{f}_{t-1} + \dots + \Phi_p \mathbf{f}_{t-p} + \boldsymbol{\eta}_t, & \mathbf{f}_t &\sim \mathcal{N}(\mathbf{0}, \Omega), & \boldsymbol{\eta}_t &\sim \mathcal{N}(\mathbf{0}, \Lambda). & (17) \end{aligned}$$

For what follows, it is important to emphasize that in the algorithms developed in this paper, Θ_0 is always either estimated or fixed a priori—i.e., $\Theta_0 = \mathbf{B}_0$ or $\Theta_0 = \mathbf{I}$. Define further the quantities:

$$\Gamma_j = \sum_{l=j}^p \Theta_l \Sigma \Theta_{l-j}', \quad j = 0, \dots, p, \quad (18)$$

$$\dot{\Gamma}_j = \sum_{l=j}^p \Phi_l \Omega \Phi_{l-j}' + \mathbf{1}(j=0) \Lambda, \quad j = 0, \dots, p. \quad (19)$$

The mapping between $(\Theta_1, \dots, \Theta_p, \Sigma)$ and $(\Phi_0, \dots, \Phi_p, \Omega, \Lambda)$ is established by equating $\Gamma_j = \dot{\Gamma}_j$ ($\equiv E(\mathbf{u}_t \mathbf{u}_{t-j}')$).

We will start with the simplest case of $p = 1$, where a draw of Θ_1, Σ can be easily recovered given a draw of $\Phi_0, \Phi_1, \Omega, \Lambda$ as follows. Proceed by first constructing $\Gamma_0 \equiv \dot{\Gamma}_0$ and $\Gamma_1 \equiv \dot{\Gamma}_1$ according to (19). Next, for $\Theta_1^* = \Theta_1 \Theta_0^{-1}$ and $\Sigma^* = \Theta_0 \Sigma \Theta_0'$, solve the quadratic matrix equation

$$\Theta_1^{*2} \Gamma_1' - \Theta_1^* \Gamma_0 + \Gamma_1 = \mathbf{0}.$$

Solving this equation is straightforward:

1. Solve the *generalized eigenvalue problem*

$$\begin{pmatrix} \Gamma_0 & -\Gamma_1' \\ \mathbf{I}_n & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{X}_{11} & \mathbf{X}_{12} \\ \mathbf{X}_{21} & \mathbf{X}_{22} \end{pmatrix} = \begin{pmatrix} \Gamma_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_n \end{pmatrix} \begin{pmatrix} \mathbf{X}_{11} & \mathbf{X}_{12} \\ \mathbf{X}_{21} & \mathbf{X}_{22} \end{pmatrix} \begin{pmatrix} \mathbf{D}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_2 \end{pmatrix}.$$

A solution to this will consist of $2n$ eigenvalues and the associated $2n$ eigenvectors.¹ Moreover, the matrices involving eigenvalues and eigenvectors can be rotated freely.

¹In our empirical work, we use the MATLAB command `eig` to efficiently solve the generalized eigenvalue problem.

- Choose \mathbf{D}_1 to contain n of the generated eigenvalues and $(\mathbf{X}'_{11}, \mathbf{X}'_{21})'$ to contain the corresponding eigenvectors. Note that by construction, exactly n of the eigenvalues will be less than one in modulus, so selecting the n smallest (in modulus) eigenvalues corresponds to enforcing invertibility (e.g., ex-post).

3. Compute

$$\begin{aligned}\Theta_1^{*'} &= \mathbf{X}_{11} \mathbf{X}_{21}^{-1}, & \Theta_1 &= \Theta_1^* \Theta_0 \\ \Sigma^* &= \Gamma_0 - \Gamma_1 \Theta_1^{*'} & \Sigma &= \Theta_0^{-1} \Sigma^* (\Theta_0^{-1})' .\end{aligned}$$

To generalize this procedure for any p , it is useful to consider the VMA(1) representation of the VMA(p), defined as

$$\underbrace{\begin{pmatrix} \mathbf{u}_t \\ \mathbf{u}_{t-1} \\ \vdots \\ \mathbf{u}_{t-p+1} \end{pmatrix}}_{\tilde{\mathbf{u}}_\tau} = \underbrace{\begin{pmatrix} \Theta_0 & \Theta_1 & \cdots & \Theta_{p-1} \\ & \ddots & \ddots & \vdots \\ & & \ddots & \Theta_1 \\ & & & \Theta_0 \end{pmatrix}}_{\tilde{\Theta}_0} \underbrace{\begin{pmatrix} \epsilon_t \\ \epsilon_{t-1} \\ \vdots \\ \epsilon_{t-p+1} \end{pmatrix}}_{\tilde{\epsilon}_\tau} + \underbrace{\begin{pmatrix} \Theta_p & & & \\ \Theta_{p-1} & \ddots & & \\ \vdots & \ddots & \ddots & \\ \Theta_1 & \cdots & \Theta_{p-1} & \Theta_p \end{pmatrix}}_{\tilde{\Theta}_1} \underbrace{\begin{pmatrix} \epsilon_{t-p} \\ \epsilon_{t-p-1} \\ \vdots \\ \epsilon_{t-2p+1} \end{pmatrix}}_{\tilde{\epsilon}_{\tau-1}} \quad (20)$$

with $\tilde{\epsilon}_\tau \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_p \otimes \Sigma)$. In this form, the corresponding autovariances may be denoted by

$$\tilde{\Gamma}_0 = \begin{pmatrix} \Gamma_0 & \Gamma_1 & \cdots & \Gamma_{p-1} \\ \Gamma'_1 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \Gamma_1 \\ \Gamma'_{p-1} & \cdots & \Gamma'_1 & \Gamma_0 \end{pmatrix} \quad \text{and} \quad \tilde{\Gamma}_1 = \begin{pmatrix} \Gamma_p \\ \Gamma_{p-1} & \ddots \\ \vdots & \ddots & \ddots \\ \Gamma_1 & \cdots & \Gamma_{p-1} & \Gamma_p \end{pmatrix}. \quad (21)$$

The algorithm used to recover $\Theta_1, \dots, \Theta_p, \Sigma$ becomes the following.

- Compute $\Gamma_0, \dots, \Gamma_p$ from draws of $\Phi_0, \dots, \Phi_p, \Omega, \Lambda$.
- Construct $\tilde{\Gamma}_0, \tilde{\Gamma}_1$ according to (21).
- Use the $p = 1$ procedure described above to compute $\tilde{\Theta}_1^*, \tilde{\Sigma}^*$, where $\tilde{\Theta}_1^* = \tilde{\Theta}_1 \tilde{\Theta}_0^{-1}$, $\tilde{\Sigma}^* = \tilde{\Theta}_0 (\mathbf{I}_p \otimes \Sigma) \tilde{\Theta}_0'$ and $\tilde{\Theta}_0, \tilde{\Theta}_1$ are defined in (20).
- Recover $\Theta_1^*, \dots, \Theta_p^*$ from the first n columns of $\tilde{\Theta}_1^*$ and Σ^* from the bottom-right $n \times n$ block of $\tilde{\Sigma}^*$; set $\Theta_j = \Theta_j^* \Theta_0$ and $\Sigma = \Theta_0^{-1} \Sigma^* (\Theta_0^{-1})'$.

Note again that solving the generalized eigenvalue problem in Step 3 above leads to $2np$ eigenvalues, of which np are less than one in modulus and correspond an invertible VMA(p) system. Thus, the idea that invertibility can be enforced ex-post extends to the general case as well.

6 Appendix E: Results using Artificial Data

In this appendix, we carry out a brief exercise with artificial data to investigate the performance of our algorithm. We focus on the identification issue and present results relating to κ for various versions of our algorithm. All the results in this section involve drawing 10 artificial data sets, each of $T = 100$ observations. Each data set is normalized to have mean zero and unit standard deviation. For each data set, 11,000 MCMC draws are taken and the first 1,000 of these discarded. All results are based on the benchmark prior described in Appendix A. We present results for the algorithm which imposes the echelon form exactly (labelled “echelon” in the tables below) versus the approximate algorithm which works with the row degrees (labelled “row degree” in the tables below). We also investigate the difference between the two different implementations of SSVS methods (labelled “hard SSVS” and “soft SSVS” in the tables) discussed in Section 3 of the paper.

The first set of artificial data exercises uses bivariate VARMA_s based on the example in equation (3) of the paper. Our first data generating process (DGP) is a standard identified VARMA(1,1) with $\kappa_1 = \kappa_2 = 1$. The second DGP is also a VARMA(1,1) but with $\kappa_1 = 1, \kappa_2 = 0$. In both cases, our estimating model is an VARMA(4,4). The starting value for κ in the MCMC algorithm is, throughout this section, always set so as to choose the VARMA(4,4). We are interested in investigating whether our algorithm can, in the context of a greatly over-parameterized model, uncover the parsimonious identified model in each case and, thus, initialize the algorithm at the over-parameterized extreme.

Precise values of the parameters used in the DGPs are:

$$\text{DGP1: } B_{1,11} = 0.7, B_{1,21} = 0.4, B_{1,12} = 0.2, B_{1,22} = 0.5, \Theta_{1,11} = 0.1, \Theta_{1,12} = 0, \Theta_{1,21} = 0.5, \Theta_{1,22} = 0.1, \Sigma = \begin{bmatrix} 0.9 & 0 \\ 0 & 0.1 \end{bmatrix}.$$

$$\text{DGP2: } B_{1,11} = 0.7, B_{1,21} = 0, B_{1,12} = 0.2, B_{1,22} = 0, \Theta_{1,11} = 0.1, \Theta_{1,12} = 0, \Theta_{1,21} = 0, \Theta_{1,22} = 0.0, \Sigma = \begin{bmatrix} 0.9 & 0 \\ 0 & 0.1 \end{bmatrix}.$$

Table 1 presents summary statistics of the various estimates of κ for the two DGPs. It can be seen that, despite working with the over-parameterized VARMA(4,4), our algorithm is accurately choosing the identified VARMA_E(1,1) and VARMA_E(1,0) for DGP₁ and DGP₂, respectively. The cross-data-set averages of κ do tend to be slightly above the true values used in the DGP. In the case of κ_2 in DGP₂, this is of necessity (since the true value of $\kappa_2 = 0$ and κ_2 cannot be negative). For other cases, this is likely due to excessively large lag length used in the estimating model. With regards to the different variants of our algorithms, there seems little difference. In particular, the approximate row degree algorithm is yielding results which are very similar to the exact algorithm which imposes the echelon form at every draw. Overall, though, the results indicate that our algorithms are working well in identifying small VARMA_s. The estimates of the parameters (not reported here) are similar to the true values used in the DGPs and the inefficiency factors for the MCMC algorithm (also not reported here) indicate the algorithms are mixing well.

Table 1: Averages across Data Sets of Posterior Mean of κ . Standard Deviation, Minimum and Maximum in Parentheses.

Algorithm details	DGP ₁		DGP ₂	
	κ_1	κ_2	κ_3	κ_4
True value	1	1	1	0
echelon, hard SSVS	1.35 (0.19) (1.14, 1.68)	1.11 (0.11) (1.02, 1.40)	1.29 (0.06) (1.21, 1.36)	0.48 (0.21) (0.31, 0.99)
row degree, hard SSVS	1.28 (0.13) (1.18, 1.63)	1.24 (0.15) (1.14, 1.63)	1.59 (0.38) (1.31, 2.54)	0.49 (0.21) (0.32, 0.91)
echelon, soft SSVS	1.28 (0.25) (1.13, 1.93)	1.09 (0.05) (1.02, 1.17)	1.30 (0.21) (1.11, 1.85)	0.49 (0.25) (0.23, 1.05)
row degree, soft SSVS	1.23 (0.13) (1.12, 1.47)	1.20 (0.14) (1.12, 1.59)	1.32 (0.21) (1.14, 1.86)	0.42 (0.30) (0.25, 1.15)

But will our algorithms be as capable of uncovering identification restrictions in larger VARMA's? And will they be computationally efficient? These are the questions we address in Tables 2 through 5. Tables 2 and 4 contain results relating to κ comparable to those in Table 1 for larger 7-variate and 12-variate VARMA's. Tables 3 and 5 contain results relating to the efficiency of the MCMC algorithm. For the sake of brevity, inefficiency factors are presented for the impulse responses of the first and second variables to a shock in the third variable four periods in the future. These are labelled "IR₁" and "IR₂" in Tables 3 and 5.

The data generating process for the 7-variate VAR is a VARMA(1,1) with the following parameter values:

DGP₃: $\mathbf{B}_0 = \mathbf{\Theta}_0 = \mathbf{I}$ and $B_{1,ii} = 0.1 \times i$, $\Theta_{1,ii} = 0.1 \times (7 - i)$, $B_{1,12} = B_{1,23} = -0.4$, $\Theta_{1,56} = \Theta_{1,67} = -0.4$ where $B_{1,ik}$ and $\Theta_{1,ik}$ are the (i, k) elements of \mathbf{B}_1 and $\mathbf{\Theta}_1$, respectively. Letting $[\sigma_{ik}]$ be the elements of $\mathbf{\Sigma}$, we set $\sigma_{ii} = 0.1 \times i$, $\sigma_{57} = \sigma_{67} = -0.3$. All elements of \mathbf{B}_1 , $\mathbf{\Theta}_1$ and $\mathbf{\Sigma}$ not specified are set to zero.

Note that DGP₃ has $\kappa_i = 1$ for $i = 1, \dots, 7$ and should be well-identified in the sense that each row of the VARMA has either an AR or an MA coefficient which is substantively different from zero.

The data generating process from the 12-variate VAR is also a VARMA(1,1) with parameter values:

DGP₄: $\mathbf{B}_0 = \mathbf{\Theta}_0 = \mathbf{I}$ and $B_{1,ii} = 0.1 \times i$ for $i = 1, \dots, 8$, $\Theta_{1,ii} = 0.1 \times (12 - i)$ for $i = 1, \dots, 10$, $B_{1,12} = B_{1,23} = -0.4$, $\Theta_{1,56} = \Theta_{1,67} = -0.4$ where $B_{1,ik}$ and $\Theta_{1,ik}$ are the (i, k) elements of \mathbf{B}_1 and $\mathbf{\Theta}_1$, respectively. Letting $[\sigma_{ik}]$ be the elements of $\mathbf{\Sigma}$, we set $\sigma_{ii} = 0.1 \times i$, $\sigma_{57} = \sigma_{67} = -0.3$. All elements of \mathbf{B}_1 , $\mathbf{\Theta}_1$ and $\mathbf{\Sigma}$ not specified are set to zero.

Note that DGP₄ has $\kappa_i = 1$ for $i = 1, \dots, 10$, with $\kappa_{11} = \kappa_{12} = 0$. However, for equations 9 and 10 in the VARMA the identification is quite weak in the sense that both

of these equations have no AR lags and the coefficient on the MA lag is quite small (i.e. $\Theta_{1,99} = 0.3$ and $\Theta_{1,10,10} = 0.2$). Hence, even through the true value $\kappa_9 = \kappa_{10} = 1$, the DGP is quite close to the $\kappa_9 = \kappa_{10} = 0$ case.

Results for the medium-sized 7-variate VARMA are similar to those for the bivariate VARMA. Table 2 indicates the variants of our algorithm are all successfully producing an estimate of $\boldsymbol{\kappa}$ near its true value. For none of the data sets do any of our algorithms go far wrong. Table 3 indicates that the efficiency of our algorithm is fairly good, producing inefficiency factors that are around 10 or 20. However, the inefficiency factors for the echelon form algorithm with hard SSVS are somewhat higher than this. One of the artificial data sets leads to an inefficiency factor of over 300 for one of the impulse responses. Hence, the researcher using our algorithm in VARMA of this size should take care with MCMC convergence issues and would probably be required to take hundreds of thousands of draws,² but MCMC convergence is unlikely to be a major worry. Indeed even the 10,000 draws (plus 1000 burn-in draws) used to produce the results in Table 2 appear to be enough to produce an accurate estimate of the true DGP in our artificial data exercise, despite the fact that the initial conditions used in our MCMC algorithm (based on the VARMA(4,4)) are far from the true VARMA(1,1).

Table 2: Averages across Data Sets of Posterior Mean of $\boldsymbol{\kappa}$ for DGP₃. Standard Deviations in Parentheses.

Algorithm details	κ_1	κ_2	κ_3	κ_4	κ_5	κ_6	κ_7
True value	1	1	1	1	1	1	1
echelon, hard SSVS	1.05 (0.10)	1.01 (0.01)	1.01 (0.01)	1.04 (0.09)	1.07 (0.13)	1.00 (0.01)	0.90 (0.32)
row degree, hard SSVS	1.07 (0.10)	1.04 (0.05)	1.02 (0.03)	1.05 (0.13)	1.03 (0.03)	1.03 (0.04)	0.80 (0.34)
echelon, soft SSVS	1.01 (0.02)	1.02 (0.05)	1.01 (0.01)	0.99 (0.08)	1.03 (0.06)	1.01 (0.01)	0.80 (0.42)
row degree, soft SSVS	1.04 (0.05)	1.04 (0.06)	1.00 (0.03)	0.99 (0.08)	1.04 (0.05)	1.02 (0.02)	0.73 (0.43)

Table 3: Inefficiency Factors for Impulse Responses for DGP₃.

Algorithm details	IR ₁	IR ₁	IR ₁	IR ₂	IR ₂	IR ₂
	ave	st dev	max	ave	st dev	max
echelon, hard SSVS	56.38	100.12	339.57	20.71	13.74	53.76
row degree, hard SSVS	23.34	6.82	38.31	20.29	7.50	30.07
echelon, soft SSVS	13.95	6.15	25.35	15.68	9.27	33.57
row degree, soft SSVS	13.31	5.74	25.09	12.55	4.11	22.41

²This statement and others which follow are based on the premise that 10,000 independent draws from a posterior would produce estimates with sufficient accuracy for the researcher's purposes.

Results for the 12-variate VARMA are also quite encouraging. In Table 4, the estimates for $\kappa_1, \dots, \kappa_n$ are almost always very close to the true values in the DGP. The only exception is for κ_9 and κ_{10} . But for the reasons noted previously, these are not surprising. The four variants of the algorithm are producing similar results, although it is worth noting that the approximate row degree algorithms are producing estimates for κ_7 which are somewhat below those for the exact echelon algorithms.

Table 4: Averages across Data Sets of Posterior Mean of $\boldsymbol{\kappa}$ for DGP₄. Standard Deviations in Parentheses.

Algorithm details	κ_1	κ_2	κ_3	κ_4	κ_5	κ_6
True value	1	1	1	1	1	1
echelon, hard SSVS	1.05 (0.11)	1.09 (0.18)	0.91 (0.23)	0.99 (0.01)	1.24 (0.31)	1.24 (0.41)
row degree, hard SSVS	1.02 (0.06)	1.00 (0.01)	0.71 (0.42)	0.98 (0.04)	0.94 (0.23)	1.00 (0.00)
echelon, soft SSVS	0.99 (0.01)	1.00 (0.00)	0.79 (0.36)	0.95 (0.14)	1.18 (0.39)	1.02 (0.03)
row degree, soft SSVS	0.99 (0.01)	1.00 (0.00)	0.67 (0.44)	0.96 (0.11)	0.92 (0.23)	1.00 (0.00)
	κ_7	κ_8	κ_9	κ_{10}	κ_{11}	κ_{12}
True value	1	1	1	1	0	0
echelon, hard SSVS	1.00 (0.00)	0.96 (0.12)	0.04 (0.01)	0.02 (0.03)	0.00 (0.00)	0.00 (0.00)
row degree, hard SSVS	0.33 (0.39)	0.98 (0.05)	0.03 (0.06)	0.02 (0.02)	0.00 (0.00)	0.00 (0.00)
echelon, soft SSVS	0.81 (0.41)	0.93 (0.16)	0.01 (0.01)	0.01 (0.01)	0.01 (0.00)	0.00 (0.00)
row degree, soft SSVS	0.21 (0.39)	0.94 (0.16)	0.01 (0.02)	0.01 (0.02)	0.00 (0.00)	0.00 (0.00)

Table 5: Inefficiency Factors for Impulse Responses for DGP₄.

Algorithm details	IR ₁	IR ₁	IR ₁	IR ₂	IR ₂	IR ₂
	ave	st dev	max	ave	st dev	max
echelon, hard SSVS	152.98	274.74	766.9	234.73	411.56	1169.9
row degree, hard SSVS	17.44	9.49	28.57	24.31	25.77	88.91
echelon, soft SSVS	36.02	58.27	179.48	31.26	54.74	178.57
row degree, soft SSVS	10.12	3.13	16.80	10.54	4.51	20.42

Table 5 presents evidence on MCMC efficiency. As expected, MCMC efficiency deteriorates somewhat in this larger VARMA, but the row degree algorithm mixes much better than the echelon form algorithm. Of course, an exact algorithm is always to be preferred to an approximate one and, hence, where computationally possible we would

recommend using the echelon form algorithm. However, Table 5 indicates that in larger VARMA, the echelon form algorithm might be excessively computationally daunting or even infeasible in a reasonable amount of time. For instance, when using the echelon form algorithm with hard SSVS, one of our artificial data sets produces an inefficiency factor of over 1000 for estimation of one of the impulse responses suggesting that millions of draws may be required in some applications with larger VARMA. In such applications, our approximate row degree algorithm, which is quite efficient even in the 12-variate VARMA, may be a good alternative.

It is also worth noting that MCMC algorithms using soft SSVS are much more efficient than hard SSVS. Even in the 12-variate VARMA, the echelon form algorithm with soft SSVS is producing inefficiency factors that are consistent with the researcher using tens (or at most a few hundred) of thousands of draws.

7 Appendix F: Additional Empirical Results from Macroeconomic Application

Tables 6–8 present the estimates of the VARMA coefficients referred to in sub-section 4.0.1 of the paper. We also present results from a VAR. Details of both specifications are given in the paper. For ease of comparison, we transform our echelon-form VARMA coefficients into the following standard VARMA specification:

$$\mathbf{y}_t = \sum_{j=1}^p \mathbf{A}_j \mathbf{y}_{t-j} + \sum_{j=1}^q \boldsymbol{\Theta}_j \boldsymbol{\epsilon}_{t-j} + \boldsymbol{\epsilon}_t.$$

Next we present results for the VARMA with $n = 12$ model using the row degree algorithm. We use 250,000 MCMC draws after 25,000 burn-in draws. Total computation time was 6.5 hours. For ease of comparison, some results from the same model but using the echelon form algorithm are also provided. Table 9 presents all values of $\boldsymbol{\kappa}$ which receive greater than 1% posterior probability using the echelon form algorithm. Table 10 repeats the analysis using the row degree algorithm, presenting all values of \mathbf{p} which receive greater than 1% posterior probability. It can be seen that the two algorithms are choosing similar, but not identical, row degrees for each equation.

Remember that the row degree and echelon algorithm differ in that the latter imposes all identifying restrictions (row degree restrictions plus the additional ones) whereas the row degree algorithm does not necessarily impose these additional ones. Figure 1 sheds light on whether these additional restrictions are being picked up by the other part of the SSVS prior (i.e. $\boldsymbol{\gamma}^S$) when using the row degree algorithm. At each MCMC draw of \mathbf{p} from this algorithm, we can count how many of the additional restrictions required to produce a valid echelon form are captured and how many are missed. Figure 1 plots histograms of the resulting draws. It can be seen that, although a large majority of the additional restrictions are captured, quite a few are not.

We also include plots of some key impulse responses, comparing the echelon and row degree algorithms. Although there are some slight differences, overall these two algorithms are producing very similar impulse responses.

Table 6: Posterior estimates of the moving average coefficients matrices $\Theta_1, \dots, \Theta_4$ in a VARMA $_E(\boldsymbol{\kappa})$. Note: * denotes that either $\Pr(\theta_{l,ij} \leq 0 | \mathbf{y}) \leq 0.1$ or $\Pr(\theta_{l,ij} > 0 | \mathbf{y}) \leq 0.1$; § denotes that either $\Pr(\theta_{l,ij} \leq 0 | \mathbf{y}) \leq 0.05$ or $\Pr(\theta_{l,ij} > 0 | \mathbf{y}) \leq 0.05$; † denotes that either $\Pr(\theta_{l,ij} \leq 0 | \mathbf{y}) \leq 0.01$ or $\Pr(\theta_{l,ij} > 0 | \mathbf{y}) \leq 0.01$.

	1	2	3	4	5	6	7	8	9	10	11	12	
Θ_1	1	-0.19*	-0.01	0.06	0.47§	-0.01	-0.04	0.20*	0.12	-0.02	0.06	0.00	-0.03
	2	-0.06	-0.08	0.07	0.03	0.01	-0.12	0.05	0.23§	0.03	0.01	0.01	0.04
	3	-0.09	-0.15	0.02	0.53§	0.00	0.04	-0.07	-0.12	-0.08	0.04	-0.03	-0.07
	4	-0.02	-0.01	0.01	0.08§	0.00	0.00	0.01	-0.01	-0.01	0.01	0.00	-0.01
	5	0.00	0.04	-0.05	0.17	0.01	-0.03	-0.03	0.05	-0.01	0.01	0.00	-0.01
	6	-0.20*	0.06	0.01	0.66†	-0.03	-0.08	0.38†	0.20*	0.02	0.03	0.03	0.00
	7	-0.08	-0.01	0.02	0.25§	-0.02	-0.04	0.03	0.13§	-0.01	0.02	0.00	-0.01
	8	-0.07	-0.02	0.00	0.30†	-0.01	-0.03	0.06	0.11*	-0.02	0.02	0.01	-0.01
	9	-0.05	-0.02	0.01	0.15§	-0.01	-0.04	0.05	0.11§	0.00	0.01	0.01	0.00
	10	-0.03	-0.03	0.01	0.26*	0.00	0.00	-0.07	-0.01	-0.04	0.06	0.01	-0.02
	11	0.07	-0.02	0.00	-0.15	0.04	0.05	-0.12	-0.26§	-0.03	0.01	-0.09	-0.01
	12	-0.01	0.01	0.03	-0.11	0.00	-0.04	0.01	0.13*	0.01	0.01	0.02	0.03
Θ_2	1	-0.03	-0.03	-0.01	0.11	-0.01	-0.02	0.12	0.02	-0.04	-0.01	0.00	-0.01
	2	0.04	-0.14	-0.02	-0.04	0.04	-0.01	-0.05	-0.11	-0.02	-0.02	-0.03	-0.09
	3	-0.02	-0.02	0.00	0.07	-0.01	-0.01	0.08	0.01	-0.03	-0.01	0.00	-0.01
	4	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00
	5	0.12	-0.05	-0.13*	-0.01	-0.04	-0.11	0.24§	0.04	0.00	-0.01	0.00	0.00
	6	-0.01	-0.02	-0.01	0.07	-0.01	-0.02	0.10	0.02	-0.03	-0.01	0.00	-0.01
	7	0.00	-0.02	-0.01	0.03	0.00	-0.02	0.05	0.00	-0.01	-0.01	0.00	-0.01
	8	0.01	-0.02	-0.02	0.02	-0.01	-0.02	0.06*	0.00	-0.01	0.00	0.00	-0.01
	9	0.01	-0.03	-0.01	0.01	0.00	-0.01	0.01	-0.02	-0.01	-0.01	-0.01	-0.02
	10	-0.02	0.00	0.01	0.03	0.00	0.00	0.01	0.00	-0.01	0.00	0.00	0.00
	11	-0.02	0.07	0.01	0.01	-0.01	0.01	0.00	0.04	0.01	0.01	0.02	0.04
	12	0.02	-0.03	0.00	-0.02	0.01	0.00	-0.03	-0.04	0.00	-0.01	-0.01	-0.02
Θ_3	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	3	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	4	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	5	-0.02	0.00	0.02	-0.05	-0.01	0.00	-0.02	0.08	0.00	0.01	0.01	0.01
	6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00
	7	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00
	8	0.00	0.00	0.00	-0.01	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00
	9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.01	0.00	0.00	0.00	0.00
	11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Θ_4	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	3	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	4	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	7	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 7: Posterior estimates of the autoregressive coefficients matrices $\mathbf{A}_1, \dots, \mathbf{A}_4$ in a VARMA $_E(\boldsymbol{\kappa})$. Note: * denotes that either $\Pr(\theta_{l,ij} \leq 0 | \mathbf{y}) \leq 0.1$ or $\Pr(\theta_{l,ij} > 0 | \mathbf{y}) \leq 0.1$; § denotes that either $\Pr(\theta_{l,ij} \leq 0 | \mathbf{y}) \leq 0.05$ or $\Pr(\theta_{l,ij} > 0 | \mathbf{y}) \leq 0.05$; † denotes that either $\Pr(\theta_{l,ij} \leq 0 | \mathbf{y}) \leq 0.01$ or $\Pr(\theta_{l,ij} > 0 | \mathbf{y}) \leq 0.01$.

	1	2	3	4	5	6	7	8	9	10	11	12	
\mathbf{A}_1	1	0.01	-0.01	0.12*	0.08	0.06	0.01	0.07	0.09	-0.01	0.11*	-0.02	-0.05
	2	0.03	-0.54†	0.04	0.01	0.01	0.00	0.01	0.12*	0.15§	-0.02	0.02	0.08
	3	0.05	-0.05	0.03	0.12*	0.00	0.10	0.09	0.05	-0.15§	0.12*	-0.07	0.00
	4	-0.04	0.03	0.04*	0.97†	-0.01	0.06§	0.01	-0.05§	-0.07†	0.04*	0.01	0.00
	5	-0.01	0.03	-0.06	0.05	-0.80†	-0.03	0.03	0.03	-0.02	0.03	0.01	0.02
	6	-0.08	-0.02	0.29†	-0.03	0.06	0.07	-0.07	0.12*	0.08	0.16§	-0.01	-0.03
	7	-0.02	-0.08§	0.11§	0.03	-0.02	0.02	0.67†	0.11§	-0.02	0.12†	-0.03	0.03
	8	0.11*	-0.02	0.07	0.05	-0.05*	-0.03	-0.15§	0.83†	-0.10§	0.10§	-0.07*	-0.01
	9	0.04	-0.10§	0.06*	0.05	0.00	-0.01	0.02	0.33†	-0.01	0.05§	-0.02	0.01
	10	0.06	0.05	0.00	-0.01	0.10*	0.04	0.04	-0.28§	-0.05	0.19§	-0.03	-0.02
	11	-0.09	-0.06	0.03	-0.02	-0.09	-0.10	0.04	0.03	-0.07	0.04	-0.15*	0.01
	12	0.01	-0.05	-0.03	0.05	0.00	0.00	-0.02	0.09	0.02	0.00	-0.01	0.05
\mathbf{A}_2	1	0.04	0.02	0.12*	0.00	-0.01	0.07	-0.06	0.01	-0.17†	0.05	0.03	-0.04
	2	0.02	-0.22§	0.09	0.00	-0.01	-0.14§	-0.02	0.05	0.02	-0.03	0.05	-0.10
	3	0.03	0.02	0.08*	0.00	-0.01	0.05	-0.04	0.00	-0.12†	0.04	0.02	-0.02
	4	0.00	0.00	0.01*	0.00	0.00	0.01	-0.01	0.00	-0.02§	0.00	0.00	-0.01
	5	0.04	0.04	-0.04	0.02	-0.70†	-0.02	0.07	0.01	-0.04	-0.03	-0.02	0.01
	6	0.03	0.03	0.08*	0.00	-0.06	0.05	-0.04	0.00	-0.12†	0.04	0.02	-0.02
	7	0.02	0.00	0.04*	0.00	-0.06§	0.01	-0.02	0.01	-0.06§	0.01	0.01	-0.02
	8	0.02	0.00	0.03	0.00	-0.09†	0.01	-0.01	0.01	-0.05§	0.01	0.01	-0.01
	9	0.01	-0.03*	0.03*	0.00	-0.03	-0.02	-0.01	0.01	-0.02	0.00	0.01	-0.02
	10	0.01	0.00	0.03*	0.00	0.05	0.02	-0.02	0.00	-0.04*	0.01	0.01	-0.01
	11	-0.01	0.08§	-0.05	0.00	0.01	0.05*	0.01	-0.02	0.01	0.01	-0.03	0.04
	12	0.00	-0.08§	0.01	0.00	0.01	-0.05*	0.00	0.02	0.03	-0.02	0.01	-0.04
\mathbf{A}_3	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	3	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	4	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	5	0.00	-0.03	-0.08	0.01	-0.53†	0.07	-0.01	-0.01	-0.07	0.05	0.09*	0.09*
	6	0.00	0.00	-0.01	0.00	-0.04§	0.01	0.00	0.00	-0.01	0.00	0.01*	0.01*
	7	0.00	0.00	-0.01	0.00	-0.04†	0.01	0.00	0.00	-0.01	0.00	0.01*	0.01*
	8	0.00	0.00	-0.01	0.00	-0.07†	0.01	0.00	0.00	-0.01	0.01	0.01*	0.01*
	9	0.00	0.00	0.00	0.00	-0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	10	0.00	0.00	0.01	0.00	0.04	-0.01	0.00	0.00	0.01	0.00	-0.01	-0.01
	11	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	12	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
\mathbf{A}_4	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	3	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	4	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	7	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 8: Posterior estimates of the autoregressive coefficients matrices $\mathbf{A}_1, \dots, \mathbf{A}_4$ in a VAR(4). Note: * denotes that either $\Pr(\theta_{l,ij} \leq 0 | \mathbf{y}) \leq 0.1$ or $\Pr(\theta_{l,ij} > 0 | \mathbf{y}) \leq 0.1$; § denotes that either $\Pr(\theta_{l,ij} \leq 0 | \mathbf{y}) \leq 0.05$ or $\Pr(\theta_{l,ij} > 0 | \mathbf{y}) \leq 0.05$; † denotes that either $\Pr(\theta_{l,ij} \leq 0 | \mathbf{y}) \leq 0.01$ or $\Pr(\theta_{l,ij} > 0 | \mathbf{y}) \leq 0.01$.

	1	2	3	4	5	6	7	8	9	10	11	12	
\mathbf{A}_1	1	-0.10	-0.03	0.12*	0.63†	0.05	-0.12	0.31§	0.15*	0.00	0.16†	-0.02	-0.06
	2	0.00	-0.64†	0.09	0.04	0.01	-0.10	0.01	0.31§	0.19†	0.02	0.02	0.17†
	3	-0.03	-0.20†	0.00	0.57†	-0.03	0.05	0.15	-0.02	-0.22†	0.17†	-0.06	-0.05
	4	-0.05	0.00	0.03	1.10†	-0.01	0.06*	0.02	-0.06*	-0.09†	0.05§	0.00	0.00
	5	-0.04	0.07	-0.09*	0.12	-0.79†	-0.05	0.05	0.10	-0.06	0.04	0.03	0.01
	6	-0.14*	0.03	0.20†	0.86†	0.05	-0.18§	0.28§	0.22§	0.12§	0.14†	0.01	-0.01
	7	-0.06	-0.09§	0.11§	0.37†	-0.06*	-0.05	0.63†	0.30†	-0.03	0.11†	-0.02	0.04
	8	0.03	-0.06	0.04	0.33†	-0.06	-0.12§	0.02	0.91†	-0.10§	0.14†	-0.03	0.00
	9	0.02	-0.19†	0.01	0.33§	-0.03	-0.06	0.05	0.58†	0.08	0.06	0.06	0.11§
	10	0.05	-0.03	0.01	0.30*	0.08	0.02	0.00	-0.35§	-0.04	0.29†	0.01	0.00
	11	-0.07	-0.02	0.07	-0.14	0.01	-0.05	-0.06	-0.30†	-0.16§	0.06	-0.33†	-0.07
	12	-0.01	0.02	0.02	-0.04	-0.02	-0.01	-0.01	0.22*	0.01	0.07	-0.03	0.15§
\mathbf{A}_2	1	-0.01	-0.02	0.07	-0.27	-0.02	-0.05	0.07	0.03	-0.16§	0.00	0.04	-0.05
	2	-0.02	-0.42†	0.05	-0.03	0.04	-0.20§	-0.04	-0.12	0.10*	-0.08*	0.05	-0.18†
	3	0.00	-0.09	0.10	-0.22	-0.06	-0.02	0.08	0.14	-0.15§	0.03	0.02	0.01
	4	-0.05	-0.04	0.05*	0.00	-0.02	0.03	-0.03	0.08*	-0.01	-0.02	-0.02	-0.01
	5	0.06	0.00	-0.11§	-0.02	-0.76†	-0.04	0.25§	0.03	-0.04	-0.03	-0.01	0.00
	6	-0.02	0.01	0.01	-0.38*	-0.06	-0.02	-0.06	-0.04	-0.04	0.02	0.04	-0.02
	7	0.05	-0.02	-0.02	-0.17	-0.08*	-0.08	0.03	-0.17§	0.02	0.05*	0.05	-0.02
	8	0.05	-0.03	-0.07*	-0.18	-0.11§	-0.11§	0.07	-0.01	0.00	-0.02	0.01	-0.04
	9	0.04	-0.06	-0.23†	-0.16	0.00	-0.09	0.25*	-0.21	-0.17§	-0.01	0.06	-0.04
	10	-0.05	-0.08	0.02	-0.19	0.06	0.00	0.04	0.01	0.11*	0.00	0.04	-0.02
	11	-0.03	0.07	0.09	0.30	0.05	0.14*	-0.09	0.27§	0.02	0.07	-0.30†	0.12§
	12	-0.10	-0.12	-0.06	0.12	0.03	-0.05	-0.01	-0.10	0.17§	-0.05	0.06	-0.13*
\mathbf{A}_3	1	-0.01	-0.01	0.00	-0.07	0.01	0.00	-0.31§	0.01	0.06	0.02	0.00	-0.02
	2	0.01	-0.05	-0.02	0.01	0.08	0.01	-0.07	0.13	-0.01	0.12§	-0.03	-0.10*
	3	0.00	-0.08	0.09	-0.11	0.05	0.03	-0.13	0.03	-0.02	-0.04	0.02	-0.10*
	4	-0.07*	-0.05*	0.03	-0.13*	0.01	0.03	0.02	-0.07*	0.00	0.00	-0.02	0.02
	5	-0.04	-0.05	-0.05	-0.02	-0.60†	0.07	-0.08	0.03	-0.10§	0.04	0.11§	0.09§
	6	-0.02	0.01	-0.04	-0.37*	-0.07	-0.01	-0.27§	0.07	0.06	0.02	0.04	0.05
	7	0.06	-0.01	-0.02	-0.09	-0.07*	-0.01	-0.05	0.08	-0.01	0.05	0.01	-0.04
	8	0.09	-0.07*	-0.01	0.00	-0.06	-0.06	-0.19§	0.04	-0.03	0.00	0.00	-0.03
	9	-0.04	-0.01	-0.02	-0.05	-0.01	-0.10	0.01	0.23*	0.05	0.06	0.08	-0.08
	10	0.03	-0.11*	0.11*	-0.05	-0.07	-0.05	-0.07	0.01	0.12*	-0.04	-0.05	-0.09*
	11	0.03	0.00	0.02	-0.13	0.02	0.05	0.00	-0.08	0.00	0.01	-0.09	0.04
	12	-0.01	0.04	-0.03	0.02	0.07	-0.06	-0.08	0.09	0.02	0.03	-0.05	-0.02
\mathbf{A}_4	1	0.07	-0.04	0.03	-0.19	0.01	-0.04	0.03	0.02	-0.05	0.01	-0.01	-0.02
	2	0.02	-0.12§	-0.03	0.03	-0.02	0.06	0.03	0.04	-0.04	-0.02	0.03	-0.09*
	3	-0.01	-0.03	-0.01	-0.10	-0.01	0.03	-0.01	0.01	-0.09	0.04	-0.05	0.03
	4	0.01	0.00	0.03	-0.01	0.03*	-0.01	0.00	0.04	0.00	-0.02	0.00	-0.02
	5	-0.02	0.01	0.00	0.00	-0.01	-0.16§	0.03	-0.04	0.05	-0.02	0.03	0.02
	6	0.07	0.00	0.14§	-0.13	-0.03	-0.02	-0.05	-0.12	0.06	0.01	0.03	-0.07*
	7	0.00	-0.02	0.05	-0.09	-0.05	-0.04	0.05	0.01	-0.04	0.01	-0.05*	-0.02
	8	-0.04	-0.02	0.12§	-0.10	-0.02	-0.04	0.00	-0.04	-0.02	0.00	-0.04	0.00
	9	-0.06	-0.02	-0.04	-0.02	-0.03	-0.03	0.00	-0.13	0.04	-0.06	0.00	-0.01
	10	0.02	-0.05	-0.02	-0.08	-0.05	-0.04	-0.01	0.08	-0.07	0.04	-0.02	-0.07
	11	-0.09	-0.03	-0.02	-0.08	0.07	-0.01	0.05	0.08	0.11*	-0.04	-0.16†	0.05
	12	0.01	-0.05	0.02	0.02	-0.02	-0.01	-0.06	0.10	0.05	0.04	0.04	-0.10*

Table 9: Posterior distribution over echelon form structures. Each column represents a particular vector $\boldsymbol{\kappa}$ of Kronecker indices. The last row is a posterior estimate of the probability mass function $\Pr(\boldsymbol{\kappa} | \mathbf{y})$ obtained with the echelon algorithm; only $\boldsymbol{\kappa}$ with estimated mass greater than 1% are shown.

κ_1	2	2	2	2	2	2
κ_2	2	2	2	2	2	2
κ_3	1	1	1	1	1	1
κ_4	1	1	1	1	1	1
κ_5	3	3	3	3	3	3
κ_6	1	1	1	1	1	1
κ_7	1	1	1	1	1	1
κ_8	1	1	1	1	1	1
κ_9	0	0	0	0	0	0
κ_{10}	0	0	1	1	1	1
κ_{11}	1	1	0	0	1	1
κ_{12}	0	1	0	1	0	1
Posterior Weight	12.71%	3.00%	1.56%	1.46%	44.59%	34.48%

Table 10: Posterior distribution over equation lag structures. Each column represents a particular vector \mathbf{p} of row degrees. The last row is a posterior estimate of the probability mass function $\Pr(\mathbf{p} | \mathbf{y})$ obtained with the row degree algorithm; only \mathbf{p} with estimated mass greater than 1% are shown.

p_1	1	1	1	1	1	1	2	2
p_2	2	2	2	2	2	2	2	2
p_3	1	1	1	1	1	1	1	1
p_4	1	1	1	1	1	1	1	1
p_5	3	3	3	3	3	3	3	3
p_6	1	1	1	1	1	1	1	1
p_7	1	1	1	1	1	1	1	1
p_8	1	1	1	1	1	1	1	1
p_9	0	0	0	0	0	0	0	0
p_{10}	0	0	0	1	1	1	0	1
p_{11}	0	1	1	0	1	1	1	1
p_{12}	0	0	1	0	0	1	0	0
Posterior Weight	1.8%	52.8%	3.4%	1.0%	28.3%	4.0%	3.5%	3.1%

Table 11: Inefficiency factors for impulse responses generated by the row degree algorithm ($n = 12$); note that the reported inefficiency factors are computed on thinned draws.

IF avg	IF st dev	IF max
2.24	3.39	22.99

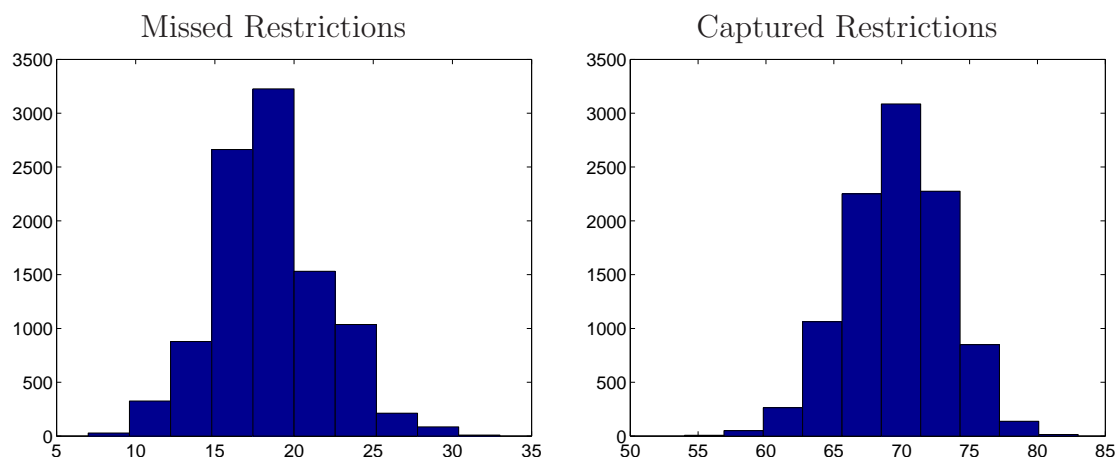


Figure 1: Effect of not enforcing the echelon form to hold at every iteration. The first column depicts the distribution of the number of echelon restrictions *missed* by the SSVS restrictions in the row degree algorithm; the second column depicts the distribution of the number of echelon restrictions correctly *captured* by the SSVS restrictions in the row degree algorithm.

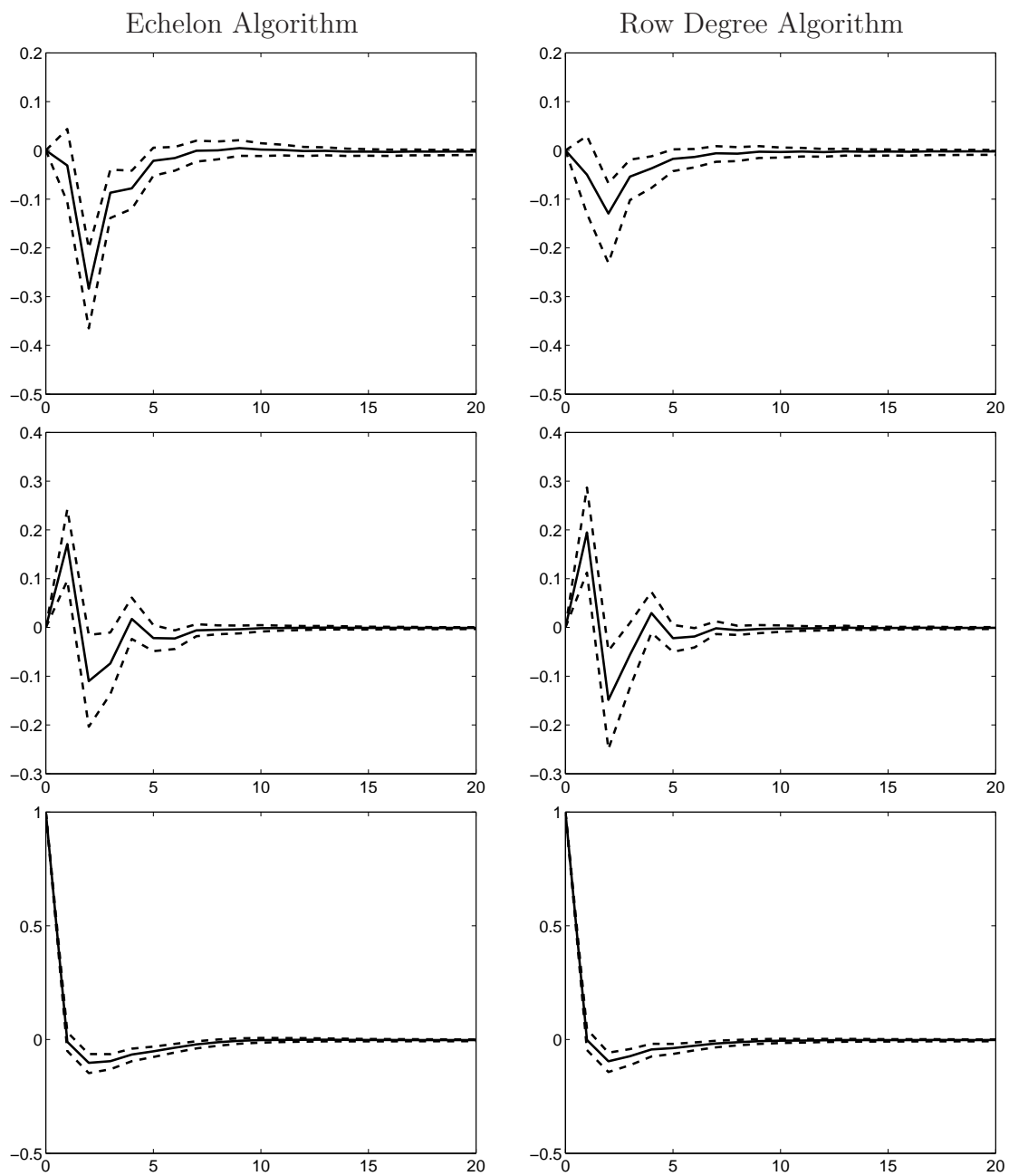


Figure 2: Comparison of impulse responses to a shock in the interest rate generated by the echelon vs. row degree algorithms ($n = 12$). The first row contains responses of GDP to a shock in the interest rate; the second row contains responses of inflation to a shock in the interest rate; the third row contains responses the interest rate to its own shock. The dotted lines depict the (10%, 90%) HPD intervals.

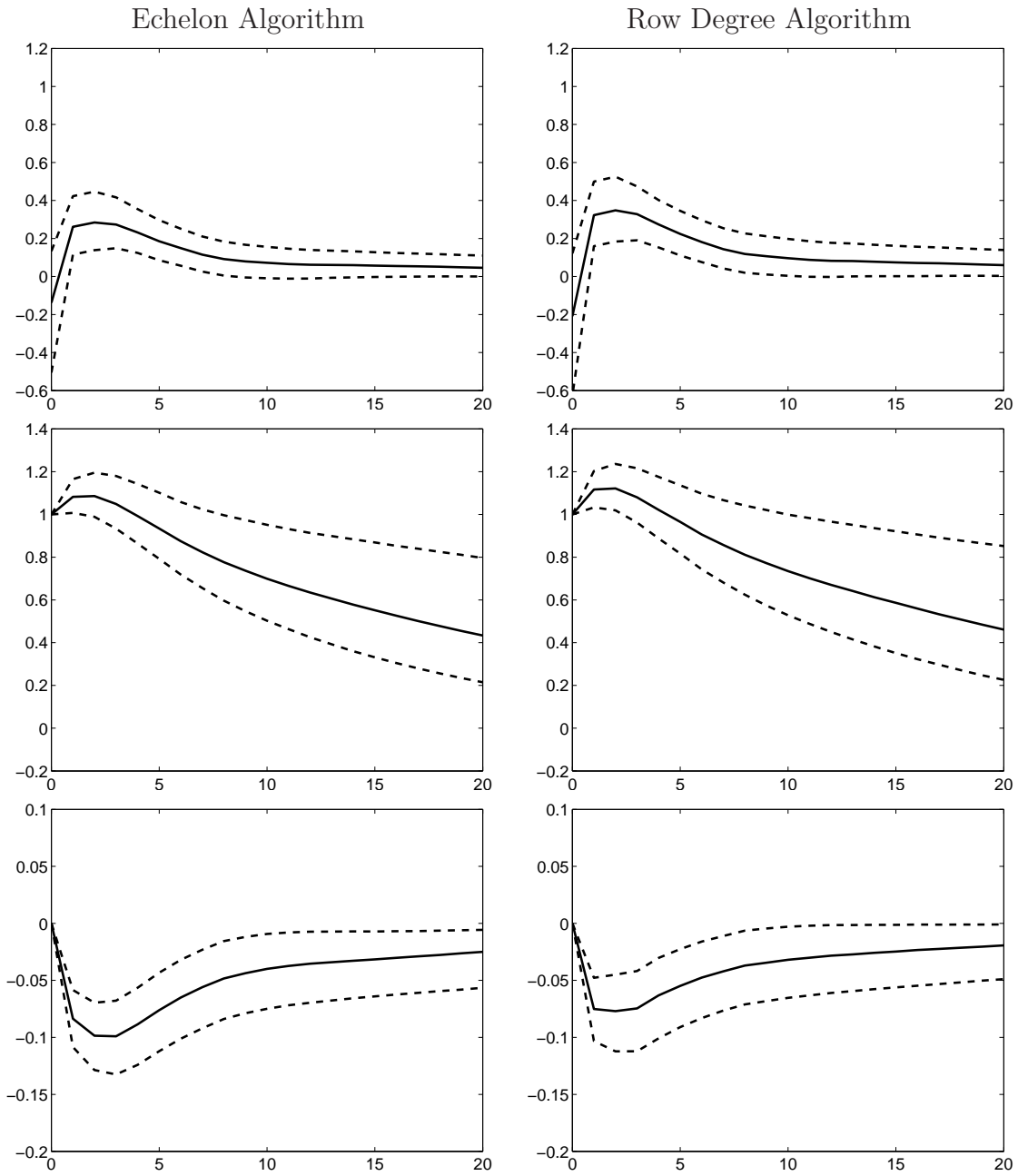


Figure 3: Comparison of impulse responses of the housing start and interest rate to shocks generated by the echelon vs. row degree algorithms ($n = 12$). The first row contains responses of the interest rate to a shock in the housing start; the second row contains responses of the housing start to its own shock; the third row contains responses of the housing start to a shock in the interest rate. The dotted lines depict the (10%, 90%) HPD intervals.

References

Celeux, G., Forbes, F., Robert, C. and Titterton, D. (2006). “Deviance information criteria for missing data models,” *Bayesian Analysis*, 1, 651-674.

Chan, J. and Eisenstat, E. (2015). “Efficient estimation of Bayesian VARMA with time-varying coefficients,” manuscript.

Chan, J. and Grant, A. (2014). “Fast computation of the deviance information criterion for latent variable models,” *Computational Statistics and Data Analysis*, forthcoming.

Spiegelhalter, D., Best, N., Carlin, B. and van der Linde, A. (2002). “Bayesian measures of model complexity and fit,” *Journal of the Royal Statistical Society Series B*, 64, 583-639.