

Bayesian Inference of Vector Autoregressions with Tensor Decompositions

Yiyong Luo & Jim E. Griffin

To cite this article: Yiyong Luo & Jim E. Griffin (26 Feb 2025): Bayesian Inference of Vector Autoregressions with Tensor Decompositions, Journal of Business & Economic Statistics, DOI: [10.1080/07350015.2024.2447302](https://doi.org/10.1080/07350015.2024.2447302)

To link to this article: <https://doi.org/10.1080/07350015.2024.2447302>



© 2025 The Author(s). Published with license by Taylor & Francis Group, LLC.



[View supplementary material](#)



Published online: 26 Feb 2025.



[Submit your article to this journal](#)



Article views: 534



[View related articles](#)



[View Crossmark data](#)



Citing articles: 1 [View citing articles](#)

Bayesian Inference of Vector Autoregressions with Tensor Decompositions

Yiyong Luo and Jim E. Griffin 

Department of Statistical Science, University College London, London, UK

ABSTRACT

Vector autoregression (VAR) is a popular model for analyzing multivariate economic time series. However, VARs can be over-parameterized if the numbers of variables and lags are moderately large. Tensor VAR, a recent solution to over-parameterization, treats the coefficient matrix as a third-order tensor and estimates the corresponding tensor decomposition to achieve parsimony. In this article, we employ the Tensor VAR structure with a CANDECOMP/PARAFAC (CP) decomposition and use Bayesian inference to estimate parameters. First, we determine the rank by imposing the Multiplicative Gamma Prior to the tensor margins, that is elements in the decomposition, and accelerate the computation with an adaptive inferential scheme. Second, to obtain interpretable margins, we propose an interweaving algorithm to improve the mixing of margins and identify the margins using a post-processing procedure. In an application to the U.S. macroeconomic data, our models outperform standard VARs in point and density forecasting and yield a summary of the dynamic of the U.S. economy.

ARTICLE HISTORY

Received June 2023
Accepted December 2024

KEYWORDS

Ancillarity-sufficiency
interweaving strategy (ASIS);
High-dimensional data;
Increasing shrinkage prior;
Markov chain Monte Carlo
(MCMC);
Over-parameterization



1. Introduction

Vector autoregression (VAR) is a multivariate time series model that describes the linear interrelationship of data. Since the advocacy of Sims (1980), VAR is a widely used tool for modeling macroeconomic variables, which are known to be temporally dependent on each other. As suggested in Korobilis and Pettenuzzo (2019), Carriero, Clark, and Marcellino (2019), Bańbura, Giannone, and Reichlin (2010), and Giannone, Lenza, and Primiceri (2015), to name a few, applying VARs to a large set of variables is advantageous for forecasting and structural analysis. However, to succeed in modeling with large VARs, one must solve over-parameterization, that is the number of parameters is high relative to the sample size. Over-parameterization is especially an issue for macroeconomic data due to the low frequency of data collection.

Methodologies to solve over-parameterization in VARs can be divided into *sparse*- and *dense*-modeling streams, according to Ng (2013). The sparse stream assumes that only small sets of predictors are important to model the time series of each variable. For example, Hsu, Hung, and Chang (2008) proposed using the Lasso penalty (Tibshirani 1996) for VARs. The dense stream relies on an opposite assumption to its sparse-modeling counterpart: all predictors could be important, but their corresponding parameters may have small magnitudes. Shrinkage priors, including the Minnesota-type priors (Doan, Litterman, and Sims 1984; Litterman 1986) and global-local shrinkage priors (Huber and Feldkircher 2019; Huber, Kastner, and Feldkircher 2019; Gruber and Kastner 2022) dominate the dense-modeling stream in a VAR framework. An alternative

methodology in this stream, called reduced-rank VAR (Carriero, Kapetanios, and Marcellino 2011), assumes that the VAR coefficient matrix has a low rank, and one can decompose this matrix to achieve parsimony. A more recent and related technique, referred to as Tensor VAR, treats the coefficient matrix as a third-order tensor and infers this tensor by its low-rank decomposition. Wang et al. (2021) was the first to introduce Tensor VAR, and this technique has been developed in Zhang et al. (2021) and Fan et al. (2022).

In this article, we contribute to the dense-modeling stream by employing the Tensor VAR structure with a CANDECOMP/PARAFAC (CP) decomposition (Kiers 2000) and conducting Bayesian inference to estimate parameters. The motivation for choosing this methodology to alleviate over-parameterization is 4-fold. First, recent work has questioned whether sparse modeling is appropriate for macroeconomic data, for example see Giannone, Lenza, and Primiceri (2021) for the “illusion of sparsity.” Second, a Tensor VAR with an appropriate choice of rank is parsimonious without imposing any penalty term or shrinkage prior (although incorporating these techniques results in further parsimony). Third, Tensor VAR is a useful model for explaining macroeconomic data since its reconstruction provides insights into the economy, and elements in its tensor decomposition (usually called *margins*) are interpretable as shown in Wang et al. (2021) and Chen, Yang, and Zhang (2022). Lastly, tensor structures with Bayesian inference have been successfully applied in time series models apart from VARs. Related work includes time-varying networks (Billio, Casarin, and Iacopini 2024) and Autoregressive Tensor Processes (ART) (Billio et al. 2023), among others.

CONTACT Yiyong Luo  yiyong.luo.20@ucl.ac.uk  Department of Statistical Science, University College London, London, WC1E 6BT, UK.

 Supplementary materials for this article are available online. Please go to www.tandfonline.com/UBES.

© 2025 The Authors. Published with license by Taylor & Francis Group, LLC.

This is an Open Access article distributed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives License (<http://creativecommons.org/licenses/by-nc-nd/4.0/>), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly cited, and is not altered, transformed, or built upon in any way. The terms on which this article has been published allow the posting of the Accepted Manuscript in a repository by the author(s) or with their consent.

Two challenges arise when making Bayesian inference in a Tensor VAR with a CP decomposition. The first challenge is about the inference of the rank, which is an important parameter in the CP decomposition because it controls the model flexibility. Unlike finding the rank in a matrix, there is no straightforward algorithm to determine the rank of a third-order tensor. Although existing literature gives rank values of some specified tensors, see Kolda and Bader (2009) and references therein, tensors for large VARs have relatively high dimensions, so they normally do not nest in these specified ones. To overcome this challenge, past literature proposed the multiway Dirichlet generalized double Pareto (M-DGDP) prior (Guhaniyogi, Qamar, and Dunson 2017) and the multiway stick breaking shrinkage prior (Guhaniyogi and Spencer 2021), based on overfitted mixture models (Rousseau and Mengersen 2011), to induce a low-rank structure in the CP decomposition and inferred the rank a posteriori. Despite being a prominent method to resolve the challenge, it is computationally expensive due to the large initialization of the rank. The second challenge is to retain the interpretability of a Tensor VAR. From a Bayesian perspective, a fundamental prerequisite for a Tensor VAR to be interpretable is the convergence of margin Markov chains, but this prerequisite cannot be achieved using the traditional MCMC scheme because the indeterminacy of the CP decomposition can lead to poorly mixing MCMC algorithm, which in turn produces posterior distributions that are difficult to interpret. One solution is to impose restrictions on margins so that they are identified (Zhou, Li, and Zhu 2013), whereas solutions in unrestricted parameter space have not been explored yet.

We tackle the above challenges with two contributions. Our first contribution is to infer the rank using an increasing shrinkage prior. We impose the Multiplicative Gamma prior (MGP) (Bhattacharya and Dunson 2011) to the margins and use an adaptive inferential scheme to infer these margins, and subsequently the rank. This idea is closely related to the recent work in Fan et al. (2022), but our prior and the criterion in the adaptive inference are different from theirs. In our second contribution, we improve the mixing of the MCMC algorithm by introducing a Gibbs sampler including a variant of the Ancillarity-Sufficiency Interweaving Strategy (ASIS) (Yu and Meng 2011) with three interweaving steps, inspired by the ASIS algorithm for factor models (Kastner, Frühwirth-Schnatter, and Lopes 2017). Unlike previous methods for tensors, dividing the margins into three blocks during inference reduces the dependence between the margins in the MCMC output. Even if the mixing of margins is not essential in some instances, for example one does not interpret margins and only regards the mixing of entries in tensor itself as important, this contribution is still beneficial because achieving good mixing of margins provides a solid foundation for entries in the VAR coefficient matrix to mix well. Additionally, we introduce a post-processing procedure aimed at identifying the margins.

We examine the utility of Tensor VARs through two US macroeconomic datasets with medium and large sizes. We consider two specifications of Tensor VARs that treat the coefficient matrix in two ways: (a) the matricization of a third-order tensor and (b) a sum of the matricization of a third-order tensor and a matrix with only nonzero entries for own lags. The first one corresponds to the original Tensor VAR idea (Wang et al.

2021), and the second one accommodates the main feature of Minnesota-type priors, that is the own lags of a variable are more informative than lags of other dependent variables. In point and density forecasting, these two Tensor VARs obtain the best results for joint forecasts and are competitive to standard VARs with a range of standard prior choices. We demonstrate how to interpret margins by applying our model to the large-scale data and constructing factors as linear combinations of lagged data. The Tensor VAR can effectively reduce the number of parameters, and the factors constructed can summarize the dynamics of the dataset. The additional own-lag matrix in the second Tensor VAR structure introduces more parameters but allows the tensor to focus on exploring the cross-variable and cross-lag effects.

The article is organized as follows. Section 2 explains the Tensor VAR and its interpretation. Section 3 provides the MCMC schemes. Section 4 introduces the post-processing procedure. Section 5 shows results from simulation experiments. Section 6 presents the forecasting performance and interpretation of Tensor VARs. Section 7 concludes the article.

2. Tensor VAR

2.1. Model Specification

Let $\mathbf{y}_t \in \mathbb{R}^N$ be the t th observation in a multivariate time series. A P -order VAR model, $\text{VAR}(P)$, describes the linear relation between \mathbf{y}_t and its lags with coefficient matrices $\mathbf{A}_1, \dots, \mathbf{A}_P \in \mathbb{R}^{N \times N}$ by

$$\mathbf{y}_t = \mathbf{A}_1 \mathbf{y}_{t-1} + \dots + \mathbf{A}_P \mathbf{y}_{t-P} + \boldsymbol{\epsilon}_t = \mathbf{A} \mathbf{x}_t + \boldsymbol{\epsilon}_t, \quad (2.1)$$

where $t = 1 \dots T$, $\mathbf{A} = (\mathbf{A}_1, \dots, \mathbf{A}_P)$ is an N -by- NP coefficient matrix linearly connecting \mathbf{y}_t and its lags, $\mathbf{x}_t = (\mathbf{y}'_{t-1}, \dots, \mathbf{y}'_{t-P})' \in \mathbb{R}^{NP}$. The error term $\boldsymbol{\epsilon}_t$ follows a multivariate normal distribution with zero mean and a time-varying covariance matrix $\boldsymbol{\Omega}_t$. In this article, we factorize $\boldsymbol{\Omega}_t$ according to Cogley and Sargent (2005), that is $\boldsymbol{\Omega}_t = \mathbf{H}^{-1} \mathbf{S}_t (\mathbf{H}^{-1})'$, where \mathbf{H}^{-1} is a lower triangular matrix with ones as diagonal entries, and \mathbf{S}_t is a time-varying diagonal matrix with diagonal terms $(s_{t,1}, \dots, s_{t,N})$.

To fit the VAR model, we must estimate the $N^2 P$ parameters in \mathbf{A} and parameters for the covariance matrix $\boldsymbol{\Omega}_t$. The number of coefficients grows quadratically as the number of time series increases, thus, VARs can become easily overparameterized. We address this problem by achieving parsimony of \mathbf{A} through tensor decomposition, in the spirit of Wang et al. (2021). Specifically, rather than modeling \mathbf{A} directly, we model a third-order tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{N \times N \times P}$, where $\boldsymbol{\mathcal{A}}_{i_1, i_2, p}$ corresponds to the (i_1, i_2) entry in \mathbf{A}_p . The model in (2.1) can be written in term of the tensor $\boldsymbol{\mathcal{A}}$ to give

$$\mathbf{y}_t = \boldsymbol{\mathcal{A}}_{(1)} \mathbf{x}_t + \boldsymbol{\epsilon}_t, \quad (2.2)$$

where $\boldsymbol{\mathcal{A}}_{(1)} = \mathbf{A}$ is the mode-1 matricization of $\boldsymbol{\mathcal{A}}$, with the i_1 th row as the vectorization of $\boldsymbol{\mathcal{A}}_{(i_1, \cdot, \cdot)}$.

So far, the number of entries in $\boldsymbol{\mathcal{A}}$ is the same as that in \mathbf{A} , but we can decompose $\boldsymbol{\mathcal{A}}$ via a rank- R CP decomposition,

$$\boldsymbol{\mathcal{A}} = \sum_{r=1}^R \boldsymbol{\mathcal{A}}^{(r)} = \sum_{r=1}^R \boldsymbol{\beta}_1^{(r)} \circ \boldsymbol{\beta}_3^{(r)} \circ \boldsymbol{\beta}_3^{(r)}, \quad (2.3)$$

where $\mathcal{A}^{(r)}$ is a third-order tensor with the same dimension as \mathcal{A} , for $r = 1, \dots, R$; $\beta_1^{(r)}, \beta_2^{(r)} \in \mathbb{R}^N$ and $\beta_3^{(r)} \in \mathbb{R}^P$ are called margins of \mathcal{A} ; $\mathcal{A}^{(r)} = \beta_1^{(r)} \circ \beta_2^{(r)} \circ \beta_3^{(r)}$ is an outer product of three vectors such that the (i_1, i_2, i_3) entry in $\mathcal{A}^{(r)}$ equals to $\beta_{1,i_1}^{(r)} \beta_{2,i_2}^{(r)} \beta_{3,i_3}^{(r)}$ for $i_1, i_2 = 1, \dots, N$ and $i_3 = 1, \dots, P$ (the definition of outer product can be found in Appendix A). We define the notation $\mathbf{B}_j = (\beta_j^{(1)}, \dots, \beta_j^{(R)}) \in \mathbb{R}^{I_j \times R}$, for $j = 1, 2, 3$, $I_1 = I_2 = N$ and $I_3 = P$, then the tensor \mathcal{A} decomposed by $\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3$ is written as $\mathcal{A} = [\![\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3]\!]_{\text{CP}}$, for the sake of brevity. Another useful representation of the margins is $\mathbf{B} = (\mathbf{B}'_1, \mathbf{B}'_2, \mathbf{B}'_3)' \in \mathbb{R}^{(2N+P) \times R}$ to which we refer as a *tensor matrix*, then $\mathcal{A}^{(r)}$ is constructed by margins in the r th column of \mathbf{B} . With an upper bound $N^2P/(2N+P)$ of R , the number of parameters reduces from N^2P in the coefficient matrix to $(2N+P)R$ in \mathbf{B} , so a low-rank structure in the CP decomposition alleviates over-parameterization.

The CP decomposition is only identified up to scaling and permutation because $\mathcal{A} = [\![\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3]\!]_{\text{CP}} = [\![\tilde{\mathbf{B}}_1, \tilde{\mathbf{B}}_2, \tilde{\mathbf{B}}_3]\!]_{\text{CP}}$, if $\tilde{\mathbf{B}}_j$ comes from the following transformations for $j = 1, 2, 3$:

1. Scaling: $\tilde{\mathbf{B}}_j = \mathbf{B}_j \mathbf{R}_j$, and \mathbf{R}_j is an R -by- R diagonal matrix satisfying $\prod_{j=1}^J \mathbf{R}_{j,(r,r)} = 1$ for $r = 1, \dots, R$, where $\mathbf{R}_{j,(r,r)}$ is the r th diagonal term in \mathbf{R}_j .
2. Permutation: $\tilde{\mathbf{B}}_j = \mathbf{B}_j \mathbf{\Pi}$ for an arbitrary R -by- R column-wise permutation matrix $\mathbf{\Pi}$.

This indeterminacy will play an important role in our algorithm in Section 3.2.2. To interpret the margins, we will identify them using a post-processing procedure described in Section 4.

The model in (2.2) represents the original Tensor VAR (Wang et al. 2021), which does not distinguish between the own-lag and cross-lag effects. In Section 6.4, we empirically find that introducing this distinction allows us to achieve better forecasting performance and interpretability, so we build an extension of (2.2), called Own-lag Tensor VAR, following the assumption of the Minnesota-type priors - the own-lag effects are more powerful than the cross-lag effects. In particular, we add a matrix \mathbf{D} , the concatenation of P N -by- N diagonal matrices, to give

$$\mathbf{y}_t = \mathcal{A}_{(1)} \mathbf{x}_t + \mathbf{D} \mathbf{x}_t + \boldsymbol{\epsilon}_t, \quad (2.4)$$

so \mathbf{D} can only affect entries corresponding to own lags.

2.2. Model Interpretation

The Tensor VAR connects $\mathbf{y}_t^* = \mathbf{y}_t - \mathbf{D} \mathbf{x}_t^1$ with past information through $\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3$ in the following reconstruction:

$$\begin{aligned} \mathbf{y}_t^* &= \mathbf{B}_1 \mathcal{I}_{(1)} \text{vec}(\mathbf{B}'_2 \mathbf{X}_t \mathbf{B}_3) + \boldsymbol{\epsilon}_t \\ &= \sum_{r=1}^R \mathbf{B}_{1,(\cdot,r)} \sum_{i_2=1}^N \sum_{i_3=1}^P \beta_{2,i_2}^{(r)} \beta_{3,i_3}^{(r)} \mathbf{y}_{t-i_3,i_2} + \boldsymbol{\epsilon}_t, \end{aligned} \quad (2.5)$$

where $\mathcal{I}_{(1)} \in \mathbb{R}^{R \times R^2}$ is the mode-1 matricization of a third-order superdiagonal tensor \mathcal{I} with ones on nonzero entries (see Appendix A for a detailed description), $\mathbf{X}_t = (\mathbf{y}_{t-1}, \dots, \mathbf{y}_{t-P})$, $\text{vec}(\cdot)$ is the vectorization operation which transforms $\mathbf{B}'_2 \mathbf{X}_t \mathbf{B}_3 \in \mathbb{R}^{R \times R}$ to an R^2 -dimensional vector,

$\mathbf{B}_{1,(\cdot,r)}$ is the r th column of \mathbf{B}_1 , $\beta_{2,i_2}^{(r)}, \beta_{3,i_3}^{(r)}$ are the (i_2, r) and (i_3, r) entries of \mathbf{B}_2 and \mathbf{B}_3 , respectively, \mathbf{y}_{t-i_3,i_2} is the i_2 th entry in \mathbf{y}_{t-i_3} .

Following Wang et al. (2021), we can relate (2.5) to a factor model (Stock and Watson 2005), where \mathbf{B}_1 is the factor loading and $\mathcal{I}_{(1)} \text{vec}(\mathbf{B}'_2 \mathbf{X}_t \mathbf{B}_3)$ contains R observable factors. Since the i_1 th row in \mathbf{B}_1 describes the linear relationship between \mathbf{y}_{t,i_1} and factors, for $i_1 = 1, \dots, N$, we refer to \mathbf{B}_1 as “response loading”. The formation of factors describes how past information is combined. We look at $\sum_{i_2=1}^N \sum_{i_3=1}^P \beta_{2,i_2}^{(r)} \beta_{3,i_3}^{(r)} \mathbf{y}_{t-i_3,i_2}$ in (2.5) to understand this formation. If $\beta_{2,i_2}^{(r)} = 0$, the r th factor will not contain information from any lagged values of \mathbf{y}_{t,i_2} . Similarly, $\beta_{3,i_3}^{(r)} = 0$ results in no information about the i_3 th lag of \mathbf{y}_t in the r th factor. Therefore, the i_2 th row of \mathbf{B}_2 contains the effect from the i_2 th variable to \mathbf{y}_t , and the i_3 th row of \mathbf{B}_3 is related to the effect from the i_3 th lag to \mathbf{y}_t . This interpretation was also discussed in Wang et al. (2021), who called \mathbf{B}_2 and \mathbf{B}_3 “predictor loading” and “temporal loading,” respectively.

Another way to explain the CP decomposition in the Tensor VAR is that it separates the lag effect from the variable-wise effect because it decomposes \mathbf{A}_p as $\mathbf{A}_p = \sum_{r=1}^R (\beta_1^{(r)} \circ \beta_2^{(r)}) \beta_{3,p}^{(r)}$, where $\beta_1^{(r)} \circ \beta_2^{(r)} \in \mathbb{R}^{N \times N}$ is the outer product of the two vectors such that the (i_1, i_2) entry of this resulting matrix equals to $\beta_{1,i_1}^{(r)} \beta_{2,i_2}^{(r)}$. The first two vectors $\beta_1^{(r)}$ and $\beta_2^{(r)}$ (for $r = 1, \dots, R$) do not depend on the index of \mathbf{A}_p , suggesting that all lagged coefficients matrices share these vectors. The only difference among these matrices reflects on the different entries in $\beta_3^{(r)}$.

3. Bayesian Inference

3.1. Prior Specification

As mentioned in Section 2.1, we aim to impose a prior on the tensor matrix \mathbf{B} , which favors a low-rank structure. A particular prior choice that meets our requirement is the MGP (Bhattacharya and Dunson 2011) because it possesses the increasing shrinkage property, enabling margins with higher column index to have higher degrees of shrinkage. As a result, the rank can be lowered if some columns in \mathbf{B} have magnitudes negligibly small. To be specific, a margin $\beta_{j,i_j}^{(r)}$ (the (i_j, r) entry of \mathbf{B}_j) follows the prior below for $j = 1, 2, 3$, $r = 1, \dots, R$, $i_1, i_2 = 1, \dots, N$ and $i_3 = 1, \dots, P$:

$$\beta_{j,i_j}^{(r)} \sim \mathcal{N}\left(0, \left(\sigma_{j,i_j}^{(r)}\right)^2\right), \left(\sigma_{j,i_j}^{(r)}\right)^2 = \phi_{(r,j,i_j)}^{-1} \tau_r^{-1},$$

$$\phi_{(r,j,i_j)} \sim \text{Gamma}(v/2, v/2), \tau_r = \prod_{l=1}^r \delta_l,$$

$$\delta_1 \sim \text{Gamma}(a_1, 1), \delta_l \sim \text{Gamma}(a_2, 1), 1 < l < R,$$

where $\phi_{(r,j,i_j)}$ is a local parameter for the margin with the same index. We store all these local parameters in a matrix Φ in which each entry corresponds to an entry in the tensor matrix \mathbf{B} with the same indices. The increasing shrinkage property is induced by τ_r since $\mathbb{E}(\tau_r) = \prod_{l=1}^r \mathbb{E}(\delta_l) = a_1 a_2^{r-1}$ increases with r , when $a_2 > 1$. Hyperparameter v is set to be known, and a_1 and a_2 will be inferred with Gamma priors. Durante (2017) showed

¹We include \mathbf{D} for completion. \mathbf{D} is a zero matrix if we apply (2.2).

that both $\mathbb{E}(\tau_r)$ and $\mathbb{E}(\tau_r^{-1})$ increase with r when $1 < a_2 < 2$. This result means that the MGP has the increasing shrinkage property only when $a_2 > 2$. Thus, we set priors for a_1 and a_2 as $\text{Gamma}(5,1)$ to have the increasing shrinkage property with a high probability. Apart from the shrinkage prior for \mathbf{B} , we follow priors in Huber and Feldkircher (2019) for \mathbf{H} and \mathbf{S}_t , see Appendix B.1 for details.

In the case of (2.4), we impose a normal-gamma prior defined in Huber and Feldkircher (2019) to each nonzero entry in \mathbf{D} . Let $d_{i,p}$ denote the own-lag coefficient for the p th lag of the i th response, then its prior is written as

$$d_{i,p} \sim \mathcal{N}\left(0, (2/\lambda_d^2) \psi_d^{(i,p)}\right),$$

$$\psi_d^{(i,p)} \sim \text{Gamma}(a_d, a_d), \text{ for } i = 1, \dots, N \text{ and } p = 1, \dots, P.$$

Priors of hyperparameters are the same as those for lower triangular matrix \mathbf{H} . All the full conditionals and their derivation can be found in Appendix B.

3.2. MCMC Scheme

3.2.1. An Overview of Inferential Scheme

To illustrate the strengths of our inferential scheme, we contrast it with the widely-used inferential scheme for tensor-structured models. In the traditional scheme (Guhaniyogi, Qamar, and Dunson 2017; Zhang et al. 2021; Fan et al. 2022; Billio et al. 2023), $\beta_j^{(r)}$ is sampled from $p\left(\beta_j^{(r)} \mid \beta_{-j}^{(r)}, \mathbf{B}_{(-,r)}, \mathbf{y}_{1:T}, \left(\sigma_j^{(r)}\right)^2\right)$, for $r = 1, \dots, R$ and $j = 1, \dots, J$ (J is 3 in our case), where $\beta_{-j}^{(r)}$ contains all $\beta_{j'}^{(r)}$ with $j' \neq j$, $\mathbf{B}_{(-,r)}$ is \mathbf{B} discarding its r th column, $\left(\sigma_j^{(r)}\right)^2$ has all prior variance corresponding to $\beta_j^{(r)}$. These full conditionals are then incorporated into a usual Gibbs sampler, so each $\beta_j^{(r)}$ sampled depends on other margins, and in turn, other margins are sampled given $\beta_j^{(r)}$ and other parameters. The rank R is fixed to a large value during the inference and can be determined to a smaller value a posteriori. This inferential scheme neglects the convergence of margin Markov chains because authors are more interested in the tensor itself, so they pay more attention to the convergence of the tensor elements rather than the margins. The convergence issue arises from the indeterminacy of margins, mentioned in Section 2.1, which leads to poor mixing of the Markov chains, consequently hindering convergence. We consider the convergence of margins to be an important aspect for two reasons. First, margins in Tensor VARs are potentially interpretable, as shown in Wang et al. (2021) and Chen, Yang, and Zhang (2022), and discussed in Section 2.2. Second, as the literature on Tensor VARs grows, one cannot guarantee that the Markov chains in a more complex model, for example including time-varying margins, still converge. Apart from the convergence issue, it is computationally expensive to infer the rank using the traditional MCMC scheme since it assumes R to be fixed during the inference. To address the issues aforementioned, we propose three modifications to our inferential framework. Two of these modifications aim to alleviate the poor mixing contributing to the convergence issue. The third modification enhances computational efficiency.

First, we reduce the dependence between columns within \mathbf{B}_j , for $j = 1, 2, 3$, by introducing a block sampler, which divides margins into three blocks according to the three loadings mentioned in Section 2.2. This block sampler is feasible because a Tensor VAR can be written as

$$\mathbf{y}_t^* = \left(\mathbf{x}_t' (\mathbf{B}_3 \otimes \mathbf{B}_2) \mathcal{I}_{(1)} \otimes \mathbf{I}_N\right) \text{vec}(\mathbf{B}_1) + \epsilon_t \quad (3.1)$$

$$= \mathbf{B}_1 \mathcal{I}_{(1)} \left((\mathbf{B}_3' \mathbf{X}_t') \otimes \mathbf{I}_R\right) \text{vec}(\mathbf{B}_2') + \epsilon_t \quad (3.2)$$

$$= \mathbf{B}_1 \mathcal{I}_{(1)} \left(\mathbf{I}_R \otimes (\mathbf{B}_2' \mathbf{X}_t)\right) \text{vec}(\mathbf{B}_3) + \epsilon_t, \quad (3.3)$$

where $\mathcal{I} \in \mathbb{R}^{R \times R \times R}$, $\mathbf{I}_R \in \mathbb{R}^{R \times R}$ is an identity matrix, \otimes is the Kronecker product. Therefore, margins in one loading can be sampled jointly to reduce their dependence on each other.

Second, we do *not* use a usual Gibbs sampler to sample loadings. Instead, we introduce a variant of the ASIS, containing four different parameterizations, to reduce the parameter autocorrelation during the sampling. Given a rank value in each sample iteration, the interweaving Gibbs sampler interweaves between full conditional distributions under a base parameterization and the other three (one for each loading).

Lastly, the rank R in our case is adaptively inferred similarly to Bhattacharya and Dunson (2011) to speed up computation. In the following three sections, we introduce the interweaving Gibbs sampler for a fixed rank in Section 3.2.2 and the adaptive inferential scheme of the rank in Section 3.2.3.

3.2.2. Interweaving Gibbs Sampler

In principle, we could run a standard Gibbs sampler to infer margins and other parameters, but in practice, Markov chains of margins suffer from poor mixing since these chains are highly autocorrelated. We circumvent margins with poor mixing by introducing a variant of the ASIS, which unfolds its strategy from its name: sampling the same block of parameters by interweaving two sampling schemes corresponding to two data augmentations—ancillary statistic and sufficient statistic. The benefit of the ASIS is that the sampling will be at least as good as the sampling from only one data augmentation, and a low correlation between these two augmentations leads to faster convergence and better mixing compared to using either augmentation alone. Because of these benefits, the ASIS has been applied to many models, including stochastic volatility (Kastner and Frühwirth-Schnatter 2014) and factor models (Kastner, Frühwirth-Schnatter, and Lopes 2017).

Our ASIS parameterizations are more related to those in Kastner, Frühwirth-Schnatter, and Lopes (2017) for sampling factor loadings and factors due to the tensor structure. The tensor structure in the Tensor VAR leads to four parameterizations instead of two in Kastner, Frühwirth-Schnatter, and Lopes (2017). The first parameterization, which we call the base one, is \mathbf{B}_1 , \mathbf{B}_2 , and \mathbf{B}_3 described in Section 2.1. The remaining three parameterizations come from specifications of scaling indeterminacy. In particular, $\mathcal{A} = \llbracket \mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3 \rrbracket_{\text{CP}} = \llbracket \mathbf{B}_1^*, \mathbf{B}_2^*, \mathbf{B}_3 \rrbracket_{\text{CP}}$ when $\mathbf{B}_1^*, \mathbf{B}_2^*$ are transformed from

$$\mathbf{B}_1^* = \mathbf{B}_1 \mathbf{D}_1^{-1}, \mathbf{B}_2^* = \mathbf{B}_2 \mathbf{D}_1, \quad (3.4)$$

where \mathbf{D}_1 is a diagonal matrix with nonzero, non-infinite diagonal entries.

There are infinite choices of \mathbf{D}_1 to get this equivalence, but since our objective is boosting the mixing of margins,

we restrict D_1 to be related to B_1 and B_2 . We choose $D_1 = \text{diag}(\beta_{1,1}^{(1)}, \dots, \beta_{1,1}^{(R)})$ for further demonstration. This choice constrains the first row of B_1^* to be ones. Other choices of D_1 will be investigated in future work. After the transformation, we are able to write the model in terms of B_1^* , B_2^* , and D_1 for the second parameterization. For $i_1, i_2 = 1, \dots, N$, we have

$$\begin{aligned}\beta_{1,1}^{*(r)} &= 1, \beta_{1,i_1}^{*(r)} \sim \mathcal{N}\left(0, \left(\frac{\sigma_{1,i_1}^{(r)}}{\beta_{1,1}^{(r)}}\right)^2\right), \\ \beta_{2,i_2}^{*(r)} &\sim \mathcal{N}\left(0, \left(\sigma_{2,i_2}^{(r)} \beta_{1,1}^{(r)}\right)^2\right).\end{aligned}\quad (3.5)$$

The above parameterization only improves the mixing of margins in B_1 and B_2 , so we also need a parameterization to improve the mixing of margins in B_3 . An obvious choice is to pair B_2 and B_3 . At this point, each B_j has been paired at least once, but we conjecture that an additional pair of B_1 and B_3 would provide better mixing than just considering three parameterizations because the mixing would be improved across margins in each pair of B_j 's. Transformations of these two pairs are similar to the one for B_1 and B_2 ,

$$\begin{aligned}B_2^{**} &= B_2 D_2^{-1}, B_3^{**} = B_3 D_2; \\ B_3^{***} &= B_3 D_3^{-1}, B_1^{***} = B_1 D_3,\end{aligned}\quad (3.6)$$

where D_2 and D_3 are diagonal matrices with nonzero, non-infinite diagonal entries.

Similarly, we choose the diagonal entries in D_2 to be the first row of B_2 , and likewise for those in D_3 (as the first row of B_3). These lead to the last two parameterizations which are presented in terms of B_2^{**} , B_3^{**} , D_2 and B_3^{***} , B_1^{***} , D_3 , respectively. For $i_1, i_2 = 1, \dots, N$, $i_3 = 1, \dots, P$, we have

$$\begin{aligned}\beta_{2,1}^{**} &= 1, \beta_{2,i_2}^{**} \sim \mathcal{N}\left(0, \left(\frac{\sigma_{2,i_2}^{(r)}}{\beta_{2,1}^{(r)}}\right)^2\right), \\ \beta_{3,i_3}^{**} &\sim \mathcal{N}\left(0, \left(\sigma_{3,i_3}^{(r)} \beta_{2,1}^{(r)}\right)^2\right),\end{aligned}\quad (3.7)$$

$$\begin{aligned}\beta_{3,1}^{***} &= 1, \beta_{3,i_3}^{***} \sim \mathcal{N}\left(0, \left(\frac{\sigma_{3,i_3}^{(r)}}{\beta_{3,1}^{(r)}}\right)^2\right), \\ \beta_{1,i_1}^{***} &\sim \mathcal{N}\left(0, \left(\sigma_{1,i_1}^{(r)} \beta_{3,1}^{(r)}\right)^2\right).\end{aligned}\quad (3.8)$$

We need to sample margins under the four parameterizations described in each iteration. The sampling using the base parameterization is stated in Appendix B.2, so we focus on sampling margins under the other three parameterizations introduced in this subsection. For $\beta_{1,1}^{(r)}$, its normal prior implies that $(\beta_{1,1}^{(r)})^2$ has a gamma prior, $\text{Gamma}\left(\frac{1}{2}, \frac{1}{2(\sigma_{1,1}^{(r)})^2}\right)$. The full conditional of $(\beta_{1,1}^{(r)})^2$ under (3.5) is a Generalized Inverse Gaussian (GIG)²,

$$\begin{aligned}(\beta_{1,1}^{(r)})^2 &| B_{1,(\cdot,r)}^*, B_{2,(\cdot,r)}^* \\ &\sim \text{GIG}\left(0, \sum_{i_2=1}^M \left(\frac{\beta_{2,i_2}^{*(r)}}{\sigma_{2,i_2}^{(r)}}\right)^2, \sum_{i_1=2}^M \left(\frac{\beta_{1,i_1}^{*(r)}}{\sigma_{1,i_1}^{(r)}}\right)^2 + \left(\frac{1}{\sigma_{1,1}^{(r)}}\right)^2\right).\end{aligned}\quad (3.9)$$

Similarly, we can get full conditionals of $(\beta_{2,1}^{(r)})^2$ under (3.7) and $(\beta_{3,1}^{(r)})^2$ under (3.8):

$$\begin{aligned}(\beta_{2,1}^{(r)})^2 &| B_{2,(\cdot,r)}^{**}, B_{3,(\cdot,r)}^{**} \\ &\sim \text{GIG}\left(\frac{M-P}{2}, \sum_{i_3=1}^P \left(\frac{\beta_{3,i_3}^{**}}{\sigma_{3,i_3}^{(r)}}\right)^2, \sum_{i_2=2}^M \left(\frac{\beta_{2,i_2}^{**}}{\sigma_{2,i_2}^{(r)}}\right)^2 + \left(\frac{1}{\sigma_{2,1}^{(r)}}\right)^2\right),\end{aligned}\quad (3.10)$$

$$\begin{aligned}(\beta_{3,1}^{(r)})^2 &| B_{3,(\cdot,r)}^{***}, B_{1,(\cdot,r)}^{***} \\ &\sim \text{GIG}\left(\frac{P-M}{2}, \sum_{i_1=1}^M \left(\frac{\beta_{1,i_1}^{***}}{\sigma_{1,i_1}^{(r)}}\right)^2, \sum_{i_3=2}^P \left(\frac{\beta_{3,i_3}^{***}}{\sigma_{3,i_3}^{(r)}}\right)^2 + \left(\frac{1}{\sigma_{3,1}^{(r)}}\right)^2\right).\end{aligned}\quad (3.11)$$

Algorithm 1 outlines how to interweave sampling under the base parameterization to the second one described in (3.5). Similar algorithms can be applied to the third and fourth parameterizations, incorporating full conditionals in (3.10) and (3.11). Combining these three algorithms leads to a Gibbs sampler, of which the full algorithm can be found in Appendix C. If we only sample margins using Step (a), the algorithm is just a standard Gibbs sampler with the base parameterization. Every interweaving step starts at the base parameterization, then switches to an alternative parameterization and swaps back to the base one. Note that B_2 in Step (d) has superscript new. This is because B_2 is included in two interweaving steps, but we only store one sample for B_2 in each iteration. It will be easier to distinguish between the one stored (with superscript “new”) and the one left (with superscript new). One can find the same superscripts in the full algorithm.

Algorithm 1 Interweave between the base parameterization and the one in (3.5).

Step (a): Update B_1^{old} under the base parameterization.

Step (b): Store the first row of B_1^{old} into D_1 and determine B_1^* and B_2^* .

Step (c): Sample $(\beta_{1,1}^{\text{new}(r)})^2$ for $r = 1, \dots, R$ using the corresponding full conditional in (3.9) and store sampled values into D_1 .

Step (d): Update B_1^{new} and B_2^{new} with transformation $B_1^{\text{new}} = B_1^* D_1$, $B_2^{\text{new}} = B_2^* D_1^{-1}$.

²A variable $x \sim \text{GIG}(\lambda, \chi, \psi)$ has probability density function $p(x) \propto x^{\lambda-1} \exp(-(\chi/x + \psi x)/2)$.

It is worth stressing that the interweaving strategy improves the mixing of entries in \mathbf{B} up to column permutations and sign-switching issues. Thus, we propose a post-processing procedure to identify the margins a posteriori in [Section 4](#).

3.2.3. Adaptive Inference of Rank

We aim to infer the rank by finding inactive columns in \mathbf{B} , that is those columns which do not contribute much to the tensor \mathcal{A} . An adaptive algorithm, inspired by Bhattacharya and Dunson (2011) and Legramanti, Durante, and Dunson (2020), is displayed in Algorithm C.1.

In this algorithm, we initialize the rank as $R^* = \lceil 5 \log N \rceil$, which is the same as for the number of factors in Bhattacharya and Dunson (2011). Empirically, this initialization is large enough to estimate the coefficient matrix. In order to meet diminishing adaptation condition (Roberts and Rosenthal 2007) for the weak law of large number in adaptive MCMC, we discard inactive columns in the m th iteration with probability $p(m) = \exp(\alpha_0 + \alpha_1 m)$, where $\alpha_0 \leq 0$, $\alpha_1 < 0$. Since $p(m)$ gets smaller as m increases, R is less likely to change during the inference. Lastly, we need to set a criterion to decide whether a column in \mathbf{B} is active. In this paper, this criterion is related to the proportion of small magnitudes in $\mathcal{A}^{(r)}$, for $r = 1, \dots, R$. For ease of explanation, we omit m here. We regard an entry in $\mathcal{A}^{(r)}$ to have a small magnitude if its absolute value is smaller than a threshold γ_1 , for example $\gamma_1 = 10^{-3}$. If the proportion of small magnitudes in $\mathcal{A}^{(r)}$ is larger than another threshold γ_2 set a-priori, for example $\gamma_2 = 0.9$, then we regard the r th column in \mathbf{B} as inactive. We use the simulation study to determine γ_1 and γ_2 so as to minimize the rank inferred while simultaneously ensuring accurate inference of the coefficient matrix. More discussion and details about choosing γ_1 and γ_2 are available in Appendix D.1.

Adaptive inference begins after the \tilde{m} th iteration to stabilize Markov chains and stops at the last iteration during the burn-in period to allow easy interpretation of margins. If the number of inactive columns is greater than 0, we remove these columns in \mathbf{B} and remove corresponding parameters in $\Phi, \delta = (\delta_1, \dots, \delta_R), \tau = (\tau_1, \dots, \tau_R)$. The rank will then be shrunk to a smaller number of active columns. If the algorithm does not detect any inactive column, we first sample a new column in Φ , a new entry in δ and subsequently compute the new entry in τ . A new column in \mathbf{B} will also be sampled using these newly-sampled hyperparameters.

4. Post-Processing Procedure

The interweaving algorithm allows Markov chains to improve mixing, but it does not completely solve the indeterminacy of tensor decomposition, which is the origin of the non-convergence of Markov chains. Therefore, we propose a post-processing procedure to identify margins a posteriori. Note that there exist methods to identify margins a priori. For example, Zhou, Li, and Zhu (2013) restricted $\mathbf{B}_{1,(1,\cdot)}$ and $\mathbf{B}_{2,(1,\cdot)}$ as ones and sorted elements in $\mathbf{B}_{3,(1,\cdot)}$ in descending order. We opt to maintain an unrestricted tensor decomposition because it can incorporate the increasing shrinkage property of the MGP, enabling us to infer the rank.

The procedure proposed is inspired by the Match-Sign-Factor (MSF) algorithm in the R package *infinitefactor* (Poworoznek, Ferrari, and Dunson 2021). The MSF performs a greedy search to rotate factor loadings and factors in factor models, and we apply a variant of this algorithm to Tensor VARs. Our algorithm is presented in 2, along with a detailed explanation divided into two parts: (a) solve column permutations by the label-matching method (up to line 11); (b) solve sign-switching issues by the sign-matching method.

Column permutations in \mathbf{B} are equivalent to those in \mathbf{B}_3 , so if we solve the equivalent issue in \mathbf{B}_3 , we will automatically solve column permutations in \mathbf{B} . There are analogous equivalences related to \mathbf{B}_1 and \mathbf{B}_2 , but the empirical finding in Figure D.2 shows that the label matching related to \mathbf{B}_3 gives the best mixing results in the simulation study. The label matching needs a *pivot* matrix $\mathbf{B}_3^{(\text{pivot})}$ as a template to align \mathbf{B}_3 sampled in each iteration, that is columns in \mathbf{B}_3 after label being matched will have the same order as that of columns in $\mathbf{B}_3^{(\text{pivot})}$. Following Poworoznek, Ferrari, and Dunson (2021), $\mathbf{B}_3^{(\text{pivot})}$ is the one with the median of the condition number $\kappa = \sigma_{\max}(\mathbf{B}_3)$, where $\sigma_{\max}(\mathbf{B}_3)$ is the maximal singular value of \mathbf{B}_3 .

After choosing the pivot, we compute the Euclidean distance between columns in \mathbf{B}_3 in each iteration and $(\mathbf{B}_3^{(\text{pivot})}, -\mathbf{B}_3^{(\text{pivot})})$, and store the distances into an R -by- $2R$ distance matrix Θ with row and column indices corresponding to columns in \mathbf{B}_3 and $(\mathbf{B}_3^{(\text{pivot})}, -\mathbf{B}_3^{(\text{pivot})})$, respectively. As shown in [Algorithm 2](#), a greedy algorithm then starts from the lowest Euclidean distance to align the corresponding column in \mathbf{B}_3 to that in $\mathbf{B}_3^{(\text{pivot})}$ or $-\mathbf{B}_3^{(\text{pivot})}$, and these columns will not be matched again. The label matching is finished after repeating the procedure for R times.

Next, we explain the sign-matching method. For $j = 1, 2, r = 1, \dots, R$, we determine whether to flip the sign of $\mathbf{B}_{j,(r,\cdot)}$ by comparing its distances to both $\mathbf{B}_{j,(r,\cdot)}^{(\text{pivot})}$ and $-\mathbf{B}_{j,(r,\cdot)}^{(\text{pivot})}$. The general guideline for flipping signs in $\mathbf{B}_{3,(r,\cdot)}$ is to do so only if this procedure identifies the tensor, that is the tensors before and after sign-matching are the same. If not, we leave the sign unflipped.

5. Simulation Results

5.1. Data and Implementation

We assess the merits of inferring ranks using the MGP and the adaptive inferential scheme in [Section 5.2](#), compared to the M-DGDP (Guhaniyogi, Qamar, and Dunson 2017) prior commonly used in tensor-structured models. [Section 5.3](#) shows that the interweaving strategy can improve the mixing of margins, and the post-processing procedure identifies the margins. We will leave the comparison of predictive performance to the real data example. The following two subsections use the same simulated data, which includes three scenarios with different combinations of the number of time series and rank (N, R): (10, 3), (20, 5), and (50, 10). The lag order is $P = 3$. We assume that the true rank increases with the number of time series. Kolda and Bader (2009) and the reference therein summarize ranks of some specific third-order tensors, but the rank of a

Algorithm 2 Match Labels and Signs

```

1: Find a pivot matrix  $\mathbf{B}_3^{(\text{pivot})}$  and its corresponding tensor matrix  $\mathbf{B}^{(\text{pivot})}$ 
2: for each iteration do
3:   Compute the  $R$ -by- $2R$  distance matrix  $\Theta$ 
4:   for  $r = 1, \dots, R$  do
5:     Find  $(r_1^*, r_2^*) = \underset{r_1, r_2}{\operatorname{argmin}} \Theta_{r_1, r_2}$ 
6:     if  $r_2^* \leq R$  then
7:       Match the  $r_1^*$ th column in  $\mathbf{B}_3$  to the  $r_2^*$ th column in  $\mathbf{B}_3^{(\text{pivot})}$ .
8:       Change the  $r_1$ th row,  $r_2$ th and  $(R + r_2)$ th columns in  $\Theta$  to infinity.
9:     else
10:      Match the  $r_1^*$ th column in  $\mathbf{B}$  to the  $(r_2^* - R)$ th column in  $\mathbf{B}_3^{(\text{pivot})}$ .
11:      Change the  $r_1$ th row,  $(r_2 - R)$ th and  $r_2$ th columns in  $\Theta$  to infinity.
12:     for  $j = 1, 2$  do
13:       Compute distance  $d_1 = d(\mathbf{B}_{j,(\cdot,r)}, \mathbf{B}_{j,(\cdot,r)}^{(\text{pivot})})$  and  $d_2 = d(\mathbf{B}_{j,(\cdot,r)}, -\mathbf{B}_{j,(\cdot,r)}^{(\text{pivot})})$ 
14:       if  $d_1 \leq d_2$  then
15:         Keep signs in  $\mathbf{B}_{j,(\cdot,r)}$ . Record  $\text{ind}_{j,r} = 1$ 
16:       else
17:         Flip signs in  $\mathbf{B}_{j,(\cdot,r)}$ . Record  $\text{ind}_{j,r} = -1$ 
18:       if  $\text{ind}_{1,r} \text{ind}_{2,r} = 1$  then
19:         Keep the signs in  $\mathbf{B}_{3,(\cdot,r)}$ 
20:       else
21:         Flip the signs in  $\mathbf{B}_{3,(\cdot,r)}$ 

```

Table 1. Uniform distributions of margins in different locations indicated by rows and different combinations of N and R indicated by columns.

	(10,3)	(20,5)	(50,10)
B_1	$U(-1,1)$	$U(-1,1)$	$U(-1,1)$
B_2	$U(-1,1)$	$U(-1,1)$	$U(-0.6,0.6)$
$B_{3,(1,\cdot)}$	$U(-1,1)$	$U(-1,1)$	$U(-0.6,0.6)$
$B_{3,(2,\cdot)}$	$U(-0.5,0.5)$	$U(-0.2,0.2)$	$U(-0.2,0.2)$
$B_{3,(3,\cdot)}$	$U(-0.1,0.1)$	$U(-0.1,0.1)$	$U(-0.1,0.1)$

tensor applied in a VAR with lag order exceeding 2 is not specified. Only an upper bound of the rank is available, which is $\min(N^2, NP)$.

In each scenario, we generate 25 datasets following VAR(3) models with independently generated parameters. The coefficient matrix of each model is the 1-mode matricization of a tensor from a CP decomposition, and the covariance matrix is an identity matrix. Margins of the CP decomposition follow uniform distributions with different parameters, see Table 1 for more details. All time series are checked for stationarity via the Dickey-Fuller test and the Kwiatkowski-Phillips-Schmidt-Shin (KPSS) tests with significance level set as 5%. All datasets are consistent with stationarity.

We apply the MGP to both simulation experiments by setting $\nu = 3$ as shown in Bhattacharya and Dunson (2011), $\gamma_1 = 10^{-3}$, and $\gamma_2 = 0.9$. A table illustrating the sensitivity to the choice of γ_1 and γ_2 is available in Appendix D.1. Our chosen combination of γ_1 and γ_2 gives the most parsimonious model and the narrowest 90% credible interval of inferred rank. Apart from the MGP, we briefly introduce the M-DGDP prior, which is a global-local shrinkage prior proposed for tensor margins with the following expression:

$$\begin{aligned}
\beta_j^{(r)} &\sim \mathcal{N}(\mathbf{0}, (\phi_r \tau) \mathbf{W}_{jr}), \quad w_{jr,k} \sim \text{Exp}(\lambda_{jr}/2), \\
\lambda_{jr} &\sim \text{Gamma}(a_\lambda, b_\lambda), \\
\Phi &= (\phi_1, \dots, \phi_R)' \sim \text{Dirichlet}(\alpha, \dots, \alpha), \\
\tau &\sim \text{Gamma}(a_\tau, b_\tau),
\end{aligned}$$

where $\mathbf{W}_{jr} = \text{diag}(w_{jr,1}, \dots, w_{jr,I_j})$, $I_j = N$ when $j = 1, 2$ and $I_j = P$ when $j = 3$ in our case. α is uniformly distributed on a grid with values equally placed on $[R^{-3}, R^{-0.01}]$, and R is the rank set in advance. We follow the same setting of hyperparameters as in Guhaniyogi, Qamar, and Dunson (2017), that is $a_\lambda = 3$ and $b_\lambda = \sqrt[3]{a_\lambda}$, $a_\tau = R\alpha$, $b_\tau = \alpha \sqrt[3]{R}$.

For both priors, the initialization of rank is $\lceil 5 \log(N) \rceil$, but the adaptive inferential scheme is only applied when using the MGP after iteration reaches 200 in the burn-in period. For the M-DGDP, the rank is determined a posteriori by removing negligible margins as in Algorithm C.1. We implement all simulations with Intel(R) Xeon(R) Gold 6140 CPU 2.30GHz and R 4.2.0.

5.2. Rank Selection

The first simulation assesses our approach to infer the rank R . Both samplers with MGP and M-DGDP were run for 10,000 iterations after 10,000 burn-in and incorporated the interweaving strategy. We record the performance of MGP and M-DGDP in Table 2 including four metrics: (a) mean squared error (MSE) of the coefficient matrix for coefficient accuracy; (b) averaged effective sample size (ESS) of coefficients for sampling efficiency; (c) averaged rank inferred (R) for rank accuracy; and (d) approximate running time for computational efficiency.

According to Table 2, both models estimate coefficient matrices with similar accuracy under the MSE. The MGP is able to infer ranks equal to or lower than the true ones. In contrast, M-DGDP can infer the true ranks after deleting redundant columns of which the corresponding averaged proportions of small magnitudes ($\gamma_1 = 10^{-3}$) are greater than $\gamma_2 = 0.9$. The MGP also explores coefficient posteriors more efficiently, as ESS results from the first two scenarios suggested. The adaptive shrinkage algorithm accelerates computation since the running time of the MGP grows more slowly with N and R compared to the growth rate of the M-DGDP. This leads to a large difference if $N = 50$ and $R = 10$, where the inference with the MGP runs more than 5 times faster than the M-DGDP.

5.3. Quality of Markov Chains

The second simulation investigates the quality of Markov chains, that is whether the interweaving strategy and the post-processing procedure contribute to the mixing and convergence of Markov chains. We choose three prior settings (standard normal, MGP, M-DGDP) to infer margins with/without interweaving. The burn-in period still has 10,000 iterations, but we change the number of iterations after burn-in to 100,000 to demonstrate results with longer chains.

We first focus on the interweaving strategy by conducting the post-processing procedure to both samples with/without interweaving. To give an insight into the effect of interweaving, Figure 1 shows trace plots of the margin $\beta_{1,1}^{(1)}$ when $N = 10$ and $R = 3$ based on different prior settings with/without interweaving. Even though we used the label- and sign-matching methods, trace plots without interweaving still suffer from the mixing problem, while the interweaving strategy substantially

improves mixing. The autocorrelations (acfs) of all draws of $\beta_{1,1}^{(1)}$ after the burn-in period, see Figure D.1, also support the merit of the interweaving strategy.

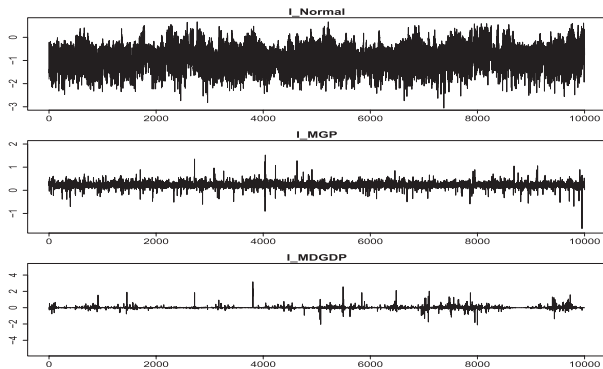
We follow the procedure in Kastner, Frühwirth-Schnatter, and Lopes (2017) to compute the inefficiency factor (IF) of each margin in different scenarios and prior settings. A smaller IF means that the sampling of a parameter is more efficient. Figure 2 displays boxplots of IFs where each panel corresponds to a scenario with a combination of (N, R) and B_j . Each boxplot contains 25 data points from the 25 simulation datasets. Each data point in a boxplot is the IF of the 1-1 entry of B_j , for $j = 1, 2, 3$, inferred from one dataset. We exclude outliers because there are only a handful of them, and this exclusion allows us to focus on the medians and quantiles of IFs. Overall, most IFs with interweaving have lower median values and less variation than their counterparts without interweaving.

We then use the Stable Gelman-Rubin method (Vats and Knudson 2021) to diagnose the convergence of the margin Markov chains. The reason why we apply the Stable Gelman-Rubin instead of the Gelman-Rubin (Gelman and Rubin 1992) is 2-fold: (a) the Gelman-Rubin is suitable when the simulation has multiple Markov chains for each parameter, while our simulation only has one Markov chain for each parameter. The Stable Gelman-Rubin can be applied to both multiple and single Markov chains; (b) The conventional Gelman-Rubin threshold of 1.1 implies an approximation of ESS of 5 according to Vats and Knudson (2021), and the authors propose a threshold depending on the parameter dimension and a significance level. The results are presented in Table 3, where each cell is the averaged proportion of margins of which the Markov chains are determined as convergent. Overall, the algorithm with interweaving achieves over 90% convergent Markov chains in all scenarios and with all prior choices. All proportions are higher based on the results from the interweaving algorithm compared to the non-interwoven ones. We also include the Geweke diagnostic (Geweke 1991) in Appendix D.1, with most interweaving results having a better convergence performance.

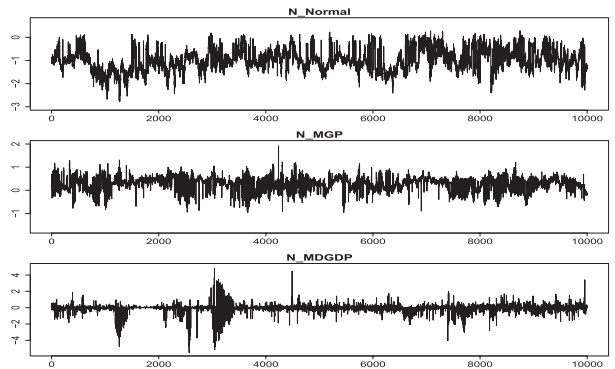
Lastly, we demonstrate the necessity of the post-processing procedure. Figure 3 displays trace plots of the whole draws (with thinning of 10) of two selected margins inferred with interweaving strategy, and we exclude the post-processing procedure at this time. All three panels in Figure 3(a) and the

Table 2. Performance of MGP and M-DGDP in 25 simulations for different dimensionality combinations.

(N, R)	method	MSE	R	ESS	Running time (hr)
(10, 3)	MGP	0.006	4	3977.539	0.45
	M-DGDP	0.006	3	3938.573	1.16
(20, 5)	MGP	0.008	4	2657.043	0.59
	M-DGDP	0.008	5	2644.262	2.60
(50, 10)	MGP	0.006	7	2125.425	2.52
	M-DGDP	0.006	10	2315.662	13.34



(a) With interweaving



(b) Without interweaving

Figure 1. Trace plots of the first 10,000 draws of $\beta_{1,1}^{(1)}$ in $N = 10, R = 3$ scenario after burn-in period. The inferential scheme adopts standard normal (top), MGP (middle), and M-DGDP (bottom) as priors and applies with (left panel) and without (right panel) interweaving strategy.

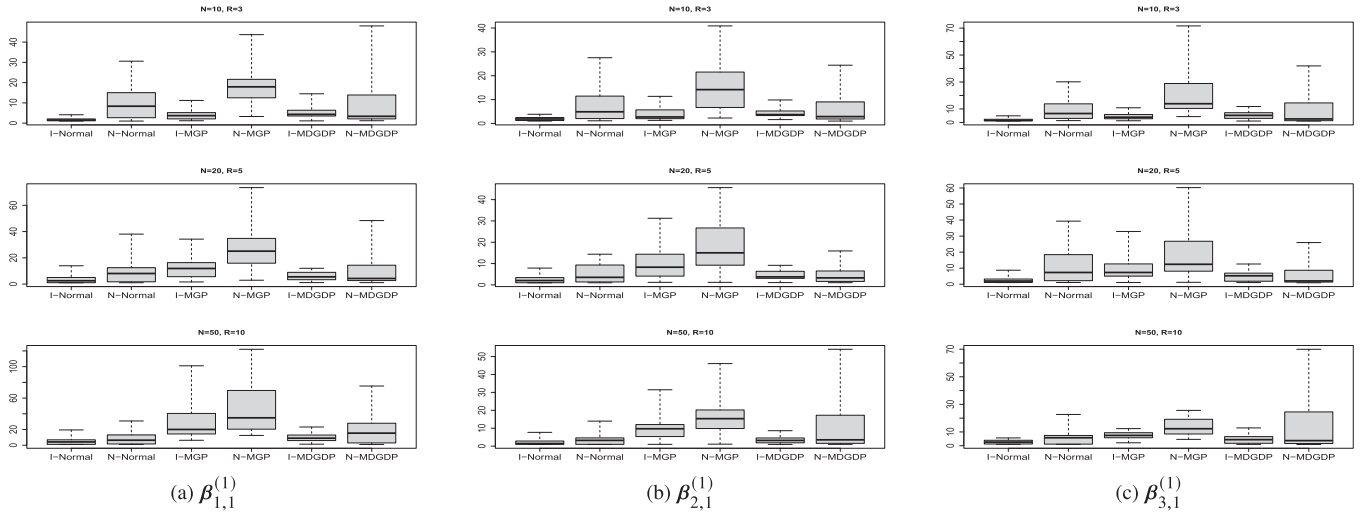


Figure 2. Boxplots of inefficiency factor of the 1–1 entry of B_1 (left), B_2 (middle), and B_3 (right) from different scenarios: $(N, R) = (10, 3)$ (top), $(N, R) = (20, 5)$ (middle), and $(N, R) = (50, 10)$ (bottom). Inferential schemes with and without interweaving are represented as "I-" and "N-", respectively, followed by a prior setting.

Table 3. Averaged proportions of margins which are convergent according to stable Gelman Rubin Statistics.

$N=10, R=3$	Interweaving	Non-interwoven	$N=20, R=5$	Interweaving	Non-interwoven	$N=50, R=10$	Interweaving	Non-interwoven
Normal	1.000	0.847	Normal	0.996	0.916	Normal	0.996	0.978
MGP	0.998	0.866	MGP	0.986	0.740	MGP	0.940	0.770
MDGDP	0.996	0.871	MDGDP	0.998	0.819	MDGDP	0.989	0.858

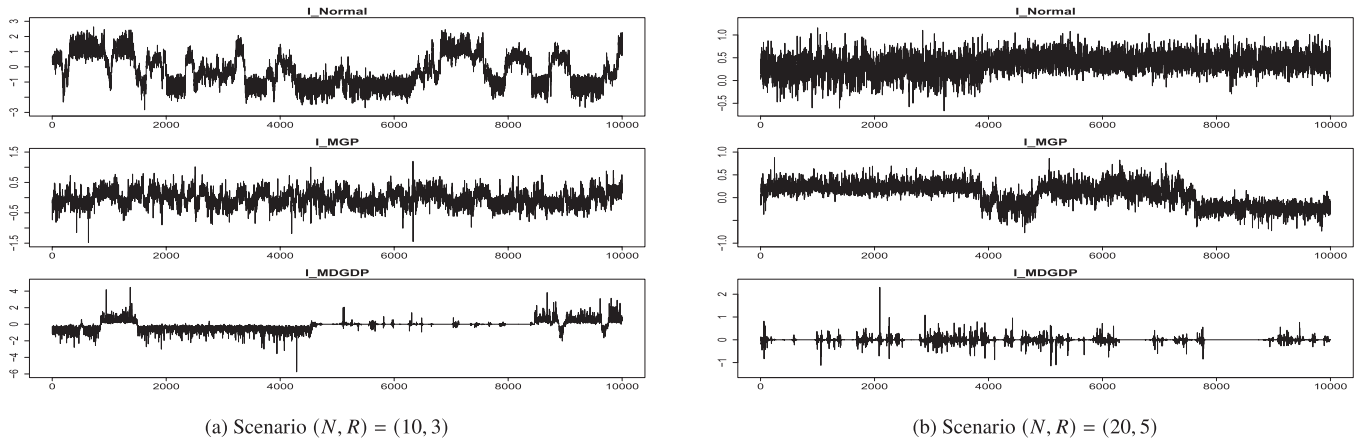


Figure 3. Trace plots of $\beta_{1,1}^{(1)}$ in $N = 10, R = 3$ scenario (left) and $\beta_{1,2}^{(1)}$ in $N = 20, R = 5$ scenario (right) after burn-in period. The inferential scheme adopts standard normal (top), MGP (middle), and M-DGDP (bottom) as priors and applies with the interweaving strategy.

middle panel in Figure 3(b) have sign-switching issues. If we do not match signs, the interpretation of margins will be infeasible because the posterior mode or mean of some margins would be zero, but they should be nonzero. The top panel in Figure 3(b) provides evidence of column permutations, with the sample mean moving from 0 to 0.5. The bottom panel in Figure 3(b) has neither sign switching nor column permutations, but the M-DGDP does not guarantee convergence only with the interweaving strategy due to the evidence provided in Figure 3(a).

6. Real Data Application

6.1. Data and Implementation

We use the U.S. macroeconomic data extracted from Federal Reserve Economic Data (FRED)³ (McCracken and Ng 2020) to assess the utility of Tensor VARs. The data spans from 1959Q1 to 2019Q4, and are transformed to stationarity and standardized to have mean zero and variance one to avoid scaling issues. We construct medium-scale and large-scale datasets by selecting 20 and 40 variables, respectively, as referred to in Korobilis and

³The data is available at <https://research.stlouisfed.org/econ/mccracken/fred-databases/>.

Pettenuzzo (2019). The selected 40 variables can be divided into eight categories: (i) output and income, (ii) consumption, orders and inventories, (iii) labor market, (iv) prices, (v) interest rate, (vi) money and credit, (vii) stock market and (viii) exchange rate. Since no variables in the categories of money and credit and the stock market are selected into the medium-scale dataset, we also construct an alternative 20-variable dataset that contains variables from all eight categories. We use this alternative dataset to examine the robustness of forecasting performance with results available in Appendix D.2. A full description of the variables selected and their transformations can be found in Appendix E. Since the decomposition of the covariance matrix $\mathbf{\Omega}_t$ has a lower triangular matrix \mathbf{H} in the model, the order of time series matters. We follow Bernanke, Boivin, and Elias (2005) by splitting time series into slow, fast groups and Federal Funds Rate (FEDFUNDS). The slow group contains variables that respond to a shock of FEDFUNDS with a lag, and variables in the fast group respond to it contemporaneously. The order is slow variables, FEDFUNDS, and fast variables.

For each dataset, we estimate various VAR models with five lags. Tensor VARs with and without the additional own-lag matrix \mathbf{D} are denoted as Tensor MGP Own-lag and Tensor MGP, respectively. For these two Tensor VARs, we use the same choice of γ_1 and γ_2 as in the simulation study. Implementation of the MGP is the same as in Section 5, and the prior of \mathbf{D} is described in Section 3.1. For competitors, we include standard VARs with the hierarchical Minnesota (Giannone, Lenza, and Primiceri 2015), Horseshoe (Carvalho, Polson, and Scott 2009) and a specification of normal-gamma (NG) prior introduced to VARs by Huber and Feldkircher (2019). All of these three priors can be written as $\mathbf{A}_{p,(i,j)} \sim \mathcal{N}(0, \mathbf{V}_{p,(i,j)})$ for (i,j) entry in \mathbf{A}_p , where $i, j = 1, \dots, N$ and $p = 1, \dots, 5$. For the

$$\text{hierarchical Minnesota, } \mathbf{V}_{p,(i,j)} = \begin{cases} \frac{\lambda_1^2}{p^2}, & \text{if } i = j \\ \frac{\lambda_1^2 \lambda_2}{p^2} \frac{\hat{\sigma}_i}{\hat{\sigma}_j}, & \text{if } i \neq j \end{cases}, \text{ where } \hat{\sigma}_i^2$$

is the variance estimate of $\mathbf{y}_{t,i}$ sequence modeled by an AR(5) process. λ_1 and λ_2 have prior Gamma(0.01, 0.01) and are inferred using a random walk Metropolis-Hastings step. For Horseshoe prior, $\mathbf{V}_{p,(i,j)} = \lambda_{p,(i,j)}^2 \tau^2$, where $\lambda_{p,(i,j)}^2$ and τ are local and global parameters, respectively, following a half Cauchy prior. We apply the NG described in Section 3.1 to the coefficient matrix. Priors of \mathbf{H} and stochastic volatility \mathbf{S}_t , for $t = 1, \dots, T$, are the same for all models. The MCMC sampler runs 10,000 iterations after the 10,000 burn-in period.

Note that the decomposition of $\mathbf{\Omega}_t$ employs a triangular system due to the lower triangular matrix \mathbf{H} , which might lead to the ordering issue when estimating the parameters. This issue has been discussed in Carriero, Clark, and Marcellino (2019), Chan, Koop, and Yu (2024), Arias, Rubio-Ramirez, and Shin (2023), among others. Thus, we also provide the forecasting performance of which we apply a nonrestrictive matrix \mathbf{H} , as defined in Chan, Koop, and Yu (2024). The results and further discussion about this order-invariant model are available in Appendix D.2.

Table 4. Averaged number of parameters and running time of Tensor MGP, Tensor MGP Own-lag and standard VARs with the NG prior.

	Number of parameters		Running time (hr)	
	Medium	Large	Medium	Large
Tensor MGP	187.18	257.361	0.95	3.14
Tensor MGP Own-lag	272.19	456.18	1.07	3.28
Standard VAR	2000	8000	1.30	10.39

6.2. Forecasting Results

Before delving into the evaluation of forecasting performance, we compare Tensor VARs and standard VARs with the NG prior in computational time and number of parameters (margins or coefficients) inferred. As shown in Table 4, fewer parameters were inferred within the Tensor VAR framework, leading to the reduced computing time of this framework compared to standard VARs. For the medium-scale dataset, Tensor VARs require at least six times fewer parameters than standard VARs. Similarly, for the large-scale dataset, Tensor VARs only need to infer fewer than 10% of the parameters compared to those inferred from standard VARs. In term of running time, Tensor and standard VARs take a similar amount of time to infer the medium-scale dataset, but the former requires approximately one-third of the time taken by the latter when we switch to the large-scale dataset. The inference using Tensor MGP is faster than Tensor MGP Own-lag because the latter necessitates additional time to infer the own-lag matrix. Note that the code for both VAR frameworks has been accelerated by the Rcpp package.

We follow the expanding window procedure to assess the forecasting performance of our models. Specifically, we first fit each VAR model with the historical data from 1959Q1 to 1984Q4, then get 1-, 2-, and 4-step-ahead forecasts for 1985Q1, 1985Q2, and 1985Q4, respectively. Next, we expand the historical data with the endpoint at 1985Q1 and conduct the multi-step-ahead forecasting again. This procedure is repeated iteratively and stops after conducting the 1-step-ahead forecast of 2019Q4.

We evaluate the forecasting performance of Tensor VARs and standard VARs with both joint and marginal results. For the marginal ones, we select seven variables which are salient to the U.S. economy, as shown in Tables 5 and 6. The metrics for the forecasting evaluation are mean squared forecast error (MSFE), mean absolute error (MAE) and averaged log predictive likelihood (ALPL), see Appendix D.2 for mathematical expressions. All marginal metrics are relative to a standard VAR with a flat prior, taking the 7 time series selected as responses.

Results about point forecasts evaluated by MSFE and MAE can be found in Appendix D.2. Overall, Tensor VARs achieve better joint and marginal performance than standard VARs. Tables 5 and 6 present density forecasting performance from the medium and large datasets. Tensor VARs have competitive performance when making joint density forecasts. They also outperform standard VARs in marginal forecasts since they are the best models in 12 and 13 out of 21 cases for medium and

Table 5. ALPL of joint and marginal variables using the medium-scale dataset.

Model	Horizon	ALPL							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
Tensor MGP	1	−16.378	0.170	0.151	0.637	0.177	0.150	0.124	0.160
	2	−17.820	0.416	0.227	0.634	0.240	0.284	0.141	0.128
	4	−19.460	0.671	0.179	0.498	0.196	0.306	0.110	0.077
Tensor MPG Own-lag	1	−16.184	0.190	0.147	0.682	0.191	0.172	0.133	0.163
	2	−17.852	0.424	0.229	0.656	0.249	0.289	0.144	0.127
	4	−19.567	0.702	0.171	0.526	0.207	0.310	0.113	0.081
Minnesota	1	−15.921	0.129	0.183	0.519	0.141	0.164	0.181	0.187
	2	−18.126	0.443	0.210	0.507	0.202	0.301	0.134	0.141
	4	−19.897	0.754	0.142	0.379	0.152	0.291	0.086	0.082
NG	1	−16.463	0.126	0.126	0.640	0.131	0.153	0.149	0.162
	2	−18.277	0.402	0.193	0.588	0.183	0.272	0.130	0.126
	4	−19.995	0.724	0.140	0.448	0.170	0.281	0.096	0.081
Horseshoe	1	−17.333	−0.164	0.090	0.633	0.112	0.048	0.168	0.152
	2	−18.394	0.214	0.199	0.626	0.162	0.223	0.146	0.130
	4	−19.464	0.632	0.141	0.495	0.156	0.257	0.104	0.108

NOTE: The best forecasts are in bold.

Table 6. ALPL of joint and marginal variables using the large-scale dataset.

Model	Horizon	ALPL							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
Tensor MGP	1	−24.520	0.078	0.126	0.670	0.151	0.135	0.103	0.178
	2	−29.790	0.401	0.231	0.686	0.213	0.286	0.133	0.151
	4	−33.847	0.703	0.171	0.532	0.172	0.353	0.108	0.099
Tensor MPG Own-lag	1	−23.809	0.101	0.143	0.688	0.159	0.172	0.116	0.175
	2	−30.338	0.389	0.240	0.673	0.217	0.298	0.138	0.151
	4	−35.631	0.686	0.176	0.533	0.171	0.334	0.113	0.101
Minnesota	1	−26.576	−0.073	0.147	0.534	0.103	0.035	0.133	0.174
	2	−29.600	0.330	0.252	0.570	0.173	0.212	0.148	0.162
	4	−32.545	0.736	0.175	0.445	0.157	0.243	0.105	0.095
NG	1	−28.455	0.081	0.133	0.518	0.107	0.167	0.130	0.172
	2	−32.823	0.421	0.218	0.518	0.163	0.316	0.136	0.145
	4	−36.715	0.793	0.154	0.386	0.159	0.312	0.104	0.085
Horseshoe	1	−27.915	0.064	0.129	0.584	0.114	0.138	0.124	0.178
	2	−31.462	0.408	0.238	0.580	0.178	0.295	0.144	0.158
	4	−34.874	0.784	0.165	0.431	0.165	0.299	0.104	0.097

NOTE: The best forecasts are in bold.

large datasets, respectively. Forecasts of FEDFUNDS, GDP, and UNRATE are more favorable when using Tensor VARs, while standard VARs have better performance in forecasting PAYEMS and GDPDEFL. In comparing the performance of the two models within Tensor VARs, Tensor MGP Own-Lag demonstrates superior results to Tensor MGP. If we focus on individual models in standard VARs, the hierarchical Minnesota prior is the best among these three priors. The superior performance of Tensor MGP Own-lag and the hierarchical Minnesota highlights the importance of own-lag effect in economic data. When comparing each marginal evaluation in these two tables, most ALPLs in Table 6 are larger than those in Figure 5, indicating that the large amount of information is advantageous for the marginal forecasting.

6.3. Interpretation

Since Tensor MGP Own-lag performs better than Tensor MGP, we demonstrate how to interpret a Tensor VAR by fitting it with the whole large-scale dataset ($N=40$). The Tensor VAR infers a rank of 3, reducing the number of parameters in the coefficient matrix from 8000 (standard VAR(5)) to 455.

According to (2.5), a Tensor VAR can be interpreted as a factor model with observable factors. Figure 4 shows these factors are consistent with recession periods reported by the National Bureau of Economic Research (NBER) (available on <https://fred.stlouisfed.org/series/USRECQ>). The first factor has wider credible intervals during or after the NBER recession periods. The second factor peaks during these recession periods and has relatively high values during the recession of 1960–1961 and the dot-com bubble in the early 2000s. The third factor peaks after recession periods, and the reason will be explained later according to Figure 5. Furthermore, we present the variables that exhibit the five highest magnitudes of correlation with these three factors in Table 7. The first factor shows a high correlation with variables from the money and credit category, while the second factor is highly correlated to the variables from the labor market and industrial production. The correlations associated with PAYEMS and UNRATE are reversed, indicating that the second factor is positively linked to unemployment. Proceeding to the third factor, M2REAL and BUSLOANx are both found in the first and third columns in Table 7, but we consider the third factor to bear a connection with the financial market due to its high correlation with the S&P price earning ratio. It may seem sur-

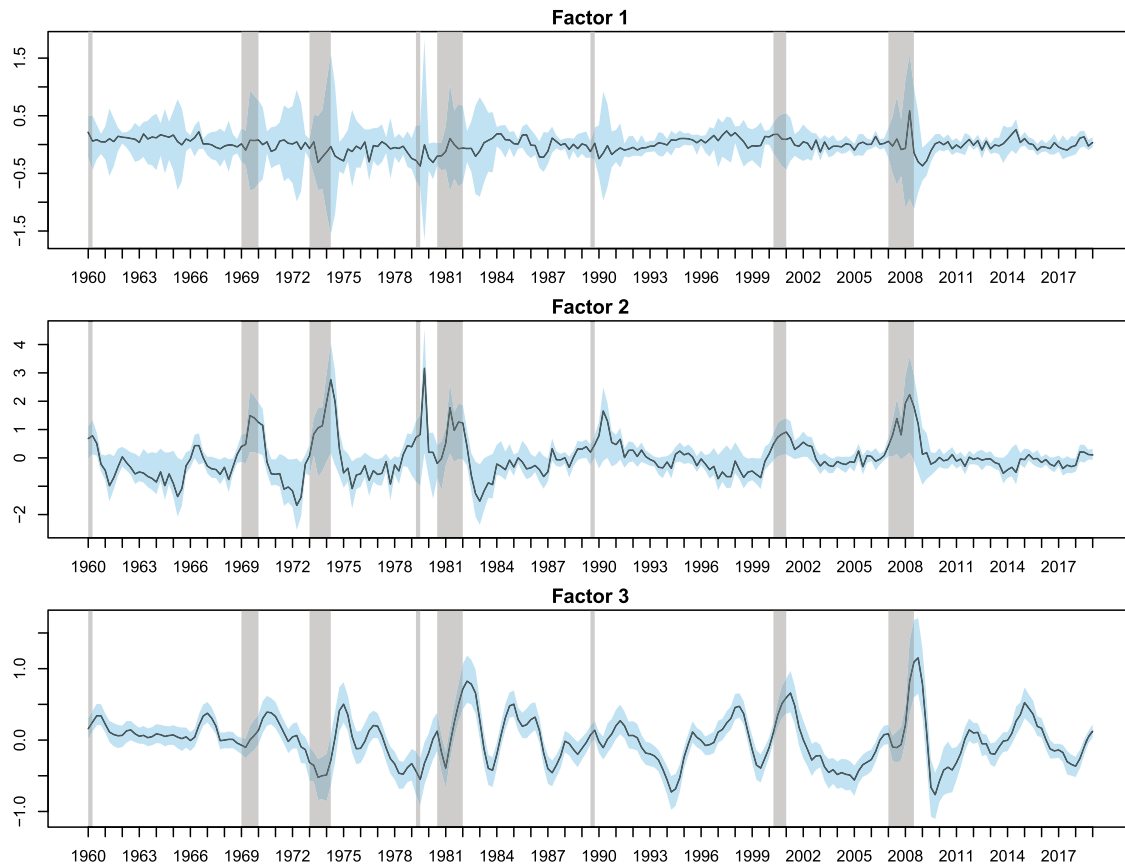


Figure 4. Time series plots of factors with median (solid line) and 80% credible interval (blue shade). The vertical gray shades correspond to the U.S. recession periods. The factors are derived from the inferential results of Tensor MGP Own-lag.

prising that none of the factors shows a strong connection with interest rates, but all three factors have a non-negligible correlation to interest rates according to the full correlation displayed in Appendix D.3.

Next, we use Figure 5 to answer two questions: (a) which lagged time series contribute to the factors; (b) what is the effect from factors to responses. Figure 5 depicts the posterior mean of response, predictor and temporal loadings. Larger margin magnitudes are associated with more deeply saturated hues.

The first question is answered by the predictor and temporal loading. The columns with the same index in these two loadings reveal how the corresponding factor is constructed. For the first factor representing money and credit, we inspect the top five margin magnitudes (M2REAL, CPIAUCSL, RPI, M2SL, and OILPRICEx) in the first column of the predictor loading, and show that price is the main category contributing to this factor. The negative margins of CPIAUCSL and OILPRICEx indicate that prices have a negative effect on the first factor. This conclusion is further strengthened by the opposite signs of M2REAL and M2SL margins since M2SL drops while M2REAL rises with decreasing prices. Additionally, the positive RPI margin, which is adjusted by inflation, supports this conclusion. In the first column of temporal loading, the first lag is suggested to be the most important one because its magnitude is the largest within the corresponding column. Combined with the findings from predictor and temporal loading, the first factor is formed by the prices one quarter ago. We follow a similar method to investigate the formation of the second factor and get the following finding:

First, A decline in real M2 money supply (M2REAL) and personal consumption expenditures (PCECC96) contributes to an increase in this factor about unemployment. Second, the factor grows with the increase of credit risk because of the opposite signs of BAA and GS10, representing the spread between the corresponding two yields. Akin to the formation of the first factor, the first lag exhibits the most significant contribution to the formation of the second factor. Lastly, we focus on the columns corresponding to the third factor and find two differences compared to other columns: (a) margins with relatively high magnitudes are related to the financial market, for example, oil price (OILPRICEx) in the commodity market, exchange rates (EXSZUsx and EXCAUSx) in the FX market; (b) the column in the temporal loading spans in all five lags, which explains why the third factor peaks after the recession periods.

The second question is answered by the response loading, which has the same definition as the factor loading in a factor model if one considers the factors from the Tensor MGP Own-lag as factor scores. Each column of the response loading shows how each factor impacts the responses. In the first column, margins corresponding to variables in the money and credit category have high magnitudes, which follows expectation because the first factor represents this category. Assume that the first factor to be positively associated with money supply given the evidence in Table 7, we can explain the negative margins of interest rates: during economic downturns, both rate cuts and quantitative easing are applied as part of the monetary policy toolkit to boost economic activity. Similarly, the positive margins in the

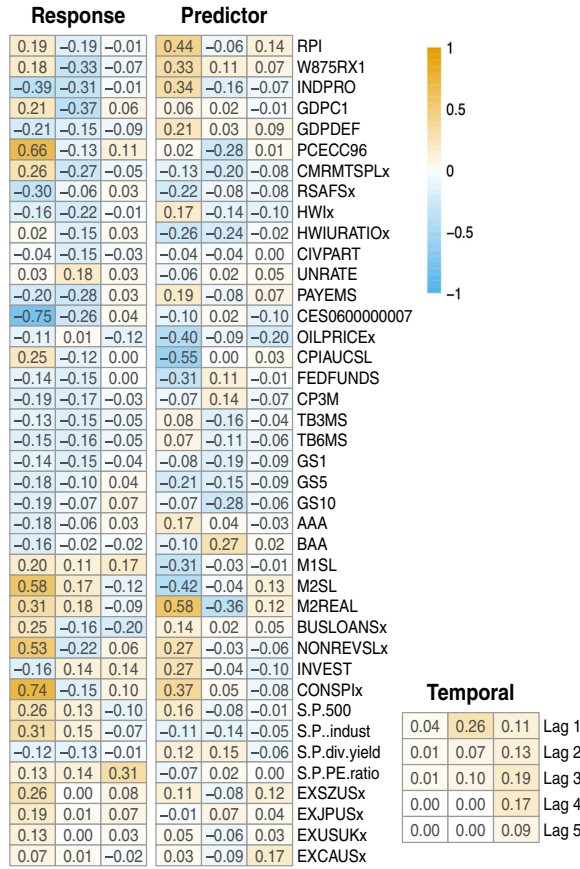


Figure 5. Posterior mean of response, predictor and temporal loadings inferred from Tensor MGP Own-lag.

Table 7. Variables with the top five correlations with the factors.

Factor 1	Factor 2	Factor 3
M2REAL (0.44)	PAYEMS (-0.84)	S&P PE ratio (0.62)
NONREVSLx (0.41)	UNRATE (0.76)	M2REAL (0.41)
CONSPlx (0.40)	INDPRO (-0.72)	BUSLOANSx (-0.31)
BUSLOANSx (0.32)	HWIURATIOx (-0.69)	INVEST (0.30)
PCECC96 (0.26)	HWIix (-0.62)	M2SL (0.28)

exchange rate category suggest the depreciation of U.S. dollars when the money supply increases in the United States. Moving to the second column, the negative margins in the income and output category have high magnitudes, suggesting an increase in this unemployment factor (the second factor) results in the slowdown of economic activities. Negative margins of interest rates show the expectation of interest rate reduction, given that the second factor rises. If we look at the loading corresponding to the third factor, it is unsurprising that the largest margin corresponds to S&P PE ratio because the third factor is highly correlated to this variable.

6.4. Effect of D

This section compares the Tensor VARs with and without the own-lag matrix D . First, we do not find a strong own-lag effect in the last subsection because the variables with high margin magnitudes in the response loading do not coincide with those counterparts in the predictor loading. Second, we use Tensor MGP (without D) to conduct the same experiment as in Section 6.3.

After the inference, we apply Welch's t-test to check whether margins inferred from these two Tensor VAR models are significantly different. Only 4 out of 255 margins cannot reject the null hypothesis that no significant difference between the two posterior samples with a 0.1% significance level. Figure D.6 depicts the posterior mean of the loadings without D . As shown in this figure, the same variable (PAYEMS) is associated with the largest margin magnitudes in the first columns of response and predictor loadings. This pattern holds for the second and third columns as well, with the corresponding variables being M2REAL and BUSLOANS. This finding indicates the additional own-lag matrix D allows the tensor to explore more cross-lag effects. In addition, these large margins in PAYEMS, M2REAL, and BUSLOANS have the potential to distort the coefficients in such a manner that the rows and columns corresponding to these three variables in the coefficient matrix exhibit a higher proportion of large magnitudes compared to their counterparts associated with other variables. Table D.16 gives a detailed analysis in Appendix D.3.

7. Conclusion and Discussion

In this article, we apply the Multiplicative Gamma Prior (MGP) to margins and use an adaptive inferential scheme to infer the rank. To overcome the convergence issue, we introduce an interweaving Gibbs sampler to allow better mixing of Markov chains and match labels and signs after the inference.

The Tensor VAR is closely related to the reduced-rank VAR (Geweke 1996; Carriero, Kapetanios, and Marcellino 2011). A detailed discussion of these two structures is available in the introduction section of Wang et al. (2021). In short, reduced-rank VAR only applies the low-rank assumption to the mode-1 matricization of the tensor, but Tensor VAR makes the same assumption to all three matricizations (model-1, -2, and -3). Following this connection, we find that reduced-rank VAR is a special case of Tensor VAR with the following expression:

$$\mathcal{A} = \sum_{r=1}^R \mathcal{A}^{(r)} = \sum_{r=1}^R \beta_1^{(r)} \circ \mathcal{C}^{(r)},$$

where $\mathcal{C}^{(r)}$ is an N -by- P matrix, and \circ is the outer product of a vector and a matrix such that $\beta_{1,i_1}^{(r)} \mathcal{C}^{(r)}$ equals to $\mathcal{A}_{i_1,\cdot}^{(r)}$, the i_1 th matrix on the first dimension of $\mathcal{A}^{(r)}$, for $i_1 = 1, \dots, N$. If we decompose $\mathcal{C}^{(r)}$ to $\beta_2^{(r)} \circ \beta_3^{(r)}$, then we retain (2.3). We leave the comparison between Tensor VAR and reduced-rank VAR to future work.

Several extensions can also be investigated. First, it will be interesting to adopt time-varying margins and rank to the Tensor VAR. Related work is studied by Zhang et al. (2021), who kept margins time-invariant and switched each column of the tensor matrix B on or off with a prior. Second, we can modify the MGP to include a local parameter corresponding to each row of the loadings to provide more interpretability. Lastly, a similar MCMC scheme can be applied to Tucker decomposition (Tucker 1966), another popular tensor decomposition with a more flexible structure compared to the CP decomposition.

Supplementary Materials

The online supplement contains basic notations, detailed descriptions of the Bayesian inference, supplementary algorithms, and additional results from both simulation study and real data application.

Acknowledgments

The authors thank the associate editor and two anonymous referees for their constructive comments and valuable suggestions, which helped improve this manuscript.

Disclosure Statement

The authors report there are no competing interests to declare.

ORCID

Jim E. Griffin  <http://orcid.org/0000-0002-4828-7368>

References

- Arias, J. E., Rubio-Ramirez, J. F., and Shin, M. (2023), "Macroeconomic Forecasting and Variable Ordering in Multivariate Stochastic Volatility Models," *Journal of Econometrics*, 235, 1054–1086. [10]
- Bañbura, M., Giannone, D., and Reichlin, L. (2010), "Large Bayesian Vector Auto Regressions," *Journal of Applied Econometrics*, 25, 71–92. [1]
- Bernanke, B. S., Boivin, J., and Elias, P. (2005), "Measuring the Effects of Monetary Policy: A Factor-Augmented Vector Autoregressive (FAVAR) Approach," *The Quarterly Journal of Economics*, 120, 387–422. [10]
- Bhattacharya, A., and Dunson, D. B. (2011), "Sparse Bayesian Infinite Factor Models," *Biometrika*, 98, 291–306. [2,3,4,6,7]
- Billio, M., Casarin, R., and Iacopini, M. (2024), "Bayesian Markov-Switching Tensor Regression for Time-Varying Networks," *Journal of the American Statistical Association*, 119, 109–121. [1]
- Billio, M., Casarin, R., Iacopini, M., and Kaufmann, S. (2023), "Bayesian Dynamic Tensor Regression," *Journal of Business & Economic Statistics*, 41, 429–439. [1,4]
- Carriero, A., Clark, T. E., and Marcellino, M. (2019), "Large Bayesian Vector Autoregressions with Stochastic Volatility and Non-conjugate Priors," *Journal of Econometrics*, 212, 137–154. [1,10]
- Carriero, A., Kapetanios, G., and Marcellino, M. (2011), "Forecasting Large Datasets with Bayesian Reduced Rank Multivariate Models," *Journal of Applied Econometrics*, 26, 735–761. [1,13]
- Carvalho, C. M., Polson, N. G., and Scott, J. G. (2009), "Handling Sparsity via the Horseshoe," *Journal of Machine Learning Research W&CP*, 5, 73–80. [10]
- Chan, J. C., Koop, G., and Yu, X. (2024), "Large Order-Invariant Bayesian VARs with Stochastic Volatility," *Journal of Business & Economic Statistics*, 42, 825–837. [10]
- Chen, R., Yang, D., and Zhang, C.-H. (2022), "Factor Models for High-Dimensional Tensor Time Series," *Journal of the American Statistical Association*, 117, 94–116. [1,4]
- Cogley, T., and Sargent, T. J. (2005), "Drifts and Volatilities: Monetary Policies and Outcomes in the Post WWII US," *Review of Economic Dynamics*, 8, 262–302. [2]
- Doan, T., Litterman, R., and Sims, C. (1984), "Forecasting and Conditional Projection Using Realistic Prior Distributions," *Econometric Reviews*, 3, 1–100. [1]
- Durante, D. (2017), "A Note on the Multiplicative Gamma Process," *Statistics & Probability Letters*, 122, 198–204. [3]
- Fan, J., Sitek, K., Chandrasekaran, B., and Sarkar, A. (2022), "Bayesian Tensor Factorized Mixed Effects Vector Autoregressive Processes for Inferring Granger Causality Patterns from High-Dimensional Neuroimaging Data," arXiv preprint arXiv:2206.10757. [1,2,4]
- Gelman, A., and Rubin, D. B. (1992), "Inference from Iterative Simulation Using Multiple Sequences," *Statistical Science*, 7, 457–472. [8]
- Geweke, J. (1991), "Evaluating the Accuracy of Sampling-based Approaches to the Calculation of Posterior Moments," Technical Report, Federal Reserve Bank of Minneapolis. [8]
- (1996), "Bayesian Reduced Rank Regression in Econometrics," *Journal of Econometrics*, 75, 121–146. [13]
- Giannone, D., Lenza, M., and Primiceri, G. E. (2015), "Prior Selection for Vector Autoregressions," *Review of Economics and Statistics*, 97, 436–451. [1,10]
- (2021), "Economic Predictions with Big Data: The Illusion of Sparsity," *Econometrica*, 89, 2409–2437. [1]
- Gruber, L., and Kastner, G. (2022), "Forecasting Macroeconomic Data with Bayesian VARs: Sparse or Dense? It Depends!," arXiv preprint arXiv:2206.04902. [1]
- Guhaniyogi, R., Qamar, S., and Dunson, D. B. (2017), "Bayesian Tensor Regression," *The Journal of Machine Learning Research*, 18, 2733–2763. [2,4,6,7]
- Guhaniyogi, R., and Spencer, D. (2021), "Bayesian Tensor Response Regression with an Application to Brain Activation Studies," *Bayesian Analysis*, 16, 1221–1249. [2]
- Hsu, N.-J., Hung, H.-L., and Chang, Y.-M. (2008), "Subset Selection for Vector Autoregressive Processes Using Lasso," *Computational Statistics & Data Analysis*, 52, 3645–3657. [1]
- Huber, F., and Feldkircher, M. (2019), "Adaptive Shrinkage in Bayesian Vector Autoregressive Models," *Journal of Business & Economic Statistics*, 37, 27–39. [1,4,10]
- Huber, F., Kastner, G., and Feldkircher, M. (2019), "Should I Stay or Should I Go? A Latent Threshold Approach to Large-Scale Mixture Innovation Models," *Journal of Applied Econometrics*, 34, 621–640. [1]
- Kastner, G., and Frühwirth-Schnatter, S. (2014), "Ancillarity-Sufficiency Interweaving Strategy (ASIS) for Boosting MCMC Estimation of Stochastic Volatility Models," *Computational Statistics & Data Analysis*, 76, 408–423. [4]
- Kastner, G., Frühwirth-Schnatter, S., and Lopes, H. F. (2017), "Efficient Bayesian Inference for Multivariate Factor Stochastic Volatility Models," *Journal of Computational and Graphical Statistics*, 26, 905–917. [2,4,8]
- Kiers, H. A. (2000), "Towards a Standardized Notation and Terminology in Multiway Analysis," *Journal of Chemometrics: A Journal of the Chemometrics Society*, 14, 105–122. [1]
- Kolda, T. G., and Bader, B. W. (2009), "Tensor Decompositions and Applications," *SIAM Review*, 51, 455–500. [2,6]
- Korobilis, D., and Pettenuzzo, D. (2019), "Adaptive Hierarchical Priors for High-Dimensional Vector Autoregressions," *Journal of Econometrics*, 212, 241–271. [1,10]
- Legramanti, S., Durante, D., and Dunson, D. B. (2020), "Bayesian Cumulative Shrinkage for Infinite Factorizations," *Biometrika*, 107, 745–752. [6]
- Litterman, R. B. (1986), "Forecasting with Bayesian Vector Autoregressions—Five Years of Experience," *Journal of Business & Economic Statistics*, 4, 25–38. [1]
- McCracken, M., and Ng, S. (2020), "FRED-QD: A Quarterly Database for Macroeconomic Research," working paper, National Bureau of Economic Research. [9]
- Ng, S. (2013), "Variable Selection in Predictive Regressions," *Handbook of Economic Forecasting*, 2, 752–789. [1]
- Poworoznek, E., Ferrari, F., and Dunson, D. (2021), "Efficiently Resolving Rotational Ambiguity in Bayesian Matrix Sampling with Matching," arXiv preprint arXiv:2107.13783. [6]
- Roberts, G. O., and Rosenthal, J. S. (2007), "Coupling and Ergodicity of Adaptive Markov Chain Monte Carlo Algorithms," *Journal of Applied Probability*, 44, 458–475. [6]
- Rousseau, J., and Mengersen, K. (2011), "Asymptotic Behaviour of the Posterior Distribution in Overfitted Mixture Models," *Journal of the Royal Statistical Society, Series B*, 73, 689–710. [2]
- Sims, C. A. (1980), "Macroeconomics and Reality," *Econometrica: Journal of the Econometric Society*, 48, 1–48. [1]
- Stock, J. H., and Watson, M. W. (2005), "Implications of Dynamic Factor Models for VAR Analysis," Working paper, National Bureau of Economic Research Cambridge, MA, USA. [3]

- Tibshirani, R. (1996), “Regression Shrinkage and Selection via the Lasso,” *Journal of the Royal Statistical Society, Series B*, 58, 267–288. [1]
- Tucker, L. R. (1966), “Some Mathematical Notes on Three-Mode Factor Analysis,” *Psychometrika*, 31, 279–311. [13]
- Vats, D., and Knudson, C. (2021), “Revisiting the Gelman–Rubin Diagnostic,” *Statistical Science*, 36, 518–529. [8]
- Wang, D., Zheng, Y., Lian, H., and Li, G. (2021), “High-Dimensional Vector Autoregressive Time Series Modeling via Tensor Decomposition,” *Journal of the American Statistical Association*, 117, 1338–1356. [1,2,3,4,13]
- Yu, Y., and Meng, X.-L. (2011), “To Center or Not to Center: That is Not the Question—An Ancillarity–Sufficiency Interweaving Strategy (ASIS) for Boosting MCMC Efficiency,” *Journal of Computational and Graphical Statistics*, 20, 531–570. [2]
- Zhang, W., Cribben, I., Guindani, M., and Petrone, S. (2021), “Bayesian Time-Varying Tensor Vector Autoregressive Models for Dynamic Effective Connectivity,” arXiv preprint arXiv:2106.14083. [1,4,13]
- Zhou, H., Li, L., and Zhu, H. (2013), “Tensor Regression with Applications in Neuroimaging Data Analysis,” *Journal of the American Statistical Association*, 108, 540–552. [2,6]