Multivariate Stochastic Volatility Model with Block Correlations

Han Chen, Yijie Fei*
College of Finance and Statistics, Hunan University
Jun Yu
Faculty of Business Administration, University of Macau

March 3, 2026

Abstract

Modeling the dynamics of correlations of multiple time series is an important yet difficult task, especially when the dimension is not confined to be low. In this paper, we propose a new multivariate stochastic volatility model featuring a block correlation structure. Our specification is built upon the new parametrization of the correlation matrix of Archakov & Hansen (2021) and extends the MSV-GFT model introduced in Chen et al. (2025). A Particle Gibbs Ancestor Sampling (PGAS) method is proposed to conduct the Bayesian analysis. It is shown to perform well for our model in finite samples. An empirical application based on a dozen U.S. stocks shows that our new model outperforms alternative specifications in terms of both the in-sample performance and the out-of-sample performance.

Keywords: Block correlation matrix; Generalized Fisher transformation; Markov chain Monte Carlo; Multivariate stochastic volatility; Particle filter;

*Email address: Han Chen, (hanchen@hnu.edu.cn), Yijie Fei (yijiefei@hnu.edu.cn) and Jun Yu (junyu@um.edu.mo). Yijie Fei acknowledges the financial support from Natural Science Foundation of Hunan Province (2025JJ60440) and National Natural Science Foundation of China (72501103).

1 Introduction

In this paper, we are concerned with modeling the dynamics of correlations in a multivariate stochastic volatility (MSV) model. Modeling correlations of multiple time series is an important task in financial econometrics. It is crucial for asset pricing, portfolio selection, risk management, and regulation. Yet, the correlation matrix is notoriously difficult to model when the dimension is high or modestly high, due to at least two reasons. First, the positive-definiteness constraint must be fulfilled, which is highly nonlinear and complicated. Second, one must strike a balance between model flexibility and computational feasibility. Such challenges have been widely explored in the MSV literature; see, for instance, the comprehensive review by Asai et al. (2006).

The MSV model with an explicit characterization of a correlation structure can be dated back to the MSV model with constant correlations in Harvey et al. (1994). A similar assumption was made in Chan et al. (2006), Asai & McAleer (2006), and Ishihara & Omori (2012). Although the assumption of constant correlations makes statistical inference simple, it is too restrictive for financial time series. To allow for time-varying correlations, Asai & McAleer (2009) consider two models, both motivated by the dynamic conditional correlation (DCC) model of Engle (2002).

Another strand of the MSV literature is built upon the Fisher z-transformation. Yu & Meyer (2006) use the Fisher z-transformation in the bivariate case. Extending their bivariate model to a higher-dimensional case is non-trivial. Several attempts have been made. For example, inspired by the dynamic equicorrelation (DECO) model of Engle & Kelly (2012), Kurose & Omori (2016) propose a stochastic volatility model with equicorrelation. The assumption, however, is obviously too restrictive and oversimplified when the number of assets is large. Kurose & Omori (2020) extend this specification to the multiple-block case and provide an easy-to-verify condition for the positivity of the correlation matrix.[1]

[1]It is a special case of the general positive-definiteness condition for the block matrix derived in Archakov & Hansen (2024).

Yamauchi & Omori (2020), on the other hand, propose to model the pairwise correlations by the Fisher z-transformation. As the resulting correlation matrix is in general not positive definite, they further derive algebraic bounds for all correlations that ensure the positive-definiteness of the correlation matrix. However, the bound for a particular correlation is conditional on all remaining correlations, and therefore, estimation methods for this model are severely limited. More recently, Chen et al. (2025) propose a highly flexible MSV model using Archakov & Hansen (2021)'s new parameterization for the correlation matrix. It is a natural and valid generalization of the Fisher z-transformation from the bivariate case to the high-dimensional case. See Section 2.1 below for a review of this parametrization, which is termed the generalized Fisher transformation (GFT). For other recent applications of GFT, see Bucci et al. (2022), Arias et al. (2023), Hafner & Wang (2023) and Archakov et al. (2024).

All the models mentioned above are designed mainly for low-dimensional cases. In practice, when the number of assets is large or modestly large, the computational cost of an unrestricted MSV model with dynamic correlations will be formidable. In such scenarios, the dimension reduction strategy is indispensable. Moreover, in practice, a completely unrestricted correlation matrix is sometimes not necessary, as there may be reasons why some correlation coefficients are related to others. An attractive method for achieving parsimony is to impose a block structure.[2] This strategy has recently gained increasing attention. To be specific, an $n \times n$ block correlation matrix with a block partition $n_1, \cdots, n_K$ and

[2]Dimension reduction can be also achieved by imposing a factor structure or sparsity. For applications in the context of MSV, see for example Poignard & Asai (2024) and Poignard & Asai (2023). Note, however, that both approaches are usually designed for modeling the covariance matrix rather than the correlation matrix.

$n_1 + \cdots + n_K = n$ is defined as

$$C^B = \begin{bmatrix} C^B_{[1,1]} & C^B_{[1,2]} & \cdots & C^B_{[1,K]} \\ C^B_{[2,1]} & C^B_{[2,2]} & & \\ \vdots & & \ddots & \\ C^B_{[K,1]} & & & C^B_{[K,K]} \end{bmatrix}, \tag{1}$$

where $C^B_{[k,l]}$ is an $n_k \times n_l$ matrix with the following structure:

$$C^B_{[k,k]} = \begin{bmatrix} 1 & \rho_{kk} & \cdots & \rho_{kk} \\ \rho_{kk} & 1 & \ddots & \\ \vdots & \ddots & \ddots & \\ \rho_{kk} & \cdots & \cdots & 1 \end{bmatrix} \text{ and } C^B_{[k,l]} = \begin{bmatrix} \rho_{kl} & \cdots & \rho_{kl} \\ \vdots & \ddots & \vdots \\ \rho_{kl} & \cdots & \rho_{kl} \end{bmatrix} \text{ if } k \neq l.$$

As pointed out in Archakov & Hansen (2024), when such a block structure is reasonable, estimation and manipulation of large correlation matrices can be greatly simplified. Furthermore, the positivity constraint can be easily handled by virtue of the matrix logarithm transformation, which has been shown to preserve the block structure. Built upon this specification, a realized multivariate GARCH model and a score-driven dynamic block correlation model have recently been proposed in Archakov et al. (2025) and Tong et al. (2025), respectively.

Motivated by the above-mentioned progress in the literature, in the present paper, we propose a new MSV model featuring a block correlation structure. The new model is an important extension of the MSV-GFT model proposed in Chen et al. (2025). In particular, we first propose a novel parametrization for block correlation matrices, which is flexible in the sense that the positivity constraint is automatically fulfilled and the block structure is always preserved. Next, we propose a Bayesian inference procedure for our new MSV model, which relies on a Markov Chain Monte Carlo (MCMC) algorithm based on particle filtering.

As the estimation procedure depends heavily on an inverse transformation that does not have a closed-form solution, we discuss in detail how to handle this step in a feasible and efficient way. The finite-sample performance of the inference for our model is checked through a comprehensive simulation study. The empirical advantages of our model are illustrated via a detailed application to U.S. stock return data.

The rest of the paper is organized as follows. Section 2 introduces a novel parametrization of a block correlation matrix and presents our new MSV model. Section 3 introduces the estimation and inferential method based on the PGAS algorithm. Section 4 reports the simulation evidence to support our proposed Bayesian method. Empirical studies are provided in Section 5 to support the proposed model. Section 6 concludes. The Appendix includes additional details and materials that complement and support our main text.

Throughout the paper, we let $diag(A)$ denote the column vector formed by the diagonal elements of a square matrix A or the diagonal matrix whose diagonal elements are elements in A if A is a column vector;[3] $vech(A)$ denote the $n(n+1)/2 \times 1$ column-vector obtained by vectorizing only the lower triangular part of a n-dimensional matrix A (including the diagonal elements); $vecl(A)$ denote the $n(n-1)/2 \times 1$ column-vector containing all lower off-diagonal elements of A (excluding the diagonal elements); $A \otimes B$ denote the Kronecker product of matrix A and B; I_n denote a n-dimensional identity matrix. To avoid confusion, the matrix logarithm defined in (2) will be represented by $\text{logm}(\cdot)$, while the conventional logarithm for positive number are denoted by $\log(\cdot)$; likewise, $\text{expm}(\cdot)$ and $\exp(\cdot)$ will represent matrix and scalar exponential, respectively.

[3] If A is a square matrix, $diag(diag(A))$ is a diagonal matrix whose diagonal elements are the diagonal elements in A.

2 Model

2.1 Generalized Fisher transformation

The specification of our new model relies heavily on the parameterization of a correlation matrix proposed in Archakov & Hansen (2021). Before introducing the new MSV model, we briefly review the GFT technique of Archakov & Hansen (2021).

When the correlation coefficient between two random variables, say ρ, is to be modeled, an essential constraint is that its value must lie in the interval $[-1, 1]$. To avoid the complexity introduced by this constraint, one can instead model the Fisher z-transformation of ρ:

$$g = \frac{1}{2} \log \frac{1+\rho}{1-\rho}.$$

Archakov & Hansen (2021) propose a valid high-dimensional extension to the Fisher z-transformation. To fix the idea, let C be a valid n-dimensional correlation matrix and

$$\operatorname{logm} C = \sum_{k=1}^{\infty} \frac{(-1)^k (C-I)^k}{k} \tag{2}$$

be its (matrix) logarithm. Note that the convergence of the infinite summation and hence, the existence and uniqueness of $\operatorname{logm} C$ are ensured by the fact that C is a correlation matrix. GFT of C is then defined by the mapping

$$q = vecl(\operatorname{logm} C). \tag{3}$$

One of the key theoretical contributions of Archakov & Hansen (2021) is to demonstrate that this mapping is bijective. Thus, given any $\frac{n(n-1)}{2}$-dimensional vector q, there exists a unique and valid n-dimensional correlation matrix C. Although the inverse mapping from q to C does not have a closed-form expression when $n > 2$, C can be obtained numerically

using an iterative algorithm.

The new transformation retains the advantages of the Fisher z-transformation and enjoys some additional desirable properties. First and foremost, it is very flexible in the sense that, when modeling q, no algebraic constraint is needed. This suggests that we can consider any reasonable dynamics for q without worrying about the positive-definiteness of the resulting correlation matrix. Second, compared with the original elements in R, the elements in q is often much closer to Gaussian due to the use of the log transformation. Hence, it is reasonable to model the elements of q via a Gaussian process. Third, the GFT transformation is invariant to the order of the variables. This is in sharp contrast to the Cholesky decomposition.

2.2 Parametrization of block correlation matrices

As the block correlation matrix is a special case of the general correlation matrix, we may also model the general correlation matrix using GFT defined by (3). However, when the number of assets increases, the number of variables to be modeled, after applying GFT, grows quadratically. This renders the inference infeasible. Fortunately, utilizing the presence of a block structure, we can reduce the dimension of our model so that it grows only with the number of groups. A critical property we leverage to construct our MSV model is that the matrix log transformation can preserve the block structure, as shown in Archakov & Hansen (2024). This allows us to model the off-diagonal elements of the log-correlation matrix,

namely $vecl\left(\operatorname{logm} C^B\right)$, whose range spans the space of $\frac{K(K+1)}{2} \times 1$ real vectors. Let[4]

$$\operatorname{logm} C^B = \begin{bmatrix} Q_{[1,1]} & Q_{[1,2]} & \cdots & Q_{[1,K]} \\ Q_{[2,1]} & Q_{[2,2]} & & \\ \vdots & & \ddots & \\ Q_{[K,1]} & & & Q_{[K,K]} \end{bmatrix}, \tag{4}$$

with

$$Q_{[k,k]} = \begin{bmatrix} y_{kk} & q_{kk} & \cdots & q_{kk} \\ q_{kk} & y_{kk} & \ddots & \\ \vdots & \ddots & \ddots & \\ q_{kk} & & & y_{kk} \end{bmatrix} \text{ and } Q_{[k,l]} = \begin{bmatrix} q_{kl} & \cdots & q_{kl} \\ \vdots & \ddots & \\ q_{kl} & & q_{kl} \end{bmatrix} \text{ if } k \neq l.$$

Define the n-dimensional group assignment vector by $s^B = \left[s_1^B, ..., s_n^B\right]$, where $s_i^B \in \{1, \ldots, K\}$ indicates the group membership of the i^{th} variable. If $s_i^B = k$, then r_i belongs to the k-th group. The corresponding $K \times n$ group matrix $S^B\left(s^B\right) = \left[S_1^B, \ldots, S_n^B\right]$ is defined by $S^B[i,j] = 1$ if $s^B[j] = i$ and 0 otherwise.

Example 1. Consider the following block correlation matrix, which consists of $n = 5$ variables belonging to $K = 2$ groups:

$$C^B = \left[\begin{array}{cc:ccc} 1 & 0.6 & 0.1 & 0.1 & 0.1 \\ 0.6 & 1 & 0.1 & 0.1 & 0.1 \\ \hdashline 0.1 & 0.1 & 1 & 0.4 & 0.4 \\ 0.1 & 0.1 & 0.4 & 1 & 0.4 \\ 0.1 & 0.1 & 0.4 & 0.4 & 1 \end{array}\right]. \tag{5}$$

[4]Note that we assume the variables are well ordered so that those belonging to the same (predetermined) group are placed next to each other.

Then we have $s^B = (1, 1, 2, 2, 2)$ and, correspondingly,

$$S^B = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}. \tag{6}$$

Meanwhile, we also have

$$\operatorname{logm} C^B = \left[\begin{array}{rrrrr} -0.229 & 0.688 & 0.059 & 0.059 & 0.059 \\ 0.688 & -0.229 & 0.059 & 0.059 & 0.059 \\ \hdashline 0.059 & 0.059 & -0.148 & 0.363 & 0.363 \\ 0.059 & 0.059 & 0.363 & -0.148 & 0.363 \\ 0.059 & 0.059 & 0.363 & 0.363 & -0.148 \end{array}\right],$$

which apparently shares the same block structure as C^B and thereby the same S^B and s^B.

The following proposition shows that GFT of a block correlation matrix can be factored into two components, with the first one being determined by the group assignments and the second one collecting the unique off-diagonal elements. In the following, let E_l and G_l be the elimination matrix and the duplication matrix defined by $E_l vec(M) = vecl(M)$ and $G_l vech(M) = vec(M)$ for any l-dimensional symmetric square matrix M.

Proposition 1. *Let* $\operatorname{logm} C^B$ *admits a block structure as in (4). Then, we have*

$$vecl(\operatorname{logm} C^B) = E_n \left(S^B \otimes S^B\right)' G_K q, \tag{7}$$

where $q = vech(Q)$ *and* $Q = \{q_{kl}\}_{k,l=1}^K \in \mathbb{R}^{K \times K}$.

The proof of Proposition 1 can be found in Section A of the Online Supplement. Transformation (7) is called B-GFT in the rest of the paper. As shown in Archakov et al. (2024), given any block structure captured by S^B, B-GFT (7) defines a one-to-one correspondence

between any $\frac{K(K+1)}{2} \times 1$ vector q and a valid $n \times n$ (block) correlation matrix C^B.[5] Notably, no algebraic constraint is necessary for vector q, so that it can vary freely in the space $\mathbb{R}^{\frac{K(K+1)}{2}}$. We emphasize that although there is a one-to-one correspondence between q and C^B, there exists no clear relationship between q_{kl} and ρ_{kl}, due to the highly nonlinear matrix logarithm transformation.

Example 1 (continued)**.** For the block correlation matrix C^B in (5), we have

$$vecl(\operatorname{logm} C^B) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}' \begin{bmatrix} 0.688 \\ 0.059 \\ 0.363 \end{bmatrix},$$

where the bit matrix can be constructed by S^B defined in (6) using formula (7).

2.3 Specification of the new model

Let $d = K(K+1)/2$. Our basic MSV model with n variables and K groups is given by[6]

$$r_t = H_t^{1/2} \epsilon_t, \ \epsilon_t \sim N\left(0, C_t^B\right), \tag{8a}$$

$$H_t = \exp\left(diag(h_t)\right), \tag{8b}$$

$$vecl\left(\operatorname{logm} C_t^B\right) = E_n \left(S^B \otimes S^B\right)' G_K q_t, \tag{8c}$$

$$h_{t+1} = \mu_h + \Phi_h(h_t - \mu_h) + \eta_{ht}, \ \eta_{ht} \sim N\left(0, \Sigma_h\right), \tag{8d}$$

$$q_{t+1} = \mu_q + \Phi_q(q_t - \mu_q) + \eta_{qt}, \ \eta_{qt} \sim N\left(0, \Sigma_q\right), \tag{8e}$$

[5] In the practical applications, it is possible that some groups contain only a single variable and hence, there is no within-group correlation for these groups. As a result, to be precise, the exact number of the unique correlation is in fact

$$K(K-1)/2 + \sum_{k=1}^{K} I(n_k \geqslant 2),$$

where $I(\cdot)$ is the indicator function.

[6] We assume that the group membership is predetermined and fixed during the whole sample period. A data-driven estimation of group assignments, as considered in Oh & Patton (2023), is beyond the scope of the present paper.

$$h_0 \sim N\left(\mu_h, \left(I_n - \Phi_h^2\right)^{-1}\Sigma_h\right),\ q_0 \sim N\left(\mu_q, \left(I_d - \Phi_q^2\right)^{-1}\Sigma_q\right), \tag{8f}$$

where $\epsilon_t = (\epsilon_{1t}, ..., \epsilon_{nt})'$, $\eta_{ht} = (\eta_{h1t}, ..., \eta_{hnt})'$, $\eta_{qt} = (\eta_{q1t}, ..., \eta_{qdt})'$, $\mu_h = (\mu_{h1}, ..., \mu_{hn})'$, $\mu_q = (\mu_{q1}, ..., \mu_{qd})'$, $\Phi_h = diag\left((\phi_{h1}, ..., \phi_{hn})'\right)$, $\Phi_q = diag\left((\phi_{q1}, ..., \phi_{qd})'\right)$, and $t = 1, ..., T$. It is assumed that ϵ_t, η_{ht} and η_{qt} are independent. This implies that no leverage (neither self-leverage nor cross-leverage) effect is allowed. It also implies that the shocks to the volatility dynamics (i.e. η_{ht}) are completely separated from those to the correlation dynamics (i.e. η_{qt}). To reduce the number of parameters, we further assume that $\Sigma_h = diag\left((\sigma_{h1}^2, ..., \sigma_{hn}^2)'\right)$ and $\Sigma_q = diag\left((\sigma_{q1}^2, ..., \sigma_{qd}^2)'\right)$. This model is referred to as MSV-B-GFT in the following.[7]

The property that no algebraic restriction on q_t is required in B-GFT is particularly convenient as far as the inference on the static parameters driving q_t is concerned. To compare with the existing method, consider the specification proposed in Kurose & Omori (2020), which also starts from the assumption of multiple-block correlation matrices. Kurose & Omori (2020) do not use the matrix logarithm transformation as they work directly with the correlation matrix C^B. Their specification can be summarized, using our notation, as

$$vecl(C_t^B) = E_n\left(S^B \otimes S^B\right)' G_K \rho_t,$$

where ρ_t is a vector collecting the unique correlations in C_t^B. They propose to apply the element-wise Fisher transformation, which only ensures that each ρ_t is within $[-1, 1]$. To further achieve the positive-definiteness of C_t^B, they derive an easy-to-verify constraint on the elements in ρ_t. When constructing their MCMC sampler, they discard all the samples fail to fulfill the constraint. Such a strategy is apparently inefficient, particularly when the number of variables is large, in which case an overwhelming proportion of samples must be wasted. Our B-GFT transformation is immune to such a problem and therefore, is a

[7] When the realized measures are available, our model can be further extended to take these additional information into account. See the Section B of Online Supplement for details.

preferable way to model the multiple-block correlation matrices.[8]

3 Estimation

3.1 Bayesian inference of MSV-B-GFT

We now present the Bayesian analysis of our MSV-GFT model. The first step is to specify the prior distributions of all the parameters $\theta = \left(\mu_h, \mu_q, \phi_h, \phi_q, \sigma_h^2, \sigma_q^2\right)'$. In this regard, we Kim et al. (1998). For μ_h and μ_q, we assume independent multivariate normal distributions. Foe the persistence parameters ϕ_h and ϕ_q we assume Beta priors. The prior distribution of σ_h and σ_q are chosen to be inverse gamma. In summary, for $i = 1, ..., n$ and $j = 1, ..., d$, we choose the following prior distributions:

- $\mu_{hi} \sim N(m_{\mu 0}, s_{\mu 0}^2)$ and $\mu_{qj} \sim N(m_{\mu 0}, s_{\mu 0}^2)$;
- $\frac{\phi_{hi}+1}{2} \sim Beta(a, b)$ and $\frac{\phi_{qj}+1}{2} \sim Beta(a, b)$;
- $\sigma_{hi}^2 \sim IG(\frac{n_{m0}}{2}, \frac{d_{m0}}{2})$ and $\sigma_{qj}^2 \sim IG(\frac{n_{m0}}{2}, \frac{d_{m0}}{2})$,

where $m_{\mu 0}, s_{\mu 0}^2, a, b, n_{m0}, d_{m0}$ are hyperparameters.[9]

To carry out the Bayesian inference, we implement a Gibbs sampler with four blocks. In the following, we use $\theta_{/\alpha}$ to denote the parameters θ excluding α. The algorithm proceeds as:

1. Initialize h, q and θ.
2. Draw $h, q|r, \theta$.
3. Draw $\mu_h, \mu_q|r, h, q, \theta_{/(\mu_h, \mu_q)}$.
4. Draw $\phi_h, \phi_q|r, h, q, \theta_{/(\phi_h, \phi_q)}$.

[8] Admittedly, this efficiency regarding MCMC samples of our method comes with a cost. Unlike the pairwise Fisher transformation, the likelihood evaluation must be conducted numerically and thereby time-consuming in our approach; see Section 3.2 below.

[9] For the implementation of the algorithm, the choices of the hyperparameters follow those in Kim et al. (1998).

5. Draw $\sigma_h^2, \sigma_q^2 | r, h, q, \theta_{/(\sigma_h^2, \sigma_q^2)}$.

Iteration over steps 2-5 consists of a complete sweep of MCMC sampler. Here, the most challenging step is to sample the latent variables h and q given all the observations r and one particular set of parameter values. A conventional method to tackle this problem is the so-called single-move sampler, which is notoriously inefficient. See Yu & Meyer (2006) and Yamauchi & Omori (2020) for applications of the single-move approach to estimate MSV models with dynamic correlations. To improve efficiency, a few papers resort to the multi-move algorithm, which is built on the early work by Watanabe & Omori (2004). See Ishihara & Omori (2012), Ishihara et al. (2016) and Kurose & Omori (2016) for the applications of the multi-move approach to estimation of MSV models. In this paper, following Chen et al. (2025), we apply the Particle Gibbs Ancestor Sampling (PGAS) introduced in Lindsten et al. (2014) to draw the latent variables; see Section C and D of the Online Supplement for a brief review of the Particle Gibbs (PG) approach and a detailed description of the PGAS algorithm. On the other hand, from the joint posterior density, it is relatively straightforward to sample each element in θ given one realization of the latent processes h and q. The details are provided in Section E of the Online Supplement.

3.2 Efficient Inversion of B-GFT

The Bayesian inference proposed above requires evaluating the Gaussian likelihood. To simply the discussion, we omit the time index t and consider the likelihood for an arbitrary period. In particular, suppose $\epsilon := H^{-1/2} r \sim N\left(0, C^B\right)$, the log-likelihood of ϵ multiplied by -2 is given by $-2l = n \log(2\pi) + \log\left|C^B\right| + \epsilon' \left(C^B\right)^{-1} \epsilon$. Therefore, a key step towards the evaluation of the likelihood is to obtain the block correlation matrix C^B from q. This is essentially the inversion of the B-GFT operation defined in (7). A natural way to proceed is to first obtain the lower off-diagonal elements of $\operatorname{logm} C^B$ via (7), and then to conduct the inverse GFT using the algorithm considered in Archakov & Hansen (2021) or Chen et al.

(2025). This procedure, however, can be rather slow, as it requires the inverse GFT of a large n-dimensional correlation matrix, which often involves a large number of n-dimensional matrix exponential operations. Since the latter is of order $O(n^3)$ in terms of time complexity, it is expected to be computationally quite expensive and even infeasible.

As the likelihood evaluation must be conducted numerous times during the Bayesian inference introduced in the last subsection, it is of importance to employ an efficient algorithm. To be specific, it is desirable that the computational cost depends only on the number of groups K and independent of the total number of variables n. Fortunately, this can be achieved by resorting to the canonical representation of block matrices proposed in Archakov & Hansen (2024). In particular, according to Theorem 1 in Archakov & Hansen (2024), a block correlation matrix C^B as defined in (1) admits the following canonical representation

$$C^B = VDV' \text{ with } \underset{n\times n}{D} = \begin{bmatrix} A & 0 & \cdots & 0 \\ 0 & \lambda_1 I_{n_1-1} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_K I_{n_K-1} \end{bmatrix} := \begin{bmatrix} A & 0 \\ 0 & \Lambda \end{bmatrix}, \tag{9}$$

where $\lambda_k = 1 - \rho_{kk}$ and $A = \{a_{kl}\}_{k,l=1}^K \in \mathbb{R}^{K\times K}$, whose elements are

$$a_{kl} = \begin{cases} \rho_{kl}\sqrt{n_k n_l} & \text{if } k \neq l \\ 1 + (n_k - 1)\rho_{kk} & \text{if } k = l \end{cases}$$

The orthogonal matrix V in the above decomposition is

$$\underset{n\times n}{V} = \left[\begin{array}{cccc:cccc} v_{n_1} & 0 & \cdots & 0 & v_{n_1\perp} & 0 & \cdots & 0 \\ 0 & v_{n_2} & & \vdots & 0 & v_{n_2\perp} & & \vdots \\ \vdots & & \ddots & & & & \ddots & \\ 0 & \cdots & & v_{n_K} & 0 & \cdots & & v_{n_K\perp} \end{array}\right] = \left[\begin{array}{c:c} V_0 & V_\perp \end{array}\right]$$

where $n_k \times 1$ vector $v_{n_k} = \left(\frac{1}{\sqrt{n_k}}, \cdots, \frac{1}{\sqrt{n_k}}\right)'$ is fully determined by the group structure and $v_{n_k\perp}$ can be obtained by applying the Gram–Schmidt process; see Section 2 in the online appendix of Creal & Kim (2024) for explicit expressions of $V_\perp$.

Based on the above representation, Archakov et al. (2024) (AHL hereafter) discuss how to recover the determinant and the inverse of the original correlation matrix efficiently. Specifically, they propose to use the following algorithm to calculate A and Λ in the representation (9) given a particular realization of q, which in turn leads to a simplified calculation of the log-likelihood as shown in Archakov & Hansen (2024).

1. Given q, construct a $K \times K$ matrix $\tilde{A}$ using

$$\tilde{A}_{k,l} = \begin{cases} q_{kk}(n_k - 1) & \text{for } k = l \\ q_{kl}\sqrt{n_k n_l} & \text{for } k \neq l \end{cases},$$

   where q_{kl} are elements of Q as defined by the identity $q = vech(Q)$.

2. Let $[M]_{kk}$ denote the k^{th} diagonal element of matrix M and $y = (y_{11}, \cdots, y_{KK})'$. From an arbitrary starting value $y^{(0)} \in \mathbb{R}^K$, evaluate the following recursion repeatedly

$$y_{kk}^{(j+1)} = y_{kk}^{(j)} + \log n_k - \log\left(\left[\text{expm}\left\{\tilde{A} + diag\left(y^{(j)}\right)\right\}\right]_{kk} + (n_k - 1)\exp\left(y_{kk}^{(j)} - q_{kk}\right)\right) \tag{10}$$

   until the convergence condition $\left|\left|y^{(J+1)} - y^{(J)}\right|\right| < \varepsilon$ is satisfied for some J.

3. Denote the final value in Step 2 by y^*. Compute $A = \text{expm}\left(\tilde{A} + diag(y^*)\right)$ and obtain the elements of Λ by $\lambda_k = (n_k - A_{kk}) / (n_k - 1)$.

4. Compute the negative log-likelihood multiplied by 2 using[10]

$$-2l = n\log(2\pi) + \log|A| + tr\left\{A^{-1}\epsilon_0\epsilon_0'\right\} + \left[\sum_{k=1}^{K}(n_k - 1)\left(\log\lambda_k + \frac{1}{\lambda_k}\frac{\epsilon_k'\epsilon_k}{n_k - 1}\right)\right],$$

[10] See Corollary 1 of Archakov & Hansen (2024).

where $\{\epsilon_k\}_{k=0}^K$ are defined by $[\epsilon_0', \epsilon_1', \cdots, \epsilon_K']' = V'\epsilon$.

Note that the complexity of this algorithm is determined completely by the number of groups K rather than the number of variables n. Crucially, no n-dimensional matrix exponential operation is involved, which is often quite time-consuming. Nevertheless, for our purpose, this algorithm still needs to be improved. This is due to the fact that it will be conducted billions of times when our particle-filter-based MCMC estimation is implemented.

Observe that the most expensive step in the AHL algorithm above is in the fixed-point iteration (10), which solves a highly nonlinear system of equations. As is well known in numerical analysis, such a problem can be solved more efficiently in many cases by taking into account of the information from Jacobian, if it is available in an analytical form and cheap to compute. Based on this idea, Chen et al. (2025) develop an improved algorithm to invert the original GFT. The key tool they use is a quasi-Newton approach called Broyden's method, which computes the Jacobian matrix only in the first iteration and then perform rank-one updates in subsequent iterations. Via extensive simulation studies, Chen et al. (2025) show that the Broyden's method exhibits superior efficacy by striking a good balance between the number of iterations and the cost per iteration.

In this paper, following Chen et al. (2025), we also use Broyden's method to accelerate the recursion in (10). A flowchart for the implementation of Broyden's method in the current context is provided in Algorithm 1, with the Jacobian matrix $\mathcal{J}$ used given in Section F of the Online Supplement.[11] To verify the performance of Broyden's method relative to AHL's, we conduct a simulation exercise based on randomly generated block correlation matrices. In particular, we assume that each group contains 10 variables, and the number of groups is $K \in \{5, 10, 15, 20\}$. Consequently, the dimension of matrices we consider is $n \in \{50, 100, 150, 200\}$. Similar to Chen et al. (2025), we generate 100,000 distinct block correlation matrices and categorize them into different groups based on the logarithm of

[11] For practical implementation, we set the maximum number of iterations J to 1000 and choose the convergence criterion to be $\varepsilon = 1 \times 10^{-6}$.

minimum eigenvalue, denoted by $\log(\lambda_{\min})$. The range of $\log(\lambda_{\min})$ is set to $[-30, 0]$, which is partitioned into 15 groups. For each group, we calculate the average iteration counts as well as the CPU time for all matrices belonging to that group.

Algorithm 1: Broyden's method for (10)

Data: q ; // $\frac{K(K+1)}{2} \times 1$ vector
Result: y ; // $K \times 1$ vector
Set initial Value: $y^{(0)} = -f(\mathbf{0}_K)$ and $f\left(y^{(0)}\right) = \inf$
for j = 0 : J **do**
  if $||f(y^{(j)})||_2 < \epsilon$ **then**
    Return y
  else
    if k = 0 **then**
      Compute Jacobian: $\mathcal{J}_{(0)} = \mathcal{J}(y^{(0)})$; // Complexity: $O(K^4)$
    else
      Update Jacobian:
      $\Delta f\left(y^{(j)}\right) = f\left(y^{(j)}\right) - f\left(y^{(j-1)}\right)$; // Complexity: $O(K^3)$
      $\Delta y^{(j)} = y^{(j)} - y^{(j-1)}$
      $\mathcal{J}_{(k)} = \mathcal{J}_{(k-1)} + \frac{\Delta f\left(y^{(k)}\right) - J_{(j-1)}\Delta y^{(j)}}{||\Delta y^{(j)}||_2}\Delta y^{(j)\prime}$
      Update y: $y^{(j+1)} = y^{(j)} - \mathcal{J}_{(j)}^{-1} f\left(y^{(k)}\right)$; // Complexity: $O(K^3)$

The results are presented in Figure 1. The subplots in the first row report the number of iterations before convergence. It is obvious that Broyden's method is much less sensitive to the minimum eigenvalue and requires much fewer recursions compared with AHL's method when the correlation matrix is close to be singular. For example, when $K = 5$ and $\log(\lambda_{\min}) \approx -30$, we observe that Broyden's method converges within 20 iterations, while the count for AHL's algorithm is almost sevenfold. As a lower iteration count does not necessarily translate to an overall reduction in computational cost, the subplots in the second row of Figure 1 further depict the CPU time taken by the two methods against minimum eigenvalue. Unsurprisingly, the expenses of both approaches rise up when the number of groups increases. Furthermore, when $\log(\lambda_{\min})$ is close to zero and hence the correlation matrix is less singular, two methods perform approximately the same. However, they differ a lot when

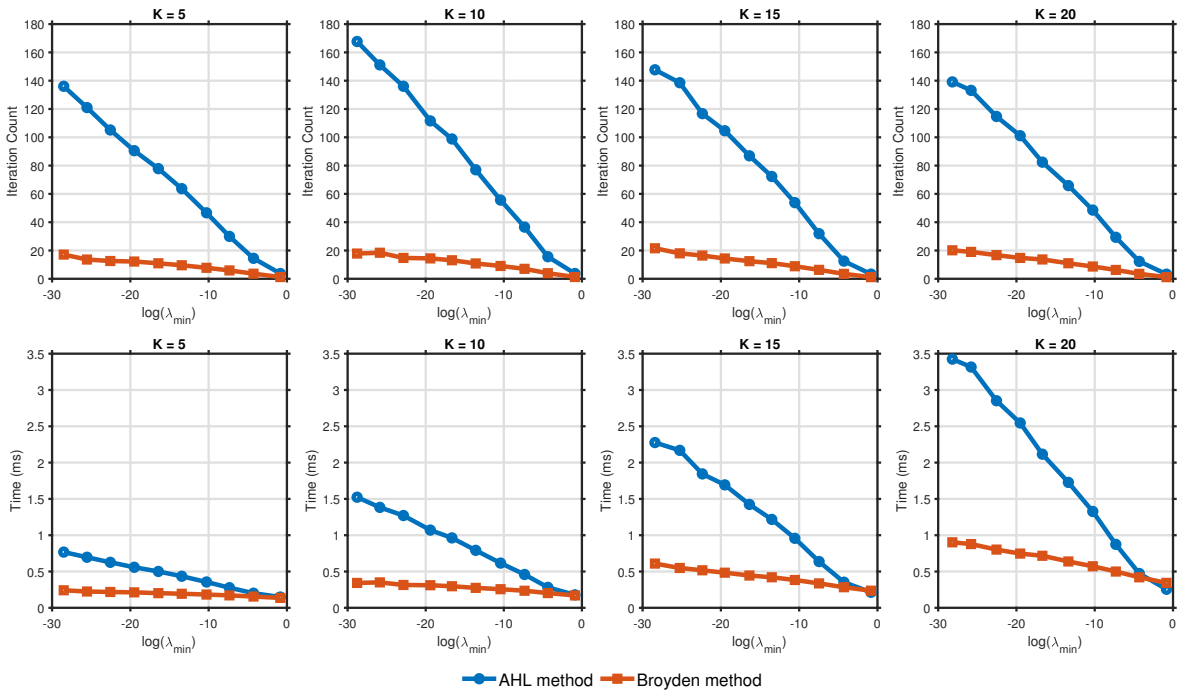

Figure 1: Comparison of AHL's method and Broyden's method for inverting B-GFT

$\log(\lambda_{\min})$ becomes more negative, with Broyden's method spends much less time and thus substantially more efficient than AHL's method. Provided that the minimum eigenvalues of block correlation matrices encountered in a typical empirical analysis are usually quite close to zero, the evidence from our exercise strongly favors the employment of Broyden's method when implementing the MCMC estimation.

4 Simulation Studies

To investigate the performance of the proposed inference procedure, we conduct some simulations in this section. The design of our experiment is frequentist in nature, as we fix the parameters at their true values and generate data from the same data generating process with 1000 replications. We use the posterior mean as a point estimator for all the parameters. Since the true values are known, we can calculate the bias (defined as the difference between the true values and the average value of the posterior means) as well as the standard

deviation.

For the purpose of evaluating the sampling efficiency of the PGAS algorithm, following Kim et al. (1998), we calculate the average inefficiency factor (IF), which is defined as the variance of the sample mean from MCMC sampling divided by that from a hypothetically independent sampler. The variance of the MCMC sample mean is the square of the numerical standard error estimated by

$$NSE = 1 + \frac{2B_M}{B_M - 1} \sum_{i=1}^{B_M} K\left(\frac{i}{B_M}\right) \hat{\rho}(i),$$

where $\hat{\rho}(i)$ is the estimated autocorrelation at lag i, B_M is the bandwidth, and $K(\cdot)$ is the Parzen kernel. We choose the bandwidth B_M to be 1000. A smaller IF indicates better mixing of a Markov chain and thereby higher sampling efficiency.

Our data generating process is the basic MSV-B-GFT model with $n = 18$. The number of groups is $K = 3$ and each group contains 6 variables. Since our main interest lies in the modeling of dynamic correlations, we assume that all variables have been standardized and hence, no estimation of volatility dynamics is necessary. There are 18 parameters to be estimated in this simplified model, whose true values are given in Table 1.

	q_1	q_2	q_3	q_4	q_5	q_6
μ_q	0.376	0.047	0.017	0.440	0.041	0.531
ϕ_q	0.700	0.700	0.700	0.700	0.700	0.700
σ_q^2	0.050	0.050	0.050	0.050	0.050	0.050

Table 1: Parameter setting used in the simulation study

All the simulation results reported in this section are based on 5000 MCMC iterations, among which the first 1000 samples are discarded as burn-in period. Examination of the autocorrelation function suggests that MCMC well converges after 1000 iterations. We consider three different sample sizes, namely $T = 500, 1000, 2000$, as well as three numbers of particles, namely $N = 50, 100, 200$. It is worthwhile to mention that, the simulated data

used across different particle numbers for a given sample size are the same, while it changes when the sample size increases. Table 2 reports the average values of the posterior means, standard deviations and IFs of μ_q, ϕ_q and σ_q^2 across replications. To save the space, only the results for first three q's are presented.

Generally speaking, the findings are in line with and similar to those in Chen et al. (2025). It can be seen that even for a small sample size (such as 500) and a relatively small number of particles (such as 50), the posterior means for μ_q are reasonably close to their respective true values, although there is an downward bias for μ_q. Nevertheless, it can be seen that the bias shrinks towards zero when T expands. As expected, the standard deviations for μ_q substantially decrease as T increases while an increasing number of particles has no effect in this regard. Meanwhile, the persistence parameters ϕ_q can be estimated accurately, even with 500 observations and 50 particles. The estimates have very small biases and low standard deviations. With 200 particles, the bias almost completely vanishes. Substantial downward biases are observed for σ_q^2 when sample size is 500. This bias is insensitive to the number of particles. Fortunately, it can be improved when more observations are available. Indeed, we observe that if $T = 2000$, the bias completely vanishes for σ_q^2. Finally, the IF varies little as we change the sample size, but improves substantially when the number of particles increases. Consistent with earlier studies, the IF is the lowest for μ and the highest for σ^2. Compared with the traditional single-move or multi-move Gibbs sampler (see, for example, Kim et al. (1998)), the PGAS sampler enjoys a much better mixing property.

In summary, the simulation results confirm that our chosen approach works well for the model considered in our study. In light of the good performance, 200 particles are used for the empirical applications reported later.

Table 2: Simulation results

T	N		μ_q			ϕ_q			σ_q^2		
			q_1	q_2	q_3	q_1	q_2	q_3	q_1	q_2	q_3
		True	0.376	0.047	0.017	0.700	0.700	0.700	0.050	0.050	0.050
		Mean	0.373	0.053	0.024	0.715	0.714	0.703	0.049	0.049	0.050
	50	Std	0.038	0.038	0.037	0.043	0.041	0.042	0.007	0.006	0.006
		IF	19.982	6.519	6.697	47.180	39.978	39.596	107.585	109.074	106.104
		Mean	0.373	0.053	0.024	0.716	0.715	0.703	0.048	0.049	0.050
500	100	Std	0.039	0.038	0.036	0.044	0.041	0.042	0.007	0.006	0.006
		IF	16.145	4.212	4.299	36.828	28.475	26.950	82.142	91.993	90.350
		Mean	0.372	0.053	0.024	0.715	0.715	0.703	0.048	0.049	0.050
	200	Std	0.039	0.038	0.037	0.044	0.041	0.042	0.007	0.006	0.006
		IF	12.546	3.272	3.028	25.045	21.637	19.203	61.077	74.447	69.575
		Mean	0.372	0.049	0.021	0.706	0.709	0.701	0.049	0.050	0.050
	50	Std	0.026	0.026	0.025	0.031	0.029	0.030	0.005	0.004	0.004
		IF	21.657	6.721	6.057	49.681	43.590	40.205	120.253	119.481	108.448
		Mean	0.372	0.049	0.021	0.707	0.709	0.701	0.049	0.049	0.050
1000	100	Std	0.026	0.026	0.025	0.031	0.029	0.030	0.005	0.004	0.004
		IF	17.932	4.713	4.449	35.680	28.909	27.830	85.456	94.595	89.494
		Mean	0.372	0.049	0.021	0.706	0.709	0.701	0.049	0.050	0.050
	200	Std	0.026	0.026	0.025	0.032	0.029	0.030	0.005	0.004	0.004
		IF	14.039	2.792	3.001	25.655	20.662	21.532	60.209	73.895	69.939
		Mean	0.374	0.046	0.019	0.703	0.705	0.699	0.050	0.050	0.050
	50	Std	0.018	0.018	0.018	0.022	0.021	0.021	0.003	0.003	0.003
		IF	21.839	7.320	6.711	49.167	41.794	38.400	110.206	110.657	111.660
		Mean	0.373	0.046	0.019	0.703	0.705	0.699	0.049	0.050	0.050
2000	100	Std	0.018	0.018	0.018	0.023	0.021	0.021	0.003	0.003	0.003
		IF	18.610	5.052	4.867	37.372	28.710	29.986	85.091	93.347	91.965
		Mean	0.374	0.046	0.018	0.704	0.704	0.698	0.049	0.050	0.050
	200	Std	0.018	0.018	0.018	0.023	0.021	0.021	0.003	0.003	0.003
		IF	13.782	3.311	3.168	26.786	22.623	21.714	61.392	70.560	70.305

Notes:
(1). The number of groups is 3 and each group contains 6 assets.
(2). T is the number of observations for each asset. N is the number of particles used in PGAS.
(3). Mean, std and IF are the average value of the posterior means, the standard error of the posterior means, and the average inefficiency factor, respectively.
(4). All these three statistics are computed across 1000 replications.

5 Empirical Studies

5.1 Data description and competing models

Durable	Non-durable	FIRE
VGR (2111)	AIMC (3568)	ARCC (6798)
LLY (2834)	NDSN (3560)	CEE (6726)
TSN (2015)	CCMP (3291)	HT (6798)
NL (2816)	ESE (3564)	FRT (6798)

Table 3: List of stocks used and their four-digit SIC code

In this section, we apply the MSV-B-GFT model to daily U.S. stock return data and compare the performance of the new model with some alternative models. The full sample period that we consider is from 2008 to 2020 and the twelve stocks involved in our analysis are listed in Table 3 accompanied by their four-digit SIC code. Following Archakov & Hansen (2024), these stocks can be categorized into three groups according to the industrial sector they belong to. Specifically, the stocks in the first column, VGR, LLY, TSN and NL, are all from the Durable Goods sector; those in the second column, AIMC, NDSN, CCMP and ESE, are all from the Non-durable Goods sector; the stocks in the last column, ARCC, CEE, HT and FRT are all from the Finance, Insurance, and Real Estate (FIRE) sector.

The time series plots for these stock returns are presented in Figure 2, in which the NBER recession periods are highlighted by shaded area. The summary statistics for these stock returns are reported in Table 4. We note that normality is clearly rejected for all stocks by the classical tests at conventional significance level. In addition, the stocks in the FIRE sector exhibit substantially higher kurtosis compared with other two sectors.

The information for sample correlations among these stocks is reported in Table 5, where the shaded regions highlight the block structure defined by industrial sectors. The numbers below the main diagonal are from the sample correlation matrix, while those above the diagonal are from the logarithm of the sample correlation matrix. Note that the estimated

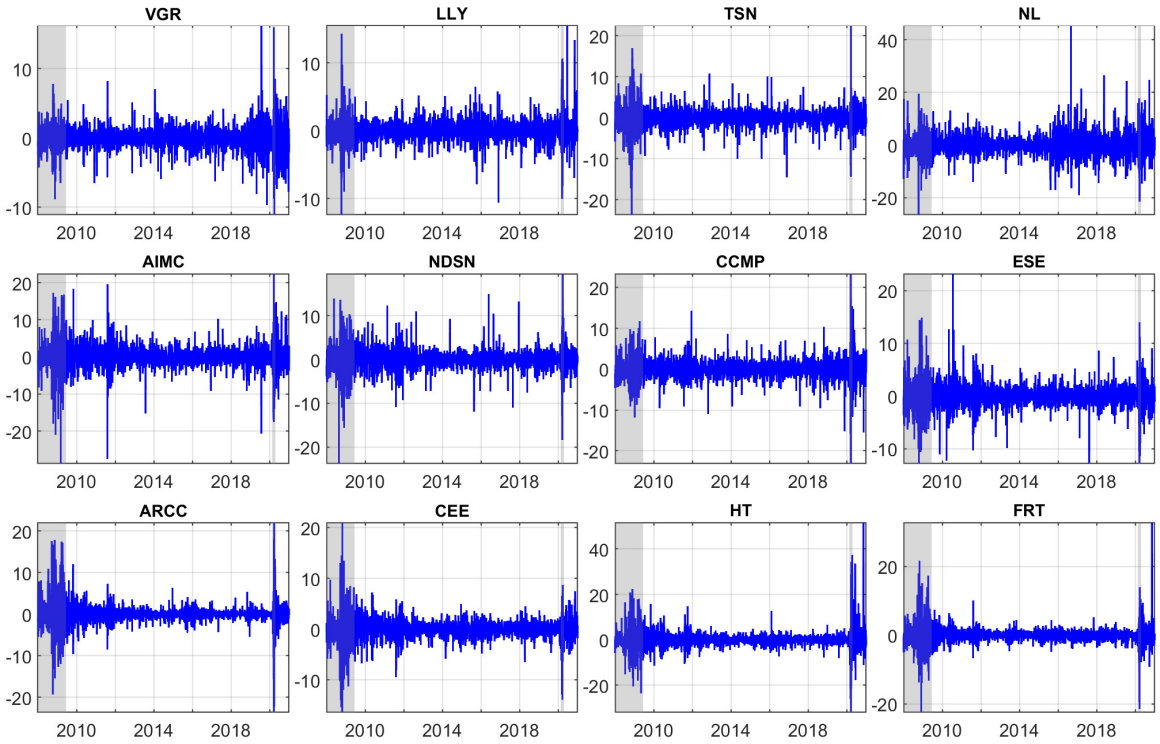

Figure 2: Time series for daily returns

Table 4: Summary statistics for daily returns

	Non-durables				Durables				FIRE			
	VGR	**LLY**	**TSN**	**NL**	**AIMC**	**NDSN**	**CCMP**	**ESE**	**ARCC**	**CEE**	**HT**	**FRT**
Mean	-0.006	-0.008	-0.002	-0.011	-0.003	0.000	0.001	0.004	-0.003	-0.004	-0.009	-0.009
Std.	1.622	1.593	2.226	3.977	3.096	2.320	2.354	2.192	2.353	2.038	3.613	2.314
Skewness	0.211	0.513	-0.292	0.790	-0.348	-0.164	-0.017	0.486	0.212	-0.245	1.660	0.880
Kurtosis	13.201	15.594	17.171	12.748	14.703	14.994	15.153	13.796	24.563	16.381	33.752	31.468
Min	-11.112	-12.418	-23.694	-26.527	-28.778	-23.819	-23.108	-12.840	-23.634	-16.401	-31.978	-22.474
Q-05%	-2.358	-2.310	-3.284	-5.874	-4.238	-3.320	-3.336	-3.002	-2.868	-2.911	-4.675	-2.948
Q-25%	-0.726	-0.728	-0.930	-1.932	-1.326	-0.916	-1.043	-0.949	-0.657	-0.810	-1.243	-0.796
Q-50%	0.007	0.002	0.065	-0.072	-0.020	-0.011	0.005	-0.040	-0.018	0.032	-0.052	-0.002
Q-75%	0.749	0.702	0.994	1.732	1.213	1.002	1.042	0.981	0.641	0.873	1.111	0.768
Q-95%	2.359	2.171	2.954	6.341	4.536	3.136	3.350	3.224	2.796	2.696	4.183	2.848
Max	16.332	15.610	22.626	45.607	22.428	19.606	23.364	23.220	22.074	20.971	51.665	32.853

Notes:
(1). The sample period is from January 2nd, 2008, to December 31st, 2020 (3274 trading days).
(2). Daily returns are all multiplied by 100 for better presentation.

Table 5: Sample correlation matrix and its logarithm

		Non-durables				Durables				FIRE			
		VGR	**LLY**	**TSN**	**NL**	**AIMC**	**NDSN**	**CCMP**	**ESE**	**ARCC**	**CEE**	**HT**	**FRT**
Non-durables	**VGR**		0.168	0.112	0.149	0.157	0.132	0.136	0.210	0.140	0.042	0.137	0.202
	LLY	0.311		0.132	0.094	0.102	0.179	0.181	0.164	0.149	0.195	0.021	0.157
	TSN	0.285	0.295		0.076	0.178	0.166	0.134	0.142	0.201	0.172	0.189	0.166
	NL	0.297	0.253	0.252		0.187	0.180	0.185	0.193	0.127	0.080	0.142	0.081
Durables	**AIMC**	0.378	0.341	0.407	0.391		0.346	0.273	0.319	0.260	0.223	0.298	0.178
	NDSN	0.367	0.394	0.405	0.389	0.591		0.338	0.309	0.312	0.322	0.146	0.175
	CCMP	0.352	0.376	0.363	0.375	0.531	0.567		0.295	0.175	0.199	0.146	0.215
	ESE	0.405	0.374	0.377	0.388	0.563	0.565	0.535		0.146	0.221	0.163	0.265
FIRE	**ARCC**	0.361	0.364	0.418	0.341	0.536	0.564	0.471	0.472		0.317	0.249	0.368
	CEE	0.289	0.380	0.387	0.301	0.500	0.555	0.467	0.487	0.544		0.171	0.266
	HT	0.342	0.260	0.388	0.329	0.520	0.450	0.421	0.447	0.504	0.438		0.465
	FRT	0.401	0.366	0.404	0.321	0.507	0.505	0.488	0.525	0.587	0.518	0.607	

Notes:
(1). The sample correlations are shown below the diagonal, while the transformed elements are shown above the diagonal.
(2). The sample period is from January 2nd, 2008, to December 31st, 2020 (3274 trading days).
(3). The sector-based block structure is illustrated with the shaded region.

unconditional correlations within each block have a similar average. The assets within the Durables sector and FIRE sector are strongly correlated, with an average correlation of around 0.55. Within-sector correlations for stock returns in Non-durables sector tend to be milder, with an average of about 0.3. The between-sector correlations tend to be of a similar magnitude within a block, but exhibit some variation across different blocks. Unsurprisingly, similar patterns can be observed for the corresponding elements of the logarithm of the unconditional correlation matrix.

5.2 Estimation results

The parameter estimation for the stochastic volatility processes, h_t, are presented in Table 6. The results are broadly consistent with the existing literature, which suggests that log-volatility is highly persistent. Indeed, for all twelve stocks, we observe that the posterior mean of the autoregressive parameter ϕ is above 0.9 and the posterior standard deviation

is relatively small. A noteworthy finding is that the volatility for stocks in the FIRE sector is uniformly more persistent than the counterparts in the Non-durables and Durables, with the posterior mean of ϕ up to 0.98 and the corresponding posterior standard deviation about half of those in other two sectors. Meanwhile, we also note that the parameter characterizing volatility of log-volatility, namely σ^2, is slightly lower in the FIRE sector. Another finding in line with existing papers is that the IF for the mean parameter μ is much lower than that for ϕ and σ^2, which implies that the former enjoys a better sampling efficiency.

The estimation results for the distinct transformed correlations, q_t, are shown in Table 7. Contrary to the case of volatility, we find that the degree of persistence for correlation variables is much more diverse, with the posterior mean ranging from 0.4386 to 0.9533. This finding reveals that some processes underlying comovements among stock returns can be quite stationary, while others are close to unit root process. Such a pattern can be also recognized from Figure 3, where we plot the filtered within-group and between-group correlation coefficients together with the corresponding 95% confidence interval. For instance, when inspecting the dynamics of these correlation coefficients, we clearly see that comovement within the FIRE sector is more persistent than that in other two sectors. Meanwhile, we can also observe that all three between-group correlations are mildly persistent with similar dynamics. As for the sampling efficiency of MCMC algorithm, it can be observed that the IFs for correlation-related parameters are in all cases significantly larger than the counterparts for volatility dynamics. Furthermore, the variation of IFs across q_t's is apparently more pronounced than that for different h_t's.

In summary, our findings above suggest that the correlations are in general more heterogeneous compared with volatilities. It is therefore of paramount importance to employ a flexible parameterization when modeling the dynamics of the block correlation matrix.

Table 6: Estimation results for stochastic volatilities

		μ				ϕ			
Non-durables	**VGR**	0.3365	(0.1054)	2.1868	[0.1312, 0.5457]	0.9334	(0.0115)	57.5896	[0.9089, 0.9539]
	LLY	0.3167	(0.0955)	4.1681	[0.1300, 0.5055]	0.9292	(0.0139)	111.3084	[0.8985, 0.9534]
	TSN	0.9050	(0.0879)	4.2047	[0.7332, 1.0804]	0.9013	(0.0158)	64.5369	[0.8684, 0.9298]
	NL	2.1685	(0.1085)	2.6024	[1.9564, 2.3830]	0.9349	(0.0115)	69.2902	[0.9107, 0.9557]
Durables	**AIMC**	1.5120	(0.1317)	2.0074	[1.2569, 1.7752]	0.9547	(0.0093)	82.7934	[0.9347, 0.9711]
	NDSN	0.9144	(0.1038)	3.7806	[0.7137, 1.1208]	0.9182	(0.0138)	66.8497	[0.8889, 0.9426]
	CCMP	1.0736	(0.0909)	4.0967	[0.8949, 1.2531]	0.9166	(0.0139)	72.6884	[0.8877, 0.9419]
	ESE	0.8975	(0.1129)	2.0501	[0.6776, 1.1214]	0.9421	(0.0108)	79.3972	[0.9195, 0.9616]
FIRE	**ARCC**	0.2450	(0.2899)	1.1344	[-0.3302, 0.8113]	0.9804	(0.0046)	40.4438	[0.9710, 0.9888]
	CEE	0.6423	(0.1839)	1.7273	[0.2810, 1.0120]	0.9768	(0.0058)	72.6033	[0.9642, 0.9869]
	HT	1.3895	(0.2294)	1.7606	[0.9441, 1.8510]	0.9784	(0.0051)	58.8827	[0.9677, 0.9878]
	FRT	0.5950	(0.2853)	1.4199	[0.0413, 1.1731]	0.9858	(0.0040)	55.6855	[0.9773, 0.9930]

		σ^2			
Non-durables	**VGR**	0.1449	(0.0243)	87.6395	[0.1023, 0.1984]
	LLY	0.1296	(0.0264)	146.7488	[0.0857, 0.1904]
	TSN	0.2093	(0.0344)	91.2828	[0.1493, 0.2824]
	NL	0.1456	(0.0247)	112.7461	[0.1026, 0.1990]
Durables	**AIMC**	0.1033	(0.0198)	122.3475	[0.0692, 0.1480]
	NDSN	0.2083	(0.0352)	92.1855	[0.1479, 0.2871]
	CCMP	0.1654	(0.0284)	103.3209	[0.1146, 0.2248]
	ESE	0.1254	(0.0225)	117.8876	[0.0848, 0.1735]
FIRE	**ARCC**	0.0941	(0.0152)	99.8139	[0.0680, 0.1278]
	CEE	0.0532	(0.0113)	128.9314	[0.0352, 0.0792]
	HT	0.0715	(0.0130)	113.4501	[0.0484, 0.0988]
	FRT	0.0448	(0.0090)	130.9466	[0.0296, 0.0658]

Notes:
(1). The sample period is from January 2nd, 2008, to December 31st, 2020 (3274 trading days).
(2). For each stock, we report (in order) the posterior mean, posterior standard deviation, IF, and 95% confidence interval for three parameters in the SV model.

Table 7: Estimation results for dynamic block correlations

	$\boldsymbol{q}_1$				$\boldsymbol{q}_2$			
$\boldsymbol{\mu}$	0.1505	(0.0097)	69.7056	[0.1307, 0.1688]	0.1878	(0.0063)	39.1352	[0.1747, 0.1998]
$\boldsymbol{\phi}$	0.4386	(0.1809)	433.7997	[0.1029, 0.7475]	0.7164	(0.1194)	486.5048	[0.4317, 0.8879]
$\boldsymbol{\sigma}^2$	0.0161	(0.0067)	443.7130	[0.0059, 0.0296]	0.0039	(0.0019)	493.3928	[0.0012, 0.0083]
	$\boldsymbol{q}_3$				$\boldsymbol{q}_4$			
$\boldsymbol{\mu}$	0.3739	(0.0091)	89.0344	[0.3561, 0.3917]	0.1685	(0.0067)	22.5015	[0.1548, 0.1813]
$\boldsymbol{\phi}$	0.4022	(0.0744)	163.4831	[0.2540, 0.5456]	0.8270	(0.1025)	539.5633	[0.5050, 0.9263]
$\boldsymbol{\sigma}^2$	0.0315	(0.0042)	168.6207	[0.0233, 0.0403]	0.0020	(0.0014)	560.2102	[0.0007, 0.0068]
	$\boldsymbol{q}_5$				$\boldsymbol{q}_6$			
$\boldsymbol{\mu}$	0.2067	(0.0071)	22.1578	[0.1924, 0.2203]	0.2178	(0.0164)	6.4943	[0.1852, 0.2502]
$\boldsymbol{\phi}$	0.8120	(0.0598)	323.1342	[0.6858, 0.9084]	0.9533	(0.0135)	257.4267	[0.9215, 0.9744]
$\boldsymbol{\sigma}^2$	0.0033	(0.0013)	350.0993	[0.0014, 0.0061]	0.0014	(0.0004)	358.0041	[0.0008, 0.0026]

Notes:
(1). The sample period is from January 2nd, 2008, to December 31st, 2020 (3274 trading days).
(2). For each latent variable q, we report (in order) the posterior mean, posterior standard deviation, IF, and 95% confidence interval for three parameters in the dynamic model.

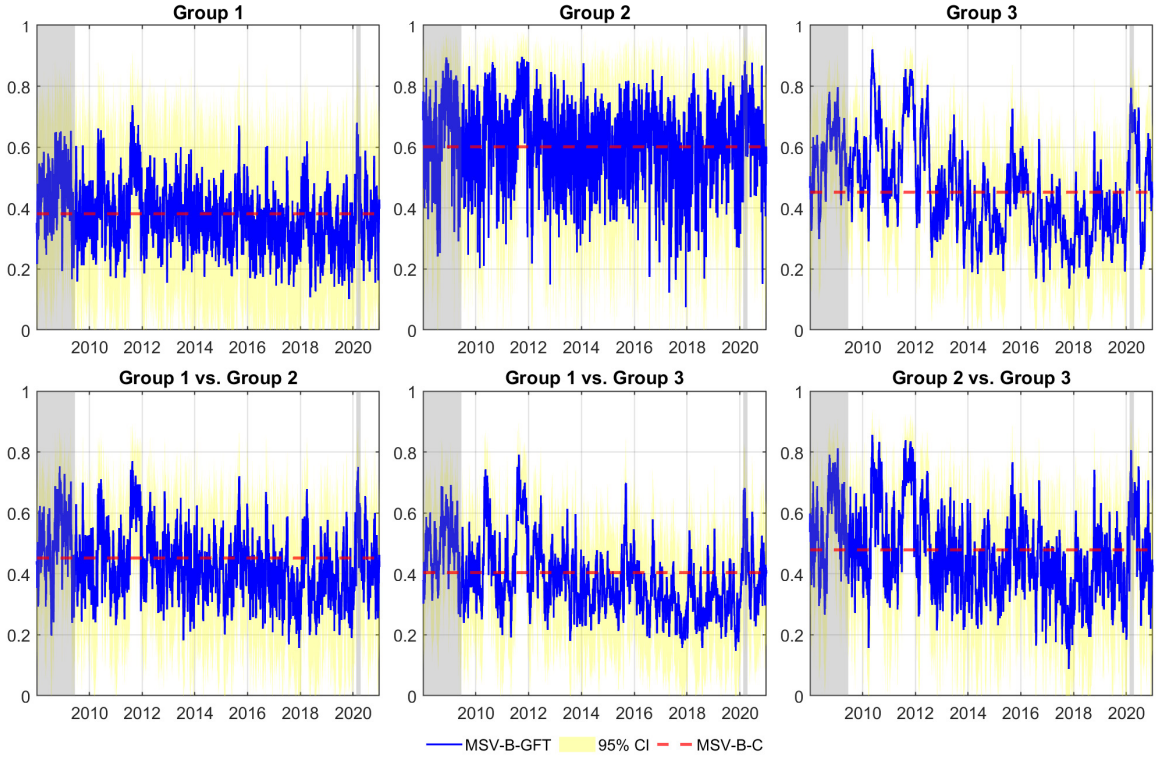

Figure 3: Filtered between-group and within-group correlation coefficients

5.3 Model comparisons

To further showcase the empirical performance of our new MSV model with a block correlation matrix, we conduct both the in-sample comparison and the out-of-sample model comparison in this subsection. The candidate models we consider can be categorized into following two groups:[12]

1. MSV models with static correlations:
   (a) MSV-C: constant correlation MSV model considered in Harvey et al. (1994).
   (b) MSV-B-C: MSV-B-GFT model with (8e) replaced by $q_t = q$ for all t.
   (c) MSV-E-C: equicorrelation MSV model with time-invariant correlations.
2. MSV models with dynamic correlations:
   (a) MSV-B-GFT.
   (b) MSV-E-D: dynamic equicorrelation SV model in Kurose & Omori (2016).

For all these models, we assume the same volatility dynamics and Gaussian innovations, so that the comparison can be concentrated on modeling of the correlation structure. Note that the MSV-C model allows all correlation coefficients to be different and impose no block structure.[13] Although the MSV-C model is flexible in terms of cross-asset correlations, it imposes a very strong restriction on the time dimension. A direct extension to dynamic correlations is computationally infeasible when n is moderately large due to the huge amount of parameters. MSV-E-C and MSV-E-D model, at the opposite end, are highly parsimonious with comovements among all stocks captured by just one correlation coefficient. Although computationally attractive, such an assumption can be overly restrictive in many applications. The MSV-B-C model and the MSV-B-GFT model, both featuring a block structure defined by sector categories, strike a good balance between flexibility and feasibility. Between these

[12] In terms of implementation, we note that all these MSV models can be treated as special cases of MSV-B-GFT model and hence estimated through the algorithm introduced in Section 3.

[13] Also note that MSV-C is a model with only one block.

two alternatives, MSV-B-GFT is particularly promising as it further allows time-varying correlations.

The results for the in-sample comparison can be found in Panel (a) of Table 8, which contains both the (negative) log marginal likelihood of each competing model and the corresponding deviance information criterion (DIC). The former is obtained using the approach proposed in Chib (1995), with the likelihood ordinate computed by auxiliary particle filter of Pitt & Shephard (1999); the latter is obtained using DIC_1 of Spiegelhalter et al. (2002) and Li et al. (2025). We highlight the model with highest log marginal likelihood and lowest DIC value by boldface and underline. Clearly, our new MSV-B-GFT model markedly outperforms all other MSV candidates based on both criteria, providing compelling in-sample evidence in favor of the new specification featuring dynamic block correlations.

To assess the predictive power of MSV-B-GFT, we first compare its predictive log-likelihood of daily returns with that of other specifications. For trading day t, this quantity is defined as $\log p(r_t|r_{1:t-1}), t \in \{T_0, +1, \cdots, T\}$, where T_0 denotes the in-sample size and $r_{1:t-1} = (r_1', \cdots, r_{T-1}')'$. We then obtain the average out-of-sample predictive log-likelihood of each candidate model. The results covering entire out-of-sample period as well as three sub-years (2018-2020) can be found in Panel (b) of Table 8, where figures in the parenthesis are the corresponding p-values of the model confidence set (MCS) of Hansen et al. (2011). Boldface and underlined values again highlight the best model. It is apparent that our flexible MSV-B-GFT model is preferred in terms of its ability in predicting the return distribution. This conclusion is statistically significant in most cases as suggested by the MCS p-values. The finding of superiority of MSV-B-GFT is robust across the different sample periods.

Next, we compare all competing models based on the empirically more relevant economic loss function. For this purpose, we construct the global minimum variance (GMV) portfolio for each model. To enable a fair comparison across models, we assume that all stocks

Table 8: Results for model comparisons

	Constant correlation			Dynamic correlation			
	MSV-C	MSV-B-C	MSV-E-C	MSV-B-GFT	MSV-E-D	DCS-B	EW
(a). In-sample fit							
Marginal Lik	132461	132744	133528	**131486**	132614	-	-
DIC	-65806	-66381	-66770	**-65772**	-66321	-	-
(b). Predictive likelihood							
All	-24.731	-24.809	-25.076	**-24.075**	-24.328	-	-
	(0.001)	(0.000)	(0.000)	(1.000)	(0.001)	-	-
2018	-22.290	-22.381	-22.544	**-21.559**	-21.741	-	-
	(0.000)	(0.000)	(0.000)	(1.000)	(0.013)	-	-
2019	-22.794	-22.826	-23.204	**-22.072**	-22.377	-	-
	(0.076)	(0.038)	(0.000)	(1.000)	(0.076)	-	-
2020	-29.082	-29.193	-29.453	**-28.567**	-28.837	-	-
	(0.095)	(0.078)	(0.002)	(1.000)	(0.095)	-	-
(c). MSE of GMV portfolio							
All	1.780	1.779	1.872	**1.703**	1.851	1.744	3.876
	(0.450)	(0.137)	(0.060)	(1.000)	(0.137)	(0.450)	(0.017)
2018	0.579	0.611	0.648	**0.578**	0.627	0.600	1.029
	(0.941)	(0.213)	(0.035)	(1.000)	(0.268)	(0.268)	(0.001)
2019	0.470	0.470	0.484	**0.463**	0.481	0.469	0.970
	(0.960)	(0.947)	(0.694)	(1.000)	(0.695)	(0.960)	(0.000)
2020	4.278	4.241	4.470	**4.054**	4.430	4.148	9.594
	(0.405)	(0.214)	(0.110)	(1.000)	(0.214)	(0.405)	(0.030)
(d). MAE of GMV portfolio							
All	0.781	0.791	0.802	**0.778**	0.797	0.787	1.171
	(0.734)	(0.225)	(0.034)	(1.000)	(0.225)	(0.225)	(0.000)
2018	**0.552**	0.570	0.581	0.556	0.566	0.566	0.743
	(1.000)	(0.044)	(0.044)	(0.702)	(0.466)	(0.234)	(0.000)
2019	**0.511**	0.520	0.523	0.517	0.521	0.522	0.756
	(1.000)	(0.803)	(0.803)	(0.803)	(0.803)	(0.688)	(0.000)
2020	1.276	1.278	1.299	**1.257**	1.301	1.271	2.009
	(0.733)	(0.317)	(0.296)	(1.000)	(0.317)	(0.733)	(0.000)

Notes:
(1). The in-sample period is from 2008 to 2017.
(2). Values in the parenthesis are the p-values for the MCS test.
(3). The underlined boldface values highlight the best model.

have equal expected returns and focus solely on the risk of the portfolio. In particular, we compare the average squared and absolute returns of the GMV portfolio constructed using each model. Beyond the MSV models discussed earlier, we also consider a portfolio with equal weights as a benchmark, which is frequently used in practice. Furthermore, to investigate the relative merits of our parameter-driven MSV models compared with existing observation-driven models, in our analysis, we also consider a simplified version of the Cluster GARCH model recently proposed in Tong et al. (2025), denoted by DCS-B.[14] The results are shown in Panel (b) and (c) of Table 8, accompanied by corresponding MCS p-values and with the best model highlighted by boldface and underline. To check the robustness of our analysis, we again present the analogous results for each of the three years in the out-of-sample period. The portfolio based on equal weights consistently displays notably higher variances in all scenarios, implying that this strategy is the least favorable option for our dataset. Among models with a time-invariant correlation matrix, MSV-E-C appears to be inferior to the other two specifications. MSV-C performs particularly well in terms of the average absolute return of the GMV portfolio, both for the full out-of-sample period and the three subperiods. Overall, our MSV-B-GFT model outperforms all other competitors under both criteria. On the one hand, the superiority of MSV-B-GFT to MSV-E-D suggests that it is important to relax the overly restricted equicorrelation assumption used in Kurose & Omori (2016). On the other hand, the better performance against MSV-B-C and DCS-B underlines the importance of allowing the correlations to have flexible dynamics.

6 Conclusion

In this paper, we introduce a novel MSV model. It extends the MSV-GFT specification proposed in Chen et al. (2025) to a higher-dimensional setting by combining a block cor-

[14] While Tong et al. (2025) propose a series of flexible distribution assumptions, we only consider their model with Gaussian innovations here for simplicity.

relation structure and the GFT technique of Archakov & Hansen (2021). There are three nice features in the new model. First, the new model ensures a block structure for both the correlation matrix and its log transformation. Second, the positive-definiteness constraint is satisfied. Third, the dimension of the model grows only with the number of blocks but not with the number of assets. This suggests that the model greatly reduces the dimensionality.

An efficient algorithm, based on Particle Gibbs Ancestor Sampling (PGAS) of Lindsten et al. (2014), is designed to conduct the Bayesian analysis of the proposed model. To reduce the computational cost of the Bayesian method, we use Broyden's method, a quasi-Newton method, to solve the fixed-point problem numerically. Extensive simulation studies confirm the performance of our inference procedure.

To demonstrate the usefulness of our model, we apply the model and the Bayesian method to twelve daily U.S. stock return series. We also compare our model with alternative models in terms of both the in-sample performance and the out-of-sample performance. The empirical results clearly show the advantage of our new model against alternative specifications.

References

Andrieu, C., Doucet, A. & Holenstein, R. (2010), 'Particle markov chain monte carlo methods', Journal of the Royal Statistical Society: Series B **72**(3), 269–342.

Archakov, I. & Hansen, P. R. (2021), 'A new parametrization of correlation matrices', Econometrica **89**(4), 1699–1715.

Archakov, I. & Hansen, P. R. (2024), 'A canonical representation of block matrices with applications to covariance and correlation matrices', Review of Economics and Statistics **106**(4), 1099–1113.

Archakov, I., Hansen, P. R. & Lunde, A. (2025), 'A multivariate realized GARCH model', Journal of Econometrics p. 106040.

Archakov, I., Hansen, P. R. & Luo, Y. (2024), 'A new method for generating random correlation matrices', The Econometrics Journal **27**(2), 188–212.

Arias, J. E., Rubio-Ramirez, J. F. & Shin, M. (2023), 'Macroeconomic forecasting and variable ordering in multivariate stochastic volatility models', Journal of Econometrics **235**(2), 1054–1086.

Arulampalam, M. S., Maskell, S., Gordon, N. & Clapp, T. (2002), 'A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking', IEEE Transactions on Signal Processing **50**(2), 174–188.

Asai, M. & McAleer, M. (2006), 'Asymmetric multivariate stochastic volatility', Econometric Reviews **25**(2-3), 453–473.

Asai, M. & McAleer, M. (2009), 'The structure of dynamic correlations in multivariate stochastic volatility models', Journal of Econometrics **150**(2), 182–192.

Asai, M., McAleer, M. & Yu, J. (2006), 'Multivariate stochastic volatility: A review', Econometric Reviews **25**(2-3), 145–175.

Bucci, A., Ippoliti, L. & Valentini, P. (2022), 'Comparing unconstrained parametrization methods for return covariance matrix prediction', Statistics and Computing **32**(5), 1–20.

Chan, D., Kohn, R. & Kirby, C. (2006), 'Multivariate stochastic volatility models with correlated errors', Econometric Reviews **25**(2-3), 245–274.

Chen, H., Fei, Y. & Yu, J. (2025), 'Multivariate stochastic volatility models based on generalized fisher transformation', Journal of Econometrics **251**, 106041.

Chib, S. (1995), 'Marginal likelihood from the gibbs output', Journal of the American Statistical Association **90**(432), 1313–1321.

Creal, D. & Kim, J. (2024), 'Bayesian estimation of cluster covariance matrices of unknown form', Journal of Econometrics **241**(1), 105725.

Engle, R. (2002), 'Dynamic conditional correlation: A simple class of multivariate generalized autoregressive conditional heteroskedasticity models', Journal of Business & Economic Statistics **20**(3), 339–350.

Engle, R. & Kelly, B. (2012), 'Dynamic equicorrelation', Journal of Business & Economic Statistics **30**(2), 212–228.

Hafner, C. M. & Wang, L. (2023), 'A dynamic conditional score model for the log correlation matrix', Journal of Econometrics **237**(2), 105176.

Hansen, P. R., Lunde, A. & Nason, J. M. (2011), 'The model confidence set', Econometrica **79**(2), 453–497.

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

Ishihara, T. & Omori, Y. (2012), 'Efficient bayesian estimation of a multivariate stochastic volatility model with cross leverage and heavy-tailed errors', Computational Statistics & Data Analysis **56**(11), 3674–3689.

Ishihara, T., Omori, Y. & Asai, M. (2016), 'Matrix exponential stochastic volatility with cross leverage', Computational Statistics & Data Analysis **100**, 331–350.

Johansen, A. M. & Doucet, A. (2008), 'A note on auxiliary particle filters', Statistics & Probability Letters **78**(12), 1498–1504.

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

Kurose, Y. & Omori, Y. (2016), 'Dynamic equicorrelation stochastic volatility', Computational Statistics & Data Analysis **100**, 795–813.

Kurose, Y. & Omori, Y. (2020), 'Multiple-block dynamic equicorrelations with realized measures, leverage and endogeneity', Econometrics and Statistics **13**, 46–68.

Li, Y., Mallick, S. K., Wang, N., Yu, J. & Zeng, T. (2025), 'Deviance information criterion for bayesian model selection: Theoretical justification and applications', Journal of Econometrics p. 105978.

Lindsten, F., Jordan, M. I. & Schön, T. B. (2014), 'Particle gibbs with ancestor sampling', The Journal of Machine Learning Research **15**(1), 2145–2184.

Linton, O. & McCrorie, J. R. (1995), 'Differentiation of an exponential matrix function', Econometric Theory **11**(5), 1182–1185.

Oh, D. H. & Patton, A. J. (2023), 'Dynamic factor copula models with estimated cluster assignments', Journal of Econometrics **237**(2), 105374.

Pitt, M. K. & Shephard, N. (1999), 'Filtering via simulation: Auxiliary particle filters', Journal of the American Statistical Association **94**(446), 590–599.

Poignard, B. & Asai, M. (2023), 'High-dimensional sparse multivariate stochastic volatility models', Journal of Time Series Analysis **44**(1), 4–22.

Poignard, B. & Asai, M. (2024), 'Factor multivariate stochastic volatility models of high dimension', arXiv preprint arXiv:2406.19033 .

Spiegelhalter, D. J., Best, N. G., Carlin, B. P. & Van Der Linde, A. (2002), 'Bayesian measures of model complexity and fit', Journal of the Royal Statistical Society: Series B **64**(4), 583–639.

Tong, C., Hansen, P. R. & Archakov, I. (2025), 'Cluster GARCH', Journal of Business & Economic Statistics **44**(1), 148—-161.

Watanabe, T. & Omori, Y. (2004), 'A multi-move sampler for estimating non-gaussian time series models: Comments on shephard & pitt (1997)', Biometrika **91**(1), 246–248.

Yamauchi, Y. & Omori, Y. (2020), 'Multivariate stochastic volatility model with realized volatilities and pairwise realized correlations', Journal of Business & Economic Statistics **38**(4), 839–855.

Yu, J. & Meyer, R. (2006), 'Multivariate stochastic volatility models: Bayesian estimation and model comparison', Econometric Reviews **25**(2-3), 361–384.

Online Supplement for "Multivariate Stochastic Volatility Model with Block Correlations" by Han Chen, Yijie Fei, Jun Yu

(Not for Publication)

A Proof of Proposition 1

First note that, by construction,

$$S^B = \begin{bmatrix} \iota'_{n_1} & & \\ & \ddots & \\ & & \iota'_{n_K} \end{bmatrix},$$

where ι_{n_k} is a column vector consisting of n_k ones. It is straightforward to verify that

$$\begin{aligned} Q_{[k,k]} &= \begin{bmatrix} q_{kk} & q_{kk} & \cdots & q_{kk} \\ q_{kk} & q_{kk} & \ddots & \\ \vdots & \ddots & \ddots & \\ q_{kk} & & & q_{kk} \end{bmatrix} + \begin{bmatrix} y_{kk}-q_{kk} & & & \\ & y_{kk}-q_{kk} & & \\ & & \ddots & \\ & & & y_{kk}-q_{kk} \end{bmatrix} \\ &= q_{kk}\iota_{n_k}\iota'_{n_k} + (y_{kk}-q_{kk})I_{n_k} \end{aligned} \tag{A.11}$$

Denote the first matrix on the right hand side by $\tilde{Q}_{[k,k]}$, we have

$$\begin{aligned} \operatorname{logm} C^B &= \begin{bmatrix} \tilde{Q}_{[1,1]} & Q_{[1,2]} & \cdots & Q_{[1,K]} \\ Q_{[2,1]} & \tilde{Q}_{[2,2]} & & \\ \vdots & & \ddots & \\ Q_{[K,1]} & & & \tilde{Q}_{[K,K]} \end{bmatrix} + \begin{bmatrix} (y_{11}-q_{11})I_{n_1} & & & \\ & (y_{22}-q_{22})I_{n_2} & & \\ & & \ddots & \\ & & & (y_{KK}-q_{KK})I_{n_K} \end{bmatrix} \\ &= \begin{bmatrix} q_{11}\iota_{n_1}\iota'_{n_1} & q_{12}\iota_{n_1}\iota'_{n_2} & \cdots & q_{1K}\iota_{n_1}\iota'_{n_K} \\ q_{21}\iota_{n_2}\iota'_{n_1} & q_{22}\iota_{n_2}\iota'_{n_2} & \cdots & q_{2K}\iota_{n_2}\iota'_{n_K} \\ \vdots & \vdots & \ddots & \vdots \\ q_{K1}\iota_{n_K}\iota'_{n_1} & q_{K2}\iota_{n_K}\iota'_{n_2} & \cdots & q_{KK}\iota_{n_K}\iota'_{n_K} \end{bmatrix} + \begin{bmatrix} (y_{11}-q_{11})I_{n_1} & & & \\ & \ddots & & \\ & & & (y_{KK}-q_{KK})I_{n_K} \end{bmatrix}. \\ &= S^{B\prime} Q S^B + \begin{bmatrix} (y_{11}-q_{11})I_{n_1} & & & \\ & \ddots & & \\ & & & (y_{KK}-q_{KK})I_{n_K} \end{bmatrix} \end{aligned} \tag{A.12}$$

As a result, we obtain that

$$vecl(\operatorname{logm} C^B) = E_n vec\left(S^{B\prime} Q S^B\right) = E_n \left(S^B \otimes S^B\right)' vec(Q) = E_n \left(S^B \otimes S^B\right)' G_K vech(Q)$$

and Proposition 1 follows immediately.

B Extension with realized measures

Our model can be extended in a straightforward manner when realized measures are available, following the setup in Chen et al. (2025). Specifically, suppose that the an $n \times n$ realized covariance matrices (denoted by RC_t) computed from intra-day high-frequency returns is available. Let

$$RC_t = (H_t^r)^{1/2} C_t^r (H_t^r)^{1/2}, \tag{B.13}$$

where the superscripts denote realized measures. H_t^r is a diagonal matrix collecting all the realized variances and C_t^r is the realized correlation matrix. Equation (B.13) is the realized version of the traditional variance-correlation decomposition. Since the latent variables in MSV-B-GFT are the transformation of the original variances and the correlation coefficients, we apply the same transformation to the realized covariance. Specifically, we define

$$h_t^r = (h_{1t}^r, ..., h_{nt}^r)' = diag\left(\operatorname{logm}(V_t^r)^{1/2}\right) \tag{B.14}$$

and

$$q_t^r = (q_{1t}^r, ..., q_{dt}^r)' = \left[E_l \left(S^B \otimes S^B\right)' L_k\right]^+ vecl(\operatorname{logm} C_t^r), \tag{B.15}$$

where $^+$ denotes the pseudo-inverse defined by $M^+ = (M'M)^{-1}M'$.[1]

It can be seen that h_t^r and q_t^r are the observed empirical measures of the latent variables, h_t and q_t, respectively. We hence expect that time variation in these realized measures contains information about the dynamics of the corresponding latent variables. As is well known in the literature, there may exist non-trivial measurement errors in h_t^r and q_t^r, because neither are perfect measurements of the latent variables due to the microstructure noise, nontrading hours, nonsynchronous trading, and so forth. Bearing this in mind, we can model the

[1] See the Footnote 5 in Archakov et al. (2025) for further discussions on the specification (B.15).

relationship between the latent variables and their realized counterparts by

$$h_t^r = \psi_h + h_t + \xi_{ht},\ \xi_{ht} \sim N\left(0, \Sigma_h^r\right), \tag{B.16a}$$

$$q_t^r = \psi_q + q_t + \xi_{qt},\ \xi_{qt} \sim N\left(0, \Sigma_q^r\right), \tag{B.16b}$$

where $\psi_h = (\psi_{h1}, ..., \psi_{hn})'$ and $\psi_q = (\psi_{q1}, ..., \psi_{qd})'$ capture potential approximation errors in the realized measures. For implementation, we may further assume

$$\Sigma_h^r = diag\left(\left(\eta_{h1}^2, ...\eta_{hn}^2\right)'\right) \text{ and } \Sigma_q^r = diag\left(\left(\eta_{q1}^2, ..., \eta_{qd}^2\right)'\right).$$

Combining equations (8a)-(8f) with equations (B.16a)-(B.16b), we may obtain the extended realized MSV-block-GFT (RMSV-B-GFT) model.

C Review of PG approach

To fix some notations, let $r = (r_1', ..., r_T')'$, $h = (h_1', ..., h_T')'$, $q = (q_1', ..., q_T')'$, $x = (h', q')' := (x_1', ..., x_T')'$ so that $x_t = (h_t', q_t')'$. Vector x contains all latent variables and vector x_t contains all latent variables at period t. We use $x_{1:t}$ to denote $(x_1', ..., x_t')'$ for any $t = 1, \cdots, T$, θ denote the set of parameters in the model, and $p(r|\theta)$ denote the likelihood function of the model.

Consider a general state-space model given by

$$\begin{aligned} r_t|x_t = x, \theta &\sim f(\cdot|x, \theta), \\ x_{t+1}|x_t = x, \theta &\sim g(\cdot|x, \theta), \text{ and } x_1 \sim \mu_\theta(\cdot). \end{aligned} \tag{C.17}$$

where $f(\cdot|x, \theta)$ is the measurement density, $g(\cdot|x, \theta)$ is the transition probability density and $\mu_\theta(\cdot)$ is the initial density.

To sample from $p(\theta, x_{1:T}|r_{1:T})$, a Gibbs sampler draws alternately from the two conditional densities, $p(\theta|x_{1:T}, r_{1:T})$ and $p(x_{1:T}|r_{1:T}, \theta)$. PG draws random samples from $p(x_{1:T}|r_{1:T}, \theta)$ based on the particle filter, which is applicable as long as the measurement density $f(\cdot|x, \theta)$ can be numerically evaluated and the transition density $g(\cdot|x, \theta)$ can be simulated.[2]

[2]Despite its general applicability, when implementing particle filter for a particular model, many subtle issues must be considered. These include how to choose a proper importance density, how many particles to use, and whether a resampling step should be added. For a thorough discussion, see Arulampalam et al. (2002) and Johansen & Doucet (2008).

The particle filter combines importance sampling and Monte Carlo simulations to approximate the target distribution. The key idea is to represent the distribution by a set of random samples with the corresponding weights and calculate the quantity of interest based on these samples and weights. Let $\left\{x_{1:t}^{(i)}, w_t^{(i)}\right\}_{i=1}^{N}$ be a random measure, where $\left\{x_{1:t}^{(i)}, i = 1, ..., N\right\}$ is a set of support points and $\left\{w_t^{(i)}, i = 1, ..., N\right\}$ are the associated weights. Each point is called a particle, and N is the number of particles used. The approximate distribution can then be written as

$$\hat{p}_\theta \left(dx_{1:t}|r_{1:t}\right) = \sum_{i=1}^{N} w_t^{(i)} \delta_{x_{1:t}^{(i)}} \left(dx_{1:t}\right),$$

where $r_{1:t}$ is similarly defined and $\delta(\cdot)$ is the Dirac delta function. $\hat{p}_\theta$ is a discrete weighted approximation to the target distribution p_θ. Apparently, the accuracy of the approximation can be improved if an increasing number of particles are included. Doing so, however, also dramatically raises the computational burden.

To obtain the weights, one resorts to importance sampling. That is, one samples N times from a candidate distribution, say $q_\theta(x_{1:t}|r_{1:t})$, and assign the weight

$$w_t^{(i)} \propto p_\theta \left(x_{1:t}^{(i)}|r_{1:t}\right) / q_\theta \left(x_{1:t}^{(i)}|r_{1:t}\right)$$

to each sample drawn. In practice, it is hard, if not impossible, to pick up a proper importance density for the joint distribution of $x_{1:t}$ conditional on the data when sample size is large. Hence, this approach usually proceeds in a sequential fashion. Specifically, the importance density is chosen to admit the factorization such that

$$q_\theta(x_{1:t}|r_{1:t}) = q_\theta(x_t|x_{t-1}, r_t) q_\theta(x_{1:t-1}|r_{1:t-1}).$$

For any existing weighted sample $\{x_{1:t-1}^{(i)}, w_{1:t-1}^{(i)}\}$ that follows from $p_\theta(x_{1:t-1}|r_{1:t-1})$, we augment it with the new state $x_t^{(i)}$ randomly drawn from $q_\theta(x_t|x_{t-1}, r_t)$. The joint sample, $(x_{t-1}^{(i)}, x_t^{(i)})$ is then a realization from the targeted joint importance density. The corresponding weight for i^{th} sample can easily be updated through

$$\tilde{w}_t^{(i)} \propto w_{t-1}^{(i)} \frac{f_\theta(r_t|x_t^{(i)}) g_\theta(x_t^{(i)}|x_{t-1}^{(i)})}{q_\theta(x_t^{(i)}|x_{t-1}^{(i)}, r_t)},$$

and normalized to be $w_t^{(i)} = \frac{1}{N} \sum_{i=1}^{N} \tilde{w}_t^{(i)}$. An unavoidable problem of this procedure, known

as degeneracy, is that after a few iterations, only one particle has a non-negligible weight, which means a large computational cost is spent on particles with almost no contribution. To alleviate this problem, a resampling step is necessary. An important by-product of this filtering strategy is an approximation to $p_\theta(r_{1:t}|r_{1:t-1})$, which has a simple formula $\hat{p}_\theta(r_{1:t}|r_{1:t-1}) = \frac{1}{N}\sum_{i=1}^{N} w_t^{(i)}$. The likelihood can then be easily obtained as $\hat{p}_\theta(r_{1:T}) = \hat{p}_\theta(r_1)\prod_{t=2}^{T}\hat{p}_\theta(r_{1:t}|r_{1:t-1})$.

One subtlety to note is that, to ensure the targeted joint density is indeed the invariant distribution of a Markov chain, we have to modify the particle filter when applying PG. Specifically, one particle trajectory must be specified a priori to serve as a reference. This modified version is known in the literature as conditional particle filter. The intuition is that this particular path can guide the simulated particles to move within a relevant region of the state space. See Theorem 5 of Andrieu et al. (2010) for more details.

D Details of PGAS algorithm

Consider a state-space model in the form of model (C.17). The output of a PGAS algorithm is a random draw from the joint distribution $p_\theta(x_{1:T}|r_{1:T})$, conditional on one particular set of parameter values. In the following, we omit parameters in all densities with an understanding that they depend on the same θ. The input of this algorithm, except for θ, is a reference trajectory of $x_{1:T}$, which is a sample from the last MCMC iteration. Let us denote that reference trajectory by $x'_{1:T}$. Then, the algorithm proceeds as following:

1. Draw $x_1^{(i)}$ from $q_1(x_1|r_1)$, for $i = 1, 2, ..., N-1$.
2. Set $x_1^{(N)} = x'_1$.
3. Set $w_1^{(i)} = f(r_1|x_1^{(i)})/q_1(x_1^{(i)}|r_1)$, for $i = 1, 2, ..., N$.
4. For $t = 2$ to T, do the following:
   (a) Generate $\{\tilde{x}_{1:t-1}^{(i)}\}_{i=1}^{N-1}$ by sampling with replacement $N-1$ times from $\{x_{1:t-1}^{(i)}\}_{i=1}^{N}$ with probabilities proportional to the importance weights $\{w_{t-1}^{(i)}\}_{i=1}^{N}$.
   (b) Draw J from $\{1, 2, ..., N\}$ with probabilities proportional to $w_{t-1}^{(i)} g(x'_t|x_{t-1}^{(i)})$ and then set $\tilde{x}_{1:t-1}^{(N)} = x_{1:t-1}^{(J)}$.
   (c) Simulate $x_t^{(i)}$ from $q_t(x_t|\tilde{x}_{t-1}^{(i)}, r_t)$, for $i = 1, 2, ..., N-1$.

(d) Set $x_t^{(N)} = x'_t$.

(e) Set $x_{i:t}^{(i)} = (\tilde{x}_{1:t-1}^{(i)}, x_t^{(i)})$.

(f) Set weight to be $w_t^{(i)} = f(r_t|x_t^{(i)})g(x_t^{(i)}|\tilde{x}_{t-1}^{(i)})/q_t(x_t^{(i)}|\tilde{x}_{t-1}^{(i)}, r_t)$, for $i = 1, 2, ..., N$.

5. Draw k from $\{1, 2, ..., N\}$ with probabilities proportional to $w_T^{(i)}$ and return $x_{1:T}^* = x_{1:T}^{(k)}$.

Note that this procedure is very similar to the original PG sampler. A major modification is in drawing J, where a new index is drawn and thus the N^{th} trajectory may not be the reference one from the last iteration. In the conditional PG, on the contrary, we fix the last particle to follow the input trajectory $x'_{1:T}$. It is also worth mentioning that the probability of drawing J depends on $g(x'_t|x_{t-1}^{(i)})$ and x'_t is drawn in the last iteration conditional on all observations $r_{1:t}$. Therefore, this step makes the algorithm more like a smoothing instead of filtering.

E Details of Sampling Model Parameters of MSV-B-GFT

The joint posterior distribution can be written as

$$
\begin{aligned}
p(\theta, h, q|r) &\propto p(r|\theta, h, q)p(\theta, h, q) \\
&= f(r|h, q)g_\theta(h)g_\theta(q)\pi(\theta) \\
&= f(r_1|h_1, q_1)g_\theta(h_1)g_\theta(q_1)\prod_{t=2}^{T}\left[f(r_t|h_t, q_t)g_\theta(h_t|h_{t-1})g_\theta(q_t|q_{t-1})\right]\pi(\theta) \\
&= \prod_{t=1}^{T}\left[\left(\sum_{i=1}^{n} h_{it}\right)|C_t|^{-1/2}\exp\left[-\frac{1}{2}r'_t\left(V_t^{1/2}C_tV_t^{1/2}\right)^{-1}r_t\right]\right] \qquad \text{(E.18)} \\
&\quad \times\prod_{t=2}^{T}\prod_{i=1}^{n}\left[(\sigma_{hi}^2)^{-1/2}\exp\left(-\frac{1}{2\sigma_{hi}^2}\left(h_{it+1}-\mu_{hi}-\phi_{hi}(h_{it}-\mu_{hi})\right)^2\right)\right] \\
&\quad \times\prod_{t=2}^{T}\prod_{j=1}^{d}\left[(\sigma_{qj}^2)^{-1/2}\exp\left(-\frac{1}{2\sigma_{qj}^2}\left(q_{jt+1}-\mu_{qj}-\phi_{qj}(q_{jt}-\mu_{qj})\right)^2\right)\right] \\
&\quad \times\prod_{i=1}^{n}\left(\frac{\sigma_{hi}^2}{1-\phi_{hi}^2}\right)^{-1/2}\exp\left(-\frac{(h_{i1}-\mu_{h1})^2}{2\sigma_{hi}^2/(1-\phi_{hi}^2)}\right) \\
&\quad \times\prod_{j=1}^{d}\left(\frac{\sigma_{qj}^2}{1-\phi_{qj}^2}\right)^{-1/2}\exp\left(-\frac{(q_{j1}-\mu_{q1})^2}{2\sigma_{qj}^2/(1-\phi_{qj}^2)}\right)\pi(\theta).
\end{aligned}
$$

To sample from the posterior distribution of parameters conditional on the realization of latent variables, we can do the following:

1. We can directly sample from the full conditional distribution of μ_{hi} and μ_{qi} which a normal distribution. For $i = 1, ..., n$ and $j = 1, ..., d$,

$$\mu_{hi}|r, h, q, \theta_{/\mu_{hi}} \sim N\left(\tilde{m}_{h\mu}, \tilde{s}^2_{h\mu}\right) \text{ and } \mu_{qj}|r, h, q, \theta_{/\mu_{qj}} \sim N\left(\tilde{m}_{q\mu}, \tilde{s}^2_{q\mu}\right), \tag{E.19}$$

where

$$\tilde{m}_{h\mu} = \tilde{s}^2_{h\mu}\left\{\frac{1-\phi^2_{hi}}{\sigma^2_{hi}}h_{i1} + \frac{1-\phi_{hi}}{\sigma^2_{hi}}\sum_{t=1}^{T-1}(h_{it+1} - \phi_{hi}h_{it})\right\},$$

$$\tilde{m}_{q\mu} = \tilde{s}^2_{q\mu}\left\{\frac{1-\phi^2_{qj}}{\sigma^2_{qj}}q_{j1} + \frac{1-\phi_{qj}}{\sigma^2_{qj}}\sum_{t=1}^{T-1}(q_{jt+1} - \phi_{qj}q_{jt})\right\},$$

and

$$\tilde{s}^2_{h\mu} = \sigma^2_{hi}\left[(T-1)(1-\phi_{hi})^2 + (1-\phi^2_{hi})\right]^{-1},$$

$$\tilde{s}^2_{q\mu} = \sigma^2_{qj}\left[(T-1)(1-\phi_{qj})^2 + (1-\phi^2_{qj})\right]^{-1}.$$

2. To draw random samples from the full conditional distribution of ϕ_{hi} and ϕ_{qi}, one can resort to the Metropolis-Hasting sampler. Since

$$\begin{aligned}\log p(\phi_{hi}|r, h, q, \theta_{/\phi_{hi}}) &\propto \log p(h_i|\phi_{hi}, \theta_{/\phi_{hi}}) + \log \pi(\phi_{hi}) \\ &= \log \pi(\phi_{hi}) - \frac{(h_{i1} - \mu_{hi})^2(1-\phi^2_{hi})}{2\sigma^2_{hi}} + \frac{1}{2}\log(1-\phi^2_{hi}) \\ &\quad - \frac{\sum_{t=1}^{T-1}\left[(h_{it+1} - \mu_{hi}) - \phi_{hi}(h_{it} - \mu_{hi})\right]^2}{2\sigma^2_{hi}},\end{aligned} \tag{E.20}$$

we draw ϕ^*_{hi} from the proposal normal density $N\left(\hat{\phi}_{hi}, V_{\phi_{hi}}\right)$, where

$$\hat{\phi}_{hi} = \frac{\sum_{t=1}^{T-1}(h_{it+1} - \mu_{hi})(h_{it} - \mu_{hi})}{\sum_{t=1}^{T-1}(h_{it} - \mu_{hi})^2},$$

is the ordinary least square estimator of ϕ_{hi} given h_i and

$$V_{\phi_{hi}} = \sigma^2_{hi}\left[\sum_{t=1}^{T-1}(h_{it} - \mu_{hi})^2\right]^{-1}.$$

Then, the draw is accepted with probability $\min\left[1, \exp\left\{g(\phi_{hi}^*)/g(\phi_{hi}^{(i-1)})\right\}\right]$, where $\phi_{hi}^{(i-1)}$ is the sample from the last MCMC iteration and

$$g(\phi_{hi}) = \log \pi(\phi_{hi}) - \frac{(h_{i1} - \mu_{hi})^2(1-\phi_{hi}^2)}{2\sigma_{hi}^2} + \frac{1}{2}\log(1-\phi_{hi}^2).$$

ϕ_{qi} can be treated in the same fashion.

3. Similar to the case for μ, due to the conjugacy, draws of σ_{hi}^2 can come from an inverse gamma distribution. For $i = 1, ..., n$ and $j = 1, ..., d$,

$$\sigma_{hi}^2 | r, h, q, \theta_{/\sigma_{hi}^2} \sim IG\left(\frac{\tilde{n}_m}{2}, \frac{\tilde{d}_{hm}}{2}\right) \text{ and } \sigma_{qj}^2 | r, h, q, \theta_{/\sigma_{qj}^2} \sim IG\left(\frac{\tilde{n}_m}{2}, \frac{\tilde{d}_{qm}}{2}\right), \tag{E.21}$$

where $\tilde{n}_m = n_{m0} + T$ and

$$\tilde{d}_{hm} = d_{m0} + (h_{i1} - \mu_{hi})^2(1-\phi_{hi}^2) + \sum_{t=1}^{T-1}\left[(h_{it+1} - \mu_{hi}) - \phi_{hi}(h_{it} - \mu_{hi})\right]^2,$$

$$\tilde{d}_{qm} = d_{m0} + (q_{j1} - \mu_{qj})^2(1-\phi_{qj}^2) + \sum_{t=1}^{T-1}\left[(q_{jt+1} - \mu_{qj}) - \phi_{qj}(q_{jt} - \mu_{qj})\right]^2.$$

F Jacobian matrix used in Broyden's method

By stacking the K equations in (10) for $k = 1, \cdots, K$, we obtain that the system of equations to be solved can be cast into matrix form as

$$f(y) \coloneqq \mathcal{N} - \log\Bigg[\underbrace{diag\left(\text{expm}\left(\tilde{A} + diag(y)\right)\right)}_{f_1(y)} + \underbrace{\mathcal{M}\exp\left(y - diag(Q)\right)}_{f_2(y)}\Bigg] = \mathbf{0}_K$$

where log and exp of a vector are understood as element-wise operations and

$$\mathcal{N} = \begin{pmatrix} \log n_1 \\ \vdots \\ \log n_K \end{pmatrix}, \quad \mathcal{M} = \begin{pmatrix} n_1 - 1 & & \\ & \ddots & \\ & & n_K - 1 \end{pmatrix}, \quad \mathbf{0}_K = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}.$$

It is straightforward to see that the Jacobian of function f with respect to the unknowns y can be expressed as

$$\mathcal{J}(y) \coloneqq \frac{\partial f(y)}{\partial y} = \left[diag\Big(f_1(y) + f_2(y) \Big) \right]^{-1} \left[\frac{\partial f_1(y)}{\partial y} + \frac{\partial f_2(y)}{\partial y} \right]$$

and two components in the bracket can be easily derived as

$$\frac{\partial f_1(y)}{\partial y} = E_d \frac{\partial vec(expm(A))}{\partial vec\,(A)} E_d' \tag{F.22}$$

$$\frac{\partial f_2(y)}{\partial y} = f_2(y) \tag{F.23}$$

where $A = \tilde{A} + diag(y)$ and E_d is an elimination matrix defined by $E_d vec(M) = diag(M)$ for any square matrix M. Note that the analytical expression for the partial derivative in (F.22) has been derived in the Linton & McCrorie (1995) and is also used in Archakov & Hansen (2021). In particular, let $A = \Gamma \Lambda \Gamma'$ be the eigenvalue decomposition of A, where Λ is the diagonal matrix with the eigenvalues $\lambda_1, \cdots, \lambda_K$. Then

$$\frac{\partial vec(expm(A))}{\partial vec\,(A)} = (\Gamma \otimes \Gamma')\, \Xi\, (\Gamma \otimes \Gamma')\,,$$

where Ξ is a $K^2 \times K^2$ diagonal matrix with elements given by

$$\Xi_{(i-1)p+j,(i-1)p+j} = \begin{cases} e^{\lambda_i} & \text{if } \lambda_i = \lambda_j \\ \frac{e^{\lambda_i} - e^{\lambda_j}}{\lambda_i - \lambda_j} & \text{if } \lambda_i \neq \lambda_j \end{cases}.$$