# Efficient Bayesian Inference for Multivariate Factor Stochastic Volatility Models

Gregor Kastner, Sylvia Frühwirth-Schnatter, Hedibert Freitas Lopes

**Research Report Series**
Report 128, February 2016

Institute for Statistics and Mathematics
**http://statmath.wu.ac.at/**

WU
WIRTSCHAFTS UNIVERSITÄT WIEN VIENNA UNIVERSITY OF ECONOMICS AND BUSINESS

EFMD EQUIS ACCREDITED

# Efficient Bayesian Inference for Multivariate Factor Stochastic Volatility Models

Gregor Kastner, Sylvia Frühwirth-Schnatter
Department of Finance, Accounting and Statistics,
WU Vienna University of Economics and Business, Austria
and
Hedibert Freitas Lopes
Insper, Brazil

February 23, 2016

**Abstract**

We discuss efficient Bayesian estimation of dynamic covariance matrices in multivariate time series through a factor stochastic volatility model. In particular, we propose two interweaving strategies (Yu and Meng, 2011) to substantially accelerate convergence and mixing of standard MCMC approaches. Similar to marginal data augmentation techniques, the proposed acceleration procedures exploit non-identifiability issues which frequently arise in factor models. Our new interweaving strategies are easy to implement and come at almost no extra computational cost; nevertheless, they can boost estimation efficiency by several orders of magnitude as is shown in extensive simulation studies. To conclude, the application of our algorithm to a 26-dimensional exchange rate data set illustrates the superior performance of the new approach for real-world data.

*Keywords:* Ancillarity-sufficiency interweaving strategy (ASIS), Curse of dimensionality, Data augmentation, Dynamic covariance matrices, Exchange rate data, Markov chain Monte Carlo (MCMC)

# 1 Introduction

Analysis of multivariate time series has become a vivid research area over the last decades where both methodological as well as computational advances have made it possible to estimate more and more complex models. In parallel, real-world applications with an ever-increasing amount of data call for the joint modeling of many simultaneous and often co-varying observations over time. However, already the number of pair-wise co-movements increases quadratically with the number of time series, let alone higher-dimensional dependency structures. This property, often referred to as the *curse of dimensionality*, can often be mitigated in various ways by imposing a lower-dimensional latent factor structure, thereby effectively reducing the number of parameters to a feasible amount. In the paper at hand, we particularly focus on the case where these factors are allowed to have time-varying variances which in turn drive the multivariate dynamics. To the best of our knowledge, models of this type have first been discussed by Jacquier et al. (1994); Shephard (1996); Kim et al. (1998). We particularly focus on the model formulation brought forward by Chib et al. (2006).

Applications of multivariate factor stochastic volatility models typically reside in the field of financial econometrics, most prominently in areas that involve accurate quantification of uncertainty and risk. Examples thereof are asset allocation (e.g. Aguilar and West, 2000; Han, 2006; Zhou et al., 2014) and asset pricing (e.g. Nardari and Scruggs, 2007). These models extend standard factor pricing models such as the arbitrage pricing theory (Ross, 1976) and the capital asset pricing model (Sharpe, 1964; Lintner, 1965) by relaxing the assumption that the multivariate volatility dynamics is constant over time.

Statistical estimation of these models can be challenging, and a variety of solutions such as quasi-maximum likelihood (e.g. Harvey et al., 1994) or simulated maximum likelihood (e.g. Liesenfeld and Richard, 2006; Jungbacker and Koopman, 2006) have been proposed. For medium to high dimensional problems, Bayesian MCMC estimation (Pitt and Shephard, 1999; Aguilar and West, 2000; Chib et al., 2006; Han, 2006; Omori et al., 2007) is probably the most efficient estimation method, however, it is associated with a considerable computational burden when the number of assets is moderate to large.

Aim of this work is to outline a reliable method for Bayesian inference that performs well

for a wide range of data sets while at the same time being easy to implement and convenient to extend. Therefore, we combine an efficient methods for estimating univariate stochastic volatility models introduced by Kastner and Frühwirth-Schnatter (2014) with a standard Gibbs sampler for regression problems. To ensure fast convergence and proper mixing of the MCMC chains we augment this simple procedure with interweaving strategies introduced by Yu and Meng (2011). Through extensive simulation studies and a real-world example, we demonstrate the effectiveness of our procedure which can boost sampling efficiency by a factor of 100 and more.

The remainder of this paper is structured as follows. Section 2 establishes notation for the factor stochastic volatility model framework and discusses questions about model specification and identification. Section 3 gives an in-depth exposure to the estimation algorithm and its implementation, whereby the focus is placed on the novel interweaving strategies employed. Section 4 presents measures of sampling efficiency for simulated data sets and compares the algorithms presented. Section 5 discusses a case study with 26 daily EUR exchange rates. Section 6 concludes.

# 2 The Multivariate Factor Stochastic Volatility Model

In a multivariate framework, the quadratic growth of the number of covariances alongside their inherent time-variability calls for a model which is sufficiently parsimoniously specified. At the same time, the model needs to be flexible enough to have the potential to capture typical features of financial and economic time series such as volatility clustering and volatility co-movement. On top of that, common irregularities in the data require the model to be robust with respect to idiosyncratic shocks.

The multivariate factor stochastic volatility (SV) model (Chib et al., 2006) aims at uniting simplicity with flexibility and robustness. It is *simple* in the sense that the potentially high-dimensional observation space is reduced to a lower-dimensional orthogonal latent factor space, just like in the case of the classic factor model. It is *flexible* in the sense that these factors are allowed to exhibit volatility clustering, and it is *robust* in the sense that idiosyncratic deviations are themselves stochastic volatility processes, thereby allowing for the degree of volatility co-movement to be time-varying.

## 2.1 Model Specification

For each point in time $t = 1, \ldots, T$, let $\boldsymbol{y}_t = (y_{1t}, \ldots, y_{mt})'$ be a vector of $m$ observed returns and let $\boldsymbol{f}_t = (f_{1t}, \ldots, f_{rt})'$ be a vector of $r$ unobserved latent factors. In analogy to the static factor model, the observations are assumed to be driven by the latent factors and the idiosyncratic innovations. In the case of the factor stochastic volatility model, however, both the idiosyncratic innovations as well as the latent factors are allowed to have time-varying variances, depending on $m + r$ latent volatilities $\boldsymbol{h}_t = (h_{1t}, \ldots, h_{m+r,t})'$. In short, we have

$$\boldsymbol{y}_t = \boldsymbol{\Lambda} \boldsymbol{f}_t + \boldsymbol{U}_t(\boldsymbol{h}_t)^{1/2} \boldsymbol{\epsilon}_t, \qquad \boldsymbol{f}_t = \boldsymbol{V}_t(\boldsymbol{h}_t)^{1/2} \boldsymbol{\zeta}_t, \tag{1}$$

where $\boldsymbol{\Lambda}$ is an unknown $m \times r$ factor loadings matrix, $\boldsymbol{U}_t(\boldsymbol{h}_t) = \text{diag}(\exp(h_{1t}), \ldots, \exp(h_{mt}))$ is a diagonal $m \times m$ matrix containing the idiosyncratic (series-specific) variances, and $\boldsymbol{V}_t(\boldsymbol{h}_t) = \text{diag}(\exp(h_{m+1,t}), \ldots, \exp(h_{m+r,t}))$ is a diagonal $r \times r$ matrix containing the factor variances. These variances are themselves modeled as latent variables whose logarithms follow independent autoregressive processes of order one, i.e. for $i = 1, \ldots, m + r$:

$$h_{it} \;\; = \;\; \mu_i + \phi_i(h_{i,t-1} - \mu_i) + \sigma_i \eta_{it}, \tag{2}$$

with unknown initial value $h_{i0}$.

All innovations are assumed to follow independent standard normal distributions, i.e. $\boldsymbol{\epsilon}_t \sim \mathcal{N}_m(\boldsymbol{0}, \boldsymbol{I}_m)$, $\boldsymbol{\zeta}_t \sim \mathcal{N}_r(\boldsymbol{0}, \boldsymbol{I}_r)$, and $\boldsymbol{\eta}_t \sim \mathcal{N}_{m+r}(\boldsymbol{0}, \boldsymbol{I}_{m+r})$, where $\boldsymbol{\eta}_t = (\eta_{1t}, \ldots, \eta_{m+r,t})'$. This implies following structure:

$$\boldsymbol{y}_t = \boldsymbol{\Lambda} \boldsymbol{f}_t + \boldsymbol{\varepsilon}_t, \qquad \boldsymbol{f}_t | \boldsymbol{h}_t \sim \mathcal{N}_r(\boldsymbol{0}, \boldsymbol{V}_t(\boldsymbol{h}_t)) \,, \tag{3}$$

with $\boldsymbol{\varepsilon}_t | \boldsymbol{h}_t \sim \mathcal{N}_m(\boldsymbol{0}, \boldsymbol{U}_t(\boldsymbol{h}_t))$. One of the main reasons for estimating a factor SV model is to reliably estimate the potentially time-varying conditional covariance matrix of $\boldsymbol{y}_t$ which, for the model at hand, is given by $\text{cov}(\boldsymbol{y}_t | \boldsymbol{h}_t) = \boldsymbol{\Sigma}_t(\boldsymbol{h}_t) = \boldsymbol{\Lambda} \boldsymbol{V}_t(\boldsymbol{h}_t) \boldsymbol{\Lambda}' + \boldsymbol{U}_t(\boldsymbol{h}_t)$. Note that because $\boldsymbol{U}_t$ is diagonal, all covariances between the component series are governed by the latent factors. Marginally with respect to $\boldsymbol{h}_t$, $\boldsymbol{y}_t$ is a process with non-Gaussian stationary distribution.

## 2.2 Identification Issues

Whenever certain combinations of parameter values result in (almost) identical maxima in the likelihood function, estimation of the corresponding parameter values from data can become impossible. Consequently, observationally equivalent parameter constellations must be ruled out for reliable statistical inference and a large body of literature dealing with this issue has arisen. In particular, Frühwirth-Schnatter and Lopes (2015) give an overview of recent advances in the context of static Bayesian factor models, and Sentana and Fiorentini (2001) specifically discusses identification for models where the factors exhibit conditional heteroscedasticity (but innovations are assumed to be homoscedastic). If not dealt with properly, usually through certain restrictions on the parameter space, sensible interpretation of the posterior distribution is not possible ("nonidentifiability"). In less severe cases ("near-nonidentifiability"), MCMC algorithms and other estimation procedures often lack convergence and thus provide unreliable results. For the model at hand, we face several issues related to this problem.

First, to prevent factor rotation and column switching, we follow the usual convention and set the upper triangular part of $\mathbf{\Lambda}$ to zero and diag($\mathbf{\Lambda}$) nonzero (e.g. Geweke and Zhou, 1996). Second, without identifying the scaling of either the $j$th column of $\mathbf{\Lambda}$ or the variance of $f_{jt}$, the model is not identified. The usual remedy (e.g. Aguilar and West, 2000; Chib et al., 2006; Han, 2006; Lopes and Carvalho, 2007; Nakajima and West, 2013; Zhou et al., 2014) is that the diagonal loading elements in model (3) are fixed to one, i.e. $\Lambda_{jj} = 1$, for $j = 1, \ldots, r$, while the level $\mu_{m+j}$ of the factor volatilities $h_{m+j,t}$ in model (2) (which corresponds to the scaling of $f_{jt}$) is modeled to be unknown. This approach implies that the first $r$ variables are leading the factors and thus makes variable ordering an important modeling decision. To alleviate this issue somewhat, we leave the diagonal elements $\Lambda_{jj}$ in model (3) unrestricted, an intuitive interpretation being that "leadership" of a factor can be passed on to lower rows (but not to higher ones). Instead, we fix the level $\mu_{m+j}$ of the factor volatilities $h_{m+j,t}$ at zero:

$$\begin{aligned} h_{it} &= (1 - \phi_i)\mu_i + \phi_i h_{i,t-1} + \sigma_i \eta_{it}, \quad i = 1, \ldots, m, \\ h_{m+j,t} &= \phi_{m+j} h_{m+j,t-1} + \sigma_{m+j} \eta_{m+j,t}, \quad j = 1, \ldots, r. \end{aligned} \tag{4}$$

Finally, each column of $\mathbf{\Lambda}$ is only identified up to a possible sign switch. We deal with this (lightweight) identification issue a posteriori, meaning that we run our MCMC sampler in the unrestricted model and identify signs afterwards, see Section A.4 in Appendix A for details.

Factor model (3) together with the $m + r$ SV models (4) define our baseline parameterization, however alternative parameterizations will be exploited in Section 3.3 in the context of efficient MCMC estimation of the factor SV model.

# 3 Bayesian Inference

We perform Bayesian inference based on a set of carefully selected proper priors which are introduced in Section 3.1 and develop efficient schemes for full conditional MCMC sampling in the remaining subsections.

## 3.1 Prior Distributions

Independently for each $i \in \{1, \ldots, m+r\}$, priors for the univariate SV processes are chosen as in Kastner and Frühwirth-Schnatter (2014): $p(\mu_i, \phi_i, \sigma_i) = p(\mu_i)p(\phi_i)p(\sigma_i)$, where the level $\mu_i \in \mathbb{R}$ is equipped with the usual normal prior $\mu_i \sim \mathcal{N}(b_\mu, B_\mu)$, the persistence parameter $\phi_i \in (-1, 1)$ is chosen according to $(\phi_i + 1)/2 \sim \mathcal{B}(a_0, b_0)$ as in Kim et al. (1998), and the volatility of log variance $\sigma_i \in \mathbb{R}^+$ is implied by $\sigma_i^2 \sim B_\sigma \times \chi_1^2 = \mathcal{G}\left(\frac{1}{2}, \frac{1}{2B_\sigma}\right)$. The initial state $h_{i0}$ is distributed according to the stationary distribution of the AR(1) process (2), i.e. $h_{i0}|\mu_i, \phi_i, \sigma_i \sim \mathcal{N}(\mu_i, \sigma_i^2/(1-\phi_i^2))$. For each element of the factor loadings matrix, we choose a zero-mean Gaussian distribution, i.e. $\Lambda_{ij} \sim \mathcal{N}(0, B_\Lambda)$ independently for each $i \in \{1, \ldots m\}$ and $j \in \{i, \ldots, r\}$.

## 3.2 Full Conditional MCMC estimation

Bayesian inference operates directly in the latent variable model (3) and (4) and relies on data augmentation by introducing the latent volatilities $\boldsymbol{h} = \{\boldsymbol{h}_{i,\bullet}\}, i = 1, \ldots, m + r$, where $\boldsymbol{h}_{i,\bullet} = (h_{i0}, h_{i1}, \ldots, h_{iT})'$, and the latent factors $\boldsymbol{f} = \{\boldsymbol{f}_{j,\bullet}\}, j = 1, \ldots, r$, where

$\boldsymbol{f}_{j,\bullet} = (f_{j1}, \ldots, f_{jT})'$, as latent data.This allows to set up a simple scheme for full conditional MCMC sampling as outlined Algorithm 1.

Conditional on knowing the latent factors $\boldsymbol{f}$ and the loadings $\boldsymbol{\Lambda}$, we are dealing with $m+r$ independent, univariate SV models where the latent state equations (4) are combined with following observation equations:

$$\log(y_{it} - \boldsymbol{\Lambda}_{i,\bullet}\boldsymbol{f}_t)^2 = h_{it} + \log \epsilon_{it}^2, \qquad i = 1, \ldots, m, \tag{5}$$

$$\log f_{jt}^2 = h_{m+j,t} + \log \zeta_{jt}^2, \qquad j = 1, \ldots, r. \tag{6}$$

Hence, sampling the latent volatilities $\boldsymbol{h}_{i,\bullet}$ as well as the parameters $(\mu_i, \phi_i, \sigma_i)$ for $i = 1, \ldots, m+r$ (with $\mu_i = 0$ for $i > m$) in Step (a) amounts to $m + r$ univariate SV updates. Consequently, the substantial amount of research on this matter which has emerged in the last two decades can directly be applied. In particular, we follow recent findings in Kastner and Frühwirth-Schnatter (2014), where an efficient sampling scheme is proposed and evaluated, and simply use the implementation in the R package **stochvol** (Kastner, 2016) as a "plug-in" for Step (a) of the factor SV sampler presented in Algorithm 1; see Appendix A.1 for more details and additional references on MCMC estimation for univariate SV models.

On the other hand, conditional on knowing the latent volatilities $\boldsymbol{h}$, we are dealing in (3) with a factor model with heteroscedastic errors. Nevertheless, given $\boldsymbol{h}$, $\boldsymbol{f}$ and $\boldsymbol{\Lambda}$ may be sampled conditionally on each other from the respective multivariate normal distributions in a similar manner as for a standard factor model (Lopes and West, 2004). This approach is conceptually straightforward, see Appendix A.2 for details how to sample in Step (b) each row $\boldsymbol{\Lambda}_{i,\bullet}$ of the factor loading matrix from $\boldsymbol{\Lambda}_{i,\bullet}|\boldsymbol{f}, \boldsymbol{y}_{i,\bullet}, \boldsymbol{h}_{i,\bullet}$, where $\boldsymbol{y}_{i,\bullet} = (y_{i1}, \ldots, y_{iT})'$, and Appendix A.3 for details how to sample in Step (c) the factor $\boldsymbol{f}_t$ from $\boldsymbol{f}_t|\boldsymbol{\Lambda}, \boldsymbol{y}_t, \boldsymbol{h}_t$ for $t = 1, \ldots, T$.

**Algorithm 1.** Choose appropriate starting values for $\mu_i$, $i \in \{1, \ldots, m\}$, $\phi_i$ and $\sigma_i$, $i \in \{1, \ldots, m+r\}$, $\Lambda_{ij}$, $i \in \{1, \ldots, m\}$, $j \in \{1, \ldots, \min(i, r)\}$ as well as for $\boldsymbol{h}$ and $\boldsymbol{f}$ and repeat the following steps:

(a) Perform in total $m + r$ univariate SV updates of the $m$ idiosyncratic variances $\boldsymbol{h}_{i,\bullet}$ as well as the parameters $(\mu_i, \phi_i, \sigma_i)$, independently for each $i = 1, \ldots, m$, and of the

$r$ factor variances $\boldsymbol{h}_{m+j,\bullet}$ as well as the parameters $(\phi_{m+j}, \sigma_{m+j})$, independently for each $j = 1, \ldots, r$.

(b) For $i = 1, \ldots, m$, sample each row $\boldsymbol{\Lambda}_{i,\bullet}$ of the factor loading matrix from $\boldsymbol{\Lambda}_{i,\bullet}|\boldsymbol{f}, \boldsymbol{y}_{i,\bullet}, \boldsymbol{h}_{i,\bullet}$. This step constitutes $m$ independent $\min(i, r)$-variate regression problems with $T$ observations.

(b*) Redraw the diagonal elements of $\boldsymbol{\Lambda}$ through interweaving into the state equation for the latent factors (*shallow interweaving*) or through interweaving into the state equation for the latent volatilities (*deep interweaving*).

(c) For $t = 1, \ldots, T$, sample $\boldsymbol{f}_t$ from $\boldsymbol{f}_t|\boldsymbol{\Lambda}, \boldsymbol{y}_t, \boldsymbol{h}_t$, constituting $T$ independent $r$-variate regression problems with $m$ observations.

After discarding a certain amount of initial draws (the *burn-in*), the *standard full conditional sampler* iterates steps (a), (b) and (c) of Algorithm 1, but not (b*), and should, in principle, yield draws from the joint posterior distribution. However, when estimating factor SV models through such an MCMC scheme, slow convergence and poor mixing (i.e. high correlation of posterior draws) can become a potentially prohibitive issue. This phenomenon substantiates in enormous autocorrelation of posterior draws – even after thinning – and can render MCMC output practically useless. For certain data sets, the burn-in phase may take extremely long and a huge amount of samples has to be discarded before the draws can be considered to emerge from the posterior distribution. Additionally, even after burn-in, these draws often show extraordinarily high autocorrelation and thus only explore the target distribution painstakingly slowly. These so-called *badly mixing* samplers do not only prolong computation time, they also frequently lead to unreliable estimates and misleading results. The simulation study in Section 4 illustrates that this can happen for the standard full conditional sampler even with data simulated from the true model, see e.g. the top of the two panels in Figure 1. Consequently, a carefully crafted posterior simulator is of utmost importance.

To overcome this problem, Chib et al. (2006) propose to sample the factor loading matrix $\boldsymbol{\Lambda}$ from the marginalized conditional posterior $p(\boldsymbol{\Lambda}|\boldsymbol{y}, \boldsymbol{h})$, without conditioning on the factors $\boldsymbol{f}$. This distribution, however, is not available in closed form, and to sample from

it requires a rather involved Metropolis-Hastings update where the proposal distribution is based on numerically maximizing the often high-dimensional conditional likelihood function and approximating its Hessian matrix at every MCMC iteration. To avoid this potential bottleneck, we employ the simpler full conditional procedure outlined in Algorithm 1 but enhance it in Step (b*) by employing two variants of an ancillarity-sufficiency interweaving strategy (ASIS) (Yu and Meng, 2011), called *shallow interweaving* and *deep interweaving*, which are explained in detail in Section 3.3.

Applications to simulated data in Section 4 as well as to exchange rate data in Section 5 illustrate how adding Step (b*) boosts MCMC dramatically, in particular for deep interweaving; compare e.g. the top panel in Figure 1 to the remaining panels. Section A.5 in Appendix A provides comments on practical implementation of the boosted Algorithm 1 using the R package **factorstochvol**.

## 3.3 Boosting Full Conditional MCMC through Interweaving

As discussed in Section 3.2, the *standard full conditional sampler* outlined in Algorithm 1 is based on data augmentation in the parameterization (3) and (4) of the factor SV model and suffers from slow convergence like so many other MCMC schemes which alternate between sampling from the full conditionals of the latent states and the model parameters. A large literature has emerged discussing various techniques to improve such algorithms, in particular reparameterization (Papaspiliopoulos et al., 2007), marginal data augmentation (van Dyk and Meng, 2001), and interweaving strategies (Yu and Meng, 2011).

Reparameterization relies on data augmentation in a different parameterization of the model with alternative latent variables. In particular, so-called non-centered parameterizations where unknown model parameters are moved from the latent state equation to the observation equation proved to be useful, see e.g. Frühwirth-Schnatter and Wagner (2010) in the context of state space modeling of time series. However, MCMC estimation based on different data augmentation schemes will often be efficient in separate regions of the parameter space, as demonstrated e.g. by Kastner and Frühwirth-Schnatter (2014) in the context of univariate SV models. This suggests to combine different data augmentation schemes to obtain an improved sampler.

Marginal data augmentation employs a randomly sampled "working parameter" to transform the baseline parameterization to an expanded, unidentified latent variable model in which the model parameters are updated conditional on the (randomly) transformed latent variables. This technique has been applied to the basic factor model, using the undefined scaling of the factors as a working parameter (Ghosh and Dunson, 2009; Frühwirth-Schnatter and Lopes, 2015), however, it is not easily extended to factor SV models, in particular if the latent volatilities should be part of the acceleration scheme.

The *ancillarity-sufficiency interweaving strategy (ASIS)*, introduced by Yu and Meng (2011), provides another principled way to interweave two different data augmentation schemes by re-sampling certain parameters conditional on the latent variables in an alternative parameterization of the model, thereby combining "best of different worlds". ASIS has been successfully employed in a variety of contexts such as univariate SV models (Kastner and Frühwirth-Schnatter, 2014) and dynamic linear state space models (Simpson, 2015; Simpson et al., 2015). To boost Algorithm 1, we apply ASIS to the factor SV model in the present paper. Two interweaving strategies called *shallow interweaving* and *deep interweaving* are derived in Section 3.3.1, where the diagonal elements $\Lambda_{11}, \ldots, \Lambda_{rr}$ of the factor loadings matrix are resampled in Step (b*) in two alternative parameterization of the factor SV model.

### 3.3.1 Shallow and Deep Interweaving

As discussed in Section 2.2, our baseline parameterization (3) and (4) is just one of several alternative ways to handle the scaling problem inherent in factor SV models and this identification issue is exploited by our schemes.

The parameterization underlying shallow interweaving constrains the diagonal elements of the factor loadings matrix to be equal to 1, whereas the variances of the factors depend on $r$ unknown scaling parameters $\boldsymbol{D} = \text{diag}(\Lambda_{11}, \ldots, \Lambda_{rr})$. The latent volatility processes are modeled as in the baseline parameterization (4), whereas the factor model takes a different form:

$$\boldsymbol{y}_t = \boldsymbol{\Lambda}^\star \boldsymbol{f}_t^\star + \boldsymbol{\varepsilon}_t, \qquad \boldsymbol{f}_t^\star | \boldsymbol{h}_t, \Lambda_{11}, \ldots, \Lambda_{rr} \sim \mathcal{N}_r\big(\boldsymbol{0}, \boldsymbol{D}^2 \boldsymbol{V}_t(\boldsymbol{h}_t)\big)\,, \tag{7}$$

with a lower triangular loading matrix $\boldsymbol{\Lambda}^\star$ where $\Lambda_{11}^\star = 1, \ldots, \Lambda_{rr}^\star = 1$. The idiosyncratic

errors $\boldsymbol{\varepsilon}_t$ are distributed as in (3). Factor model (3) in the baseline parameterization can be transformed into factor model (7) through a simple linear transformation:

$$\boldsymbol{f}_t^\star = \boldsymbol{D}\boldsymbol{f}_t, \qquad t = 1, \ldots, T, \qquad \boldsymbol{\Lambda}^\star = \boldsymbol{\Lambda}\boldsymbol{D}^{-1}. \tag{8}$$

Boosting through shallow interweaving consists of three parts. First, transformation (8) is used to move the current posterior draws of the latent factors $\boldsymbol{f}_t$ and the factor loading matrix $\boldsymbol{\Lambda}$ from the baseline parameterization to parameterization (7). Second, the scale parameters $\Lambda_{11}, \ldots, \Lambda_{rr}$, contained in $\boldsymbol{D}$, are resampled in parameterization (7), conditionally on the transformed values $\boldsymbol{f}^\star$, from $p(\Lambda_{11}, \ldots, \Lambda_{rr} | \boldsymbol{f}^\star, \boldsymbol{\Lambda}^\star, \boldsymbol{h})$. Finally, the new values $\Lambda_{11}^{\text{new}}, \ldots, \Lambda_{rr}^{\text{new}}$ are used in transformation (8) to move $\boldsymbol{f}_t^\star$ and $\boldsymbol{\Lambda}^\star$ back to new draws $\boldsymbol{f}_t^{\text{new}}$ and $\boldsymbol{\Lambda}^{\text{new}}$ in the baseline parameterization.

It is evident from transformation (8) that shallow interweaving only affects the factors and the factor loading matrix, whereas the latent volatilities remain untouched. This makes shallow interweaving also applicable to static factor models. However, to achieve boosting also for the $r$ factor volatilities, deep interweaving is based on an alternative SV model for the factor volatilities where the level is assumed to be unknown. The parameterization underlying deep interweaving relies on the factor model

$$\boldsymbol{y}_t = \boldsymbol{\Lambda}^\star \boldsymbol{f}_t^\star + \boldsymbol{\varepsilon}_t, \qquad \boldsymbol{f}_t^\star | \boldsymbol{h}_{m+j,\bullet}^\star \sim \mathcal{N}_r\big(\boldsymbol{0}, \operatorname{diag}\big(e^{h_{m+1,t}^\star}, \ldots, e^{h_{m+r,t}^\star}\big)\big), \tag{9}$$

where $\boldsymbol{\Lambda}^\star$ has the same structure as in the factor model (7) for shallow interweaving and the idiosyncratic errors $\boldsymbol{\varepsilon}_t$ are distributed as before, with the univariate SV models for the $m$ underlying volatilities following (4). However, the $r$ latent factor volatilities $h_{m+j,t}^\star$ follow alternative univariate SV models where the level is $\mu_{m+j} = \log \Lambda_{jj}^2$ rather than zero:

$$h_{m+j,t}^\star \;=\; \mu_{m+j}(1 - \phi_{m+j}) + \phi_{m+j} h_{m+j,t-1}^\star + \sigma_{m+j}\eta_{m+j,t}. \tag{10}$$

This parameterization can be motivated by moving the parameters $\Lambda_{11}, \ldots, \Lambda_{rr}$ from factor model (7) into SV model (10), since:

$$f_{jt}^\star | \Lambda_{jj}, h_{m+j,t} \sim \mathcal{N}\big(0, \Lambda_{jj}^2 e^{h_{m+j,t}}\big) = \mathcal{N}\Big(0, e^{\log \Lambda_{jj}^2 + h_{m+j,t}}\Big) = \mathcal{N}\big(0, e^{h_{m+j,t}^\star}\big).$$

Hence, the baseline parameterization can be transformed into parameterization (9) and (10) by applying transformation (8) to the factors and the factor loadings, as well as the

following transformation to the factor volatilities:

$$h^{\star}_{m+j,t} = h_{m+j,t} + \log \Lambda^2_{jj}, \quad t = 0, \ldots, T, \quad j = 1, \ldots, r. \tag{11}$$

Boosting through deep interweaving also consists of three parts: first, transformations (8) and (11) are used to move from the current draws of $\boldsymbol{f}_t$, $\boldsymbol{\Lambda}$ and the factor log-variances $h_{m+j,t}$ from the baseline parameterization to parameterization (9) and (10). Second, the scale parameters $\Lambda_{11}, \ldots, \Lambda_{rr}$ are resampled in parameterization (10) conditionally on the transformed values $\boldsymbol{h}^{\star}_{m+j,\bullet} = (h^{\star}_{m+j,0}, \ldots, h^{\star}_{m+j,T})'$ from $p(\Lambda_{11}, \ldots, \Lambda_{rr} | \boldsymbol{h}^{\star}_{m+1,\bullet}, \cdots, \boldsymbol{h}^{\star}_{m+r,\bullet}, \boldsymbol{\Lambda}^{\star})$. Based on the new values $\Lambda^{\text{new}}_{11}, \ldots, \Lambda^{\text{new}}_{rr}$, transformations (8) and (11) are inverted to move $\boldsymbol{f}^{\star}_t$, $\boldsymbol{\Lambda}^{\star}$, $h^{\star}_{m+j,t}$ back to new draws $\boldsymbol{f}^{\text{new}}_t$, $\boldsymbol{\Lambda}^{\text{new}}$, $h^{\text{new}}_{m+j,t}$ in the baseline parameterization.

Both interweaving strategies are summarized in Algorithm 2. Details on resampling $\Lambda^{\text{new}}_{jj}$ are provided in Section 3.3.2. It is evident that deep interweaving affects the factors, the factor loading matrix as well as the latent factor volatilities and for this reason is more effective in boosting MCMC for factor SV models than shallow interweaving.

**Algorithm 2** (**Shallow and Deep Interweaving**)**.** Denote the original posterior draws for $\boldsymbol{\Lambda}_{\bullet,j}$, $\boldsymbol{f}_{j,\bullet}$, and $\boldsymbol{h}_{m+j,\bullet}$ in Algorithm 1 by $\boldsymbol{\Lambda}^{\text{old}}_{\bullet,j}$, $\boldsymbol{f}^{\text{old}}_{j,\bullet}$, and $\boldsymbol{h}^{\text{old}}_{m+j,\bullet}$ and perform following steps independently for each $j = 1, \ldots, r$ in Step (b*):

(b*-1) Determine the vector $\boldsymbol{\Lambda}^{\star}_{\bullet,j} = (\Lambda^{\star}_{j+1,j}, \ldots, \Lambda^{\star}_{j+1,m})'$, containing the free parameters $\Lambda^{\star}_{ij} = \Lambda^{\text{old}}_{ij} / \Lambda^{\text{old}}_{jj}$ in the $j$th column of the transformed factor loading matrix $\boldsymbol{\Lambda}^{\star}$.

(b*-2) For shallow interweaving, define $\boldsymbol{f}^{\star}_{j,\bullet} = \Lambda^{\text{old}}_{jj} \boldsymbol{f}^{\text{old}}_{j,\bullet}$ and sample a new value $\Lambda^{\text{new}}_{jj}$ from $p(\Lambda_{jj} | \boldsymbol{\Lambda}^{\star}_{\bullet,j}, \boldsymbol{f}^{\star}_{j,\bullet}, \boldsymbol{h}_{m+j,\bullet})$. For deep interweaving, define $\boldsymbol{h}^{\star}_{m+j,\bullet} = \boldsymbol{h}^{\text{old}}_{m+j,\bullet} + 2 \log |\Lambda^{\text{old}}_{jj}|$ and sample $\Lambda^{\text{new}}_{jj}$ from $p(\Lambda_{jj} | \boldsymbol{\Lambda}^{\star}_{\bullet,j}, \boldsymbol{h}^{\star}_{m+j,\bullet}, \phi_{m+j}, \sigma_{m+j})$; see Section 3.3.2 for details.

(b*-3) Update $\boldsymbol{\Lambda}_{\bullet,j}$, $\boldsymbol{f}_{j,\bullet}$, and, for deep interweaving, also $\boldsymbol{h}_{m+j,\bullet}$:

$$\boldsymbol{\Lambda}_{\bullet,j} = \frac{\Lambda^{\text{new}}_{jj}}{\Lambda^{\text{old}}_{jj}} \boldsymbol{\Lambda}^{\text{old}}_{\bullet,j}, \qquad \boldsymbol{f}_{j,\bullet} = \frac{\Lambda^{\text{old}}_{jj}}{\Lambda^{\text{new}}_{jj}} \boldsymbol{f}^{\text{old}}_{j,\bullet}, \qquad \boldsymbol{h}_{m+j,\bullet} = \boldsymbol{h}^{\text{old}}_{m+j,\bullet} + 2 \log \left| \frac{\Lambda^{\text{old}}_{jj}}{\Lambda^{\text{new}}_{jj}} \right|.$$

### 3.3.2 Sampling the scaling parameters in the alternative representations

To derive the full conditional posterior distribution of $\Lambda_{jj}$, we combine the appropriate full conditional likelihood function with the Gaussian prior $\Lambda_{jj} \sim \mathcal{N}(0, B_{\Lambda})$. In addition,

the prior $\boldsymbol{\Lambda}^{\star}_{\bullet,j}|\Lambda^2_{jj} \sim \mathcal{N}_{m-j}\big(0, B_\Lambda/\Lambda^2_{jj}\boldsymbol{I}_{m-j}\big)$ of the transformed factor loadings in column $j$ contributes to the posterior distribution of $\Lambda^2_{jj}$ because its scale depends on $\Lambda^2_{jj}$.

For shallow interweaving, we sample $\Lambda^2_{jj}$ and define $\Lambda^{\text{new}}_{jj}$ as the square root of $\Lambda^2_{jj}$. Combining the likelihood obtained from factor model (7) with the implied prior $\Lambda^2_{jj} \sim \mathcal{G}(1/2, 1/(2B_\Lambda))$ and $p(\boldsymbol{\Lambda}^{\star}_{\bullet,j}|\Lambda^2_{jj})$ yields

$$p(\Lambda^2_{jj}|\boldsymbol{\Lambda}^{\star}_{\bullet,j}, \boldsymbol{f}^{\star}_{j,\bullet}, \boldsymbol{h}_{m+j,\bullet}) \;\; \propto \;\; p(\boldsymbol{f}^{\star}_{j,\bullet}|\boldsymbol{h}_{m+j,\bullet}, \Lambda^2_{jj})p(\boldsymbol{\Lambda}^{\star}_{\bullet,j}|\Lambda^2_{jj})p(\Lambda^2_{jj}),$$

which is the product of $T$ univariate Gaussian densities with $\Lambda^2_{jj}$ appearing as part of the variance, $(m-j)$ univariate Gaussian densities with $\Lambda^2_{jj}$ appearing as part of the precision, and one Gamma density with $\Lambda^2_{jj}$ appearing as argument. Thus, the resulting posterior distribution of $\Lambda^2_{jj}$ is Generalized Inverse Gaussian, i.e.

$$\Lambda^2_{jj}|\boldsymbol{\Lambda}^{\star}_{\bullet,j}, \boldsymbol{f}^{\star}_{j,\bullet}, \boldsymbol{h}_{m+j,\bullet} \sim \text{GIG}\left(\frac{1+m-j-T}{2}, \frac{1}{B_\Lambda}\left(1+\sum_{i=j+1}^{m}{\Lambda^{*\,2}_{ij}}\right), \sum_{t=1}^{T}\frac{f^{*\,2}_{jt}}{e^{h_{m+j,t}}}\right), \quad (12)$$

where $\text{GIG}(p, a, b)$ has a density proportional to $x^{p-1}\exp\left\{-\frac{1}{2}(ax+b/x)\right\}$. Given an efficient method to draw from the GIG such as the adaptive rejection sampling algorithm provided by Hörmann and Leydold (2013), sampling from (12) is straightforward. For practical implementation, we use the R package **GIGrvg** (Leydold and Hörmann, 2015) which provides a C/C++ interface to avoid the cost of interpreting code at every MCMC iteration, thereby rendering the re-updating negligible in terms of overall computation time.

For deep interweaving, we sample $\Lambda_{jj}$ indirectly through $\mu_{m+j} = \log\Lambda^2_{jj}$. Combining the implied prior $p(\mu_{m+j}) \propto \exp\{\mu_{m+j}/2 - e^{\mu_{m+j}}/(2B_\Lambda)\}$with the likelihood obtained from SV model (10) and the priors $h^{\star}_{m+j,0}|\mu_{m+j}, \phi_{m+j}, \sigma^2_{m+j} \sim \mathcal{N}\big(\mu_{m+j}, \sigma^2_{m+j}/(1-\phi^2_{m+j})\big)$ and $\boldsymbol{\Lambda}^{\star}_{\bullet,j}|\mu_{m+j} \sim \mathcal{N}_{m-j}(0, B_\Lambda e^{-\mu_{m+j}}\boldsymbol{I}_{m-j})$ yields the posterior

$$p(\mu_{m+j}|\boldsymbol{\Lambda}^{\star}_{\bullet,j}, \boldsymbol{h}^{\star}_{m+j,\bullet}, \phi_{m+j}, \sigma_{m+j}) \propto p(\boldsymbol{h}^{\star}_{m+j,\bullet}|\mu_{m+j}, \phi_{m+j}, \sigma^2_{m+j})p(\boldsymbol{\Lambda}^{\star}_{\bullet,j}|\mu_{m+j})p(\mu_{m+j}),$$

which has a non-standard form. To generate draws from this density, we consider an independence Metropolis-Hastings update in the spirit of Kastner and Frühwirth-Schnatter (2014). Since the likelihood $p(h^{\star}_{m+j,1}, \ldots, h^{\star}_{m+j,T}|h^{\star}_{m+j,0}, \mu_{m+j}, \phi_{m+j}, \sigma^2_{m+j})$ is the kernel of a Gaussian density in $\mu_{m+j}$, it can be used to construct an auxiliary posterior under a conjugate auxiliary prior $p_{\text{aux}}(\mu_{m+j}|\sigma^2_{m+j}, \phi_{m+j}) \sim \mathcal{N}\big(0, B_0\sigma^2_{m+j}/(1-\phi_{m+j})^2\big)$ with $B_0$

large. Consequently, we draw a proposal $\mu_{m+j}^{\text{prop}}$ from the $\mathcal{N}\big(m_j^\mu, S_j^\mu\big)$ distribution with:

$$m_j^\mu = \frac{\sum_{t=1}^{T-1} h_{m+j,t}^\star + (h_{m+j,T}^\star - \phi_{m+j} h_{m+j,0}^\star)/(1-\phi_{m+j})}{T + 1/B_0}, \quad S_j^\mu = \frac{\sigma_{m+j}^2/(1-\phi_{m+j})^2}{T + 1/B_0}.$$

Denoting the old value of $\mu_{m+j}$ by $\mu_{m+j}^{\text{old}}$, this proposal gets accepted with probability $\min(1, R)$, where

$$R = \frac{p(\boldsymbol{\Lambda}_{\bullet,j}^\star | \mu_{m+j}^{\text{prop}}) p(h_{m+j,0}^\star | \mu_{m+j}^{\text{prop}}, \phi_{m+j}, \sigma_{m+j}^2) p(\mu_{m+j}^{\text{prop}})}{p(\boldsymbol{\Lambda}_{\bullet,j}^\star | \mu_{m+j}^{\text{old}}) p(h_{m+j,0}^\star | \mu_{m+j}^{\text{old}}, \phi_{m+j}, \sigma_{m+j}^2) p(\mu_{m+j}^{\text{old}})} \times \frac{p_{\text{aux}}(\mu_{m+j}^{\text{old}} | \sigma_{m+j}^2, \phi_{m+j})}{p_{\text{aux}}(\mu_{m+j}^{\text{prop}} | \sigma_{m+j}^2, \phi_{m+j})}.$$

In case of acceptance, set $\Lambda_{jj}^{\text{new}} = e^{\mu_{m+j}^{\text{prop}}/2}$; otherwise, let $\Lambda_{jj}^{\text{new}} = \Lambda_{jj}^{\text{old}}$.

# 4 Simulation Study

In order to compare the different algorithms in terms of sampling efficiency, a simple simulation experiment is conducted. We use $m = 10$ (simulated) series and $r = 2$ (simulated) factors to generate $T = 1000$ observations, thereby imposing the usual lower triangular constraint. The data generating parameter values – listed in Table B.1 in Appendix B – are kept constant, whereas the data generating process as well as the estimation procedure is repeated 100 times. Each time, the draws are initialized at the data generating values; then, 5100 000 draws are obtained of which 100 000 are discarded as burn-in. Prior hyperparameters are set as follows: $B_\Lambda = 1$, $b_\mu = 0$, $B_\mu = 100$, $a_0 = 20$, $b_0 = 1.5$, and $B_\sigma = 1$.

To gain insight about the mixing behavior of the different sampling strategies, trace plots (i.e. time series plots of the MCMC draws) for $\Lambda_{11}$ are displayed in the left hand panel of Figure 1. Even though the plots depict only the first 10 000 iterations after burn-in, it becomes very clear that the mixing of the non-interwoven sampler is extremely slow. The algorithm doesn't seem to explore the posterior distribution within a reasonable amount of draws which renders this output practically useless in terms of posterior inference. Moreover, the burn-in period for this sampler would need to be chosen extremely long to avoid strong dependence on the starting values. This situation is slightly mitigated when using shallow interweaving; nevertheless, mixing is still poor and for reliable posterior inference many draws are required. Turning towards the deeply interwoven sampler, one can observe quick mixing and hardly any visible autocorrelation.

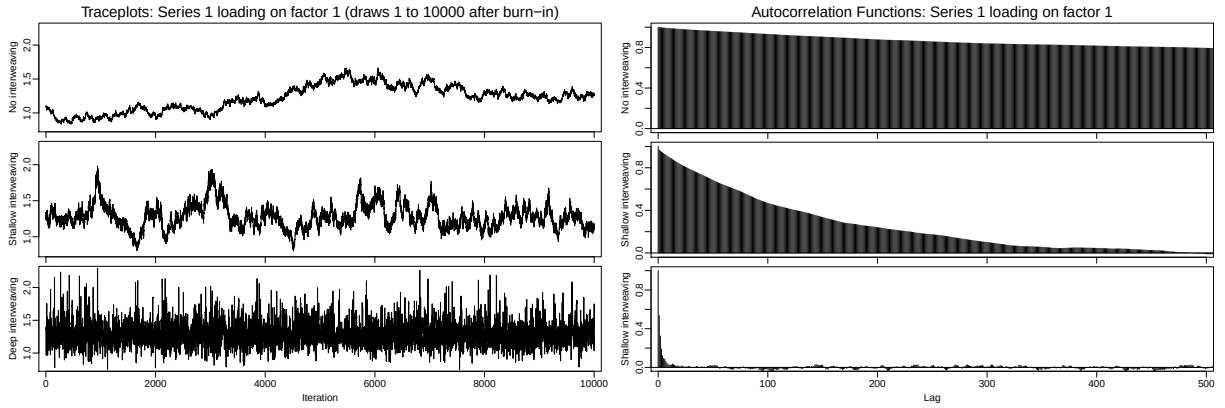


Figure 1: Trace plots of 10 000 draws from $p(\Lambda_{11}|\boldsymbol{y})$ (left hand side) and empirical autocorrelation functions of all 5000 000 draws (right hand side) obtained via the standard sampler (top), shallow interweaving (middle), and deep interweaving (bottom).

Investigating autocorrelations of the draws via the empirical autocorrelation function confirms this picture; the right hand panel of Figure 1 shows that the empirical autocorrelation function for draws from $p(\Lambda_{11}|\boldsymbol{y})$ decays very quickly for the sampler using deep interweaving which is not the case for the other two samplers, where visible autocorrelation remains even at large lags.

A convenient and common way of measuring sampling (in)efficiency is by means of the *inefficiency factor (IF)*, sometimes called the *(integrated) autocorrelation time*. It is defined as the ratio of the numerical variance of a statistic which is estimated from the Markov chain to the variance of that statistic when estimated from independent draws, thereby quantifying the relative loss of efficiency when inferring from correlated as opposed to independent samples. In other words, to achieve the same inferential accuracy about some posterior moment of some parameter as with $k$ independent samples, IF $\times\, k$ MCMC draws are required. For the paper at hand, we use the R package **coda** (Plummer et al., 2006) to estimate the inefficiency factors.

Moreover, when investigating performance of MCMC samplers through simulation studies, it is of great importance to take sample variation into account; even when identical parameter values are used for the generation of latent variables and data, sampling (in)efficiency may vary greatly. To illustrate this, we show box plots of the inefficiency factors stemming from repeated data generating processes in the left panel of Figure 2.

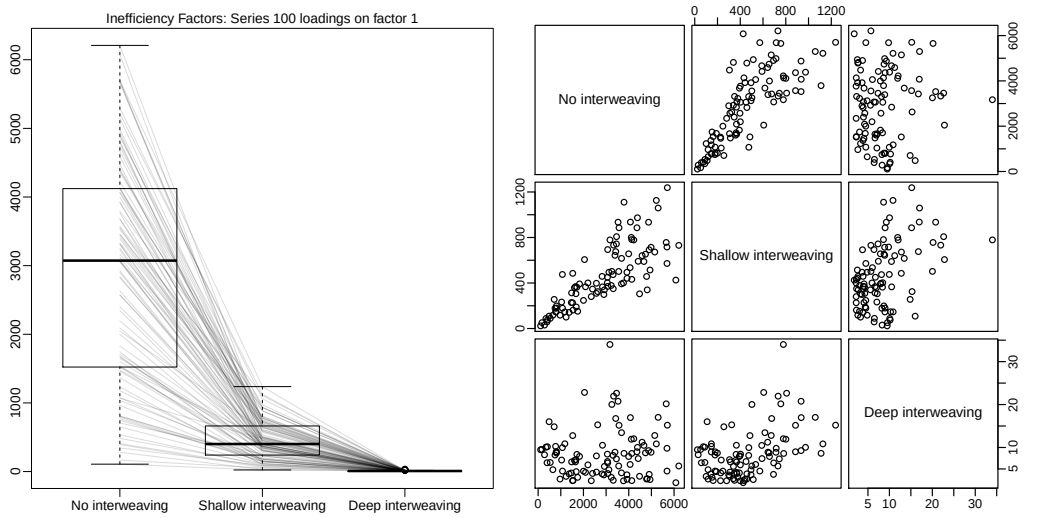


Figure 2: Left: Boxplots of estimated inefficiency factors for posterior draws from $p(\Lambda_{11}|\boldsymbol{y}^{[i]})$ where $\boldsymbol{y}^{[i]}, i \in \{1, \ldots, 100\}$, denote artificially generated data sets whose underlying parameters are identical, cf. Table B.1 in Appendix B. Right: Pairwise scatter plots thereof.

Note the enormous range for the standard sampler; depending on the data, IFs of 5000 or more are not uncommon, while at the same time IFs of around 100 can be observed. However, independently of the actual data, interweaving attenuates this effect drastically and increases efficiency uniformly. The right panel of Figure 2 shows pairwise scatter plots of these IFs. Note that shallow interweaving yields efficiency improvements which are more or less independent of the actual data (around five-fold for all data sets), whereas deep interweaving IFs appear less clearly correlated.

To provide a more complete picture, we list the inefficiency factors for all elements of $\boldsymbol{\Lambda}$ for the various algorithms in Table 1, averaged over all 100 runs. Note that shallow interweaving permits efficiency gains of around two- to eight-fold as opposed to the standard sampler, whereas deep interweaving delivers gains up to almost 1000-fold.

It comes as no surprise that sampling (in)efficiencies of draws for the volatility parameters $\mu_i$, $i \in \{1, \ldots, m\}$ as well as $\phi_i$ and $\sigma_i$, $i \in \{1, \ldots, m+r\}$ are not affected substantially by this interweaving strategy, thus they are not reported here. It is however worth noting that inefficiency of factor $\boldsymbol{f}_{j,\bullet}$ and factor log-variance draws $\boldsymbol{h}_{m+j,\bullet}$, $j \in \{1, \ldots, r\}$ may be influenced by bad mixing of $\boldsymbol{\Lambda}$. For illustration, IFs are reported for the final factors $f_{1T}$ and $f_{2T}$ and their log-variances $h_{m+1,T}$ and $h_{m+2,T}$ in Table 2.

| | 1 | 2 |
|---|---|---|
| 1 | 2901.87 | |
| 2 | 2630.89 | 999.88 |
| 3 | 2931.02 | 233.08 |
| 4 | 2936.83 | 673.08 |
| 5 | 2909.98 | 876.57 |
| 6 | 2772.16 | 934.87 |
| 7 | 2303.80 | 958.02 |
| 8 | 1463.70 | 968.13 |
| 9 | 605.16 | 974.56 |
| 10 | 113.16 | 976.10 |

(a) No interweaving

| | 1 | 2 |
|---|---|---|
| 1 | 462.07 | |
| 2 | 434.09 | 186.39 |
| 3 | 460.59 | 80.57 |
| 4 | 457.71 | 141.30 |
| 5 | 451.84 | 164.85 |
| 6 | 437.95 | 174.34 |
| 7 | 406.73 | 178.91 |
| 8 | 337.78 | 181.54 |
| 9 | 215.53 | 183.30 |
| 10 | 67.89 | 184.30 |

(b) Shallow interweaving

| | 1 | 2 |
|---|---|---|
| 1 | 8.56 | |
| 2 | 10.81 | 8.69 |
| 3 | 8.48 | 10.92 |
| 4 | 8.55 | 9.00 |
| 5 | 8.79 | 8.46 |
| 6 | 9.33 | 8.25 |
| 7 | 10.38 | 8.19 |
| 8 | 12.36 | 8.17 |
| 9 | 16.07 | 8.14 |
| 10 | 22.07 | 8.18 |

(c) Deep interweaving

Table 1: Average IFs for factor loadings matrix $\mathbf{\Lambda}$.

| | $f_{1,1000}$ | $f_{2,1000}$ | $h_{11,1000}$ | $h_{12,1000}$ |
|---|---|---|---|---|
| No interweaving | 121.54 | 53.30 | 232.23 | 21.59 |
| Shallow interweaving | 48.98 | 20.42 | 127.24 | 16.27 |
| Deep interweaving | 3.79 | 3.76 | 5.44 | 5.85 |

Table 2: Average IFs for the final factors $f_{jT}$ and their log-variances $h_{m+j,T}$, $j \in \{1, 2\}$.

# 5 Application to Exchange Rate Data

In this section, we analyze exchange rates with respect to EUR. Data was obtained from the European Central Bank's Statistical Data Warehouse and ranges from April 1, 2005 to August 6, 2015. It contains $m = 26$ (all which were available for this time frame) daily exchange rates on 2650 days listed in Table 3. For further analysis, we thus use $T = 2649$ demeaned percentage log returns which are displayed in Figure 3. Common "stylized facts" of financial time series are clearly visible; note e.g. the obvious volatility clustering during 2008 and 2009 and again throughout late 2014 and early 2015. To put the robustness of our sampler to the test, we use the data as-is, i.e. without preprocessing or without excluding

series containing extreme outliers such as the notedly CHF spike on January 14, 2015 or the near collapse of RUB around December 16, 2014.

| | | | | | |
|---|---|---|---|---|---|
| AUD | Australia dollar | CAD | Canada dollar | CHF | Switzerland franc |
| CNY | China yuan renminbi | CZK | Czech R. koruna | DKK | Denmark krone |
| GBP | UK pound | HKD | Hong Kong dollar | HRK | Croatia kuna |
| HUF | Hungary forint | IDR | Indonesia rupiah | JPY | Japan yen |
| KRW | South Korea won | MYR | Malaysia ringgit | NOK | Norway krone |
| NZD | New Zealand dollar | PHP | Philippines peso | PLN | Poland zloty |
| RON | Romania fourth leu | RUB | Russia ruble | SEK | Sweden krona |
| SGD | Singapore dollar | THB | Thailand baht | TRY | Turkey lira |
| USD | US dollar | ZAR | South Africa rand | | |

Table 3: Currency abbreviations.

## 5.1 Model Specification

An important modeling decision concerns the number of factors. We analyze models with up to 10 factors and conclude that a 4-factor model fits the data well without observing "spurious" factors, i.e. factors which are significantly loaded on by only one or two series. When identification of $\mathbf{\Lambda}$ is of no concern (i.e. for covariance matrix estimation only) higher-order models are clearly possible and computationally not substantially more involved. For the paper at hand, we restrict ourselves to 4 factors mainly to simplify presentation.

A further important step of preprocessing is the appropriate ordering of the variables. As outlined in Section 2.2, we impose the usual lower triangular structure for the factor loadings matrix to guarantee identifiability of the factor loadings draws. This, however, makes inference on the factor loadings matrix dependent on the ordering of the variables. For the remainder of this section, we use a preliminary maximum-likelihood factor analysis via the R (R Core Team, 2016) function `factanal` to determine the ordering by placing the variable with the highest loading on factor 1 first, the variable with the highest loading on factor 2 second (unless this variable is already placed first, in which case the variable with the second highest loading is taken). For the data at hand, HKD is placed first, HUF

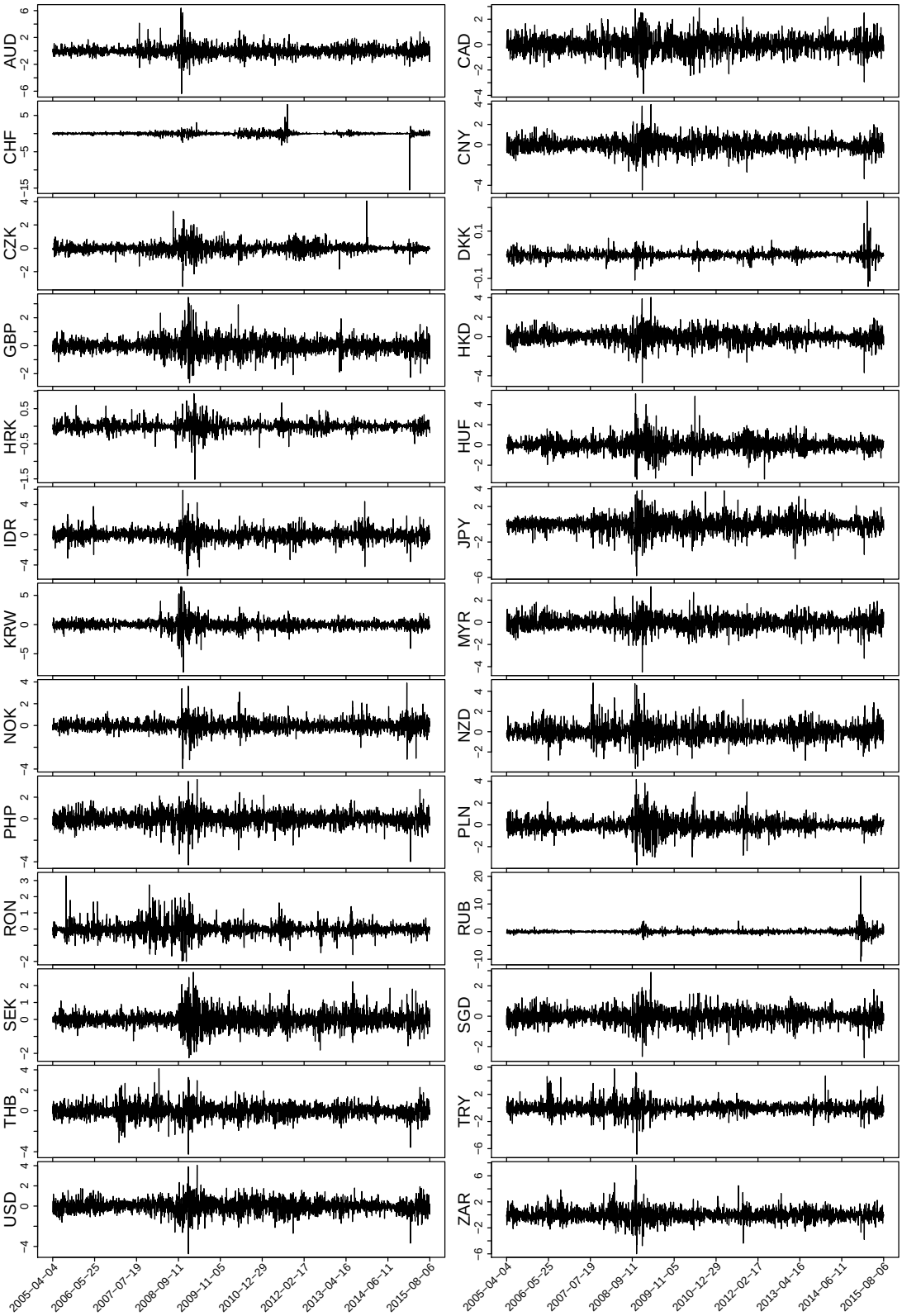


Figure 3: Demeaned log returns of EUR exchange rates.

is placed second, AUD is placed third, and MYR leads the forth factor, see also Figure C.1 in Appendix C. Note that this procedure is fast and it can easily be automated.

Prior hyperparameters are the same as for the simulation study in Section 4. We run each sampler for 110 000 iterations, whereas the first 10 000 draws are discarded as burn-in. Even after this substantial amount of iterations, it is not clear that the sampler without interweaving has properly converged; we therefore omit its presentation. IFs from the interwoven samplers are presented in Table C.1 in Appendix C.

Finally, we identify the signs of the loadings in the post-processing phase by investigating the MCMC draws. For each factor, the series whose posterior absolute loadings distribution is furthest away from zero is assigned a positive sign, the other loadings are aligned thereafter, see also Section A.4 in Appendix C. Again, this can trivially be automated.

## 5.2 Posterior Factor Volatilities and their Loadings

We begin by discussing the log-variances of the latent factors, visualized in Figure 4, alongside the corresponding factor loadings whose posterior distribution is depicted in Figure 5, with means presented in Table 4 and medians visualized in Figure C.1 in Appendix C.

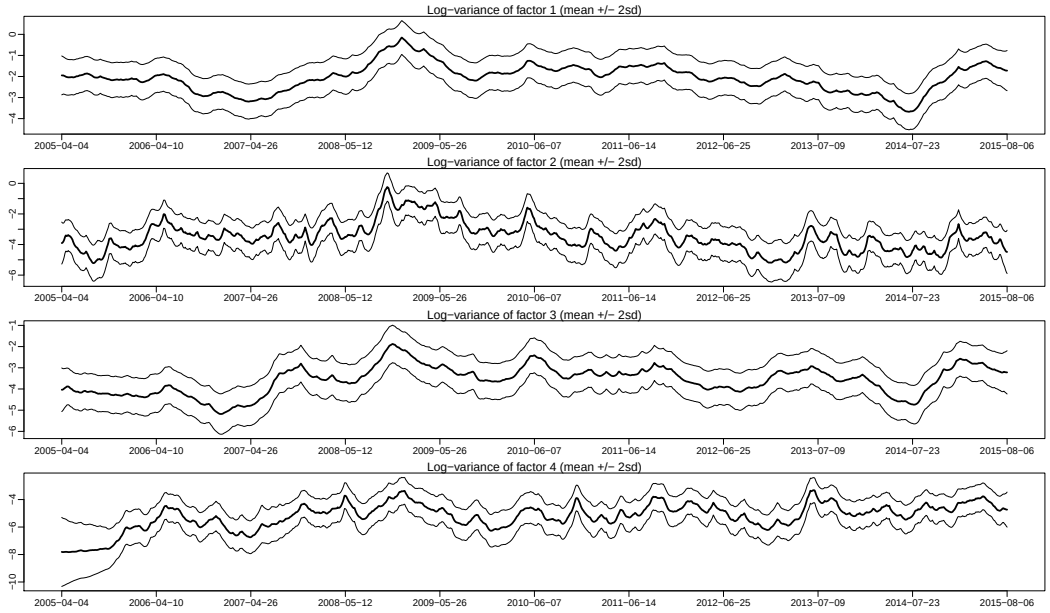


Figure 4: Marginal posteriors of the factor log-variances $h_{m+j,t}, j = 1, \ldots, 4$ (mean$\pm 2\times$sd).

| | $\mathbf{\Lambda}_{\bullet,1}$ | $\mathbf{\Lambda}_{\bullet,2}$ | $\mathbf{\Lambda}_{\bullet,3}$ | $\mathbf{\Lambda}_{\bullet,4}$ | | $\mathbf{\Lambda}_{\bullet,1}$ | $\mathbf{\Lambda}_{\bullet,2}$ | $\mathbf{\Lambda}_{\bullet,3}$ | $\mathbf{\Lambda}_{\bullet,4}$ |
|---|---|---|---|---|---|---|---|---|---|
| AUD | 0.406 | 1.170 | 2.701 | * | MYR | 1.252 | 0.382 | 0.573 | 2.441 |
| CAD | 0.849 | 0.805 | 1.354 | | NOK | | 0.615 | 0.687 | |
| CHF | | -0.192 | | | NZD | 0.331 | 1.082 | 2.592 | |
| CNY | 1.551 | | | 0.067 | PHP | 1.297 | 0.444 | 0.375 | 1.697 |
| CZK | -0.096 | 0.591 | | | PLN | -0.277 | 1.783 | | 0.279 |
| DKK | 0.002 | | | | RON | -0.049 | 0.538 | | |
| GBP | 0.589 | 0.237 | 0.612 | | RUB | 0.793 | 0.104 | 0.136 | 0.248 |
| HKD | 1.570 | * | * | * | SEK | -0.048 | 0.523 | 0.515 | |
| HRK | | | | | SGD | 1.036 | 0.256 | 0.628 | 1.460 |
| HUF | -0.336 | 2.045 | * | * | THB | 1.323 | 0.088 | 0.264 | 1.037 |
| IDR | 1.359 | 0.419 | 0.329 | 1.150 | TRY | 0.825 | 1.722 | 0.518 | 0.930 |
| JPY | 1.145 | -0.894 | 0.318 | 0.850 | USD | 1.572 | | -0.003 | -0.006 |
| KRW | 1.071 | 0.607 | 0.732 | 1.950 | ZAR | 0.421 | 2.310 | 1.164 | 1.432 |

Table 4: Posterior means of $p(\mathbf{\Lambda}|\boldsymbol{y})$, in alphabetical order. Blank entries signify that the respective marginal distribution is not bound away from zero with at least 99% posterior probability. Starred entries are those which have been set to zero a priori.

The first factor can clearly be interpreted as the USD-driven one, as the pegged triplet USD, CNY and HKD loads very highly on this factor, alongside many other currencies. Its volatility is generally very smooth, rising in the aftermath of the 2008 financial crisis and going down again after 2009; a second increase can be seen in the second half of 2014, possibly in connection with the Greek government-debt crisis. Factor 2's log-variance appears slightly less persistent and more volatile, it is driven by ZAR, the only African currency in the sample, alongside Eastern Europe's / Southwestern Asia's HUF, PLN and TRY. Interestingly, JPY loads negatively on this factor. The third factor shows a similar overall pattern as the first. The highest loading series for this factor are AUD and NZD, emphasizing the Trans-Tasman relations. Other commodity currencies such as ZAR and CAD also load highly on this factor. Factor 4 is clearly driven by the currencies of the Tiger Cub economies such as MYR, KRW, PHP and SGD.

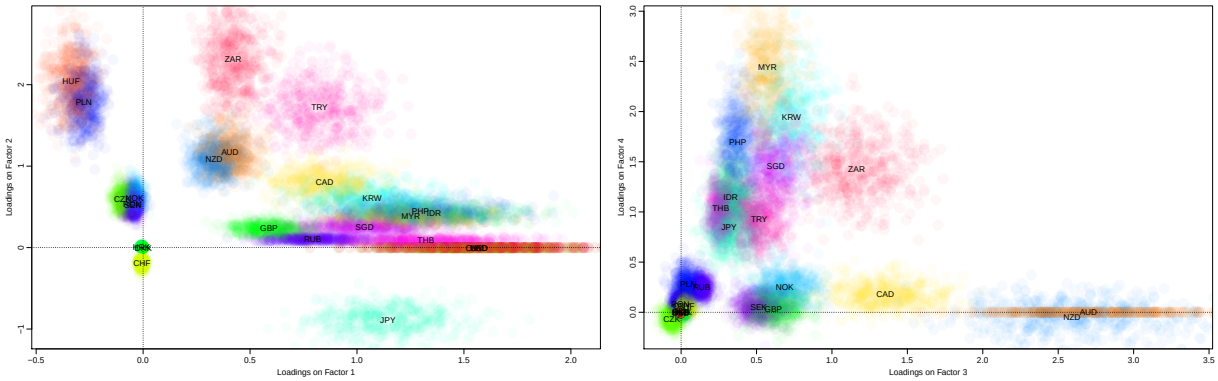


Figure 5: Marginal posterior distribution of the factor loadings, visualized through arbitrarily colored scatterplots of MCMC draws.

## 5.3 Posterior Volatilities and Correlations

In order not to overload the graphical displays used to visualize the results of the analysis, we display the results for a two-year period only for the rest of this section. More specifically, we look at the years 2008 and 2009, covering the most volatile span during the financial crisis. Irrespectively of that, the full data set has been used for estimation and other time spans could be displayed analogously.

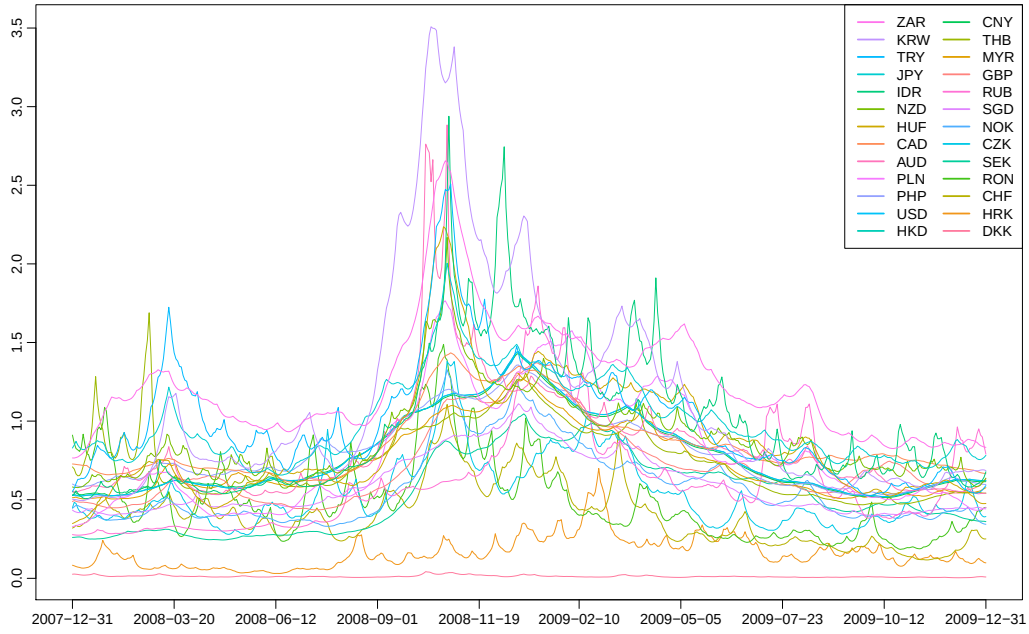


Figure 6: Implied volatilities of exchange rates with respect to the EUR.

We start out by visualizing the marginal posterior means of univariate volatilities for all 26 currencies in Figure 6 from the last day of 2007 until the last day of 2009. Series such as DKK or HRK are (very) closely pegged to EUR and unsurprisingly show very low volatility throughout the crisis. Other European currencies (CHF[1], RON, SEK, CZK) follow suit. Tiger Cub economies such as PHP, HKD, THB, and MYR align very closely with USD and CNY. Most volatile currencies during this period are KRW, ZAR, IDR, JPY, but also TRY, followed by NZD, AUD, and CAD. Overall, it stands out that even though some series-specific ups and downs can be spotted, a common trend is clearly visible.

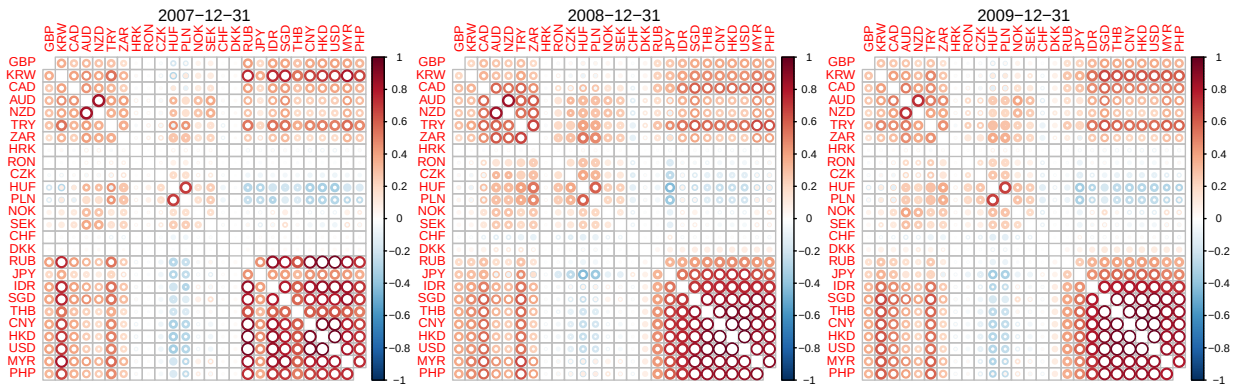


Figure 7: Posterior correlations matrices on the last trading days of 2006, 2007, and 2008. For each element, the size of the outer/inner circle is determined by the posterior mean plus/minus two posterior standard deviations, thereby indicating posterior uncertainty. Color and opacity are determined by the posterior mean.

Next, implied correlation matrices are displayed displayed in Figure 7, exemplified for the last day of 2007, 2008, and 2009. Additionally to displaying the mean posterior pairwise correlations (at the given dates) via color and shading, these plots visualize posterior uncertainty; the outer and inner circles' sizes correspond to posterior mean $\pm$ 2 standard deviations, respectively. The images were generated using the R package **corrplot** (Wei, 2013); its option `hclust` (hierarchical clustering) was used for ordering the series to emphasize the blocks of currencies.

[1]It is interesting to note that the Swiss franc stays comparably stable from a EUR perspective throughout 2007-2009. Very differently during summer 2011, where CHF shows atypical and very high volatility until the Swiss Central Bank sets the minimum exchange rate at CHF 1.20 per EUR 1 on September 6.

To further illustrate variability over time, we determine the posterior means of the pairwise time-varying correlations of USD against the other currencies which are plotted in Figure C.2 in Appendix C. As was to be expected, correlations of CNY and HKD with USD are almost always very close to one; IDR, THB, SGD, MYR, and PHP show rather high correlation throughout. The correlation between USD and RUB on the other hand falls from around 0.9 in early 2008 to around 0.4 in late 2009, whereas THB moves in the opposite direction; its correlation with USD is around 0.5 at the beginning of the time window and increases quickly to around 0.9. Eastern European non-euro currencies, in particular PLN and HUF, appear to be slightly negatively correlation with USD throughout the entire period.

# 6 Conclusion

Estimating time-varying (dynamic) covariance and correlation matrices of financial and economic time series constitutes a current and active area of research. One of the main challenges thereby is the curse of dimensionality, i.e. the fact that the number of elements of these matrices grows quadratically with the number of observed series. We address this issue by imposing a low-dimensional latent factor structure where the factors are allowed to exhibit stochastic volatility and thereby govern co-movement of volatility over time. To conduct reliable statistical inference, we propose novel Bayesian MCMC algorithms which exploit the model-inherent identifiability constraints. By interweaving different (but mathematically equivalent) parameterizations, the proposed strategies substantially improve mixing of draws obtained from the posterior distribution, in particular for the factor loadings matrix. The method proposed is fully automatic in the sense that the end-user is not required to manually adjust any tuning parameters.

In an extensive case study discussing exchange rates with respect to EUR we show that the algorithm plays well with real-world data that exhibits a fair degree of outliers (e.g. CHF, RUB) which are captured through the idiosyncratic stochastic volatility components. The model structure allows for a covariance decomposition in four interpretable factors (USD/CNY driven, Eastern Europe, commodity currencies, Tiger Cub economies). These, alongside the idiosyncratic volatilities, drive the dynamics of the joint correlation

structure. The pairwise correlations with USD range from "almost perfect" (CNY, HKD) over "hardly existent" (CHF, HRK) to "slightly negative" (PLN, HUF) with a varying and time-dependent degree of variability.

## SUPPLEMENTARY MATERIAL

**Web Appendix** containing (A) details of the various sampling steps of Algorithm 1 and details concerning the R package **factorstochvol**, (B) the data generating parameter values for the simulation study in Section 4, and (C) further results for the exchange rate data discussed in Section 5. (.pdf file)

**Video** displaying the time-varying conditional correlation matrix distribution for the full data set (cf. Figure 7). Currently available at `http://statmath.wu.ac.at/~kastner/postcor.avi`. (.avi file)

# References


Aguilar, O. and M. West (2000). Bayesian dynamic factor models and portfolio allocation. *Journal of Business & Economic Statistics 18*(3), 338–357.

Chib, S., F. Nardari, and N. Shephard (2006). Analysis of high dimensional multivariate stochastic volatility models. *Journal of Econometrics 134*(2), 341–371.

Frühwirth-Schnatter, S. and H. F. Lopes (2015). Parsimonious Bayesian Factor Analysis when the Number of Factors is Unknown. Research Report.

Frühwirth-Schnatter, S. and H. Wagner (2010). Stochastic model specification search for Gaussian and partially non-Gaussian state space models. *Journal of Econometrics 154*, 85–100.

Geweke, J. and G. Zhou (1996). Measuring the pricing error of the arbitrage pricing theory. *The Review of Financial Studies 9*(2), 557–587.

Ghosh, J. and D. B. Dunson (2009). Default prior distributions and efficient posterior computation in Bayesian factor analysis. *Journal of Computational and Graphical Statistics 18*, 306–320.

Han, Y. (2006). Asset allocation with a high dimensional latent factor stochastic volatility model. *Review of Financial Studies 19*(1), 237–271.

Harvey, A. C., E. Ruiz, and N. Shephard (1994). Multivariate stochastic variance models. *Review of Economic Studies 61*(2), 247–264.

Hörmann, W. and J. Leydold (2013). Generating generalized inverse Gaussian random variates. *Statistics and Computing 24*(4), 1–11.

Jacquier, E., N. G. Polson, and P. E. Rossi (1994). Bayesian analysis of stochastic volatility models. *Journal of Business & Economic Statistics 12*(4), 371–389.

Jungbacker, B. and S. J. Koopman (2006). Monte Carlo likelihood estimation for three multivariate stochastic volatility models. *Econometric Reviews 25*(2–3), 385–408.

Kastner, G. (2016). Dealing with stochastic volatility in time series using the R package stochvol. *Journal of Statistical Software 69*(5), 1–30.

Kastner, G. and S. Frühwirth-Schnatter (2014). Ancillarity-sufficiency interweaving strategy (ASIS) for boosting MCMC estimation of stochastic volatility models. *Computational Statistics and Data Analysis 76*, 408–423.

Kim, S., N. Shephard, and S. Chib (1998). Stochastic volatility: Likelihood inference and comparison with ARCH models. *Review of Economic Studies 65*(3), 361–393.

Leydold, J. and W. Hörmann (2015). *GIGrvg: Random variate generator for the GIG distribution*. R package version 0.4.

Liesenfeld, R. and J.-F. Richard (2006). Classical and Bayesian analysis of univariate and multivariate stochastic volatility models. *Econometric Reviews 25*(2–3), 335–360.

Lintner, J. (1965). The valuation of risk assets and the selection of risky investments in stock portfolios and capital budgets. *The Review of Economics and Statistics 47*(1), 13–37.

Lopes, H. F. and C. M. Carvalho (2007). Factor stochastic volatility with time varying loadings and Markov switching regimes. *Journal of Statistical Planning and Inference 137*(10), 3082–3091.

Lopes, H. F. and M. West (2004). Bayesian model assessment in factor analysis. *Statistica Sinica 14*(1), 41–67.

Nakajima, J. and M. West (2013). Dynamic factor volatility modeling: A Bayesian latent threshold approach. *Journal of Financial Econometrics 11*(1), 116–153.

Nardari, F. and J. T. Scruggs (2007). Bayesian analysis of linear factor models with latent factors, multivariate stochastic volatility, and APT pricing restrictions. *Journal of Financial and Quantitative Analysis 42*(4), 857–891.

Omori, Y., S. Chib, N. Shephard, and J. Nakajima (2007). Stochastic volatility with leverage: Fast and efficient likelihood inference. *Journal of Econometrics 140*(2), 425–449.

Papaspiliopoulos, O., G. Roberts, and M. Sköld (2007). A general framework for the parameterization of hierarchical models. *Statistical Science 22*, 59–73.

Pitt, M. K. and N. Shephard (1999). Time-varying covariances: A factor stochastic volatility approach. In J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith (Eds.), *Bayesian Statistics 6 – Proceedings of the Sixth Valencia International Meeting*, pp. 547–570. Oxford University Press.

Plummer, M., N. Best, K. Cowles, and K. Vines (2006). CODA: Convergence diagnosis and output analysis for MCMC. *R News 6*(1), 7–11.

R Core Team (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing.

Ross, S. A. (1976, December). The arbitrage theory of capital asset pricing. *Journal of Economic Theory 13*(3), 341–360.

Sentana, E. and G. Fiorentini (2001). Identification, estimation and testing of conditionally heteroskedastic factor models. *Journal of Econometrics 102*(2), 143–164.

Sharpe, W. F. (1964). Capital asset prices: A theory of market equilibrium under conditions of risk. *The Journal of Finance 19*(3), 425–442.

Shephard, N. (1996). Statistical aspects of ARCH and stochastic volatility. In D. R. Cox, O. E. Barndorff-Nielsen, and D. V. Hinkley (Eds.), *Time Series Models in Econometrics, Finance and Other Fields*, pp. 1–67. Chapman & Hall.

Simpson, M. (2015). Application of interweaving in DLMs to an exchange and specialization experiment. In S. Frühwirth-Schnatter, A. Bitto, G. Kastner, and A. Posekany (Eds.), *Bayesian Statistics from Methods to Models and Applications – Research from BAYSM 2014*, Volume 126 of *Springer Proceedings in Mathematics & Statistics*, pp. 75–90. Springer.

Simpson, M., J. Niemi, and V. Roy (2015). Interweaving Markov chain Monte Carlo strategies for efficient estimation of dynamic linear models. *Journal of Computational and Graphical Statistics (forthcoming)*.

van Dyk, D. and X.-L. Meng (2001). The art of data augmentation. *Journal of Computational and Graphical Statistics 10*, 1–50.

Wei, T. (2013). *corrplot: Visualization of a correlation matrix*. R package version 0.73.

Yu, Y. and X.-L. Meng (2011). To center or not to center: that is not the question—an ancillarity-suffiency interweaving strategy (ASIS) for boosting MCMC efficiency. *Journal of Computational and Graphical Statistics 20*(3), 531–570.

Zhou, X., J. Nakajima, and M. West (2014). Bayesian forecasting and portfolio decisions using dynamic dependent sparse factor models. *International Journal of Forecasting 30*(4), 963–980.

# Efficient Bayesian Inference for Multivariate Factor Stochastic Volatility Models: Web Appendix

Gregor Kastner,

*Department of Finance, Accounting and Statistics,*

*WU Vienna University of Economics and Business, Austria;*

*E-mail: gregor.kastner@wu.ac.at*

Sylvia Frühwirth-Schnatter,

*Department of Finance, Accounting and Statistics,*

*WU Vienna University of Economics and Business, Austria;*

*E-mail: sfruehwi@wu.ac.at*

Hedibert Freitas Lopes,

*Insper, Brazil;*

*E-mail: hedibertfl@insper.edu.br*

February 23, 2016

**Abstract**

This document contains supplementary material for the paper "Efficient Bayesian Inference for Multivariate Factor Stochastic Volatility Models". It encloses (A) details of the various sampling steps of Algorithm 1 and details concerning the R package **factorstochvol**, (B) the data generating parameter values for the simulation study in Section 4, and (C) further results for the exchange rate data discussed in Section 5.

# A Details on MCMC Sampling

## A.1 Details on Updating the Volatilities

The latent equations

$$h_{it} = (1-\phi_i)\mu_i + \phi_i h_{i,t-1} + \sigma_i \eta_{it}, \qquad i = 1, \ldots, m, \tag{A.1}$$

$$h_{m+j,t} = \phi_{m+j} h_{m+j,t-1} + \sigma_{m+j} \eta_{m+j,t}, \qquad j = 1, \ldots, r, \tag{A.2}$$

are combined with the (augmented) observation equation (3), yielding

$$\log(y_{it} - \boldsymbol{\Lambda}_{i,\bullet} \boldsymbol{f}_t)^2 = h_{it} + \log \varepsilon_{it}^2, \qquad i = 1, \ldots, m, \tag{A.3}$$

$$\log f_{jt}^2 = h_{m+j,t} + \log \zeta_{jt}^2, \qquad j = 1, \ldots, r. \tag{A.4}$$

Due to the modular nature of MCMC methods, updating the latent log-variances $\boldsymbol{h}$ and the corresponding parameters appearing in (A.1) and (A.2) amounts to $m+r$ independent updates with augmented data appearing on the left hand side of (A.3) and (A.4).

Each of these $m+r$ models is a univariate stochastic volatility (SV) model as introduced by Taylor (1982). References about its efficient Bayesian estimation include Jacquier et al. (1994); Shephard (1994); Shephard and Pitt (1997); Kim et al. (1998); Omori et al. (2007); Strickland et al. (2008); McCausland et al. (2011); Kastner and Frühwirth-Schnatter (2014); Shestopaloff and Neal (2014). Consequently, the substantial amount of research on this matter which has emerged in the last two decades can directly be applied. In particular, we follow Kastner and Frühwirth-Schnatter (2014) and simply use the implementation in the R (R Core Team, 2016) package **stochvol** (Kastner, 2016b) as a "plug-in" for the factor SV sampler discussed in this paper (set `dontupdatemu` to `TRUE` for sampling factor volatilities). The SV update in **stochvol** can be accessed from R through the functions `svsample` and `svsample2`, whereas the latter is a stripped-down version of the former that omits input checking and post-processing. Moreover, in order to maximize execution speed when large models are to be fitted, C/C++ level access to the core `update` function is provided. Note that the number of function calls may be in the range of billions and more, thus even tiny costs for code interpretation can quickly accumulate. The R level interface was employed for prototyping and proof-of-concept implementations (cf. Kastner et al., 2014). For the

results reported in this paper, direct access to `update` was used. A detailed description of this procedure is given in Kastner (2016a) and the package manual.

## A.2 Details on Sampling the Loadings

Because $y_{it} \sim \mathcal{N}\big(\boldsymbol{\Lambda}_{i,\bullet}\boldsymbol{f}_t, e^{h_{it}}\big)$, sampling the loadings conditionally on $\boldsymbol{f}$ constitutes a Bayesian regression problem with heteroscedastic errors. Letting $\tilde{r} = \min(i, r)$, rewriting yields

$$\tilde{\boldsymbol{y}}_i \sim \mathcal{N}_T\big(\boldsymbol{X}_i\boldsymbol{\Lambda}_{i,\bullet}', \boldsymbol{I}_T\big)\,,$$

where $\tilde{\boldsymbol{y}}_i = (y_{i1}e^{-h_{i1}/2}, \ldots, y_{iT}e^{-h_{iT}/2})'$ denotes the $i$th normalized observation vector and

$$\boldsymbol{X}_i = \begin{bmatrix} f_{11}e^{-h_{i1}/2} & \cdots & f_{\tilde{r}1}e^{-h_{i1}/2} \\ \vdots & & \vdots \\ f_{1T}e^{-h_{iT}/2} & \cdots & f_{\tilde{r}T}e^{-h_{iT}/2} \end{bmatrix}$$

is the $T \times \tilde{r}$ design matrix. Thus, independently for each $i$, sampling from $\boldsymbol{\Lambda}_{i,\bullet}|\boldsymbol{f}, \boldsymbol{y}_{i,\bullet}, \boldsymbol{h}_{i,\bullet}$ is achieved by performing a Gibbs-update from

$$\boldsymbol{\Lambda}_{i,\bullet}'|\boldsymbol{f}, \boldsymbol{y}_{i,\bullet}, \boldsymbol{h}_{i,\bullet}, \sim \mathcal{N}_{\tilde{r}}(\boldsymbol{b}_{iT}, \boldsymbol{B}_{iT})\,,$$

with $\boldsymbol{B}_{iT} = (\boldsymbol{X}_i'\boldsymbol{X}_i + \mathrm{diag}(1/B_\Lambda))^{-1}$ and $\boldsymbol{b}_{iT} = \boldsymbol{B}_{iT}\boldsymbol{X}_i'\tilde{\boldsymbol{y}}_i$.

## A.3 Details on Sampling the Factors

Sampling the factors $\boldsymbol{f}_t$ for each $t = 1, \ldots, T$ conditionally on the factor loadings $\boldsymbol{\Lambda}$ and the volatilities $\boldsymbol{h}_t$, i.e. from $\boldsymbol{f}_t|\boldsymbol{\Lambda}, \boldsymbol{y}_t, \boldsymbol{h}_t$, is again a standard Bayesian regression problem. We have

$$\tilde{\boldsymbol{y}}_t \sim \mathcal{N}_m(\boldsymbol{X}_t\boldsymbol{f}_t, \boldsymbol{I}_m)\,,$$

where $\tilde{\boldsymbol{y}}_t = (y_{1t}e^{-h_{1t}/2}, \ldots, y_{mt}e^{-h_{mt}/2})'$ denotes the normalized observation vector at time $t$ and

$$\boldsymbol{X}_t = \begin{bmatrix} \Lambda_{11}e^{-h_{1t}/2} & \cdots & \Lambda_{1r}e^{-h_{1t}/2} \\ \vdots & & \vdots \\ \Lambda_{m1}e^{-h_{mt}/2} & \cdots & \Lambda_{mr}e^{-h_{mt}/2} \end{bmatrix},$$

is the $m \times r$ design matrix. Independently for each $t$, the posterior is consequently given by

$$\boldsymbol{f}_t | \boldsymbol{\Lambda}, \boldsymbol{y}_t, \boldsymbol{h}_t \sim \mathcal{N}_r(\boldsymbol{b}_{mt}, \boldsymbol{B}_{mt}) \,,$$

where $\boldsymbol{B}_{mt} = (\boldsymbol{X}_t' \boldsymbol{X}_t + \boldsymbol{V}_t(\boldsymbol{h}_t)^{-1})^{-1}$ and $\boldsymbol{b}_{mt} = \boldsymbol{B}_{mt}(\boldsymbol{X}_t' \tilde{\boldsymbol{y}}_t)$.

## A.4 Sign Identification

For sensible interpretation of the factor loadings their signs have to be identified. If done a posteriori, this requires the selection of one series per factor whose loadings distribution is sufficiently bound away from zero. For the paper at hand, we simply investigate the absolute values of the posterior draws from the loadings distribution. For each factor, the series whose smallest absolute MCMC draw is largest gets assigned a positive sign which in turn implies the signs for all other loadings on this factor. In other words, letting $K$ denote the number of MCMC draws after burn-in, in order to identify factor $j = 1, \ldots, r$, we assign a positive sign to the posterior distribution of $|\boldsymbol{\Lambda}_{ij}|$, where $i$ is chosen to be

$$\underset{i \in \{1,\ldots,m\}}{\arg\max} \left( \min_{k \in \{1,\ldots,K\}} \left| \boldsymbol{\Lambda}_{ij}^{(k)} \right| \right) .$$

Subsequently, we align the other loadings on factor $j$ accordingly. To achieve this behavior in **factorstochvol**, use `signident` with `method = "maximin"`. In order to simply use the leading factors to identify the loadings, use `signident` with `method = "diagonal"`.

For the exchange rate data, the outcome of this procedure is visualized in Figure C.1 through triangular symbols. It can be seen that factor 1 is identified via USD, factor 2 via ZAR, factor 3 via AUD, and factor 4 via MYR.

## A.5 A Note on Implementation

High-dimensional models, in particular models with many latent variables, pose a non-negligible computational challenge to those aiming for efficient MCMC implementations. In principle, each individual MCMC step can be straightforwardly computed in parallel. In practice, however, doing so is only useful in shared memory environments (e.g. through multithreading/multiprocessing) as the increased communication overhead in distributed memory environments easily outweighs the speed gains. Apart from within-steps-parallelization,

MCMC is of intrinsically iterative nature in the sense that posterior draws are generated conditionally on older draws. Thus, they cannot be parallelized straightforwardly and call for compiled and optimized programming languages to avoid the cost of code interpretation at every iteration. Moreover, memory access needs to be optimized, as large amounts of latent variable draws must be stored either temporary (if required only for the next conditional draws) or more permanently (if required for direct posterior inference). In this paper, we tackle the computational burden by using high-performance C and C++ code, interfaced to R via **RcppArmadillo** (Eddelbuettel and Sanderson, 2014). Additionally to providing an interface between R and C++, **RcppArmadillo** also accommodates fast linear algebra routines by means of the **Armadillo** library (Sanderson, 2010). For ease-of-use, all code is bundled in the R package **factorstochvol** which is available from the authors and in preparation to be published on the Comprehensive R Archive Network (CRAN).

# B Simulation Study

The data generating parameter values for the simulation study in Section 4 are listed in Table B.1.

# C Application to Exchange Rate Data

This section provides additional results for exchange rate data analyzed in Section 5. Figure C.1 visualizes the posterior medians of the factor loadings for all factors. The pairwise correlations of USD against the other currencies are plotted in Figure C.2. Finally, inefficiency factors for posterior draws from the factor loadings using the interwoven samplers, i.e. deep and shallow interweaving, are presented in Table C.1.

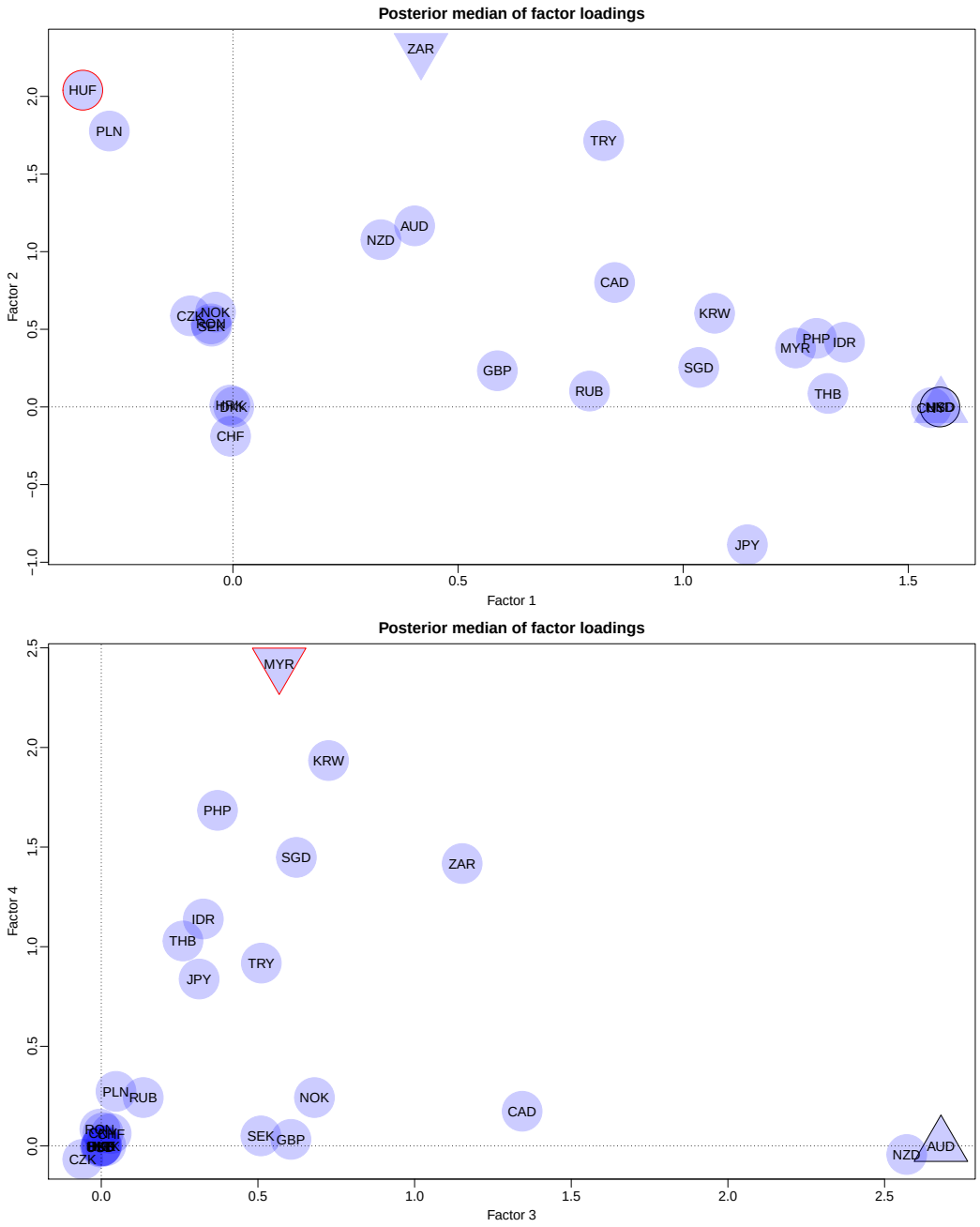


Figure C.1: Posterior medians of the factor loadings. Black border means that this series was used as the leading factor in the horizontal direction, red border means that this series was used as the leading factor in the vertical direction. Leading factors were determined via a preliminary factor analysis. Up-pointing triangle means that this series was used for sign-identification in the horizontal direction, down-pointing triangle means that this series was used for sign-identification in the vertical direction.

| $\mathbf{\Lambda}_{\text{true}}$ | 1 | 2 |
|---|---|---|
| 1 | 1.00 | |
| 2 | 0.90 | 1.00 |
| 3 | 0.80 | 0.10 |
| 4 | 0.70 | 0.20 |
| 5 | 0.60 | 0.30 |
| 6 | 0.50 | 0.40 |
| 7 | 0.40 | 0.50 |
| 8 | 0.30 | 0.60 |
| 9 | 0.20 | 0.70 |
| 10 | 0.10 | 0.80 |

(a) Factor loadings

| | $\mu_{\text{true}}$ | $\phi_{\text{true}}$ | $\sigma_{\text{true}}$ |
|---|---|---|---|
| 1 | -2.00 | 0.80 | 0.60 |
| 2 | -1.90 | 0.82 | 0.55 |
| 3 | -1.80 | 0.84 | 0.50 |
| 4 | -1.70 | 0.86 | 0.45 |
| 5 | -1.60 | 0.88 | 0.40 |
| 6 | -1.50 | 0.90 | 0.35 |
| 7 | -1.40 | 0.92 | 0.30 |
| 8 | -1.30 | 0.94 | 0.25 |
| 9 | -1.20 | 0.96 | 0.20 |
| 10 | -1.10 | 0.98 | 0.15 |

(b) Idiosyncratic volatility parameters

| | $\phi_{\text{true}}$ | $\sigma_{\text{true}}$ |
|---|---|---|
| 1 | 0.99 | 0.10 |
| 2 | 0.95 | 0.30 |

(c) Factor volatility parameters

Table B.1: Data generating values.

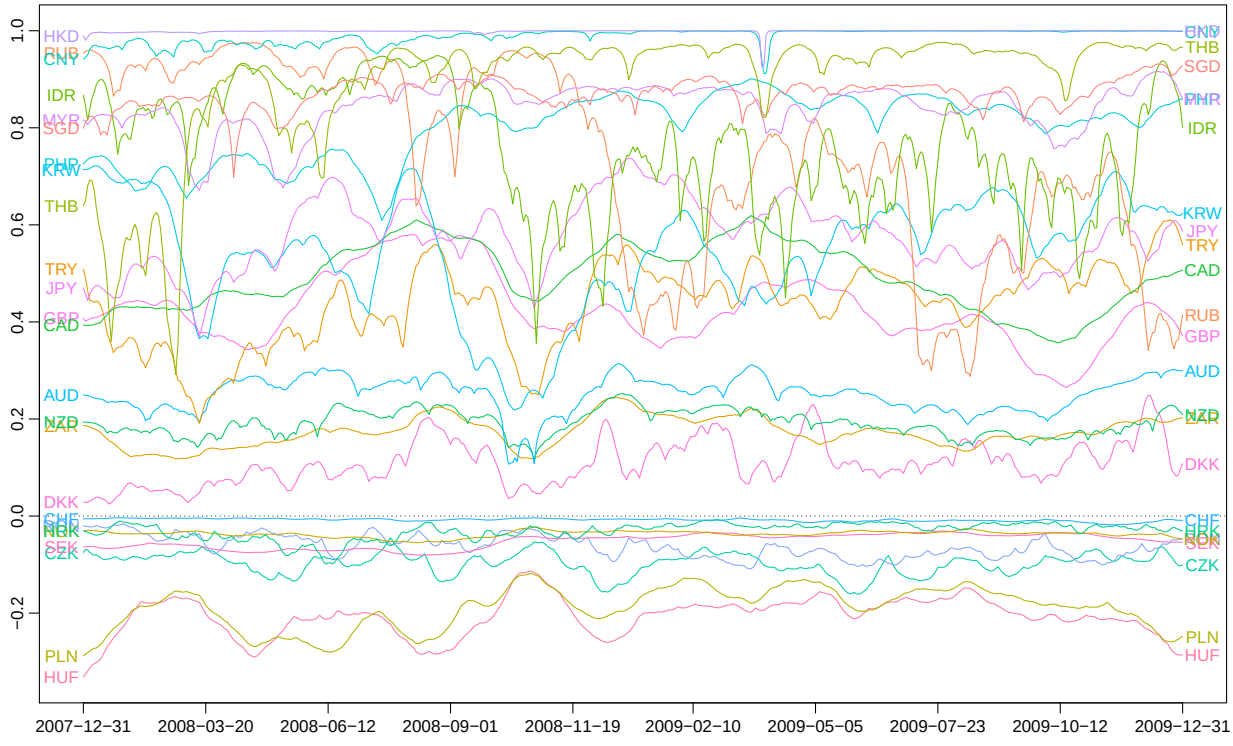


Figure C.2: Posterior means of pairwise correlations with USD.

| | $\mathbf{\Lambda}_{\bullet,1}$ | $\mathbf{\Lambda}_{\bullet,2}$ | $\mathbf{\Lambda}_{\bullet,3}$ | $\mathbf{\Lambda}_{\bullet,4}$ |
|---|---|---|---|---|
| AUD | 335 | 607 | 853 | * |
| CAD | 636 | 618 | 867 | 27 |
| CHF | 6 | 173 | 28 | 21 |
| CNY | 722 | 14 | 13 | 100 |
| CZK | 172 | 747 | 26 | 12 |
| DKK | 76 | 5 | 19 | 5 |
| GBP | 630 | 169 | 677 | 7 |
| HKD | 713 | * | * | * |
| HRK | 10 | 10 | 12 | 5 |
| HUF | 350 | 877 | * | * |
| IDR | 689 | 496 | 345 | 603 |
| JPY | 694 | 618 | 112 | 185 |
| KRW | 651 | 543 | 643 | 720 |
| MYR | 671 | 392 | 561 | 849 |
| NOK | 25 | 604 | 648 | 31 |
| NZD | 270 | 534 | 890 | 44 |
| PHP | 689 | 569 | 389 | 738 |
| PLN | 332 | 822 | 33 | 45 |
| RON | 60 | 677 | 13 | 14 |
| RUB | 665 | 218 | 167 | 135 |
| SEK | 34 | 607 | 565 | 10 |
| SGD | 681 | 344 | 782 | 929 |
| THB | 700 | 68 | 453 | 735 |
| TRY | 571 | 756 | 296 | 221 |
| USD | 713 | 55 | 103 | 95 |
| ZAR | 298 | 793 | 609 | 346 |

(a) Shallow Interweaving

| | $\mathbf{\Lambda}_{\bullet,1}$ | $\mathbf{\Lambda}_{\bullet,2}$ | $\mathbf{\Lambda}_{\bullet,3}$ | $\mathbf{\Lambda}_{\bullet,4}$ |
|---|---|---|---|---|
| AUD | 30 | 50 | 27 | * |
| CAD | 25 | 41 | 26 | 16 |
| CHF | 6 | 37 | 15 | 12 |
| CNY | 25 | 10 | 9 | 21 |
| CZK | 17 | 44 | 11 | 8 |
| DKK | 10 | 5 | 7 | 5 |
| GBP | 24 | 24 | 23 | 7 |
| HKD | 25 | * | * | * |
| HRK | 5 | 6 | 6 | 5 |
| HUF | 25 | 48 | * | * |
| IDR | 25 | 38 | 22 | 29 |
| JPY | 24 | 40 | 22 | 22 |
| KRW | 25 | 40 | 25 | 30 |
| MYR | 25 | 39 | 27 | 32 |
| NOK | 15 | 39 | 24 | 10 |
| NZD | 30 | 45 | 27 | 49 |
| PHP | 25 | 41 | 22 | 30 |
| PLN | 23 | 48 | 28 | 16 |
| RON | 15 | 43 | 12 | 9 |
| RUB | 25 | 27 | 17 | 22 |
| SEK | 16 | 42 | 23 | 11 |
| SGD | 25 | 39 | 27 | 31 |
| THB | 25 | 24 | 26 | 29 |
| TRY | 25 | 46 | 24 | 22 |
| USD | 25 | 47 | 36 | 39 |
| ZAR | 24 | 47 | 25 | 22 |

(b) Deep Interweaving

Table C.1: Estimated inefficiency factors for posterior draws of the factor loadings.

# References


Eddelbuettel, D. and C. Sanderson (2014). RcppArmadillo: Accelerating R with high-performance C++ linear algebra. *Computational Statistics and Data Analysis 71*, 1054–1063.

Jacquier, E., N. G. Polson, and P. E. Rossi (1994). Bayesian analysis of stochastic volatility models. *Journal of Business & Economic Statistics 12*(4), 371–389.

Kastner, G. (2016a). Dealing with stochastic volatility in time series using the R package stochvol. *Journal of Statistical Software 69*(5), 1–30.

Kastner, G. (2016b). *stochvol: Efficient Bayesian inference for stochastic volatility (SV) models*. R package version 1.2.3.

Kastner, G. and S. Frühwirth-Schnatter (2014). Ancillarity-sufficiency interweaving strategy (ASIS) for boosting MCMC estimation of stochastic volatility models. *Computational Statistics and Data Analysis 76*, 408–423.

Kastner, G., S. Frühwirth-Schnatter, and H. F. Lopes (2014). Analysis of exchange rates via multivariate Bayesian factor stochastic volatility models. In E. Lanzarone and F. Ieva (Eds.), *The Contribution of Young Researchers to Bayesian Statistics – Proceedings of BAYSM2013*, Volume 63 of *Springer Proceedings in Mathematics & Statistics*, pp. 181–186. Springer.

Kim, S., N. Shephard, and S. Chib (1998). Stochastic volatility: Likelihood inference and comparison with ARCH models. *Review of Economic Studies 65*(3), 361–393.

McCausland, W. J., S. Miller, and D. Pelletier (2011). Simulation smoothing for state-space models: A computational efficiency analysis. *Computational Statistics and Data Analysis 55*(1), 199–212.

Omori, Y., S. Chib, N. Shephard, and J. Nakajima (2007). Stochastic volatility with leverage: Fast and efficient likelihood inference. *Journal of Econometrics 140*(2), 425–449.

R Core Team (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing.

Sanderson, C. (2010). *Armadillo: An open source C++ linear algebra library for fast prototyping and computationally intensive experiments*. Technical Report, NICTA.

Shephard, N. (1994). Partial non-Gaussian state space. *Biometrika 81*(1), 115–131.

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

Shestopaloff, A. Y. and R. M. Neal (2014). Efficient Bayesian inference for stochastic volatility models with ensemble MCMC methods. arXiv pre-print 1412.3013.

Strickland, C. M., G. M. Martin, and C. S. Forbes (2008). Parameterisation and efficient MCMC estimation of non-Gaussian state space models. *Computational Statistics and Data Analysis 52*(6), 2911–2930.

Taylor, S. J. (1982). Financial returns modelled by the product of two stochastic processes: A study of daily sugar prices 1691–79. In O. D. Anderson (Ed.), *Time Series Analysis: Theory and Practice 1*, pp. 203–226. Amsterdam: North-Holland.