Bayesian High-Rank Hankel Matrix Completion for Nonlinear Synchrophasor Data Recovery

Ming Yi, *Student Member, IEEE,* Meng Wang, *Senior Member, IEEE*, Tianqi Hong, *Member, IEEE,* Dongbo Zhao, *Senior Member, IEEE,*

***Abstract*—Phasor measurement units (PMUs) provide high temporal-resolution synchrophasor measurements for power system monitoring and control. The frequent data quality issues, such as missing and bad data, prevent the incorporation of synchrophasor data in real-time operations. Most existing data-driven data recovery methods assume the power system dynamics can be approximated by a linear dynamical system, and the recovery performance degrades significantly when the power system is experiencing nonlinear dynamics during significant events. This paper proposes a data-driven Bayesian nonlinear synchrophasor data recovery method (Ba-NSDR) that can recover a consecutive time period of simultaneous data losses or errors across all channels, even when the underlying system is highly nonlinear. The idea is to lift the Hankel matrix of the spatial-temporal synchrophasor data to a higher dimension such that the lifted Hankel matrix is low-rank in that space and can be processed with the kernel trick. Our proposed Bayesian method then infers the probabilistic distributions of synchrophasor from the partial observations. Some distinctive features of Ba-NSDR include an uncertainty index to measure the accuracy of the recovery result and the robustness to parameter selections. Our method is verified on both synthetic and recorded event datasets.**

***Index Terms*—PMU data recovery, high-rank matrix completion, Bayesian robust matrix completion, kernel method, uncertainty modeling**

I. Introduction

PHasor Measurement Units (PMUs) provide synchronized voltage and current phasor measurements across different locations in the electric power system. With a high sampling rate of thirty or sixty samples per second per channel, synchrophasor data provide great visibility of power system dynamics, which is typically difficult to observe in the supervisory control and data acquisition (SCADA) system. Synchrophasor data have been employed for event classification [1], [2], state estimation [3]–[5] and system identification [6], [7]. Synchrophasor data, however, suffer from quality issues such as missing and bad data, because of various reasons like PMU malfunctions, communication failure, and false data injections. Synchrophasor data usually have missing and bad data issues. The quality issues prevent synchrophasor data from being employed in real-time control operations.

This work was supported in part by the NSF grant # 1932196, AFOSR FA9550-20-1-0122, Electric Power Research Institute (EPRI) and the U.S. Department of Energy Solar Energy Technologies Office (SETO) OEDI project. (Corresponding author: Meng Wang.)

M. Yi, and M. Wang are with the Dept. of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY. Email:{yim3, wangm7}@rpi.edu. T. Hong is with Energy System Division, Argonne National Laboratory, Lemont, IL. Email: thong@anl.gov. D. Zhao is affiliated with Eaton Corporation, Golden, Colorado. Email: dongbozhao@eaton.com.

Various approaches have been developed to handle missing and bad data. The model-based methods utilize a dynamic model [8] to fill the missing data or estimate the dynamic states based on the Kalman filter [9], [10]. The performance critically depends on accurate model estimation. Refs. [11]–[13] train deep neural networks to recover missing data. Refs. [14]–[17] formulate the error correction as a hypothesis testing problem. Ref. [18] exploits spatial-temporal similarities in the synchrophasor measurements to correct bad data. Ref. [19] employs the independent component analysis to obtain the measurement structure and remove the errors. Refs. [20]–[23] exploit the low-rank property of the spatial-temporal PMU data matrix to correct missing and bad data. These data-driven methods, however, cannot handle simultaneous and consecutive data issues across all channels.

When the power system dynamics can be approximated by a linear dynamical system, [24]–[27] exploit the resulting low-rank property of the Hankel matrix of PMU data to recover simultaneous and consecutive data issues. The linear dynamical model, however, becomes inaccurate when the power system is experiencing nonlinear dynamics. To the best of our knowledge, only Ref. [28] considers missing data recovery in nonlinear dynamical systems and proposes a lifted low-rank Hankel property to characterize the data dynamics without explicitly modeling the dynamical system. This approach cannot handle bad data, and its performance is very sensitive to parameter selection. Moreover, the recovery performance drops significantly for long consecutive data loss. One major limitation of most methods mentioned above is that they only provide an estimation of the actual data without any evaluation of the accuracy of the estimation. Only Ref. [27] provides an uncertainty evaluation of the recovered data.

This paper proposes a Bayesian high-rank Hankel matrix recovery method (Ba-NSDR) to recover missing data and correct bad data when the power system exhibits significant nonlinear dynamics. The main idea is to lift the original high-rank Hankel matrix into a higher-dimensional space so that the lifted matrix becomes low-rank. The nonlinear lifting function can be characterized implicitly by the kernel function [29], which has been exploited in high-rank matrix completion [30]–[32]. [33] employs the kernel function to incorporate the prior knowledge into the matrix completion, but it does not consider the nonlinear dynamics. [34] uses the Gaussian process with kernel functions to model the linear time-invariant (LTI) systems and approximate the nonlinear dynamical systems by LTI systems. [34] is not modeless and requires detailed system information. A prior probabilistic distribution is imposed over

the Hankel matrix, and Ba-NSDR computes the approximate posterior distributions using variational inference based on the observed data. Ba-NSDR has multiple distinctive features. First, it can handle simultaneous and consecutive missing/bad data across all PMU channels. When the system is experiencing nonlinear dynamics, the recovery accuracy by Ba-NSDR is much higher than the existing methods. Second, Ba-NSDR returns an uncertainty index that reflects the accuracy of recovered data, while the recovery accuracy of most existing methods cannot be measured without the ground-truth value. Third, Ba-NSDR does not require any prior knowledge of the unknown ground-truth matrix rank and is robust to the initial rank selection. It can effectively estimate the rank from the observed data through pruning from a large rank.

The rest of the paper is organized as follows. The problem formulation, low-rank Hankel property, and low-rank lifted Hankel property of synchrophasor data are described in Section II. The methodology is introduced in Section III. Section IV reports the numerical results. Section V concludes the paper. The derivation details of our method are shown in the supplementary materials.

II. Problem Formulation

Let a matrix Y denote the ground truth of PMU measurements of m channels at different locations during n time instants,

$$\boldsymbol{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_n] \in \mathbb{R}^{m\times n}, \tag{1}$$

where $\boldsymbol{y}_i \in \mathbb{R}^m$ denotes the measurement of m channels at time instant i. Let $\boldsymbol{N} \in \mathbb{R}^{m\times n}$ denote the measurement noise. Let $\boldsymbol{E} \in \mathbb{R}^{m\times n}$ denote the additive bad data. The entries in $\boldsymbol{E}$ can be arbitrarily large, modeling significant bad data. We assume such bad data only happen at a small fraction of measurements, i.e., $\boldsymbol{E}$ is sparse.

Let a matrix $\boldsymbol{Y}^o \in \mathbb{R}^{m\times n}$ denote the observed measurements. Each entry $Y^o_{i,j}$ in the set $\boldsymbol{\Omega}$ of observed entries is given by

$$Y^o_{i,j} = Y_{i,j} + E_{i,j} + N_{i,j} \quad (i,j) \in \boldsymbol{\Omega}, \tag{2}$$

where $\boldsymbol{\Omega}$ denotes the set of observed entries. The unobserved entries in $\boldsymbol{Y}^o$ are irrelevant and set as zeroes for the completeness of the definition.

The objective of this paper is to recover data $\boldsymbol{Y}$ with measurable accuracy from measurements $\boldsymbol{Y}^o$ that are corrupted by missing data, bad data, and noise. This is particularly challenging when the power system is under nonlinear dynamics.

Our proposed Ba-NSDR method exploits the low-rank property of the lifted Hankel matrix of the PMU data in nonlinear dynamical systems. We first introduce the low-rank Hankel property for linear dynamical systems in Section II-A and then generalize to the lifted Hankel matrix for nonlinear dynamical systems in Section II-B. Detailed analyses of low-rank Hankel property can be found in Refs. [24] and [28], respectively.

A. Low-Rank Hankel Property of PMU Data

Let $\mathcal{H}_{n_2}(\boldsymbol{Y}) \in \mathbb{R}^{mn_2\times n_1}$ $(n_1 + n_2 = n + 1)$ denote the Hankel matrix of $\boldsymbol{Y}$, where the jth column of $\mathcal{H}_{n_2}(\boldsymbol{Y})$ includes all the measurements in m channels from time j to $j + n_2 - 1$, i.e.,

$$\mathcal{H}_{n_2}(\boldsymbol{Y}) = \begin{bmatrix} \boldsymbol{y}_1 & \boldsymbol{y}_2 & \cdots & \boldsymbol{y}_{n_1} \\ \boldsymbol{y}_2 & \boldsymbol{y}_3 & \cdots & \boldsymbol{y}_{n_1+1} \\ \vdots & \vdots & \cdots & \vdots \\ \boldsymbol{y}_{n_2} & \boldsymbol{y}_{n_2+1} & \cdots & \boldsymbol{y}_n \end{bmatrix} \in \mathbb{R}^{mn_2\times n_1}. \tag{3}$$

As shown in [24], if the underlying system that produces output $\boldsymbol{y}_1$ to $\boldsymbol{y}_n$ can be approximated by an order-r (integer $r \geq 1$) linear dynamical system, then $\mathcal{H}_{n_2}(\boldsymbol{Y})$ can be approximated by a rank-r matrix. $\mathcal{H}_{n_2}(\boldsymbol{Y})$ is low-rank because r can be much smaller than m and n_1. The rank-r approximation $\mathcal{Q}^r(\mathcal{H}_{n_2}(\boldsymbol{Y}))$ to $\mathcal{H}_{n_2}(\boldsymbol{Y})$ can be computed by

$$\mathcal{Q}^r(\mathcal{H}_{n_2}(\boldsymbol{Y})) = \boldsymbol{A}_1\boldsymbol{S}_1^r{\boldsymbol{B}_1}^T, \tag{4}$$

where $\mathcal{H}_{n_2}(\boldsymbol{Y}) = \boldsymbol{A}_1\boldsymbol{S}_1{\boldsymbol{B}_1}^T$ is the singular value decomposition of $\mathcal{H}_{n_2}(\boldsymbol{Y})$. $\boldsymbol{A}_1$, $\boldsymbol{B}_1$, and $\boldsymbol{S}_1$ represent the left singular vectors, right singular vectors, and singular values, respectively. $\boldsymbol{S}_1^r$ keeps the largest r singular values in $\boldsymbol{S}_1$ and sets all the others to zero. The corresponding normalized approximation error is computed by

$$\frac{||\mathcal{Q}^r(\mathcal{H}_{n_2}(\boldsymbol{Y})) - \mathcal{H}_{n_2}(\boldsymbol{Y})||_F}{||\mathcal{H}_{n_2}(\boldsymbol{Y})||_F} = \frac{||\boldsymbol{S}_1^r - \boldsymbol{S}_1||_F}{||\boldsymbol{S}_1||_F}. \tag{5}$$

where $||.||_F$ represents the Frobenious norm.

B. Low-Rank lifted Hankel Property in Nonlinear Dynamical System

When the underlying system is highly nonlinear such as immediately after a significant event, approximating a nonlinear system using a linear dynamical model usually requires a large order r. Thus, the corresponding $\mathcal{H}_{n_2}(\boldsymbol{Y})$ is no longer low-rank. The idea is to lift the measurements $\boldsymbol{y}_i$ to a higher dimensional space using a mapping function $\phi(\cdot) : \mathbb{R}^m \to \mathbb{R}^M$, where M is much larger than m and can be infinite. As described in [28], there exists a mapping $\phi(\cdot)$ such that the nonlinear dynamical system can be a linear dynamical system in the lifted space. Let $\mathcal{H}_{n_2}(\boldsymbol{Z})$ be

$$\mathcal{H}_{n_2}(\boldsymbol{Z}) = \begin{bmatrix} \boldsymbol{z}_1 & \boldsymbol{z}_2 & \cdots & \boldsymbol{z}_{n_1} \\ \boldsymbol{z}_2 & \boldsymbol{z}_3 & \cdots & \boldsymbol{z}_{n_1+1} \\ \vdots & \vdots & \cdots & \vdots \\ \boldsymbol{z}_{n_2} & \boldsymbol{z}_{n_2+1} & \cdots & \boldsymbol{z}_n \end{bmatrix} \in \mathbb{R}^{Mn_2\times n_1}, \tag{6}$$

where $\boldsymbol{z}_i = \phi(\boldsymbol{y}_i)$. The rank of $\mathcal{H}_{n_2}(\boldsymbol{Z})$ can be smaller than that of $\mathcal{H}_{n_2}(\boldsymbol{Y})$ for a proper ϕ.

The rank-r approximation of $\mathcal{H}_{n_2}(\boldsymbol{Z})$ can be written as

$$\mathcal{Q}^r(\mathcal{H}_{n_2}(\boldsymbol{Z})) = \boldsymbol{A}_2\boldsymbol{S}_2^r{\boldsymbol{B}_2}^T, \tag{7}$$

where $\boldsymbol{S}_2^r$ contains the largest r singular values of $\mathcal{H}_{n_2}(\boldsymbol{Z})$, $\boldsymbol{A}_2$ and $\boldsymbol{B}_2$ contain left and right singular vectors, respectively. The normalized approximation error of $\mathcal{Q}^r(\mathcal{H}_{n_2}(\boldsymbol{Z}))$ to $\mathcal{H}_{n_2}(\boldsymbol{Z})$ can be computed by

$$\frac{||\mathcal{Q}^r(\mathcal{H}_{n_2}(\boldsymbol{Z})) - \mathcal{H}_{n_2}(\boldsymbol{Z})||_F}{||\mathcal{H}_{n_2}(\boldsymbol{Z})||_F} = \sqrt{\frac{\sum_{i=r+1}^{n_1}\sigma_i^2}{\sum_{i=1}^{n_1}\sigma_i^2}}, \tag{8}$$

This article has been accepted for publication in IEEE Transactions on Power Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2023.3254909

where σ_i denotes the ith largest singular value of $\boldsymbol{S}_2$. σ_i cannot be computed directly from the SVD of $\mathcal{H}_{n_2}(\boldsymbol{Z})$ because $\mathcal{H}_{n_2}(\boldsymbol{Z})$ is unknown. Instead, one can compute $\mathcal{H}_{n_2}(\boldsymbol{Z})^T\mathcal{H}_{n_2}(\boldsymbol{Z})$ explicitly using the kernel trick [28] without knowing $\phi(\cdot)$. The (i,j)th entry in $\mathcal{H}_{n_2}(\boldsymbol{Z})^T\mathcal{H}_{n_2}(\boldsymbol{Z})$ is computed by

$$\big(\mathcal{H}_{n_2}(\boldsymbol{Z})^T\mathcal{H}_{n_2}(\boldsymbol{Z})\big)_{i,j} = \begin{bmatrix}\boldsymbol{z}_i & \dots & \boldsymbol{z}_{i+n_2-1}\end{bmatrix}\begin{bmatrix}\boldsymbol{z}_j \\ \vdots \\ \boldsymbol{z}_{j+n_2-1}\end{bmatrix} \tag{9}$$

$$= \sum_{p=0}^{p=n_2-1}\phi(\boldsymbol{y}_i)^T\phi(\boldsymbol{y}_j) = \sum_{p=0}^{p=n_2-1}\mathcal{K}_{YY}(i+p,j+p), \tag{10}$$

where $\boldsymbol{\mathcal{K}}_{YY}$ is the kernel function. The most popular kernel functions are the Gaussian kernel and the polynomial kernel. Reference [32] reports that the matrix completion methods with the Gaussian kernel perform better than the polynomial kernel. The Gaussian kernel corresponds to an infinite dimensional ϕ. We employ the Gaussian kernel as follows,

$$\boldsymbol{\mathcal{K}}_{YY}(i,j) = \phi(\boldsymbol{y}_i)^T\phi(\boldsymbol{y}_j) = \exp(-\frac{1}{2c}||\boldsymbol{y}_i - \boldsymbol{y}_j||_2^2), \tag{11}$$

where c is a pre-defined scalar. One then solves the eigen-decomposition of $\mathcal{H}_{n_2}(\boldsymbol{Z})^T\mathcal{H}_{n_2}(\boldsymbol{Z})$. The eigenvalues of $\mathcal{H}_{n_2}(\boldsymbol{Z})^T\mathcal{H}_{n_2}(\boldsymbol{Z})$ are σ_i^2, i.e.,

$$\mathcal{H}_{n_2}(\boldsymbol{Z})^T\mathcal{H}_{n_2}(\boldsymbol{Z}) = \boldsymbol{B}_2\boldsymbol{S}_2^2\boldsymbol{B}_2^T. \tag{12}$$

Remark. When $\boldsymbol{Y}$ is obtained from a nonlinear dynamical system, to achieve the same normalized low-rank approximation error, it often requires a smaller rank to approximate the lifted Hankel matrix $\mathcal{H}_{n_2}(\boldsymbol{Z})$ (with a properly selected kernel function) than to approximate $\mathcal{H}_{n_2}(\boldsymbol{Y})$ with the same n_2. Therefore, the low-rank lifted Hankel property is more desirable in recovering PMU data in nonlinear dynamics.

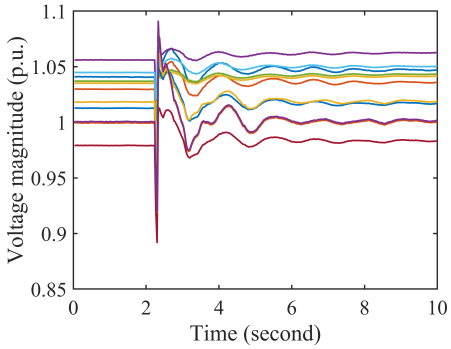


Fig. 1: The measurements of voltage magnitude [24]

To illustrate the low-rank lifted Hankel property, we consider a recorded generator trip event in New York State [24]. Fig. 1 shows the 10 seconds of voltage magnitude measurements in 11 channels at different locations. The data rate is 30 samples per second per channel. Let $\boldsymbol{Y} \in \mathbb{R}^{11\times 300}$ contain all the measurements. Fig. 2 (a) shows the normalized approximation errors of rank-r matrices to $\mathcal{H}_{n_2}(\boldsymbol{Z})$ and $\mathcal{H}_{n_2}(\boldsymbol{Y})$. $c = 200$ in (11). For example, the normalized error of rank-5 approximation to $\mathcal{H}_{10}(\boldsymbol{Z})$ is 0.0015. In comparison, the matrix rank needs to be as least 10 to achieve a similar approximation error to $\mathcal{H}_{10}(\boldsymbol{Y})$. Moreover, with a large n_2, the dimension of $\mathcal{H}_{n_2}(\boldsymbol{Z})$ is very large but could be approximated by a matrix with a small rank. For instance, $\mathcal{H}_{20}(\boldsymbol{Y})$ is in $\mathbb{R}^{220\times 281}$, and $\mathcal{H}_{20}(\boldsymbol{Z})$ is even higher-dimensional due to the lifting. Still, $\mathcal{H}_{20}(\boldsymbol{Z})$ can be approximated by a rank-15 matrix with a normalized error of 0.00083.

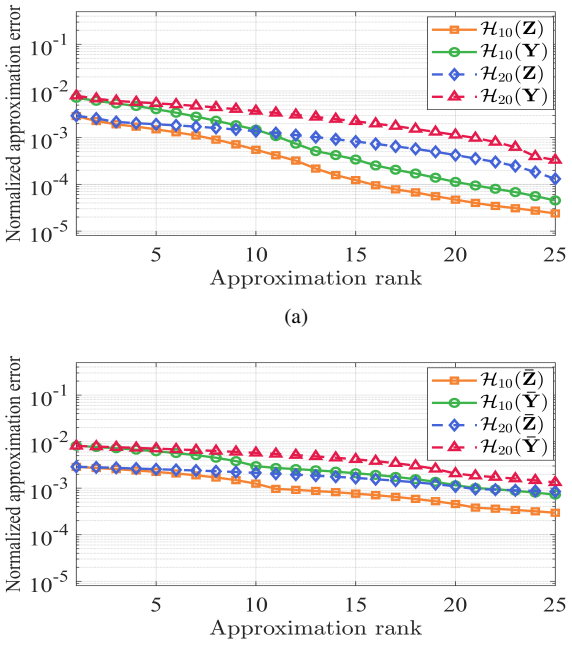


Fig. 2: (a)The normalized approximation errors of the original Hankel matrices $\mathcal{H}_{n_2}(\boldsymbol{Y})$ and the corresponding lifted Hankel matrices $\mathcal{H}_{n_2}(\boldsymbol{Z})$. (b) The normalized approximation errors of column-wise permuted Hankel matrices $\mathcal{H}_{n_2}(\bar{\boldsymbol{Y}})$ and the corresponding lifted Hankel matrices $\mathcal{H}_{n_2}(\bar{\boldsymbol{Z}})$

To illustrate that the low-rank (lifted) Hankel property is special for data from dynamical systems rather than an arbitrary matrix, we permute the columns in $\boldsymbol{Y}$ randomly and let $\bar{\boldsymbol{Y}}$ be the resulting matrix. Then $\boldsymbol{Y}$ and $\bar{\boldsymbol{Y}}$ have the same rank, but each row of $\bar{\boldsymbol{Y}}$ is no longer a time series. Fig. 2 (b) shows the normalized approximation errors of $\mathcal{H}_{n_2}(\bar{\boldsymbol{Y}})$ and $\mathcal{H}_{n_2}(\bar{\boldsymbol{Z}})$, which are Hankel and lifted Hankel matrices constructed from $\bar{\boldsymbol{Y}}$. In contrast to Fig. 2 (a), the approximation errors in Fig. 2 (b) remain significant even when the rank is very large, because the low-rank (lifted) Hankel property does not hold for $\bar{\boldsymbol{Y}}$, which is not obtained from a dynamical system.

III. Bayesian High-Rank Hankel Matrix Recovery (Ba-NSDR) Method

The main idea of our proposed Ba-NSDR method is to estimate a matrix $\boldsymbol{Y}$ from partial observations $\boldsymbol{Y}^o$ such that the

This article has been accepted for publication in IEEE Transactions on Power Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2023.3254909

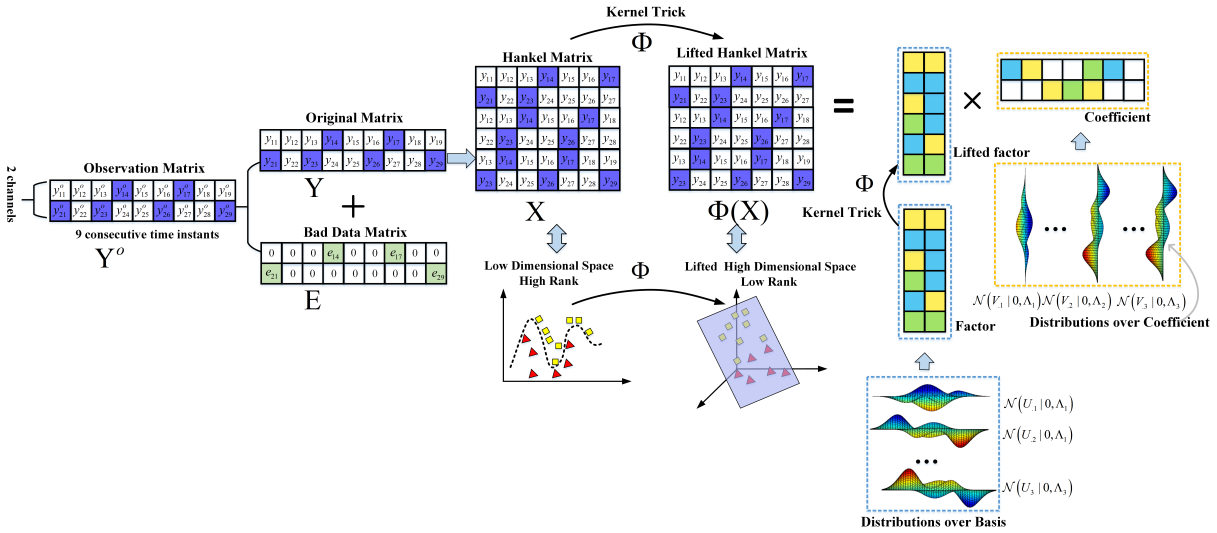


Fig. 3: An overall framework of the proposed method. The method maps the estimated data $\boldsymbol{Y}$ into a Hankel matrix $\boldsymbol{X}$ and then lifts $\boldsymbol{X}$ into higher dimensional space $\Phi(\boldsymbol{X})$. $\Phi(\boldsymbol{X})$ is decomposed with a lifted factor $\Phi(\boldsymbol{U})$, and the coefficient matrix $\boldsymbol{V}$.

lifted Hankel matrix of $\boldsymbol{Y}$ is low-rank. To simplify representation, given n_2, we use $\boldsymbol{X}$ and $\Phi(\boldsymbol{X})$ to denote $\mathcal{H}_{n_2}(\boldsymbol{Y})$ and $\mathcal{H}_{n_2}(\boldsymbol{Z})$, respectively. With a bit of abuse of notation, $\Phi(\boldsymbol{A})$ means dividing each column of the matrix $\boldsymbol{A}$ into multiple vectors in $\mathbb{R}^m$ and lifting each vector to $\mathbb{R}^M$ by the lifting function ϕ. Assuming $\Phi(\boldsymbol{X})$ is rank K, we view $\Phi(\boldsymbol{X})$ as the product of two matrix factors, $\Phi(\boldsymbol{U})$ in $\mathbb{R}^{Mn_2\times K}$ and $\boldsymbol{V}$ in $\mathbb{R}^{K\times n_1}$, where $\Phi(\boldsymbol{U})$ is a lifted matrix to $\mathbb{R}^{Mn_2\times K}$ from a matrix $\boldsymbol{U}$ in $\mathbb{R}^{mn_2\times K}$. Because the rank K is small, the degree of freedom $K(mn_2+n_1)$ is much less than mn, the ambient dimension of $\boldsymbol{Y}$. Therefore, we could accurately recover $\boldsymbol{U}$, $\boldsymbol{V}$, and thus $\boldsymbol{Y}$ from partial observations that contain bad data. Note that every column of $\Phi(\boldsymbol{X})$ includes all data from m channels in n_2 consecutive steps. Then as long as there exist K reliable measurements in all channels in a length-n_2 window, all the remaining measurements in that window can be accurately recovered. Thus, by exploiting the low-rank lifted Hankel property, one can recover data losses/errors in all m channels consecutively.

As a Bayesian approach, Ba-NSDR first imposes a prior distribution on $\boldsymbol{Y}$ and $\Phi(\boldsymbol{X})$ (Section III-A) and then computes the posterior distribution based on partial observations $\boldsymbol{Y}^o$ (Section III-B). Ba-NSDR then uses the posterior distribution of $\boldsymbol{Y}$ to estimate the data and compute the uncertainty index that reflects the estimation accuracy (Section III-C). Section III-D discusses the parameter selection.

A. Proposed Probabilistic Model

Equations (13) to (20) show our hierarchical probabilistic model of the prior distributions. Readers can refer to [35] for prerequisites of the proposed Bayesian model. The latent variables are inferred using observations based on this probabilistic model. Equation (13) is a probabilistic version of equation (2), where $Y_{i,j}$ can be written as the Hankel inverse $(\mathcal{H}^\dagger \boldsymbol{X})_{i,j}$, where the Hankel inverse operator $\mathcal{H}^\dagger$ is defined in equation (35) in the supplementary materials. $\Phi(\boldsymbol{X})_{\cdot q} \in \mathbb{R}^{Mn_2}$ is the qth column of $\Phi(\boldsymbol{X})$. The prior knowledge of rank K might be unavailable. Ba-NSDR sets the initial K as a relatively large number and gradually prunes the basis based on learned coefficients $\boldsymbol{V}$.

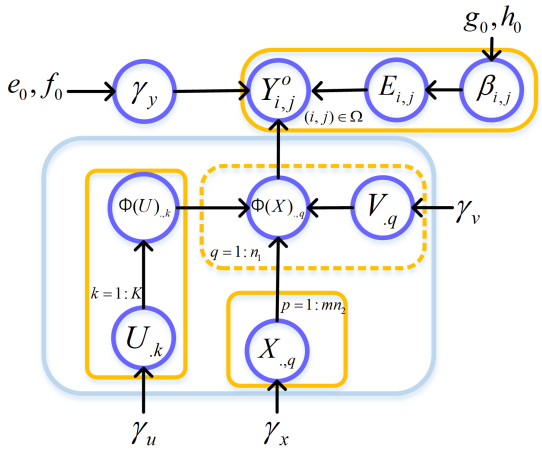


Fig. 4: The Graphical model of the proposed Bayesian high-rank Hankel matrix completion method

The prior distributions of $\boldsymbol{U}_{\cdot k}$, $\boldsymbol{X}_{\cdot q}$, and $\boldsymbol{V}_{\cdot q}$ are drawn from multivariate Gaussian distributions $\mathcal{N}(\boldsymbol{0}, \gamma_u^{-1}\boldsymbol{I}_K)$, $\mathcal{N}(\boldsymbol{0}, \gamma_x^{-1}\boldsymbol{I}_K)$, and $\mathcal{N}(\boldsymbol{0}, \gamma_v^{-1}\boldsymbol{I}_K)$, respectively. $\boldsymbol{I}_{mn_2}$ is an mn_2 by mn_2 identity matrix. γ_u, γ_x, and γ_v are three predefined scalars. Each element in the error matrix $\boldsymbol{E}$ is drawn from a Gaussian distribution $\mathcal{N}(0, \beta_{i,j}^{-1})$. Each element in the noise matrix $\boldsymbol{N}$ is drawn from a Gaussian distribution $\mathcal{N}(0, \gamma_y^{-1})$. The Gamma prior distribution is placed on γ_y and $\beta_{i,j}$, following parameters $(e_0,\ f_0)$ and $(g_0,\ h_0)$, respectively. The mathematical definition of the Gamma distribution is shown in the supplementary material. The conjugate priors are placed on $\boldsymbol{V}_{\cdot q}$, γ_y, $E_{i,j}$, and $\beta_{i,j}$ to derive analytical solutions of posterior distributions. The graphical representation of the proposed probabilistic model is shown in Fig. 4.

For all $q = 1, 2, 3, ..., n_1$, and $k = 1, 2, 3, ..., K$,

$$Y^o_{i,j} \sim \mathcal{N}((\mathcal{H}^\dagger \boldsymbol{X})_{i,j} + E_{i,j}, \frac{1}{\gamma_y}) \quad (i,j) \in \Omega \tag{13}$$

$$\Phi(\boldsymbol{X})_{.q} \sim \mathcal{N}(\Phi(\boldsymbol{U})\boldsymbol{V}_{.q}, \frac{1}{\gamma_\epsilon}\boldsymbol{I}_{mn_2}) \tag{14}$$

$$\boldsymbol{U}_{.k} \sim \mathcal{N}(0, \frac{1}{\gamma_u}\boldsymbol{I}_{mn_2}) \tag{15}$$

$$\boldsymbol{X}_{.q} \sim \mathcal{N}(0, \frac{1}{\gamma_x}\boldsymbol{I}_{mn_2}) \tag{16}$$

$$\boldsymbol{V}_{.q} \sim \mathcal{N}(0, \frac{1}{\gamma_v}\boldsymbol{I}_K) \tag{17}$$

$$\gamma_y \sim \Gamma(e_0, f_0) \tag{18}$$

$$E_{i,j} \sim \mathcal{N}(0, \frac{1}{\beta_{i,j}}) \quad (i,j) \in \Omega \tag{19}$$

$$\beta_{i,j} \sim \Gamma(g_0, h_0) \tag{20}$$

B. Variational Inference for Approximating the Posterior Distributions

To simplify representation, we denote $\boldsymbol{\Theta} = \{\boldsymbol{U}_{.k}, \boldsymbol{V}_{.q}, \boldsymbol{X}_{.q}, \gamma_y, E_{i,j}, \beta_{i,j}, q = 1, 2, 3, ..., n_1, k = 1, 2, 3, ..., K, (i,j) \in \boldsymbol{\Omega}\}$ as the set of all the latent variables. Let $\boldsymbol{\Theta}_i$ denote one arbitrary variable in $\boldsymbol{\Theta}$. Given partial observation $\boldsymbol{Y}^o_\Omega$, the goal is to compute the posterior distribution $p(\boldsymbol{\Theta}, \boldsymbol{Y}|\boldsymbol{Y}^o_\Omega)$. Based on Bayes' theorem,

$$p(\boldsymbol{\Theta}, \boldsymbol{Y}|\boldsymbol{Y}^o_\Omega) = \frac{p(\boldsymbol{\Theta}, \boldsymbol{Y}, \boldsymbol{Y}^o_\Omega)}{p(\boldsymbol{Y}^o_\Omega)}. \tag{21}$$

Computing (21) requires marginalizing out all the latent variables, which is usually intractable.

As a popular approach to approximate the complicated posterior distribution, the mean field variational inference [35] employs a simple distribution $q(\boldsymbol{\Theta})$ to approximate $p(\boldsymbol{\Theta}, \boldsymbol{Y}|\boldsymbol{Y}^o_\Omega)$. The mean field assumption assumes that each element in $\boldsymbol{\Theta}$ is mutually independent. Then $q(\boldsymbol{\Theta})$ can be factorized as the product of each element, i.e.,

$$q(\boldsymbol{\Theta}) = \prod_{k=1}^{K} q(\boldsymbol{U}_{.k}) \prod_{q=1}^{n_1} q(\boldsymbol{V}_{.q}) q(\boldsymbol{X}_{.q}) \prod_{(i,j)\in\boldsymbol{\Omega}} q(E_{i,j}) q(\beta_{i,j}) q(\gamma_y). \tag{22}$$

The best $q(\boldsymbol{\Theta})$ to approximate $p(\boldsymbol{\Theta}, \boldsymbol{Y}_\Omega|\boldsymbol{Y}^o_\Omega)$ is found by minimizing the Kullback–Leibler (KL) divergence, which measures the similarity of two probabilistic distributions. Specifically,

$$\begin{aligned} q(\boldsymbol{\Theta}) &= \arg\min_{q(\boldsymbol{\Theta})} \mathbb{KL}(q(\boldsymbol{\Theta})||p(\boldsymbol{\Theta}, \boldsymbol{Y}|\boldsymbol{Y}^o_\Omega)) \\ &= \arg\max_{q(\boldsymbol{\Theta})} \mathbb{E}[\ln p(\boldsymbol{\Theta}, \boldsymbol{Y}, \boldsymbol{Y}^o_\Omega)] - \mathbb{E}[\ln q(\boldsymbol{\Theta})]. \end{aligned} \tag{23}$$

where $\mathbb{KL}(x||y)$ denotes the KL divergence of distribution x and y, and $\mathbb{E}$ is the expectation over $q(\boldsymbol{\Theta})$. The second equality follows from the definition of KL divergence and removes the term unrelated to $q(\boldsymbol{\Theta})$.

Because it is intractable to solve (23), a typical approach is to optimize each variable $\boldsymbol{\Theta}_i$ in $\boldsymbol{\Theta}$ via solving (23) while keeping all other variables fixed using the most recent distributions.

$$\begin{aligned} &q(\boldsymbol{\Theta}_i) \\ &= \arg\max_{q(\boldsymbol{\Theta}_i)} \Big(\int q(\boldsymbol{\Theta}_i) \mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{\Theta}_i)}[\ln p(\boldsymbol{\Theta}, \boldsymbol{Y}, \boldsymbol{Y}^o_\Omega)] d(\boldsymbol{\Theta}_i) \\ &- \int q(\boldsymbol{\Theta}_i) \ln q(\boldsymbol{\Theta}_i) d\boldsymbol{\Theta}_i \Big) \end{aligned} \tag{24}$$

where $\mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{\Theta}_i)}$ represents that the expectation is taken with respect to all the latent variables excluding $\boldsymbol{\Theta}_i$. These approximate distributions of variational inference finally converge to a local optimum of (23) [35], [36].

Because the conjugate priors are placed on latent variables $\boldsymbol{V}_{.q}$, $E_{i,j}$, $\beta_{i,j}$ and γ_y, (24) has analytical solutions for these variables. Please refer to steps (I), (IV), (V), and (VI) in supplementary materials for the respective updating equations. Because $\boldsymbol{U}_{.k}$ and $\boldsymbol{X}_{.q}$ are lifted to a higher dimensional space via the kernel method, (24) does not have analytical forms for these variables. To solve (24), we assume $\boldsymbol{U}_{.k}$ and $\boldsymbol{X}_{.q}$ are drawn from Gaussian distributions, and then the problem is simplified to find the corresponding mean and the variance of each variable. Then the reparameterization trick [37] is employed to differentiate and optimize the objective in (24) with respect to the mean and variance, respectively. Please refer to steps (II) and (III) in supplementary materials for the updating equations of $\boldsymbol{U}_{.k}$ and $\boldsymbol{X}_{.q}$.

Computing the objective function in (24) for $\boldsymbol{V}_{.q}$, $\boldsymbol{U}_{.k}$ and $\boldsymbol{X}_{.q}$ requires computing the inner product of the lifting function. We employ three Gaussian kernels $\boldsymbol{\mathcal{K}}_{XX}$, $\boldsymbol{\mathcal{K}}_{XU}$ and $\boldsymbol{\mathcal{K}}_{UU}$ in (25)-(27) when updating $\boldsymbol{V}_{.q}$, $\boldsymbol{X}_{.q}$, and $\boldsymbol{U}_{.k}$, respectively.

$$\boldsymbol{\mathcal{K}}_{XX}(p,q) = \Phi(\boldsymbol{X})^T_{.p}\Phi(\boldsymbol{X})_{.q} = \exp(-\tfrac{1}{2c_1}||\boldsymbol{X}_{.p} - \boldsymbol{X}_{.q}||_2^2), \tag{25}$$

$$\boldsymbol{\mathcal{K}}_{XU}(q,k) = \Phi(\boldsymbol{X})^T_{.q}\Phi(\boldsymbol{U})_{.k} = \exp(-\tfrac{1}{2c_2}||\boldsymbol{X}_{.q} - \boldsymbol{U}_{.k}||_2^2), \tag{26}$$

$$\boldsymbol{\mathcal{K}}_{UU}(i,j) = \Phi(\boldsymbol{U})^T_{.i}\Phi(\boldsymbol{U})_{.j} = \exp(-\tfrac{1}{2c_3}||\boldsymbol{U}_{.i} - \boldsymbol{U}_{.j}||_2^2), \tag{27}$$

where c_1, c_2 and c_3 are pre-defined scalars.

Initialization. Each entry in $\boldsymbol{U}$ is initialized from a Gaussian distribution $\mathcal{N}(0,1)$. $\boldsymbol{V}$ is initialized as an all-zero matrix. All the elements in initial variances for $\boldsymbol{U}_{.k}$ and $\boldsymbol{X}_{.q}$ are set as $\exp(-2)$. The initialization $\bar{\boldsymbol{X}}^0$ of $\boldsymbol{X}$ is initialized as the rank-r approximation to $\mathcal{H}_{n_2}(\boldsymbol{Y}^o)$, where the missing entries are set as zero. The initial $\boldsymbol{E}$ is set as $\boldsymbol{Y}^o - \mathcal{P}_{\boldsymbol{\Omega}}(\mathcal{H}^\dagger \bar{\boldsymbol{X}}^0)$. The γ_y is initialized as 10^6.

Estimating the rank of the lifted Hankel matrix. Because the actual rank of $\Phi(\boldsymbol{X})$ is unknown, one selects K that is guaranteed to be larger than the actual rank. The deterministic methods such as [32] require K to be an accurate estimation of the rank and often overfit when K is larger than the actual rank. Here we propose to estimate the rank and remove the redundant factor by thresholding the entries in $\mathbb{E}[\boldsymbol{V}]$. If the sum of absolute values of $\mathbb{E}[V_{kq}]$ for all q is less than a threshold (e.g., 10^{-2}), the algorithm removes the kth column in $\mathbb{E}[\boldsymbol{U}]$, the kth row in $\mathbb{E}[\boldsymbol{V}]$, and reduces the rank K by one. That is because the kth column in $\mathbb{E}[\boldsymbol{U}]$ is not selected to represent $\Phi(\boldsymbol{X})$ and is no longer needed. Therefore, our method is robust to the initial rank and can effectively infer the actual rank.

Estimating the sparsity of the error matrix $\boldsymbol{E}$. Reference [38] shows that the Gaussian distribution with Gamma priors promotes the sparsity of $\boldsymbol{E}$. We can make $\mathbb{E}[\boldsymbol{E}]$ sparser through thresholding, because significant errors do not happen frequently. When entries in $\mathbb{E}[\boldsymbol{E}]$ are very small (e.g., 10^{-1}), the corresponding entries are set as zeroes.

Convergence criteria. Let $\bar{\boldsymbol{X}}^t$ and $\bar{\boldsymbol{X}}^{t-1}$ denote the estimation of $\boldsymbol{X}$ at the tth and $t-1$th iteration, respectively. The algorithm terminates if $\frac{\|\bar{\boldsymbol{X}}^t-\bar{\boldsymbol{X}}^{t-1}\|_F}{\|\bar{\boldsymbol{X}}^{t-1}\|_F} < \xi$ where ξ is a predetermined threshold, or if the maximum iterations $T_{\max}$ is reached.

Missing data recovery only. The algorithm can be simplified when the objective is to recover missing data only, assuming the observations do not contain bad data. Equations (19) and (20) in the prior model characterize the bad data distribution and can be removed. One can also skip steps (IV) and (V) (in the supplementary materials) that update $E_{i,j}$ and $\beta_{i,j}$.

Computational complexity. The computational complexity per iteration is $\mathcal{O}(Lmn_2n_1Kt^{\max})$, where L and $t^{\max}$ are the Monte-Carlo samples and maximum iterations of inner loops, respectively, when computing $\boldsymbol{U}$ and $\boldsymbol{X}$. Thus, the computational complexity scales at most linearly in the size of the Hankel matrix. The details of derivation are provided in Section F in the supplementary materials. Our algorithm is a block processing method and is most suitable for offline data recovery. It could possibly be used for online processing with sufficient computational power.

C. Data Recovery and Uncertainty Index

With the computed posterior distributions, we use the mean of the distribution of $Y_{i,j}$ as an estimate of the corresponding entry in $\boldsymbol{Y}$ for every $i = 1, ..., m$, and $j = 1, ..., n$. The variance of $Y_{i,j}$ is employed to estimate the accuracy of data recovery. Because the mean and variance do not have closed-form solutions, the Monte Carlo integration [39] is employed to compute them approximately. The predictive mean is derived as follows:

$$\mathbb{E}[Y_{i,j}] \approx \frac{1}{J}\sum_{l=1}^{J}(\mathcal{H}^\dagger \boldsymbol{X}^{(l)})_{i,j} \quad \boldsymbol{X}^{(l)} \sim q(\boldsymbol{X}|\boldsymbol{Y}_\Omega^o), \tag{28}$$

where J is the number of Monte-Carlo samples. Each $\boldsymbol{X}^{(l)}$ is sampled from learned posterior distributions. The predictive variance is computed by:

$$\begin{aligned}\text{Var}[Y_{i,j}] &= \mathbb{E}[Y_{i,j}^2] - \mathbb{E}[Y_{i,j}]^2 \\ &\approx \frac{1}{J}\sum_{l=1}^{J}\frac{1}{\gamma_y^{(l)}} + \frac{1}{J}\sum_{l=1}^{J}(\mathcal{H}^\dagger \boldsymbol{X}^{(l)})_{i,j}^2 - (\frac{1}{J}\sum_{l=1}^{J}(\mathcal{H}^\dagger \boldsymbol{X}^{(l)})_{i,j})^2,\end{aligned} \tag{29}$$

where each $\gamma_y^{(l)}$ is sampled from learned posterior distribution $q(\gamma_y|\boldsymbol{Y}_\Omega^o)$. We use the average variance as an uncertainty index of the data estimation, i.e.,

$$U_{\text{index}} = (\sum_{i=1}^{m}\sum_{j=1}^{n}\text{Var}[Y_{i,j}])/(mn). \tag{30}$$

A higher average variance leads to a larger uncertainty index. That means the algorithm is less confident about the recovery results.

D. Parameter Selection

The prior distributions (18) and (20) require setting parameters (e_0, f_0) and (g_0, h_0). When e_0 is fixed, a larger f_0 corresponds to a smaller γ_y, which in turn increases the variance $1/\gamma_y$ of the noise $\boldsymbol{N}$. When h_0 is fixed, a larger g_0 corresponds to a larger $\beta_{i,j}$, which in turn decreases the value of $E_{i,j}$. Note that (e_0, f_0) and (g_0, h_0) have a minor impact on the recovery results. Another important parameter is the Hankel size n_2. With a larger n_2, the method can recover consecutive data losses and errors for all channels for a longer time window (close to n_2 time steps). On the other hand, increasing n_2 leads to a higher computational cost. In our experiments, setting n_2 as at most 80 is sufficient to obtain accurate recovery performance. In Section IV-C3, we show that Ba-NSDR is not sensitive to these parameter selections.

IV. Numerical Experiments

A. Experimental Setup

We compare our proposed Ba-NSDR approach with the following nine methods: the Bayesian robust Hankel matrix completion method (BRHMC) in [27], the Bayesian robust Hankel matrix completion method employing the sliding window (BRHMC-S), the Bayesian Hankel matrix completion method (BHMC) in [27], the Bayesian Hankel matrix completion method employing the sliding window (BHMC-S), the deterministic kernel-based matrix completion method (KMC) in [32], the deterministic Hankel matrix completion method (AM-FIHT) in [25], the deterministic robust Hankel matrix completion method (SAP) in [26], the deterministic streaming data recovery method (SDR) in [40], the deterministic streaming data recovery method considering the nonlinear dynamics (SDR-K) in [28], The streaming methods "SDR" and "SDR-K" require that the observations in the first time window contain no missing and bad data, which is one disadvantage compared with offline methods. In the following experiments, we do not include missing and bad data in the first time window of these two methods to make a fair comparison.

Some parameters of Ba-NSDR are set as follows for all experiments if not otherwise stated: $\gamma_\epsilon = 10^5$, $\gamma_v = 10^2$, $J = 50$, $L = 1$, $\gamma_x = \gamma_u = 1$, $e_0 = 10^{-6}$, $f_0 = 10^{-4}$, $g_0 = 1$, $h_0 = 10^{-6}$, $\xi = 10^{-4}$. $\lambda_1 = 10$, $\lambda_2 = \lambda_4 = 0.1$, $\lambda_3 = 1$. $T_{\max} = 100$. $t_1^{\max} = t_2^{\max} = t_3^{\max} = t_4^{\max} = 100$. The experiments are conducted on Matlab 2019 with a desktop with 3.1 GHz Intel i9-9900 and 32 GB memory.

Fig. 5 shows three modes of missing/bad data considered in this experiment. For example, M3 represents Mode 3 of missing data. B2 represents Mode 2 of bad data.

- Mode 1: Missing/bad entries independently and randomly distribute across all channels and time instants.
- Mode 2: Missing/bad entries distribute across all channels, and the time instants are randomly selected.

This article has been accepted for publication in IEEE Transactions on Power Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2023.3254909

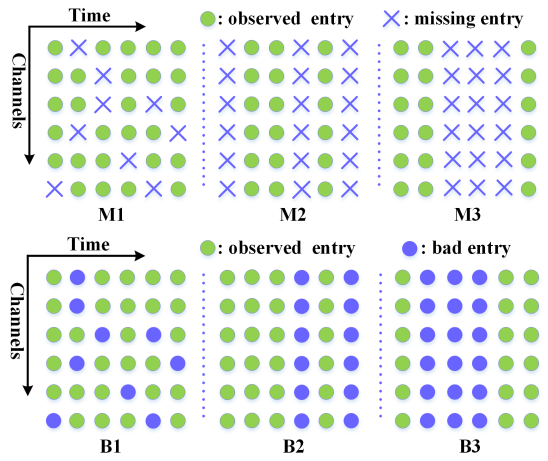


Fig. 5: Three modes of missing/bad data generation. "M" stands for missing data. "B" stands for bad data.

- Mode 3: Missing/bad entries distribute across all channels. The time instants are consecutive instants and the starting instant is randomly selected.

Evaluation Metrics: The Normalized Estimation Error (NEE) is employed to evaluate the data recovery performance. The NEE is defined as

$$\text{NEE} = \|\hat{\boldsymbol{Y}} - \boldsymbol{Y}\|_F / \|\boldsymbol{Y}\|_F, \tag{31}$$

where $\hat{\boldsymbol{Y}}$ in $\mathbb{R}^{m\times n}$ is the estimate and $\boldsymbol{Y}$ in $\mathbb{R}^{m\times n}$ is the ground-truth data. Note that the computation of NEE requires ground-truth data and can only be used for evaluation. When there is no ground-truth data provided, the uncertainty index reflects the estimation accuracy. We will report both the uncertainty index and NEE in the following experiments.

B. Performance on Synthetic Datasets

1) Dataset generation: We first evaluated the data recovery performance on synthetic data where each row of $\boldsymbol{Y}$ is a weighted sum of r time-varying damping noisy sinusoids. Each entry $Y_{i,j}$ in $\boldsymbol{Y}$ is generated by

$$Y_{i,j} = \sum_{k=1}^{r} b_{k,j} e^{-a_i t_j} \sin(2\pi f_{k,j} t_j) \quad i = 1, ..., m, j = 1, ..., n, \tag{32}$$

where $f_{k,j}$ is the time-varying frequency, $b_{k,j}$ is the time-varying amplitude of the kth sinusoid. The general form of time-varying frequency and amplitude can characterize the dynamic transitions during a significant disturbance in power systems. The frequency $f_{k,j}$ is randomly selected from $(100, 102)$. The amplitude $b_{j,k}$ is randomly selected from $(1, 1.3)$. $r = 2$, $a_1 = 30, a_2 = 40, a_3 = 35$. The generated matrix $\boldsymbol{Y}$ has three rows and 300 columns. Fig. 7 shows the normalized approximation errors of rank-r matrices to the Hankel matrix $\mathcal{H}_{n_2}(\boldsymbol{Y})$ and the lifted Hankel matrix $\mathcal{H}_{n_2}(\boldsymbol{Z})$. $c = 200$ in (11). One can see that it requires a much smaller rank to approximate $\mathcal{H}_{n_2}(\boldsymbol{Z})$ than $\mathcal{H}_{n_2}(\boldsymbol{Y})$ with the same normalized approximation error. For example, a rank-2 approximation to $\mathcal{H}_{10}(\boldsymbol{Z})$ is 0.019, while it requires at least rank-27 to achieve a similar error to approximate $\mathcal{H}_{10}(\boldsymbol{Y})$.

Table I: The recovery error and the uncertainty index by Ba-NSDR on M2 missing data of synthetic data

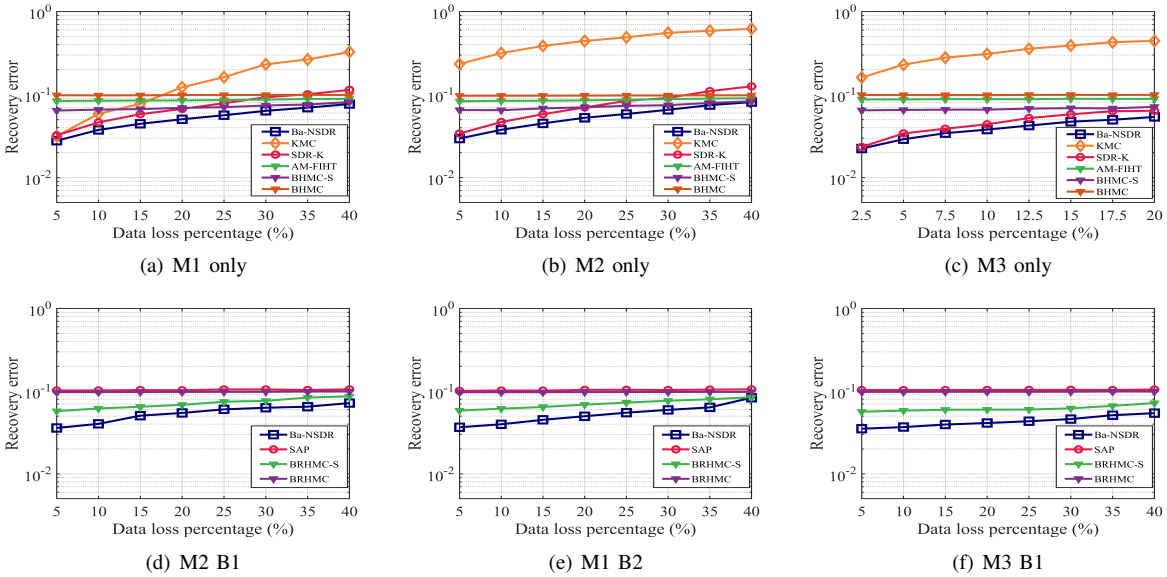
Missing rate %	5	15	25	35
NEE	0.028	0.044	0.057	0.071
U_{index}	1.2×10^{-3}	1.6×10^{-3}	2.0×10^{-3}	2.9×10^{-3}
Missing rate %	45	55	65	
NEE	0.10	0.28	0.54	
U_{index}	3.8×10^{-3}	7.2×10^{-3}	8.3×10^{-2}	

We used a simple signal with nonlinear dynamics in (32) to verify the performance of our algorithm. The signals in (32) simulate the nonlinear dynamics from a nonlinear dynamical system. As stated in reference [41], a linear dynamical system should hold homogeneity property and additive property at the same time. Therefore, if an input is a sinusoidal signal $x(t) = \sin(2\pi f t)$, where f is the frequency and t is the time instant, the output of a linear dynamical system should be $y(t) = A\sin(2\pi f t + \alpha)$, where A is a scaling amplitude and A is a scalar, and α is the time-shifting phase. Because the amplitude in (32) is time-varying, the resulting signals are not generated from a linear dynamical system but from a nonlinear system.

2) Recovery performance: Some parameters of Ba-NSDR are: $c_2 = c_3 = 200$, $\xi = 10^{-4}$, $K = 50$, $T_{\max} = 150$. $n_2 = 20$ for all cases except that $n_2 = 30$ for M3 missing mode (Figs. 6 (c)(f)). The results are averaged over 10 trails. Figs. 6 (a)-(c) compare the missing data recovery performance of Ba-NSDR with KMC, SDR-K, AM-FIHT, BHMC-S, and BHMC on three missing data modes. Ba-NSDR achieves the lowest recovery error among all the methods. Specifically, the conventional kernel-based method KMC does not consider the Hankel structure and, thus, performs poorly on M2 and M3 modes. Deterministic Hankel-based method AM-FIHT and Bayesian Hankel-based methods BHMC, BHMC-S, approximate the data generated from nonlinear dynamical systems using linear dynamical systems and, thus, cannot accurately recover the highly nonlinear components. SDR-K employs the low-rank lifted Hankel property to characterize nonlinear dynamics and performs better than all other methods except our method Ba-NSDR. SDR-K does not provide any uncertainty index and cannot handle bad data. Moreover, SDR-K is sensitive to parameter selections, especially the selection of rank. Table I shows the NEE and the corresponding uncertainty indices when the missing data follow M2 mode. The uncertain index increases when the recovery error increases. This indicates that the uncertainty index is able to differentiate reliable estimations from unreliable estimations.

Figs. 6 (d)-(f) compare the data recovery performance of Ba-NSDR with SAP, BRHMC-S and BRHMC when data contain both missing and bad data. Except for Ba-NSDR, all other methods do not characterize nonlinear dynamics. One can see from Fig. 6 (d)-(f) that Ba-NSDR performs the best among all the methods. Note that the signal in (32) does not include the phase for simplicity. We also tested the performance of our algorithm on a sinusoid with a time-varying phase, and the recovery results are shown in Fig. 11 in supplementary materials. Our method achieves similar performance as the results in Fig. 6.

This article has been accepted for publication in IEEE Transactions on Power Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2023.3254909



(a) M1 only (b) M2 only (c) M3 only

(d) M2 B1 (e) M1 B2 (f) M3 B1

Fig. 6: Comparison of Ba-NSDR with other methods. (a)-(c) show the missing data recovery results with three missing modes. (d)-(f) show the recovery results with both missing and bad data.

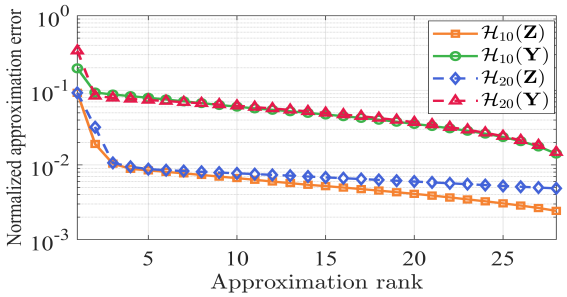


Fig. 7: Low-rank approximations to Hankel matrices $\mathcal{H}_{n_2}(\boldsymbol{Y})$ and the corresponding lifted Hankel matrices $\mathcal{H}_{n_2}(\boldsymbol{Z})$ for synthetic data.

Table II: The recovery performance of recorded PMU data on 6.7% M3 mode

Method	Ba-NSDR	BHMC	BHMC-S	AM-FIHT	SDR-K
NEE	8.3×10^{-4}	5.6×10^{-3}	3.0×10^{-3}	6.0×10^{-3}	2.1×10^{-3}
NEE_{2-4}	1.9×10^{-3}	1.2×10^{-2}	6.6×10^{-3}	1.3×10^{-2}	4.7×10^{-3}
Time(sec.)	28.5	13.1	281.5	0.30	0.45

Table III: The recovery performance of recorded PMU data on 5% M1 and 3.7% B3 mode

Method	Ba-NSDR	BRHMC	BRHMC-S	SAP	SDR
NEE	9.8×10^{-4}	7.1×10^{-3}	3.6×10^{-3}	6.0×10^{-3}	5.6×10^{-3}
NEE_{2-4}	1.9×10^{-3}	1.5×10^{-2}	7.9×10^{-3}	1.3×10^{-2}	1.2×10^{-2}
Time(sec.)	19.5	2.2	276.8	0.054	0.13

C. Performance on practical PMU dataset

We then conducted the experiments on the recorded dataset as shown in Fig. 1 in Central New York Power System[1]. The PMU data type is voltage in rectangular coordinates. The proposed method can also be easily extended to other data types such as current and frequency. Observations in all channels are available in this 10-second window and are treated as ground-truth data. We remove some data points and add bad data following different patterns. The recovered data are evaluated by comparing them with the ground-truth data.

1) Recovery performance: We first evaluated our method on two case studies.

- Case 1: 6.7% data are removed following Mode M3. The length of M3 missing data is 20 consecutive time instants, which correspond to 0.67 seconds.
- Case 2: 5% data are removed following Mode M1 and 3.7% bad data following Mode B3 are added. The length of B3 bad data is 10 consecutive time instants, which correspond to 0.33 seconds. The bad data is randomly sampled from $(0.1, 0.4)$.

The parameter setting of Ba-NSDR is as follows. The initial rank is set as 10. $n_2 = 30$, $c_2 = c_3 = 40$, $f_0 = 10^{-6}$ in Case 1. $n_2 = 80$, $n_2 = 5$, $c_2 = c_3 = 7$, $f_0 = 10^{-4}$ in Case 2.

Figs. 8 and 9 compare the recovery performance of Ba-NSDR with other methods on Case 1 and Case 2, respectively. Ba-NSDR can accurately recover the nonlinear dynamics during the event and clearly outperform all the existing methods. Tables II and III report the NEE over the whole ten-second window, the NEE of a window between 2-4 seconds where missing data occur, denoted by NEE_{2-4}, and the computational time of these methods over the whole ten-second window. Ba-NSDR achieves a great balance of recovery accuracy and computational cost. AM-FIHT, SAP, SDR, and SDR-K are computationally efficient, but their recovery performances are worse than our method. BHMC-S and BRHMC-S truncate the data into small windows and approximate each window using low-rank Hankel matrices and thus are much more computationally expensive than other methods.

The major disadvantage of our method is that it is more computationally expensive than the deterministic low-rank Hankel methods. However, we can see from Tables II and III that the proposed Ba-NSDR method achieves a great balance of recovery accuracy and computational cost. Moreover, the

[1]We provide an additional case study on the recorded PMU data of a transformer failure event in Central New York in the supplementary materials.

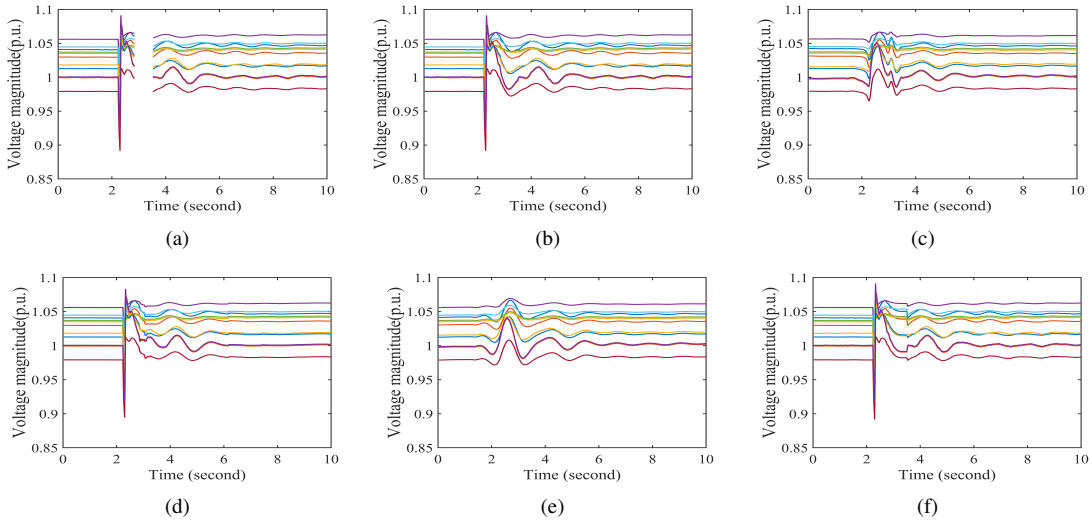


Fig. 8: The recovery performance on 6.7% M3 missing data. (a) the observed data, (b) the estimated data by the proposed Ba-NSDR method, (c) the estimated data by the BHMC method, (d) the estimated data by the BHMC-S method, (e) the estimated data by the AM-FIHT method, (f) the estimated data by the SDR-K method.

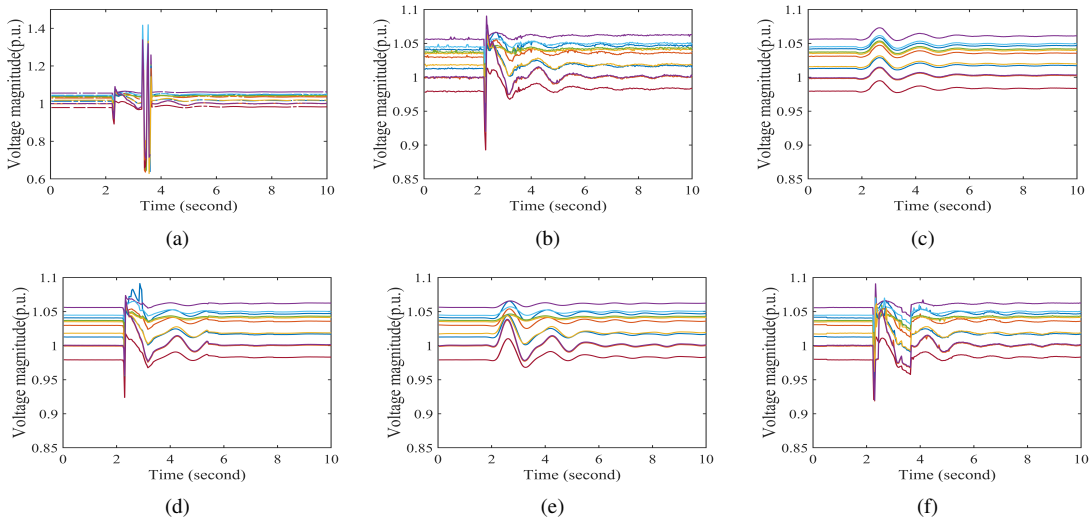


Fig. 9: The recovery performance on 5% M1 missing data and 3.7%B3 bad data. (a) the observed data, (b) the estimated data by the proposed Ba-NSDR method, (c) the estimated data by the BRHMC method, (d) the estimated data by the BRHMC-S method, (e) the estimated data by the SAP method, (f) the estimated data by the SDR method.

proposed probabilistic framework is able to model the uncertainty of the recovery results, while other works cannot provide such an uncertainty index.

2) Uncertainty modeling: One major advantage of Ba-NSDR over existing PMU data recovery methods is that it provides an uncertainty index, which can be employed to evaluate the reliability of the recovery results. Table IV shows the recovery performance and the corresponding uncertainty index on 5% B1 with varying missing data percentages of mode M2. Table V shows the recovery performance and the corresponding uncertainty index on 5% M2 with varying bad data percentages of B1. One can see that the recovery error and uncertainty index increase when the missing/bad data percentage increases. In Table IV, the recovery error is large when the missing data percentage is 45%, and the corresponding uncertainty index is significantly larger than the values at other missing data percentages when the recovery errors are small. This verifies the effectiveness of our proposed uncertainty index.

Table IV: The recovery error and the uncertainty index on 5% B1 with varying missing data percentage of M2

Missing rate	5	15	25	35	45
NEE	0.0019	0.0037	0.0057	0.0060	0.18
U_{index}	2.6×10^{-5}	4.8×10^{-5}	1.5×10^{-4}	4.5×10^{-4}	1.1×10^{-2}

Table V: The recovery error and the uncertainty index on 5% M2 with varying bad data percentage of B1

Bad rate	5	15	25	35	45
NEE	0.0019	0.0091	0.016	0.017	0.032
U_{index}	2.6×10^{-5}	5.8×10^{-5}	7.0×10^{-5}	8.3×10^{-4}	6.2×10^{-3}

3) The impact of parameter selections: We evaluated the impact of parameter selections in recovering 5% M2 missing and 5%B2 bad data. The bad data are randomly selected from $(0.3, 0.5)$. As discussed in Section III-D, we fixed e_0 and vary f_0 to show the impact of (e_0, f_0). One can see from Table VI that Ba-NSDR maintains a very small recovery error with a wide range of f_0. We also fixed h_0 and varied g_0 to show the impact of (g_0, h_0). Table VII shows that Ba-NSDR is not sensitive to the selection of g_0.

Table VIII shows the recovery performance when the initial rank varies. The recovery error NEE of Ba-NSDR remains very small with different ranks. Moreover, the estimated final ranks are consistent and much smaller than the initial rank, indicating that Ba-NSDR prunes the rank effectively.

The Hankel parameter n_2 is increased from 1 to 25 and the results are shown in Table IX. When $n_2 = 1$, the Hankel matrix reduces to the original data matrix. One can see from Table IX that increasing n_2 indeed leads to more accurate recovery results.

Table X shows the performance when the Gaussian kernel parameters c_2 and c_3 increase. The numerical results indicate that the proposed method is not sensitive to the Gaussian kernel parameters c_2 and c_3.

Table VI: The impact of f_0 (e_0 is fixed and $e_0 = 10^{-6}$)

f_0	10^{-6}	10^{-5}	10^{-3}	10^{-3}	10^{-2}	10^{-1}
NEE	9.4×10^{-4}	1.3×10^{-3}	1.1×10^{-3}	1.7×10^{-3}	2.3×10^{-3}	3.9×10^{-3}

Table VII: The impact of g_0 (h_0 is fixed and $h_0 = 10^{-3}$)

g_0	10^{-6}	10^{-5}	10^{-3}	10^{-3}	10^{-2}	10^{-1}
NEE	1.4×10^{-3}	1.7×10^{-3}	1.4×10^{-3}	1.1×10^{-3}	1.4×10^{-3}	1.1×10^{-3}

Table VIII: The impact of the initial K

Initial rank K	10	15	20	25	30
NEE	1.1×10^{-3}	1.1×10^{-3}	1.1×10^{-3}	1.4×10^{-3}	1.1×10^{-3}
estimated rank	8	9	11	12	13

Table IX: The impact of Hankel parameter n_2

n_2	1	5	10	15	20	25
NEE	0.075	6.5×10^{-3}	3.0×10^{-3}	1.4×10^{-3}	1.1×10^{-3}	1.2×10^{-3}

Table X: The impact of Gaussian kernel parameter $c_2 = c_3$

$c_2 = c_3$	40	50	60	70	80
NEE	1.3×10^{-3}	1.2×10^{-3}	1.1×10^{-3}	1.4×10^{-3}	1.4×10^{-3}

V. Conclusions

This paper proposes a Bayesian high-rank Hankel matrix recovery (Ba-NSDR) method to recover the synchrophasor measurements with missing and bad data. The proposed method maps the constructed Hankel matrix into a higher dimensional space by employing the kernel method and exploits the lifted low-rank Hankel property in recovering synchrophasor data under significant nonlinear dynamics. Ba-NSDR clearly outperforms the existing methods, especially when the data contain long consecutive missing or bad data. The distinctive features of Ba-NSDR include an uncertainty index that reflects the reliability of recovery results and the robustness to the initial rank selection. One future direction is to explore the effect of different kernels so that the method can pick the best kernel automatically for different scenarios.

References

[1] W. Li, M. Wang, and J. H. Chow, "Real-time event identification through low-dimensional subspace characterization of high-dimensional synchrophasor data," *IEEE Trans. Power Syst.*, vol. 33, no. 5, pp. 4937–4947, Jan. 2018.

[2] W. Li and M. Wang, "Identifying overlapping successive events using a shallow convolutional neural network," *IEEE Trans. Power Syst.*, vol. 34, no. 6, pp. 4762–4772, Nov. 2019.

[3] J. Zhao, G. Zhang, K. Das, G. N. Korres, N. M. Manousakis, A. K. Sinha, and Z. He, "Power system real-time monitoring by using pmu-based robust state estimation method," *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 300–309, 2015.

[4] F. Aminifar, M. Shahidehpour, M. Fotuhi-Firuzabad, and S. Kamalinia, "Power system dynamic state estimation with synchronized phasor measurements," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 2, pp. 352–363, 2013.

[5] A. S. Dobakhshari, M. Abdolmaleki, V. Terzija, and S. Azizi, "Robust hybrid linear state estimator utilizing scada and pmu measurements," *IEEE Transactions on Power Systems*, vol. 36, no. 2, pp. 1264–1273, 2020.

[6] I. Kamwa and L. Gerin-Lajoie, "State-space system identification-toward mimo models for modal analysis and optimization of bulk power systems," *IEEE Transactions on Power Systems*, vol. 15, no. 1, pp. 326–335, 2000.

[7] N. Zhou, J. W. Pierre, and J. F. Hauer, "Initial results in power system identification from injected probing signals using a subspace method," *IEEE Transactions on Power Systems*, vol. 21, no. 3, pp. 1296–1302, 2006.

[8] B. Foggo and N. Yu, "Online pmu missing value replacement via event-participation decomposition," *IEEE Transactions on Power Systems*, pp. 1–1, 2021.

[9] N. Zhou, D. Meng, Z. Huang, and G. Welch, "Dynamic state estimation of a synchronous machine using pmu data: A comparative study," *IEEE Transactions on Smart Grid*, vol. 6, no. 1, pp. 450–460, 2014.

[10] K. D. Jones, A. Pal, and J. S. Thorp, "Methodology for performing synchrophasor data conditioning and validation," *IEEE Transactions on Power Systems*, vol. 30, no. 3, pp. 1121–1130, 2014.

[11] J. James, A. Y. Lam, D. J. Hill, Y. Hou, and V. O. Li, "Delay aware power system synchrophasor recovery and prediction framework," *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 3732–3742, 2018.

[12] J. James, D. J. Hill, V. O. Li, and Y. Hou, "Synchrophasor recovery and prediction: A graph-based deep learning approach," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 7348–7359, 2019.

[13] C. Ren and Y. Xu, "A fully data-driven method based on generative adversarial networks for power system dynamic security assessment with missing data," *IEEE Transactions on Power Systems*, vol. 34, no. 6, pp. 5044–5052, 2019.

[14] O. Kosut, L. Jia, R. J. Thomas, and L. Tong, "Malicious data attacks on the smart grid," *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 645–658, 2011.

[15] K. R. Mestav, J. Luengo-Rozas, and L. Tong, "Bayesian state estimation for unobservable distribution systems via deep learning," *IEEE Transactions on Power Systems*, vol. 34, no. 6, pp. 4910–4920, 2019.

[16] K. R. Mestav and L. Tong, "Universal data anomaly detection via inverse generative adversary network," *IEEE Signal Processing Letters*, vol. 27, pp. 511–515, 2020.

[17] T. Huang, B. Satchidanandan, P. Kumar, and L. Xie, "An online detection framework for cyber attacks on automatic generation control," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 6816–6827, 2018.

[18] M. Wu and L. Xie, "Online detection of low-quality synchrophasor measurements: A data-driven approach," *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 2817–2827, 2016.

[19] M. Esmalifalak, H. Nguyen, R. Zheng, L. Xie, L. Song, and Z. Han, "A stealthy attack against electricity market using independent component analysis," *IEEE Systems Journal*, vol. 12, no. 1, pp. 297–307, 2015.

[20] P. Gao, M. Wang, S. G. Ghiocel, J. H. Chow, B. Fardanesh, and G. Stefopoulos, "Missing data recovery by exploiting low-dimensionality in power system synchrophasor measurements," *IEEE Transactions on Power Systems*, vol. 31, no. 2, pp. 1006–1013, March 2016.

[21] P. Gao, M. Wang, J. H. Chow, S. G. Ghiocel, B. Fardanesh, G. Stefopoulos, and M. P. Razanousky, "Identification of successive "unobservable" cyber data attacks in power systems." *IEEE Transactions on Signal Processing*, vol. 64, no. 21, pp. 5557–5570, Nov. 2016.

[22] M. Liao, D. Shi, Z. Yu, Z. Yi, Z. Wang, and Y. Xiang, "An alternating direction method of multipliers based approach for pmu data recovery," *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 4554–4565, 2018.

[23] C. Genes, I. Esnaola, S. M. Perlaza, L. F. Ochoa, and D. Coca, "Robust recovery of missing data in electricity distribution systems," *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 4057–4067, 2018.

[24] Y. Hao, M. Wang, J. H. Chow, E. Farantatos, and M. Patel, "Model-less data quality improvement of streaming synchrophasor measurements by exploiting the low-rank Hankel structure," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 6966–6977, June 2018.

[25] S. Zhang, Y. Hao, M. Wang, and J. H. Chow, "Multichannel hankel matrix completion through nonconvex optimization," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 4, pp. 617–632, 2018.

[26] S. Zhang and M. Wang, "Correction of corrupted columns through fast robust hankel matrix completion," *IEEE Transactions on Signal Processing*, vol. 67, no. 10, pp. 2580–2594, 2019.

[27] M. Yi, M. Wang, E. Farantatos, and T. Barik, "Bayesian robust hankel matrix completion with uncertainty modeling for synchrophasor data recovery," *ACM SIGENERGY Energy Informatics Review*, vol. 2, no. 1, pp. 1–19, 2022.

[28] Y. Hao, M. Wang, and J. H. Chow, "Modeless streaming synchrophasor data recovery in nonlinear systems," *IEEE Transactions on Power Systems*, vol. 35, no. 2, pp. 1166–1177, 2019.

[29] T. Hofmann, B. Schölkopf, and A. J. Smola, "Kernel methods in machine learning," *The annals of statistics*, vol. 36, no. 3, pp. 1171–1220, 2008.

[30] G. Ongie, R. Willett, R. D. Nowak, and L. Balzano, "Algebraic variety models for high-rank matrix completion," in *International Conference on Machine Learning*. PMLR, 2017, pp. 2691–2700.

[31] J. Fan and T. W. Chow, "Non-linear matrix completion," *Pattern Recognition*, vol. 77, pp. 378–394, 2018.

[32] J. Fan and M. Udell, "Online high rank matrix completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8690–8698.

[33] J. A. Bazerque and G. B. Giannakis, "Nonparametric basis pursuit via sparse kernel-based learning: A unifying view with advances in blind methods," *IEEE Signal Processing Magazine*, vol. 30, no. 4, pp. 112–125, 2013.

[34] M. Jalali, V. Kekatos, S. Bhela, H. Zhu, and V. A. Centeno, "Inferring power system dynamics from synchrophasor data using gaussian processes," *IEEE Transactions on Power Systems*, vol. 37, no. 6, pp. 4409–4423, 2022.

[35] C. M. Bishop, "Pattern recognition," *Machine learning*, vol. 128, no. 9, 2006.

[36] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, "Variational inference: A review for statisticians," *Journal of the American statistical Association*, vol. 112, no. 518, pp. 859–877, 2017.

[37] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[38] S. D. Babacan, M. Luessi, R. Molina, and A. K. Katsaggelos, "Sparse bayesian methods for low-rank matrix estimation," *IEEE Transactions on Signal Processing*, vol. 60, no. 8, pp. 3964–3977, 2012.

[39] J. Paisley, D. M. Blei, and M. I. Jordan, "Variational bayesian inference with stochastic search," in *International Conference on Machine Learning*, 2012, pp. 1363–1370.

[40] Y. Hao, M. Wang, and J. H. Chow, "Modeless streaming synchrophasor data recovery in nonlinear systems," *IEEE Transactions on Power Systems*, vol. 35, no. 2, pp. 1166–1177, 2019.

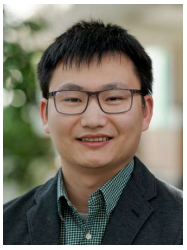
[41] A. V. Oppenheim, A. S. Willsky, S. H. Nawab, G. M. Hernández *et al.*, *Signals & systems*. Pearson Educación, 1997.

Ming Yi (S'17) received the B.E. degree in automation from Harbin Engineering University, Harbin, China, in 2016, and the M.S. degrees in control science and engineering from Harbin Institute of Technology, Harbin, China, in 2018, respectively.

He is currently a Ph.D. student in Rensselaer Polytechnic Institute, Troy, NY, USA. His research interests include signal processing, machine learning, power systems monitoring, and high-dimensional data analysis.

Meng Wang (M'12-SM'22) received B.S. and M.S. degrees from Tsinghua University, China, in 2005 and 2007, respectively. She received the Ph.D. degree from Cornell University, Ithaca, NY, USA, in 2012.

She is an Associate Professor in the department of Electrical, Computer, and Systems Engineering at Rensselaer Polytechnic Institute, Troy, NY, USA, where she joined in Dec. 2012. Before that, she was a postdoc scholar at Duke University, Durham, NC, USA. Her research interests include high-dimensional data analytics, machine learning and artificial intelligence, power systems monitoring, and synchrophasor technologies. She serves as an Associate Editor for IEEE Transactions on Smart Grid.



Tianqi Hong (S'13–M'16) received the B.Sc. degree in electrical engineering from Hohai University, China, in 2011, and the M.Sc. degree in electrical engineering from the Southeast University, China and the Engineering school of New York University in 2013. He received a Ph.D. degree from New York University in 2016. His main research interests are power system analysis, power electronics systems, microgrid, and electromagnetic design.

Currently, he is an Energy System Scientist at Argonne National Laboratory. Prior to this, he was a Postdoc Fellow in the Engineering school of New York University and a Senior Research Scientist at Unique Technical Services, LLC, responsible for transportation electrification, battery energy storage integration, and medium capacity microgrid. Dr. Hong is an active reviewer in the power engineering area, and he serves as an Editorial Board Member of International Transactions on Electrical Energy Systems, IEEE Transactions on Power Delivery, IEEE Transactions on Industry Applications, and IEEE Power Engineering Letters. He also serves as Special Activity Co-Chair of the IEEE IAS Industrial Power Converters Committee (IPCC).

Dongbo Zhao (SM'16) is currently a Global Technology Manager with Eaton in Eaton Research Lab. He was a Team Lead of DER Integration and a Principal Energy System Scientist with Argonne National Laboratory, Lemont, IL, and also an Institute Fellow of Northwestern Argonne Institute of Science and Engineering of Northwestern University, before joining Eaton. His research interests include power system control, protection, reliability analysis, transmission and distribution automation, and electric market optimization.

Dr. Zhao is a Senior Member of IEEE, and a member of IEEE PES, IAS and IES Societies. He has been the editor of IEEE Transactions on Power Delivery, IEEE Transactions on Sustainable Energy, and IEEE Power Engineering Letters.

This article has been accepted for publication in IEEE Transactions on Power Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2023.3254909

SUPPLEMENTARY MATERIALS

A. Gamma distribution

The Gamma function and the Gamma distribution are introduced here. The definition of the Gamma function with parameter e_0 is

$$\Gamma(e_0) = \int_0^\infty x^{e_0-1}e^{-x}dx. \tag{33}$$

The definition of the Gamma distribution of γ_y with parameters (e_0, f_0) is

$$\Gamma(\gamma_y|e_0, f_0) = \frac{{f_0}^{e_0}(\gamma_y)^{e_0-1}e^{-f_0\gamma_y}}{\Gamma(e_0)} \propto (\gamma_y)^{e_0-1}e^{-f_0\gamma_y}, \tag{34}$$

where e_0 and f_0 are positive scalers. The symbol "$\propto$" denotes "proportional to."

B. The Hankel operation and the mapping set

The Moore-Penrose pseudo-inverse of $\mathcal{H}$ is denoted as $\mathcal{H}^\dagger$. The entry (i, j) of inverse Hankel matrix $(\mathcal{H}^\dagger \boldsymbol{X}) \in \mathbb{R}^{m\times n}$ is defined as

$$\begin{aligned}(\mathcal{H}^\dagger \boldsymbol{X})_{i,j} &= \langle \mathcal{H}^\dagger \boldsymbol{X}, e_i e_j^T\rangle = \frac{1}{\kappa_j}\sum_{\frac{u-i}{m}+v=j} X_{u,v} \\ &= \begin{cases} \frac{1}{\kappa_j}\sum_{j_1=1}^{j} X_{(j_1-1)m+i,j+1-j_1} & j \le n_2 \\ \frac{1}{\kappa_j}\sum_{j_2=j+1-n_2}^{n_j} X_{(j-j_2)m+i,j_2} & j \ge n_2+1 \end{cases},\end{aligned} \tag{35}$$

where κ_j is the total number of entries in the jth anti-diagonal of an $n_2 \times n_1$ matrix. Mathematically, it can be written as $\kappa_j = \#\{(j_1, j_2)|j_1 + j_2 = j + 1 \quad 1 \le j_1 \le n_2, 1 \le j_2 \le n_j, n_j = \min(j, n_1)\}$

Let $\mathcal{H}_{n_2}(\mathbf{Y})$ denote the Hankel matrix of the data matrix $\mathbf{Y}$. $\Psi_{i,j}$ denotes the set of indices of entries in $\mathcal{H}_{n_2}(\mathbf{Y})$, where the entry values are $Y_{i,j}$. Mathematically, we have

$$\begin{aligned}&\Psi_{i,j} = \{(u, v)|(u, v) = ((j_1 - 1)m + i, j + 1 - j_1) \text{ for every } \\ &j_1 = 1, 2, ..., j, \quad \text{under the case when } j \le n_2; \\ &(u, v) = ((j - j_2)m + i, j_2) \text{ for every } j_2 = j + 1 - n_2, ..., n_j, \\ &\text{where } n_j = \min(j, n_1), \quad \text{under the case when } j \ge n_2 + 1;\} \\ &(i, j) \in \Omega.\end{aligned} \tag{36}$$

C. The derivation details of updating rule for variational inference

Algorithm 1 shows the complete updating process of our proposed method. Note that Algorithm 1 can be simplified when the objective is to recover missing data only. One can skip the updating steps for $\mathbb{E}[E_{i,j}]$ and $\mathbb{E}[\beta_{i,j}]$ in lines 24-25 and all the other updating steps remain the same.

As discussed in equation (21), computing $p(\boldsymbol{Y}_\Omega^o)$ is intractable. It is hard to directly minimize the KL divergence in (23). Instead, we can solve an equivalent maximization problem. To see this,

Algorithm 1 Bayesian High Rank Hankel Matrix Completion

Require: The observation matrix $\boldsymbol{Y}^o$; The parameters e_0, f_0, g_0, h_0, γ_u and γ_x for prior distributions; The initial rank K; The maximum iterations $T_{\max}$ for the outer loop; The maximum iterations $t_1^{\max}$, $t_2^{\max}$, $t_3^{\max}$ and $t_4^{\max}$ for the inner loops; The convergence threshold ξ; The Hankel parameter n_2.

1: **Initialization**: Entries in $\boldsymbol{U}$ are randomly initialized by $\mathcal{N}(0, 1)$. $\boldsymbol{V}$ is initialized by an all zeroes matrix. Use the rank-r approximation of $\boldsymbol{X}$ as $\bar{\boldsymbol{X}}^0$. The initial $\boldsymbol{E}$ is $\boldsymbol{Y}^o - \mathcal{P}_{\boldsymbol{\Omega}}(\mathcal{H}^\dagger\bar{\boldsymbol{X}}^0)$. $\eta = 1, t = t_1 = t_2 = t_3 = t_4 = 1$.
2: **while** $\eta > \xi$ and $t < T_{\max}$ **do**
3: Compute kernels $\boldsymbol{\mathcal{K}}_{XU}$ and $\boldsymbol{\mathcal{K}}_{UU}$ by (26) and (27);
4: Compute $\mathbb{E}[\boldsymbol{V}_{.q}]$ and $\boldsymbol{\Sigma}_{\boldsymbol{V}_{.q}}$ from $q(\boldsymbol{V}_{.q})$ by (44) and (45) for each $q = 1, 2, 3, ..., n_1$;
5: **repeat**
6: Compute $\nabla_{a_{\boldsymbol{U}_{.k}}}\ell_1$ by (53);
7: $a_{\boldsymbol{U}_{.k}}^{t_1+1} = a_{\boldsymbol{U}_{.k}}^{t_1} + \lambda_1\nabla_{a_{\boldsymbol{U}_{.k}}}\ell_1$;
8: $t_1 = t_1 + 1$
9: **until** converged or $t_1 = t_1^{\max}$; for all k
10: **repeat**
11: Compute $\nabla_{\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}}\ell_1$ and $\nabla^2_{\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}}\ell_1$ by (51) and (52);
12: $\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}^{t_2+1} = \boldsymbol{\mu}_{\boldsymbol{U}_{.k}}^{t_2} + \lambda_2(\nabla^2_{\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}}\ell_1)^{-1}\nabla_{\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}}\ell_1$;
13: $t_2 = t_2 + 1$
14: **until** converged or $t_2 = t_2^{\max}$ for all k;
15: **repeat**
16: Compute $\nabla_{b_{\boldsymbol{X}_{.q}}}\ell_2$ by (59) ;
17: $b_{\boldsymbol{X}_{.q}}^{t_3+1} = b_{\boldsymbol{X}_{.q}}^{t_3} + \lambda_3\nabla_{b_{\boldsymbol{X}_{.q}}}\ell_2$;
18: $t_3 = t_3 + 1$
19: **until** converged or $t_3 = t_3^{\max}$ for all q;
20: **repeat**
21: Compute $\nabla_{\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}}\ell_2$ and $\nabla^2_{\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}}\ell_2$ by (57) and (58) ;
22: $\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}^{t_4+1} = \boldsymbol{\mu}_{\boldsymbol{X}_{.q}}^{t_4} + \lambda_4(\nabla^2_{\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}}\ell_2)^{-1}\nabla_{\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}}\ell_2$;
23: $t_4 = t_4 + 1$;
24: **until** converged or $t_4 = t_4^{\max}$ for all q;
25: Compute $\mathbb{E}[E_{i,j}]$ and $\Sigma_{E_{i,j}}$ by (63) and (64) for all $(i, j) \in \boldsymbol{\Omega}$;
26: Compute $\mathbb{E}[\beta_{i,j}]$ by (70) for all $(i, j) \in \boldsymbol{\Omega}$;
27: Compute $\mathbb{E}[\gamma_y]$ from $q(\gamma_y)$ by (77);
28: **if** $\mathbb{E}[V_{kq}] < \xi_1$ for all k **then**
29: Remove $\mathbb{E}[\boldsymbol{U}_{.k}]$ in $\mathbb{E}[\boldsymbol{U}]$, $\mathbb{E}[V_{kq}]$ for all q;
30: $K = K - 1$;
31: **end if**
32: $\boldsymbol{X}_{.q}^{(l)} = \boldsymbol{\mu}_{\boldsymbol{X}_{.q}} + \exp(0.5b_{\boldsymbol{X}_{.q}})\boldsymbol{\epsilon}^{(l)}$ for all q, $\boldsymbol{U}_{.k}^{(l)} = \boldsymbol{\mu}_{\boldsymbol{U}_{.k}} + \exp(0.5a_{\boldsymbol{U}_{.q}})\boldsymbol{\epsilon}^{(l)}$ for all k, $\boldsymbol{\epsilon}^{(l)} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_{mn_2})$;
33: $\bar{\boldsymbol{X}}^t = \frac{1}{L}\sum_{l=1}^{L}\boldsymbol{X}_{.q}^{(l)}$;
34: $\eta = \frac{\|\bar{\boldsymbol{X}}^t - \bar{\boldsymbol{X}}^{t-1}\|_F}{\|\bar{\boldsymbol{X}}^{t-1}\|_F}$;
35: $\bar{\boldsymbol{X}}^{t-1} = \bar{\boldsymbol{X}}^t$;
36: $t = t + 1$;
37: **end while**
38: Compute the predictive mean $\mathbb{E}[Y_{i,j}]$ and the uncertainty index by (28) and (30), respectively
39: **return** The estimation $\mathbb{E}[Y_{i,j}]$ and the uncertainty index.

This article has been accepted for publication in IEEE Transactions on Power Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2023.3254909

$$\begin{aligned}
&\mathbb{KL}(q(\boldsymbol{\Theta})||p(\boldsymbol{\Theta},\boldsymbol{Y}|\boldsymbol{Y}_\Omega^o))\\
&=-\int q(\boldsymbol{\Theta})\ln\frac{p(\boldsymbol{\Theta},\boldsymbol{Y}|\boldsymbol{Y}_\Omega^o)}{q(\boldsymbol{\Theta})}d\boldsymbol{\Theta}\\
&=\mathbb{E}[\ln q(\boldsymbol{\Theta})]-\mathbb{E}[\ln p(\boldsymbol{\Theta},\boldsymbol{Y}|\boldsymbol{Y}_\Omega^o)]\\
&=\mathbb{E}[\ln q(\boldsymbol{\Theta})]-\mathbb{E}[\ln p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o)]+\ln(p(\boldsymbol{Y}_\Omega^o))\\
&=-(\mathbb{E}[\ln p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o)]-\mathbb{E}[\ln q(\boldsymbol{\Theta})]).
\end{aligned}\tag{37}$$

where $\ln(p(\boldsymbol{Y}_\Omega^o))$ represents taking the logarithm of $p(\boldsymbol{Y}_\Omega^o)$. The terms $\mathbb{E}[\ln p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o)]-\mathbb{E}[\ln q(\boldsymbol{\Theta})]$ are the so-called evidence lower bound (ELBO). The term $p(\boldsymbol{Y}_\Omega^o)$ is unrelated to the optimization problem and can be removed. Thus, the KL divergence minimization problem is equivalent to the ELBO maximization problem.

The joint probability distribution of observed data, inferred data, and all the latent variables is given by (38),

$$\begin{aligned}
&p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o)\\
&=p(\boldsymbol{Y}_{\boldsymbol{\Omega}}^o|\boldsymbol{X},\boldsymbol{E},\gamma_y)p(\Phi(\boldsymbol{X})|\Phi(\boldsymbol{U}),\boldsymbol{V})p(\Phi(\boldsymbol{U})|\gamma_u)p(\boldsymbol{V}|\gamma_v)p(\gamma_y)\\
&\quad p(\boldsymbol{X})p(\boldsymbol{U})p(\boldsymbol{E}|\boldsymbol{\beta})p(\boldsymbol{\beta})\\
&=\prod_{j=1}^{n}\mathcal{N}(\boldsymbol{Y}_{\cdot j}^o|\mathcal{P}_{\Omega_j}(\mathcal{H}^\dagger\boldsymbol{X}+\boldsymbol{E})_{\cdot j},\frac{1}{\gamma_y}I_{|\boldsymbol{\Omega}_j|})\\
&\quad\prod_{q=1}^{n_1}\mathcal{N}(\boldsymbol{V}_{\cdot q}|0,\tfrac{1}{\gamma_v}\boldsymbol{I}_K)\mathcal{N}(\boldsymbol{X}_{\cdot q}|0,\tfrac{1}{\gamma_x}\boldsymbol{I}_{mn_2})\mathcal{N}(\Phi(\boldsymbol{X})_{\cdot q}|\Phi(\boldsymbol{U})\boldsymbol{V}_{\cdot q},\tfrac{1}{\gamma_\epsilon}I_{mn_2})\\
&\quad\prod_{k=1}^{K}\mathcal{N}(\boldsymbol{U}_{\cdot k}|0,\frac{1}{\gamma_u}\boldsymbol{I}_{mn_2})\prod_{(i,j)\in\Omega}p(E_{i,j}|\beta_{i,j})p(\beta_{i,j}|g_0,h_0)\\
&\quad\Gamma(\gamma_y|e_0,f_0),
\end{aligned}\tag{38}$$

where $\mathcal{N}(\boldsymbol{Y}_{\cdot j}^o|(\mathcal{H}^\dagger\boldsymbol{X}+\boldsymbol{E})_{\cdot j},\frac{1}{\gamma_y}I_{|\boldsymbol{\Omega}_j|})$ represents that the jth column of $\boldsymbol{Y}_{\cdot j}^o$ follows a Gaussian distribution with mean $(\mathcal{H}^\dagger\boldsymbol{X}+\boldsymbol{E})_{\cdot j}$ and covariance $\frac{1}{\gamma_y}I_{|\boldsymbol{\Omega}_j|}$, where $\boldsymbol{\Omega}_j$ is the set of observed entries in jth column and $|\boldsymbol{\Omega}_j|$ denotes the cardinality of $\boldsymbol{\Omega}_j$. In the following part, we show the derivation details for each updating rule of variational inference.

(I) The approximate posterior distribution of $\boldsymbol{V}_{\cdot q}$ follows a Gaussian distribution (for all $q=1,...,n_1$).

Note that

$$\begin{aligned}
&\mathcal{N}(\Phi(\boldsymbol{X})_{\cdot q}|(\Phi(\boldsymbol{U})\boldsymbol{V}_{\cdot q},\frac{1}{\gamma_\epsilon}\boldsymbol{I}_{n_1})\\
&\propto\exp(\frac{-\gamma_\epsilon}{2}(\Phi(\boldsymbol{X})_{\cdot q}-\Phi(\boldsymbol{U})\boldsymbol{V}_{\cdot q})^T(\Phi(\boldsymbol{X})_{\cdot q}-\Phi(\boldsymbol{U})\boldsymbol{V}_{\cdot q}))\\
&\propto\exp(\tfrac{-\gamma_\epsilon}{2}(\Phi(\boldsymbol{X})_{\cdot q}^T\Phi(\boldsymbol{X})_{\cdot q}-2\boldsymbol{V}_{\cdot q}^T\Phi(\boldsymbol{U})^T\Phi(\boldsymbol{X})_{\cdot q}+\boldsymbol{V}_{\cdot q}^T\Phi(\boldsymbol{U})^T\Phi(\boldsymbol{U})\boldsymbol{V}_{\cdot q}))\\
&\propto\exp(\tfrac{-\gamma_\epsilon}{2}\boldsymbol{\mathcal{K}}_{XX}(q,q)+\gamma_\epsilon\boldsymbol{V}_{\cdot q}^T\boldsymbol{\mathcal{K}}_{XU}(q,:)^T-\tfrac{1}{2}\boldsymbol{V}_{\cdot q}^T\gamma_\epsilon\boldsymbol{\mathcal{K}}_{UU}\boldsymbol{V}_{\cdot q}))
\end{aligned}\tag{39}$$

where $\boldsymbol{\mathcal{K}}_{XU}(q,:)$ represents the qth row in $\boldsymbol{\mathcal{K}}_{XU}$.

Also note that

$$\mathcal{N}(\boldsymbol{V}_{\cdot q}|0,\frac{1}{\gamma_v}\boldsymbol{I}_K)\propto\exp(\frac{-\gamma_v}{2}(\boldsymbol{V}_{\cdot q}^T\boldsymbol{V}_{\cdot q})).\tag{40}$$

Therefore,

$$\begin{aligned}
&p(\boldsymbol{V}_{\cdot q}|-)\propto\mathcal{N}(\Phi(\boldsymbol{X})_{\cdot q}|(\Phi(\boldsymbol{U})\boldsymbol{V}_{\cdot q},\frac{1}{\gamma_\epsilon}\boldsymbol{I}_{n_1})\mathcal{N}(\boldsymbol{V}_{\cdot q}|0,\frac{1}{\gamma_v}\boldsymbol{I}_K)\\
&\propto\exp(\tfrac{-\gamma_\epsilon}{2}\boldsymbol{\mathcal{K}}_{XX}(q,q)+\gamma_\epsilon\boldsymbol{V}_{\cdot q}^T\boldsymbol{\mathcal{K}}_{XU}(q,:)^T-\tfrac{1}{2}\boldsymbol{V}_{\cdot q}^T(\gamma_\epsilon\boldsymbol{\mathcal{K}}_{UU}+\gamma_v\boldsymbol{I}_K)\boldsymbol{V}_{\cdot q})
\end{aligned}\tag{41}$$

Then the following derivations are straightforward expansions. The logarithm of $q(\boldsymbol{V}_{\cdot q})$ can be expressed as

$$\begin{aligned}
&\ln(q(\boldsymbol{V}_{\cdot q}))\\
&=\mathbb{E}_{\boldsymbol{\Theta}\setminus\boldsymbol{V}_{\cdot q}}[\ln\,p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o)]+\text{const.}\\
&=\mathbb{E}_{\boldsymbol{\Theta}\setminus\boldsymbol{V}_{\cdot q}}[\ln\,p(\Phi(\boldsymbol{X})_{\cdot q}|\boldsymbol{U},\boldsymbol{V},\gamma_\epsilon)p(\boldsymbol{V}_{\cdot q})]+\text{const.}\\
&=\mathbb{E}_{\boldsymbol{\Theta}\setminus\boldsymbol{V}_{\cdot q}}[\ln\,\mathcal{N}(\Phi(\boldsymbol{X})_{\cdot q}|(\Phi(\boldsymbol{U})\boldsymbol{V}_{\cdot q},\tfrac{1}{\gamma_\epsilon}I_{mn_2})\mathcal{N}(\boldsymbol{V}_{\cdot q}|0,\tfrac{1}{\gamma_v}\boldsymbol{I}_K)]\\
&\quad+\text{const.}\\
&=\mathbb{E}_{\boldsymbol{\Theta}\setminus\boldsymbol{V}_{\cdot q}}[\ln\exp(\tfrac{-\gamma_\epsilon}{2}(\Phi(\boldsymbol{X})_{\cdot q}-\Phi(\boldsymbol{U})\boldsymbol{V}_{\cdot q})^T(\Phi(\boldsymbol{X})_{\cdot q}-\Phi(\boldsymbol{U})\boldsymbol{V}_{\cdot q}))\\
&\quad\exp(\tfrac{-\gamma_v}{2}(\boldsymbol{V}_{\cdot q}^T\boldsymbol{V}_{\cdot q}))]+\text{const.}\\
&=\mathbb{E}_{\boldsymbol{\Theta}\setminus\boldsymbol{V}_{\cdot q}}[\ln\exp(\tfrac{-\gamma_\epsilon}{2}(\Phi(\boldsymbol{X})_{\cdot q}^T\Phi(\boldsymbol{X})_{\cdot q}-2\boldsymbol{V}_{\cdot q}^T\Phi(\boldsymbol{U})^T\Phi(\boldsymbol{X})_{\cdot q}\\
&\quad-\boldsymbol{V}_{\cdot q}^T\Phi(\boldsymbol{U})^T\Phi(\boldsymbol{U})\boldsymbol{V}_{\cdot q}))\exp(\tfrac{-\gamma_v}{2}(\boldsymbol{V}_{\cdot q}^T\boldsymbol{V}_{\cdot q}))]+\text{const.}\\
&=\mathbb{E}_{\boldsymbol{\Theta}\setminus\boldsymbol{V}_{\cdot q}}[\ln\exp(\frac{-\gamma_\epsilon}{2}(\boldsymbol{\mathcal{K}}_{XX}(q,q)-2\boldsymbol{V}_{\cdot q}^T\boldsymbol{\mathcal{K}}_{XU}(q,:)^T+\\
&\quad\boldsymbol{V}_{\cdot q}^T\boldsymbol{\mathcal{K}}_{UU}\boldsymbol{V}_{\cdot q}))\exp(\tfrac{-\gamma_v}{2}(\boldsymbol{V}_{\cdot q}^T\boldsymbol{V}_{\cdot q}))]+\text{const.}\\
&=\mathbb{E}_{\boldsymbol{\Theta}\setminus\boldsymbol{V}_{\cdot q}}[\ln\exp(\frac{-\gamma_\epsilon}{2}\boldsymbol{\mathcal{K}}_{XX}(q,q)+\gamma_\epsilon\boldsymbol{V}_{\cdot q}^T\boldsymbol{\mathcal{K}}_{XU}(q,:)^T\\
&\quad-\frac{1}{2}\boldsymbol{V}_{\cdot q}^T(\gamma_\epsilon\boldsymbol{\mathcal{K}}_{UU}+\gamma_v\boldsymbol{I}_K)\boldsymbol{V}_{\cdot q}))]+\text{const.}\\
&=(\tfrac{-\gamma_\epsilon}{2}\mathbb{E}[\boldsymbol{\mathcal{K}}_{XX}(q,q)]+\gamma_\epsilon\boldsymbol{V}_{\cdot q}^T\mathbb{E}[\boldsymbol{\mathcal{K}}_{XU}(q,:)]^T-\tfrac{1}{2}\boldsymbol{V}_{\cdot q}^T(\gamma_\epsilon\mathbb{E}[\boldsymbol{\mathcal{K}}_{UU}]\\
&\quad+\gamma_v\boldsymbol{I}_K)\boldsymbol{V}_{\cdot q}))+\text{const.}
\end{aligned}\tag{42}$$

Then $\boldsymbol{V}_{\cdot q}$ follows a Gaussian distribution:

$$q(\boldsymbol{V}_{\cdot q})\sim\mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{V}_{\cdot q}},\boldsymbol{\Sigma}_{\boldsymbol{V}_{\cdot q}}),\tag{43}$$

with mean $\boldsymbol{\mu}_{\boldsymbol{V}_{\cdot q}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{V}_{\cdot q}}$, where

$$\boldsymbol{\Sigma}_{\boldsymbol{V}_{\cdot q}}=[\gamma_\epsilon\mathbb{E}[\boldsymbol{\mathcal{K}}_{UU}]+\gamma_v\boldsymbol{I}_K]^{-1},\tag{44}$$

$$\boldsymbol{\mu}_{\boldsymbol{V}_{\cdot q}}=\gamma_\epsilon\boldsymbol{\Sigma}_{\boldsymbol{V}_{\cdot q}}\mathbb{E}[\boldsymbol{\mathcal{K}}_{XU}(q,:)]^T.\tag{45}$$

(II) The approximate posterior distribution of $\boldsymbol{U}_{\cdot k}$ follows a Gaussian distribution (for all $k=1,...,K$) .

$$q(\boldsymbol{U}_{\cdot k})\sim\mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{U}_{\cdot k}},\boldsymbol{\Sigma}_{\boldsymbol{U}_{\cdot k}}),\tag{46}$$

The mean $\boldsymbol{\mu}_{\boldsymbol{U}_{\cdot k}}$ and the variance $\boldsymbol{\Sigma}_{\boldsymbol{U}_{\cdot k}}$ needs to be computed. From (37), we know that the approximate distributions of all latent variables $q(\boldsymbol{\Theta})$ are obtained by solving the following equivalent maximization problem,

$$\begin{aligned}
q(\boldsymbol{\Theta})&=\arg\min_{q(\boldsymbol{\Theta})}\mathbb{KL}(q(\boldsymbol{\Theta})||p(\boldsymbol{\Theta},\boldsymbol{Y}|\boldsymbol{Y}_\Omega^o))\\
&=\arg\max_{q(\boldsymbol{\Theta})}\mathbb{E}[\ln p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o)]-\mathbb{E}[\ln q(\boldsymbol{\Theta})].
\end{aligned}\tag{47}$$

Based on the mean-field assumption, we assume that $q(\boldsymbol{\Theta})$ can be factorized as $q(\boldsymbol{\Theta})=\prod_{i=1}^N q(\boldsymbol{\Theta}_i)$ where N is the number of all latent variables. Then the maximization problem with respect to one latent variable $q(\boldsymbol{\Theta}_i)$ becomes

$$
\begin{aligned}
&q(\boldsymbol{\Theta}_i)\\
&= \arg\max_{q(\boldsymbol{\Theta}_i)} \int q(\boldsymbol{\Theta}_i)\mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{\Theta}_i)}[\ln p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o)]d(\boldsymbol{\Theta}_i)\\
&- \int q(\boldsymbol{\Theta}_i)\ln q(\boldsymbol{\Theta}_i)d\boldsymbol{\Theta}_i\\
&= \arg\max_{q(\boldsymbol{\Theta}_i)} \int q(\boldsymbol{\Theta}_i)\mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{\Theta}_i)}[\ln p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o|\boldsymbol{\Theta}_i)p(\boldsymbol{\Theta}_i)]d(\boldsymbol{\Theta}_i)\\
&- \int q(\boldsymbol{\Theta}_i)\ln q(\boldsymbol{\Theta}_i)d\boldsymbol{\Theta}_i.\\
&= \arg\max_{q(\boldsymbol{\Theta}_i)} \int q(\boldsymbol{\Theta}_i)\mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{\Theta}_i)}[\ln p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o|\boldsymbol{\Theta}_i)]d(\boldsymbol{\Theta}_i)\\
&+ \int q(\boldsymbol{\Theta}_i)\ln p(\boldsymbol{\Theta}_i)d\boldsymbol{\Theta}_i - \int q(\boldsymbol{\Theta}_i)\ln q(\boldsymbol{\Theta}_i)d\boldsymbol{\Theta}_i\\
&= \arg\max_{q(\boldsymbol{\Theta}_i)} \int q(\boldsymbol{\Theta}_i)\mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{\Theta}_i)}[\ln p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o|\boldsymbol{\Theta}_i)]d(\boldsymbol{\Theta}_i)\\
&+ \int q(\boldsymbol{\Theta}_i)\ln\frac{p(\boldsymbol{\Theta}_i)}{q(\boldsymbol{\Theta}_i)}d\boldsymbol{\Theta}_i\\
&= \arg\max_{q(\boldsymbol{\Theta}_i)} \int q(\boldsymbol{\Theta}_i)\mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{\Theta}_i)}[\ln p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o|\boldsymbol{\Theta}_i)]d(\boldsymbol{\Theta}_i)\\
&- \mathbb{KL}(q(\boldsymbol{\Theta}_i)|p(\boldsymbol{\Theta}_i)).
\end{aligned}
\tag{48}
$$

The derivations above are straightforward expansions. When the objective function is for $\boldsymbol{U}_{.k}$, one can replace the latent variable $\boldsymbol{\Theta}_i$ in (48) with $\boldsymbol{U}_{.k}$. We employ the reparameterization trick [37] to make the objective function differentiable. The approximate distribution $q(\boldsymbol{U}_{.k})$ is assumed to follow a Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}, \boldsymbol{\Sigma}_{\boldsymbol{U}_{.k}})$, where the variance $\boldsymbol{\Sigma}_{\boldsymbol{U}_{.k}}$ is a diagonal matrix and the diagonal entry ${\sigma^j_{\boldsymbol{U}_{.k}}}^2$ is the same for $j = 1, ...mn_2$. The prior distribution $p(\boldsymbol{U}_{.k})$ also follows a Gaussian distribution $\mathcal{N}(0, \frac{1}{\gamma_u}\boldsymbol{I}_{mn_2})$.

Plug in $p(\boldsymbol{U}_{.k})$ and $q(\boldsymbol{U}_{.k})$, the negative KL divergence can be derived in a straightforward way as follows

$$
\begin{aligned}
&-\mathbb{KL}(q(\boldsymbol{U}_{.k})|p(\boldsymbol{U}_{.k}))\\
&= -\int q(\boldsymbol{U}_{.k})\ln\frac{q(\boldsymbol{U}_{.k})}{p(\boldsymbol{U}_{.k})}d\boldsymbol{U}_{.k}.\\
&= \frac{mn_2}{2}\ln(\gamma_u) + \frac{1}{2}\ln(|\boldsymbol{\Sigma}_{\boldsymbol{U}_{.k}}|) + \frac{mn_2}{2} - \frac{1}{2}\gamma_u\boldsymbol{\mu}^T_{\boldsymbol{U}_{.k}}\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}\\
&- \frac{1}{2}\gamma_u\text{trace}(\boldsymbol{\Sigma}_{\boldsymbol{U}_{.k}})\\
&= \frac{mn_2}{2}\ln(\gamma_u) - \frac{1}{2}\gamma_u\boldsymbol{\mu}^T_{\boldsymbol{U}_{.k}}\boldsymbol{\mu}_{\boldsymbol{U}_{.k}} + \frac{1}{2}\sum_{j=1}^{mn_2}\ln({\sigma^j_{\boldsymbol{U}_{.k}}}^2) + \frac{mn_2}{2}\\
&- \frac{1}{2}\gamma_u\sum_{j=1}^{mn_2}{\sigma^j_{\boldsymbol{U}_{.k}}}^2\\
&= \frac{mn_2}{2}\ln(\gamma_u) - \frac{1}{2}\gamma_u\boldsymbol{\mu}^T_{\boldsymbol{U}_{.k}}\boldsymbol{\mu}_{\boldsymbol{U}_{.k}} + \frac{1}{2}mn_2a_{\boldsymbol{U}_{.k}} + \frac{mn_2}{2}\\
&- \frac{1}{2}mn_2\gamma_u\text{exp}(a_{\boldsymbol{U}_{.k}}),
\end{aligned}
\tag{49}
$$

where $a_{\boldsymbol{U}_{.k}}$ is the logarithm of the variance and $a_{\boldsymbol{U}_{.k}} = \ln({\sigma^j_{\boldsymbol{U}_{.k}}}^2)$ for all $j = 1, 2, ...mn_2$. Optimization over $a_{\boldsymbol{U}_{.k}}$ prevents the negative solutions of the variance.

$$
\begin{aligned}
&q(\boldsymbol{U}_{.k})\\
&\overset{(a)}{=} \arg\max_{q(\boldsymbol{U}_{.k})} \int q(\boldsymbol{U}_{.k})\mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{U}_{.k})}[\ln p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o)]d(\boldsymbol{U}_{.k})\\
&- \int q(\boldsymbol{U}_{.k})\ln q(\boldsymbol{U}_{.k})d\boldsymbol{U}_{.k}.\\
&\overset{(b)}{=} \arg\max_{q(\boldsymbol{U}_{.k})} \int q(\boldsymbol{U}_{.k})\mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{U}_{.k})}[\ln p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o|\boldsymbol{U}_{.k})]d(\boldsymbol{U}_{.k})\\
&-\mathbb{KL}(q(\boldsymbol{U}_{.k})|p(\boldsymbol{U}_{.k})).\\
&\overset{(c)}{\approx} \arg\max_{q(\boldsymbol{U}_{.k})} \tfrac{1}{L}\textstyle\sum_l^L \mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{U}_{.k})}[\ln p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o|\boldsymbol{U}_{.k}^{(l)})]\\
&-\mathbb{KL}(q(\boldsymbol{U}_{.k})|p(\boldsymbol{U}_{.k})).\\
&\overset{(d)}{=} \arg\max_{q(\boldsymbol{U}_{.k})} \tfrac{1}{L}\textstyle\sum_l^L \mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{U}_{.k})}[\ln p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_\Omega^o|\boldsymbol{U}_{.k}^{(l)})] - \tfrac{1}{2}\gamma_u\boldsymbol{\mu}^T_{\boldsymbol{U}_{.k}}\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}\\
&+\tfrac{1}{2}mn_2a_{\boldsymbol{U}_{.k}} - \tfrac{1}{2}mn_2\gamma_u\text{exp}(a_{\boldsymbol{U}_{.k}}) + \text{const.}\\
&\overset{(e)}{=} \arg\max_{q(\boldsymbol{U}_{.k})} -\tfrac{1}{L}\textstyle\sum_l^L(\frac{\gamma_\epsilon}{2}\mathbb{E}[||\Phi(\boldsymbol{X}) - \Phi(\boldsymbol{U}^{(l)})\boldsymbol{V}||_F^2])\\
&-\tfrac{1}{2}\gamma_u\boldsymbol{\mu}^T_{\boldsymbol{U}_{.k}}\boldsymbol{\mu}_{\boldsymbol{U}_{.k}} + \tfrac{1}{2}mn_2a_{\boldsymbol{U}_{.k}} - \tfrac{1}{2}mn_2\gamma_u\text{exp}(a_{\boldsymbol{U}_{.k}}) + \text{const.}\\
&= \arg\max_{q(\boldsymbol{U}_{.k})} -\tfrac{1}{L}\textstyle\sum_l^L(\frac{\gamma_\epsilon}{2}\mathbb{E}[\text{trace}(\Phi(\boldsymbol{X})^T\Phi(\boldsymbol{X}) + 2\Phi(\boldsymbol{X})^T\Phi(\boldsymbol{U}^{(l)})\boldsymbol{V}\\
&+ \boldsymbol{V}^T\Phi(\boldsymbol{U}^{(l)})^T\Phi(\boldsymbol{U}^{(l)})\boldsymbol{V})]) - \tfrac{1}{2}\gamma_u\boldsymbol{\mu}^T_{\boldsymbol{U}_{.k}}\boldsymbol{\mu}_{\boldsymbol{U}_{.k}} + \tfrac{1}{2}mn_2a_{\boldsymbol{U}_{.k}}\\
&- \frac{1}{2}mn_2\gamma_u\text{exp}(a_{\boldsymbol{U}_{.k}}) + \text{const.}\\
&= \arg\max_{q(\boldsymbol{U}_{.k})} -\tfrac{1}{L}\textstyle\sum_l^L(\frac{\gamma_\epsilon}{2}\mathbb{E}[\text{trace}(\boldsymbol{\mathcal{K}}_{XX} + 2\boldsymbol{\mathcal{K}}^{(l)}_{XU}\boldsymbol{V} + \boldsymbol{V}^T\boldsymbol{\mathcal{K}}^{(l)}_{UU}\boldsymbol{V})]\\
&-\tfrac{1}{2}\gamma_u\boldsymbol{\mu}^T_{\boldsymbol{U}_{.k}}\boldsymbol{\mu}_{\boldsymbol{U}_{.k}} + \tfrac{1}{2}mn_2a_{\boldsymbol{U}_{.k}} - \tfrac{1}{2}mn_2\gamma_u\text{exp}(a_{\boldsymbol{U}_{.k}}) + \text{const.}\\
&\overset{(f)}{=} \arg\max_{q(\boldsymbol{U}_{.k})} -(\tfrac{\gamma_\epsilon}{2}\mathbb{E}[\text{trace}(2(\tfrac{1}{L}\textstyle\sum_l^L\boldsymbol{\mathcal{K}}^{(l)}_{XU})\boldsymbol{V} + \boldsymbol{V}^T(\tfrac{1}{L}\textstyle\sum_l^L\boldsymbol{\mathcal{K}}^{(l)}_{UU})\boldsymbol{V})]\\
&-\tfrac{1}{2}\gamma_u\boldsymbol{\mu}^T_{\boldsymbol{U}_{.k}}\boldsymbol{\mu}_{\boldsymbol{U}_{.k}} + \tfrac{1}{2}mn_2a_{\boldsymbol{U}_{.k}} - \tfrac{1}{2}mn_2\gamma_u\text{exp}(a_{\boldsymbol{U}_{.k}}) + \text{const.}\\
&\overset{(g)}{\approx} \arg\max_{q(\boldsymbol{U}_{.k})} -(\tfrac{\gamma_\epsilon}{2}\text{trace}(2\mathbb{E}[\boldsymbol{\mathcal{K}}_{XU}]\mathbb{E}[\boldsymbol{V}] + \mathbb{E}[\boldsymbol{V}]^T\mathbb{E}[\boldsymbol{\mathcal{K}}_{UU}]\mathbb{E}[\boldsymbol{V}])\\
&-\tfrac{1}{2}\gamma_u\boldsymbol{\mu}^T_{\boldsymbol{U}_{.k}}\boldsymbol{\mu}_{\boldsymbol{U}_{.k}} + \tfrac{1}{2}mn_2a_{\boldsymbol{U}_{.k}} - \tfrac{1}{2}mn_2\gamma_u\text{exp}(a_{\boldsymbol{U}_{.k}}) + \text{const.}\\
&= \arg\max_{q(\boldsymbol{U}_{.k})} \ell_1\ (\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}, a_{\boldsymbol{U}_{.k}}).
\end{aligned}
\tag{50}
$$

One can replace the latent variable $\boldsymbol{\Theta}_i$ in (48) with $\boldsymbol{U}_{.k}$ to get steps (a) and (b) in (50). Step (c) follows the Monte-Carlo approximation. The reparameterization trick [37] is employed here to make the Monte-Carlo estimation differentiable with respect to $\boldsymbol{U}_{.k}$. The reparametrization of $\boldsymbol{U}_{.k}$ is $\boldsymbol{U}_{.k} = \boldsymbol{\mu}_{\boldsymbol{U}_{.k}} + \text{exp}(0.5a_{\boldsymbol{U}_{.k}})\boldsymbol{\epsilon}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_{mn_2})$, where $\text{exp}(0.5a_{\boldsymbol{U}_{.k}})$ is the standard deviation. $\boldsymbol{\epsilon}$ is the auxiliary noise vector. Step (d) follows the definition of KL divergence in (49). Steps (e)-(f) follow from the straightforward expansion, and we use kernels to replace the inner product of mapping functions. Step (f) holds because the Kernel function $\boldsymbol{\mathcal{K}}_{XX}$ is not related to the $q(\boldsymbol{U}_{.k})$ and is removed from the objective function. The definition of the other two Kernel matrices is $\boldsymbol{\mathcal{K}}^{(l)}_{XU}(q,k) = \text{exp}(-\frac{1}{2c_2}||\boldsymbol{X}_{.q} - \boldsymbol{U}^{(l)}_{.k}||_2^2) = \text{exp}(-\frac{1}{2c_2}||\boldsymbol{X}_{.q} - \boldsymbol{\mu}_{\boldsymbol{U}_{.k}} - \text{exp}(0.5a_{\boldsymbol{U}_{.k}})\boldsymbol{\epsilon}^{(l)}||_2^2)$ and $\boldsymbol{\mathcal{K}}^{(l)}_{UU}(i,k) = \text{exp}(-\frac{1}{2c_3}||\boldsymbol{U}_{.i} - \boldsymbol{\mu}_{\boldsymbol{U}_{.k}} - \text{exp}(0.5a_{\boldsymbol{U}_{.k}})\boldsymbol{\epsilon}^{(l)}||_2^2)$ $\boldsymbol{\epsilon}^{(l)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_{mn_2})$. At step (g), $\mathbb{E}[\boldsymbol{\mathcal{K}}_{XU}]$ and $\mathbb{E}[\boldsymbol{\mathcal{K}}_{UU}]$ are approximated by sampling $\boldsymbol{X}$ and $\boldsymbol{U}$ to compute the kernel L times and then taking the average.

The gradient of the objective function over $\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}$ is

$$\nabla_{\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}}\ell_1 = \frac{\gamma_\epsilon}{L}\sum_l^L[\frac{1}{c_2}(-\boldsymbol{X}^{(l)}\boldsymbol{Q}_1^k + (\mathbf{1}_{n_1}^T\boldsymbol{Q}_1^k)\boldsymbol{U}_{.k}^{(l)}) + \frac{2}{c_3}(-\boldsymbol{U}^{(l)}\boldsymbol{Q}_2^k + \mathbf{1}_K^T\boldsymbol{Q}_2^k\boldsymbol{U}_{.k}^{(l)})] - \gamma_u\boldsymbol{\mu}_{\boldsymbol{U}_{.k}} \quad (51)$$

where $\mathbf{1}_{n_1} \in \mathbb{R}^{n_1}, \mathbf{1}_K \in \mathbb{R}^K$ are all-one vectors. $\boldsymbol{Q}_1^k = (-\mathbb{E}[\boldsymbol{V}_{k.}]^T \odot \boldsymbol{k}_{XU}^1)$, $\boldsymbol{Q}_2^k = (\frac{1}{2}\mathbb{E}[\boldsymbol{V}]\mathbb{E}[\boldsymbol{V}_{k.}]^T \odot \boldsymbol{k}_{UU})$, $\boldsymbol{k}_{XU}^1$ and $\boldsymbol{k}_{UU}$ are the kth column in $\boldsymbol{\mathcal{K}}_{XU}^{(l)}$ and $\boldsymbol{\mathcal{K}}_{UU}^{(l)}$, respectively. Computing the kernels $\boldsymbol{\mathcal{K}}_{XU}^{(l)}$ and $\boldsymbol{\mathcal{K}}_{UU}^{(l)}$ at each sub-iteration of the gradient ascent is computationally expensive. We approximate $\boldsymbol{\mathcal{K}}_{XU}^{(l)}$ and $\boldsymbol{\mathcal{K}}_{UU}^{(l)}$ by $\mathbb{E}[\boldsymbol{\mathcal{K}}_{XU}]$ and $\mathbb{E}[\boldsymbol{\mathcal{K}}_{UU}]$, respectively.

As discussed in Ref. [32], the term $\boldsymbol{X}\boldsymbol{Q}_1^k$ in (51) is nearly a constant and can be neglected when computing Hessian. The approximate Hessian of the objective function over $\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}$ is

$$\nabla^2_{\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}}\ell_1 = [\gamma_\epsilon(\frac{1}{c_2}(\mathbf{1}_{n_1}^T\boldsymbol{Q}_1^k) - \frac{2}{c_3}(q_2^k - \mathbf{1}_K^T\boldsymbol{Q}_2^k)) - \gamma_u]\boldsymbol{I}_{mn_2}, \quad (52)$$

where q_2^k is the kth entry in vector $\boldsymbol{Q}_2^k$.

We employ the relaxed Newton method [32] to achieve a faster convergence rate. The step size of the relaxed Newton method is $(\nabla^2_{\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}}\ell_1)^{-1}\nabla_{\boldsymbol{\mu}_{\boldsymbol{U}_{.k}}}\ell_1$. Compared with the standard Newton method, the relaxed Newton method has a relaxed hyper-parameter before the step size when computing the gradient.

The steepest gradient ascent is employed to update $a_{\boldsymbol{U}_{.k}}$. The gradient of the objective function over $a_{\boldsymbol{U}_{.k}}$ is

$$\begin{aligned}&\nabla_{a_{\boldsymbol{U}_{.k}}}\ell \\ &= 0.5\exp(0.5a_{\boldsymbol{U}_{.k}})\epsilon^T\frac{\gamma_\epsilon}{L}\sum_l^L[\frac{1}{c_2}(-\boldsymbol{X}^{(l)}\boldsymbol{Q}_1^k + (\mathbf{1}_n^T\boldsymbol{Q}_1^k)\boldsymbol{U}_{.k}^{(l)}) \\ &+\frac{2}{c_3}(-\boldsymbol{U}^{(l)}\boldsymbol{Q}_2^k + \mathbf{1}_K^T\boldsymbol{Q}_2^k\boldsymbol{U}_{.k}^{(l)})] + \frac{1}{2}mn_2(1-\gamma_u\exp(a_{\boldsymbol{U}_{.k}})).\end{aligned} \quad (53)$$

(III) The approximate posterior distribution of $\boldsymbol{X}_{.q}$ follows a Gaussian distribution (for all $q = 1, ..., n_1$).

$$q(\boldsymbol{X}_{.q}) \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}, \boldsymbol{\Sigma}_{\boldsymbol{X}_{.q}}), \quad (54)$$

where the variance $\boldsymbol{\Sigma}_{\boldsymbol{X}_{.q}}$ is a diagonal matrix and all diagonal entries ${\sigma^j_{\boldsymbol{X}_{.q}}}^2$ are the same. The prior distribution $p(\boldsymbol{X}_{.q})$ satisfies a Gaussian distribution $\mathcal{N}(0, \frac{1}{\gamma_x}\boldsymbol{I}_{mn_2})$.

Plug in the $p(\boldsymbol{X}_{.q})$ and $q(\boldsymbol{X}_{.q})$, the negative KL divergence can be straightforwardly derived as

$$\begin{aligned}&-\mathbb{KL}(q(\boldsymbol{X}_{.q})|p(\boldsymbol{X}_{.q})) \\ &= -\int q(\boldsymbol{X}_{.q})\ln\frac{q(\boldsymbol{X}_{.q})}{p(\boldsymbol{X}_{.q})}d\boldsymbol{X}_{.q}. \\ &= \frac{mn_2}{2}\ln(\gamma_x) + \frac{1}{2}\ln(|\boldsymbol{\Sigma}_{\boldsymbol{X}_{.q}}|) + \frac{mn_2}{2} - \frac{1}{2}\gamma_x\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}^T\boldsymbol{\mu}_{\boldsymbol{X}_{.q}} \\ &- \frac{1}{2}\gamma_x\text{trace}(\boldsymbol{\Sigma}_{\boldsymbol{X}_{.q}}) \\ &= \frac{mn_2}{2}\ln(\gamma_x) - \frac{1}{2}\gamma_x\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}^T\boldsymbol{\mu}_{\boldsymbol{X}_{.q}} + \frac{1}{2}\sum_{j=1}^{mn_2}\ln({\sigma^j_{\boldsymbol{X}_{.q}}}^2) + \frac{mn_2}{2} \\ &- \frac{1}{2}\gamma_x\sum_{j=1}^{mn_2}{\sigma^j_{\boldsymbol{X}_{.q}}}^2 \\ &= \frac{mn_2}{2}\ln(\gamma_x) - \frac{1}{2}\gamma_x\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}^T\boldsymbol{\mu}_{\boldsymbol{X}_{.q}} + \frac{1}{2}mn_2b_{\boldsymbol{X}_{.q}} + \frac{mn_2}{2} \\ &- \frac{1}{2}mn_2\gamma_x\exp(b_{\boldsymbol{X}_{.q}})\end{aligned} \quad (55)$$

where $b_{\boldsymbol{X}_{.q}}$ is the logarithm of the variance and $b_{\boldsymbol{X}_{.q}} = \ln({\sigma^j_{\boldsymbol{X}_{.q}}}^2)$ for all $j = 1, 2, ...mn_2$.

$$\begin{aligned}&q(\boldsymbol{X}_{.q}) \\ &\overset{\text{(h)}}{=} \arg\max_{q(\boldsymbol{X}_{.q})}\int q(\boldsymbol{X}_{.q})\mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{X}_{.q})}[\ln p(\boldsymbol{\Theta}, \boldsymbol{Y}, \boldsymbol{Y}_\Omega^o)]d\boldsymbol{X}_{.q} \\ &-\int q(\boldsymbol{X}_{.q})\ln q(\boldsymbol{X}_{.q})d\boldsymbol{X}_{.q}. \\ &\overset{\text{(i)}}{=} \arg\max_{q(\boldsymbol{X}_{.q})}\int q(\boldsymbol{X}_{.q})\mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{X}_{.q}))}[\ln p(\boldsymbol{\Theta}, \boldsymbol{Y}, \boldsymbol{Y}_\Omega^o|\boldsymbol{X}_{.q})]d\boldsymbol{X}_{.q} \\ &-\mathbb{KL}(q(\boldsymbol{X}_{.q})|p(\boldsymbol{X}_{.q})). \\ &\overset{\text{(j)}}{=} \arg\max_{q(\boldsymbol{X}_{.q})}\frac{1}{L}\sum_l^L\mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{X}_{.q}))}[\ln p(\boldsymbol{\Theta}, \boldsymbol{Y}, \boldsymbol{Y}_\Omega^o|\boldsymbol{X}_{.q}^{(l)})] \\ &-\mathbb{KL}(q(\boldsymbol{X}_{.q})|p(\boldsymbol{X}_{.q})). \\ &\overset{\text{(k)}}{=} \arg\max_{q(\boldsymbol{X}_{.q})}\frac{1}{L}\sum_l^L\mathbb{E}_{q(\boldsymbol{\Theta}\setminus\boldsymbol{X}_{.q}))}[\ln p(\boldsymbol{\Theta}, \boldsymbol{Y}, \boldsymbol{Y}_\Omega^o|\boldsymbol{X}_{.q}^{(l)})] \\ &-\frac{1}{2}\gamma_x\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}^T\boldsymbol{\mu}_{\boldsymbol{X}_{.q}} + \frac{1}{2}mn_2b_{\boldsymbol{X}_{.q}} - \frac{1}{2}mn_2\gamma_x\exp(b_{\boldsymbol{X}_{.q}}) + \text{const.} \\ &\overset{\text{(l)}}{=} \arg\max_{q(\boldsymbol{X}_{.q})} -\frac{1}{L}\sum_l^L(\mathbb{E}[\frac{\gamma_\epsilon}{2}||\Phi(\boldsymbol{X}^{(l)})_{.q} - \Phi(\boldsymbol{U})\boldsymbol{V}_{.q}||_2^2] \\ &+\frac{\mathbb{E}[\gamma_y]}{2}\mathbb{E}[||\boldsymbol{Y}_{.j}^o - \mathcal{P}_{\Omega_j}(\mathcal{H}^\dagger\boldsymbol{X}^{(l)} + \boldsymbol{E})_{.j}||_2^2]) - \frac{1}{2}\gamma_x\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}^T\boldsymbol{\mu}_{\boldsymbol{X}_{.q}} + \frac{1}{2}mn_2b_{\boldsymbol{X}_{.q}} \\ &-\frac{1}{2}mn_2\gamma_x\exp(b_{\boldsymbol{X}_{.q}}) + \text{const.} \\ &\overset{\text{(m)}}{=} \arg\max_{q(\boldsymbol{X}_{.q})} -\frac{1}{L}\sum_l^L(\frac{\gamma_\epsilon}{2}\mathbb{E}[||\Phi(\boldsymbol{X}^{(l)})_{.q} - \Phi(\boldsymbol{U})\boldsymbol{V}_{.q}||_2^2] \\ &+\frac{\mathbb{E}[\gamma_y]}{2}\mathbb{E}[||\mathcal{H}(\boldsymbol{Y}_{\boldsymbol{\Omega}}^o - \boldsymbol{E})_{.q} - \mathcal{P}_{\mathcal{H}\Omega_q}\boldsymbol{X}_{.q}^{(l)}||_2^2]) - \frac{1}{2}\gamma_x\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}^T\boldsymbol{\mu}_{\boldsymbol{X}_{.q}} + \frac{1}{2}mn_2b_{\boldsymbol{X}_{.q}} \\ &-\frac{1}{2}mn_2\gamma_x\exp(b_{\boldsymbol{X}_{.q}}) + \text{const.} \\ &= \arg\max_{q(\boldsymbol{X}_{.q})} -\frac{1}{L}\sum_l^L(\frac{\gamma_\epsilon}{2}\mathbb{E}[\text{trace}(\Phi(\boldsymbol{X}^{(l)})_{.q}^T\Phi(\boldsymbol{X}^{(l)})_{.q} - 2\Phi(\boldsymbol{X}^{(l)})_{.q}^T\Phi(\boldsymbol{U})\boldsymbol{V}_{.q} \\ &+\boldsymbol{V}_{.q}^T\Phi(\boldsymbol{U})^T\Phi(\boldsymbol{U})\boldsymbol{V}_{.q})] + \frac{\mathbb{E}[\gamma_y]}{2}\mathbb{E}[||\mathcal{H}(\boldsymbol{Y}_{\boldsymbol{\Omega}}^o - \boldsymbol{E})_{.q} - \mathcal{P}_{\mathcal{H}\Omega_q}\boldsymbol{X}_{.q}^{(l)}||_2^2]) - \frac{1}{2}\gamma_x\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}^T\boldsymbol{\mu}_{\boldsymbol{X}_{.q}} \\ &+\frac{1}{2}mn_2b_{\boldsymbol{X}_{.q}} - \frac{1}{2}mn_2\gamma_x\exp(b_{\boldsymbol{X}_{.q}}) + \text{const.} \\ &\overset{\text{(n)}}{=} \arg\max_{q(\boldsymbol{X}_{.q})} -(\frac{\gamma_\epsilon}{2}\mathbb{E}[\text{trace}(\frac{1}{L}\sum_l^L\boldsymbol{\mathcal{K}}_{xx}^{q(l)} - 2(\frac{1}{L}\sum_l^L\boldsymbol{\mathcal{K}}_{XU}^{q(l)})\boldsymbol{V}_{.q} + \boldsymbol{V}_{.q}^T\boldsymbol{\mathcal{K}}_{UU}\boldsymbol{V}_{.q})] \\ &+\frac{\mathbb{E}[\gamma_y]}{2}\mathbb{E}[||\mathcal{H}(\boldsymbol{Y}_{\boldsymbol{\Omega}}^o - \boldsymbol{E})_{.q} - \mathcal{P}_{\mathcal{H}\Omega_q}\boldsymbol{X}_{.q}^{(l)}||_2^2]) - \frac{1}{2}\gamma_x\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}^T\boldsymbol{\mu}_{\boldsymbol{X}_{.q}} + \frac{1}{2}mn_2b_{\boldsymbol{X}_{.q}} \\ &-\frac{1}{2}mn_2\gamma_x\exp(b_{\boldsymbol{X}_{.q}}) + \text{const.} \\ &\overset{\text{(o)}}{\approx} \arg\max_{q(\boldsymbol{X}_{.q})} -\frac{\gamma_x}{2}\text{trace}(-2\mathbb{E}[\boldsymbol{\mathcal{K}}_{XU}^q]\mathbb{E}[\boldsymbol{V}_q]) - \frac{\mathbb{E}[\gamma_y]}{2}\frac{1}{L}\sum_l^L||\mathcal{H}(\boldsymbol{Y}_{\boldsymbol{\Omega}}^o - \mathbb{E}[\boldsymbol{E}])_{.q} - \mathcal{P}_{\mathcal{H}\Omega_q}\boldsymbol{X}_{.q}^{(l)}||_2^2 \\ &-\frac{1}{2}\gamma_x\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}^T\boldsymbol{\mu}_{\boldsymbol{X}_{.q}} + \frac{1}{2}mn_2b_{\boldsymbol{X}_{.q}} - \frac{1}{2}mn_2\gamma_x\exp(b_{\boldsymbol{X}_{.q}}) + \text{const.} \\ &= \arg\max_{q(\boldsymbol{X}_{.q})}\ell_2\ (\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}, b_{\boldsymbol{X}_{.q}}),\end{aligned} \quad (56)$$

One can replace the latent variable $\boldsymbol{\Theta}_i$ in (48) with $\boldsymbol{X}_{.q}$ to get steps (h) and (i) in (56). Step (j) follows the Monte-Carlo approximation. The reparameterization trick [37] is also employed here to make the Monte-Carlo estimation differentiable with respect to $\boldsymbol{X}_{.q}$. The reparametrization of $\boldsymbol{X}_{.q}$ is $\boldsymbol{X}_{.q} = \boldsymbol{\mu}_{\boldsymbol{X}_{.q}} + \exp(0.5b_{\boldsymbol{X}_{.q}})\boldsymbol{\epsilon}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_{mn_2})$, where $\exp(0.5b_{\boldsymbol{X}_{.q}})$ is the standard deviation. $\boldsymbol{\epsilon}$ is the auxiliary noise vector. Step (k) follows the definition of KL divergence in (55). Steps (l) to (n) follow from the straightforward expansion and we use kernels to replace the inner product of mapping functions. At step (l), $\mathcal{P}_\Omega(.)$ is the sampling operator with $(\mathcal{P}_\Omega(Y))_{ij} = Y_{i,j}$ if $(i,j) \in \Omega$ and 0 otherwise. $\mathcal{P}_{\Omega_j}$ denotes the jth column in $\mathcal{P}_\Omega$. At step (d), $\mathcal{H}\Omega$ denotes the index set of observed entries in the constructed Hankel matrix. $\mathcal{P}_{\mathcal{H}\Omega}(.)$ is the sampling operator with $(\mathcal{P}_{\mathcal{H}\Omega}(\boldsymbol{X}))_{ij} = \boldsymbol{X}_{i,j}$ if

$(i,j) \in \mathcal{H}\Omega$ and 0 otherwise. $\mathcal{P}_{\mathcal{H}\Omega_q}$ denotes the qth column in $\mathcal{P}_{\mathcal{H}\Omega}$. At step (n), $\boldsymbol{\mathcal{K}}^q_{xx}$ is a scaler and denotes the qth diagonal element in $\boldsymbol{\mathcal{K}}_{XX}$. $\boldsymbol{\mathcal{K}}^q_{XU} \in \mathbb{R}^{1\times K}$ denotes the qth row in $\boldsymbol{\mathcal{K}}_{XU}$. At step (o), $\boldsymbol{\mathcal{K}}^q_{xx}$ is removed because it is a scaler. $\mathbb{E}[\boldsymbol{\mathcal{K}}_{XU}]$ is approximated by sampling $\boldsymbol{X}$ and $\boldsymbol{U}$ to compute the kernel L times and then taking the average.

The gradient of the objective function over $\boldsymbol{\mu_X}$ is

$$\begin{aligned}&\nabla_{\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}}\ell_2 = \tfrac{1}{L}\textstyle\sum_l^L[\tfrac{1}{c_2}\gamma_\epsilon(-\boldsymbol{U}^{(l)}\boldsymbol{Q}^q_3 + \boldsymbol{X}^{(l)}_{.q}\mathbf{1}^T_K\boldsymbol{Q}^q_3)\\ &+\gamma_y(\mathcal{H}(\boldsymbol{Y}^o_{\boldsymbol{\Omega}} - \boldsymbol{E})_{.q} - \mathcal{P}_{\mathcal{H}\Omega_q}\boldsymbol{X}^{(l)}_{.q}) \odot \mathcal{P}_{\mathcal{H}\Omega_q})] - \gamma_x\boldsymbol{\mu}_{\boldsymbol{X}_{.q}},\end{aligned} \tag{57}$$

where $\boldsymbol{Q}^q_3 = (-\mathbb{E}[\boldsymbol{V}_{.q}] \odot \boldsymbol{k}^{qT}_{XU})$, $\boldsymbol{k}^q_{XU} \in \mathbb{R}^{1\times K}$ denotes the qth row in $\boldsymbol{\mathcal{K}}^{(l)}_{XU}$. We approximate $\boldsymbol{\mathcal{K}}^{(l)}_{XU}$ by $\mathbb{E}[\boldsymbol{\mathcal{K}}_{XU}]$.
The approximate Hessian of objective function over $\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}$ is

$$\nabla^2_{\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}}\ell_2 = (\frac{1}{c_2}\gamma_\epsilon(\mathbf{1}^T_K\boldsymbol{Q}^q_3) - \gamma_x)\boldsymbol{I}_{mn_2} - \gamma_y\text{Diag}(\mathcal{P}_{\mathcal{H}\Omega_q}), \tag{58}$$

where the diagonal matrix $\text{Diag}(\mathcal{P}_{\mathcal{H}\Omega_q}) = (\mathcal{P}_{\mathcal{H}\Omega_q}\mathbf{1}^T_{mn_2}) \odot \boldsymbol{I}_{mn_2}$, where $\mathbf{1}_{mn_2} \in \mathbb{R}^{mn_2}$ and $\odot$ represents the element-wise product. The diagonal entries of $\text{Diag}(\mathcal{P}_{\mathcal{H}\Omega_q})$ are constructed from the vector $\mathcal{P}_{\mathcal{H}\Omega_q}$. The relaxed Newton method is also employed to update $\boldsymbol{\mu}_{\boldsymbol{x}_{.q}}$ The step size of the relaxed Newton method is $(\nabla^2_{\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}}\ell_2)^{-1}\nabla_{\boldsymbol{\mu}_{\boldsymbol{X}_{.q}}}\ell_2$.

The steepest gradient ascent is employed to update $b_{\boldsymbol{X}_{.q}}$. The gradient of the objective function ℓ_2 over $b_{\boldsymbol{X}_{.q}}$ is

$$\begin{aligned}&\nabla_{b_{\boldsymbol{X}_{.q}}}\ell_2\\ &= 0.5\text{exp}(0.5b_{\boldsymbol{X}_{.q}})\boldsymbol{\epsilon}^T\tfrac{1}{L}\textstyle\sum_l^L[\tfrac{1}{c_2}\gamma_\epsilon(-\boldsymbol{U}^{(l)}\boldsymbol{Q}^q_3 + \boldsymbol{X}^{(l)}_{.q}\mathbf{1}^T_K\boldsymbol{Q}^q_3)\\ &+\gamma_y\mathcal{P}_{\mathcal{H}\Omega_q} \odot (\mathcal{H}(\boldsymbol{Y}^o_{\boldsymbol{\Omega}} - \boldsymbol{E})_{.q} - \mathcal{P}_{\mathcal{H}\Omega_q}\boldsymbol{X}^{(l)}_{.q})] + \tfrac{1}{2}mn_2(1 - \gamma_x\text{exp}(b_{\boldsymbol{X}_{.q}})).\end{aligned} \tag{59}$$

(IV) The approximate probabilistic distribution of $E_{i,j}$ follows a Gaussian distribution (for all $(i,j) \in \boldsymbol{\Omega}$).
Note that

$$\begin{aligned}&p(E_{i,j}|-)\\ &\propto \mathcal{N}(Y_{i,j}|(\mathcal{H}^\dagger\boldsymbol{X})_{i,j} + E_{i,j}, \frac{1}{\gamma_y})\mathcal{N}(E_{i,j}|0, \frac{1}{\beta_{i,j}})\\ &\propto (\gamma_y)^{\frac{1}{2}}\text{exp}(\tfrac{-\gamma_y}{2}(E_{i,j} - (Y_{i,j} - (\mathcal{H}^\dagger\boldsymbol{X})_{i,j}))^2)\text{exp}(\tfrac{-\beta_{i,j}}{2}E^2_{i,j})\\ &\propto \text{exp}(\tfrac{-(\gamma_y+\beta_{i,j})}{2}E^2_{i,j} + \gamma_yE_{i,j}(Y_{i,j} - (\mathcal{H}^\dagger\boldsymbol{X})_{i,j}) - \tfrac{\gamma_y}{2}(Y_{i,j} - (\mathcal{H}^\dagger\boldsymbol{X})_{i,j})^2).\end{aligned} \tag{60}$$

Then the logarithm of $q(E_{i,j})$ is

$$\begin{aligned}&\ln(q(E_{i,j}))\\ &= \mathbb{E}_{\boldsymbol{\Theta}\setminus E_{i,j}}[\ln p(\boldsymbol{Y},\boldsymbol{\Theta})] + \text{const.}\\ &= \mathbb{E}_{\boldsymbol{\Theta}\setminus E_{i,j}}[\ln\ p(\boldsymbol{Y}|\boldsymbol{U},\boldsymbol{V},\boldsymbol{X},\gamma_y)p(E_{i,j})] + \text{const.}\\ &= \mathbb{E}_{\boldsymbol{\Theta}\setminus E_{i,j}}[\ln\ \mathcal{N}(Y_{i,j}|(\mathcal{H}^\dagger\boldsymbol{X})_{i,j} + E_{i,j}, \tfrac{1}{\gamma_y})\mathcal{N}(E_{i,j}|0, \tfrac{1}{\beta_{i,j}})] + \text{const.}\\ &= \mathbb{E}[\frac{-(\gamma_y+\beta_{i,j})}{2}E^2_{i,j} + \gamma_yE_{i,j}(Y_{i,j} - (\mathcal{H}^\dagger\boldsymbol{X})_{i,j})\\ &- \frac{\gamma_y}{2}(Y_{i,j} - (\mathcal{H}^\dagger\boldsymbol{X})_{i,j})^2] + \text{const.}\\ &= \tfrac{-(\mathbb{E}[\gamma_y]+\mathbb{E}[\beta_{i,j}])}{2}E^2_{i,j} + \mathbb{E}[\gamma_y]E_{i,j}(Y_{i,j} - \mathbb{E}[(\mathcal{H}^\dagger\boldsymbol{X})_{i,j}]\\ &- \mathbb{E}[\frac{\gamma_y}{2}(Y_{i,j} - (\mathcal{H}^\dagger\boldsymbol{X})_{i,j})^2))] + \text{const.}\end{aligned} \tag{61}$$

Then $q(E_{i,j})$ follows a Gaussian distribution with mean $\mu_{E_{i,j}}$ and variance $\Sigma_{E_{i,j}}$, i.e.,

$$q(E_{i,j}) \sim \mathcal{N}(\mu_{E_{i,j}}, \Sigma_{E_{i,j}}), \tag{62}$$

where

$$\mu_{E_{i,j}} = \mathbb{E}[\gamma_y]\Sigma_{E_{i,j}}(Y^o_{i,j} - (\mathcal{H}^\dagger\mathbb{E}[\boldsymbol{X}])_{i,j}), \tag{63}$$

$$\Sigma_{E_{i,j}} = \frac{1}{\mathbb{E}[\gamma_y] + \mathbb{E}[\beta_{i,j}]}. \tag{64}$$

(V) The approximate distribution $\beta_{i,j}$ follows a Gamma distribution (for all $(i,j) \in \boldsymbol{\Omega}$).
Because

$$\begin{aligned}\Gamma(\beta_{i,j}|g_0,h_0) &= \frac{h^{g_0}_0(\beta_{i,j})^{g_0-1}e^{-h_0\beta_{i,j}}}{\Gamma(g_0)}\\ &\propto (\beta_{i,j})^{g_0-1}e^{-h_0\beta_{i,j}},\end{aligned} \tag{65}$$

Also note that

$$\mathcal{N}(E_{i,j}|0, \frac{1}{\beta_{i,j}}) \propto (\beta_{i,j})^{\frac{1}{2}}\text{exp}(\frac{-\beta_{i,j}}{2}E^2_{i,j}). \tag{66}$$

From (65) and (65), we can get

$$\begin{aligned}p(\beta_{i,j}|-) &\propto \mathcal{N}(E_{i,j}|0, \frac{1}{\beta_{i,j}})\Gamma(\beta_{i,j}|g_0,h_0)\\ &\propto (\beta_{i,j})^{\frac{1}{2}+g_0-1}\text{exp}(-\beta_{i,j}(\frac{1}{2}E^2_{i,j} + h_0)).\end{aligned} \tag{67}$$

Thus

$$\begin{aligned}&\ln(q(\beta_{i,j}))\\ &= \mathbb{E}_{\boldsymbol{\Theta}\setminus\beta_{i,j}}[\ln\ p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}^o_\Omega)] + \text{const.}\\ &= \mathbb{E}_{\boldsymbol{\Theta}\setminus\beta_{i,j}}[\ln\ p(E_{i,j}|\beta_{i,j})p(\beta_{i,j})] + \text{const.}\\ &= \mathbb{E}_{\boldsymbol{\Theta}\setminus\beta_{i,j}}[\ln\ \mathcal{N}(E_{i,j}|0, \frac{1}{\beta_{i,j}})\Gamma(\beta_{i,j}|g_0,h_0)] + \text{const.}\\ &= [(\frac{1}{2} + g_0 - 1)\ln(\beta_{i,j}) - \beta_{i,j}(\frac{1}{2}\mathbb{E}[E^2_{i,j}] + h_0)] + \text{const..}\end{aligned} \tag{68}$$

where $\mathbb{E}[E^2_{i,j}] = \mathbb{E}[E_{i,j}]^2 + \Sigma_{E_{i,j}}$.
This reveals that the $\beta_{i,j}$ is from a Gamma distribution.

$$q(\beta_{i,j}) \sim \Gamma(\frac{1}{2} + g_0, \frac{1}{2}\mathbb{E}[E^2_{i,j}] + h_0), \tag{69}$$

and its mean is

$$\mathbb{E}[\beta_{i,j}] = \frac{\frac{1}{2} + g_0}{\frac{1}{2}\mathbb{E}[E^2_{i,j}] + h_0}, \tag{70}$$

where $\mathbb{E}[E^2_{i,j}] = \mathbb{E}[E_{i,j}]^2 + \boldsymbol{\Sigma}_{E_{i,j}}$.
(VI) The approximate posterior distribution of γ_y follows a Gamma distribution.
Note that

$$\Gamma(\gamma_y|e_0,f_0) = \frac{f^{e_0}_0(\gamma_y)^{e_0-1}e^{-f_0\gamma_y}}{\Gamma(e_0)} \propto (\gamma_y)^{e_0-1}e^{-f_0\gamma_y}, \tag{71}$$

Also

This article has been accepted for publication in IEEE Transactions on Power Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2023.3254909

$$\begin{aligned}
&\prod_{j=1}^{n}\mathcal{N}(\boldsymbol{Y}_{\cdot j}^{o}|\mathcal{P}_{\boldsymbol{\Omega}_j}(\mathcal{H}^{\dagger}\boldsymbol{X}+\boldsymbol{E})_{\cdot j},\frac{1}{\gamma_y}\boldsymbol{I}_{|\boldsymbol{\Omega}_j|})\\
&\propto \textstyle\prod_{j=1}^{n}\frac{1}{(2\pi)^{1/2}\sqrt{|\gamma_y|^{-|\boldsymbol{\Omega}_j|}}}\exp(\frac{-\gamma_y}{2}||\boldsymbol{Y}_{\cdot j}^{o}-\mathcal{P}_{\Omega_j}(\mathcal{H}^{\dagger}\boldsymbol{X}+\boldsymbol{E})_{\cdot j}||_2^2)\\
&\propto \prod_{j=1}^{n}(\gamma_y)^{\frac{|\boldsymbol{\Omega}_j|}{2}}\exp(\frac{-\gamma_y}{2}||\boldsymbol{Y}_{\cdot j}^{o}-\mathcal{P}_{\Omega_j}(\mathcal{H}^{\dagger}\boldsymbol{X}+\boldsymbol{E})_{\cdot j}||_2^2)\\
&\propto (\gamma_y)^{\frac{|\boldsymbol{\Omega}|}{2}}\exp(\frac{-\gamma_y}{2}||\boldsymbol{Y}_{\boldsymbol{\Omega}}^{o}-\mathcal{P}_{\Omega}(\mathcal{H}^{\dagger}\boldsymbol{X}+\boldsymbol{E})||_F^2).
\end{aligned}\tag{72}$$

Combine (71) and (72), we can obtain

$$\begin{aligned}
p(\gamma_y|-)&\propto \textstyle\prod_{j=1}^{n}\mathcal{N}(Y_{\cdot j}^{o}|\mathcal{P}_{\boldsymbol{\Omega}_j}(\mathcal{H}^{\dagger}\boldsymbol{X}+\boldsymbol{E})_{\cdot j},\frac{1}{\gamma_y}\boldsymbol{I}_{|\boldsymbol{\Omega}_j|})\Gamma(\gamma_y|e_0,f_0)\\
&\propto (\gamma_y)^{\frac{|\boldsymbol{\Omega}|}{2}}\exp(\frac{-\gamma_y}{2}||\boldsymbol{Y}_{\boldsymbol{\Omega}}^{o}-\mathcal{P}_{\Omega}(\mathcal{H}^{\dagger}\boldsymbol{X}+\boldsymbol{E})||_F^2).
\end{aligned}\tag{73}$$

Then the following derivations are straightforward expansions. The $\ln(q(\gamma_y))$ can be derived as

$$\begin{aligned}
&\ln(q(\gamma_y))\\
&=\mathbb{E}_{\boldsymbol{\Theta}\setminus\gamma_y}[\ln\, p(\boldsymbol{\Theta},\boldsymbol{Y},\boldsymbol{Y}_{\Omega}^{o})]+\text{const.}\\
&=\mathbb{E}_{\boldsymbol{\Theta}\setminus\gamma_y}[\ln p(\boldsymbol{Y}_{\boldsymbol{\Omega}}^{o}|\boldsymbol{X},\boldsymbol{E},\gamma_y)P(\gamma_y)]+\text{const.}\\
&=\mathbb{E}_{\boldsymbol{\Theta}\setminus\gamma_y}[\ln\textstyle\prod_{j=1}^{n}\mathcal{N}(\boldsymbol{Y}_{\cdot j}^{o}|(\mathcal{H}^{\dagger}\boldsymbol{X}+\boldsymbol{E})_{\cdot j},\frac{1}{\gamma_y})\Gamma(\gamma_y|e_0,f_0)]+\text{const.}\\
&=\mathbb{E}_{\boldsymbol{\Theta}\setminus\gamma_y}[\ln(\gamma_y)^{\frac{|\boldsymbol{\Omega}|}{2}}\exp(\tfrac{-\gamma_y}{2}\textstyle\sum_{(i,j)\in\Omega}(Y_{i,j}^{o}-E_{i,j}-\frac{1}{\kappa_j}\sum_{(u,v)\in\Psi_{i,j}}X_{u,v})^2)]\\
&+\ln((\gamma_y)^{e_0-1}e^{-f_0\gamma_y})+\text{const.}\\
&=\mathbb{E}_{\boldsymbol{\Theta}\setminus\gamma_y}[(\tfrac{|\boldsymbol{\Omega}|}{2}+e_0-1)\ln(\gamma_y)-f_0\gamma_y+(\tfrac{-\gamma_y}{2}\textstyle\sum_{(i,j)\in\Omega_i}(Y_{i,j}^{o}-E_{i,j}\\
&-\frac{1}{\kappa_j}\sum_{(u,v)\in\Psi_{i,j}}X_{u,v})^2)]+\text{const.}\\
&=[(\tfrac{|\boldsymbol{\Omega}|}{2}+e_0-1)\ln(\gamma_y)+(\tfrac{-\gamma_y}{2}\mathbb{E}[||\boldsymbol{Y}_{\boldsymbol{\Omega}}^{o}-\mathcal{P}_{\Omega}(\mathcal{H}^{\dagger}\boldsymbol{X}+\boldsymbol{E})||_F^2])\\
&+\text{const.}\\
&=[(\tfrac{|\boldsymbol{\Omega}|}{2}+e_0-1)\ln(\gamma_y)+(\tfrac{-\gamma_y}{2}\textstyle\sum_{(i,j)\in\Omega_i}\mathbb{E}[Y_{i,j}^{o}-E_{i,j}\\
&-\frac{1}{\kappa_j}\sum_{(u,v)\in\Psi_{i,j}}X_{u,v})^2]+\text{const.},
\end{aligned}\tag{74}$$

with

$$\begin{aligned}
&\mathbb{E}[(Y_{i,j}^{o}-E_{i,j}-\tfrac{1}{\kappa_j}\textstyle\sum_{(u,v)\in\Psi_{i,j}}X_{u,v})^2]\\
&=\mathbb{E}[(Y_{i,j}^{o}-E_{i,j})^2-2(Y_{i,j}-E_{i,j})\frac{1}{\kappa_j}\sum_{(u,v)\in\Psi_{i,j}}X_{u,v}+(\frac{1}{\kappa_j}\sum_{(u,v)\in\Psi_{i,j}}X_{u,v})^2]\\
&\approx (Y_{i,j}^{o}-\mathbb{E}[E_{i,j}])^2+\Sigma_{E_{i,j}}-2(Y_{i,j}^{o}-\mathbb{E}[E_{i,j}])\mathbb{E}[X_{u,v}]+\mathbb{E}[X_{u,v}^2]\\
&=(Y_{i,j}^{o}-\mathbb{E}[E_{i,j}])^2-2(Y_{i,j}^{o}-\mathbb{E}[E_{i,j}])\mathbb{E}[X_{u,v}]+\mathbb{E}[X_{u,v}]^2+\text{Var}(X_{u,v})\\
&=(Y_{i,j}^{o}-\mathbb{E}[E_{i,j}]-\mathbb{E}[X_{u,v}])^2+\Sigma_{E_{i,j}}+\text{Var}(X_{u,v})\\
&\approx (Y_{i,j}^{o}-\mathbb{E}[E_{i,j}]-\tfrac{1}{\kappa_j}\textstyle\sum_{(u,v)\in\Psi_{i,j}}\mathbb{E}[X_{u,v}])^2+\Sigma_{E_{i,j}}+\frac{1}{\kappa_j}\sum_{(u,v)\in\Psi_{i,j}}\text{Var}(X_{u,v}).
\end{aligned}\tag{75}$$

where $\text{Var}(X_{u,v})$ denotes the variance of $X_{u,v}$.
This reveals that the γ_y follows a Gamma distribution.

$$q(\gamma_y)\sim\Gamma(\tfrac{|\boldsymbol{\Omega}|}{2}+e_0,\tfrac{1}{2}\mathbb{E}[||\boldsymbol{Y}_{\boldsymbol{\Omega}}^{o}-P_{\Omega}(\mathcal{H}^{\dagger}\boldsymbol{X}+\boldsymbol{E})||_F^2]+f_0),\tag{76}$$

and its mean is

$$\mathbb{E}[\gamma_y]=\frac{\frac{|\boldsymbol{\Omega}|}{2}+e_0}{\frac{1}{2}\mathbb{E}[||\boldsymbol{Y}_{\boldsymbol{\Omega}}^{o}-P_{\Omega}(\mathcal{H}^{\dagger}\boldsymbol{X}+\boldsymbol{E})||_F^2]+f_0}.\tag{77}$$

D. Data Recovery and Uncertainty Index

We define a set $\boldsymbol{\psi}=\{\boldsymbol{X},\gamma_y\}$ to represent the related latent variables for the estimation of $Y_{i,j}$. The predictive mean is derived as follows:

$$\begin{aligned}
\mathbb{E}[Y_{i,j}]&=\int p(Y_{i,j}|\boldsymbol{Y}_{\Omega}^{o})Y_{i,j}dY_{i,j}\\
&=\int(\int p(Y_{i,j}|\boldsymbol{\psi})p(\boldsymbol{\psi}|\boldsymbol{Y}_{\Omega}^{o})d\boldsymbol{\psi})Y_{i,j}dY_{i,j}\\
&=\int(\int p(Y_{i,j}|\boldsymbol{\psi})Y_{i,j}dY_{i,j})p(\boldsymbol{\psi}|\boldsymbol{Y}_{\Omega}^{o})d\boldsymbol{\psi}\\
&=\int\mathbb{E}_{p(Y_{i,j}|\boldsymbol{\psi})}[Y_{i,j}]p(\boldsymbol{\psi}|\boldsymbol{Y}_{\Omega}^{o})d\boldsymbol{\psi}\\
&=\int(\mathcal{H}^{\dagger}\boldsymbol{X})p(\boldsymbol{\psi}|\boldsymbol{Y}_{\Omega}^{o})d\boldsymbol{\psi}\\
&\approx\frac{1}{J}\sum_{l=1}^{J}(\mathcal{H}^{\dagger}\boldsymbol{X}^{(l)})_{i,j}\quad\boldsymbol{X}^{(l)}\sim q(\boldsymbol{X}|\boldsymbol{Y}_{\Omega}^{o}).
\end{aligned}\tag{78}$$

where $\mathbb{E}_{p(Y_{i,j}|\boldsymbol{\psi})}[Y_{i,j}]$ denotes the expectation of $Y_{i,j}$ over the probability $p(Y_{i,j}|\boldsymbol{\psi})$. Each $\boldsymbol{X}^{(l)}$ is sampled J times from the learned posterior distributions. The Monte Carlo integration is employed in the last step in equation (78) to approximate the exact integration.

The $\mathbb{E}[Y_{i,j}^2]$ is computed as follows:

$$\begin{aligned}
&\mathbb{E}_{p(Y_{i,j}|\boldsymbol{Y}_{\Omega}^{o})}[Y_{i,j}^2]\\
&=\int p(Y_{i,j}|\boldsymbol{Y}_{\Omega}^{o})Y_{i,j}^2dY_{i,j}\\
&=\int(\int p(Y_{i,j}|\boldsymbol{\psi})p(\boldsymbol{\psi}|\boldsymbol{Y}_{\Omega}^{o})d\boldsymbol{\psi})Y_{i,j}^2dY_{i,j}\\
&=\int(\int p(Y_{i,j}|\boldsymbol{\psi})Y_{i,j}^2dY_{i,j})p(\boldsymbol{\psi}|\boldsymbol{Y}_{\Omega}^{o})d\boldsymbol{\psi}\\
&=\int(\mathbb{E}_{p(Y_{i,j}|\boldsymbol{\psi})}[Y_{i,j}^2])p(\boldsymbol{\psi}|\boldsymbol{Y}_{\Omega}^{o})d\boldsymbol{\psi}\\
&=\int(\text{Var}_{p(Y_{i,j}|\boldsymbol{\psi})}[Y_{i,j}]+\mathbb{E}_{p(Y_{i,j}|\boldsymbol{\psi})}^2[Y_{i,j}]))p(\boldsymbol{\psi}|\boldsymbol{Y}_{\Omega}^{o})d\boldsymbol{\psi}\\
&=\int(\frac{1}{\gamma_y}+(\mathcal{H}^{\dagger}\boldsymbol{X})_{i,j}^2)p(\boldsymbol{\psi}|\boldsymbol{Y}_{\Omega}^{o})d\boldsymbol{\psi}\\
&\approx\frac{1}{J}\sum_{l=1}^{J}\frac{1}{\gamma_y^{(l)}}+\frac{1}{J}\sum_{l=1}^{J}(\mathcal{H}^{\dagger}\boldsymbol{X}^{(l)})_{i,j}^2,
\end{aligned}\tag{79}$$

where each $\gamma_y^{(l)}$ is sampled from the learned posterior distribution $q(\gamma_y|\boldsymbol{Y}_{\Omega}^{o})$. The predictive variance is computed by combining (78) and (79), i.e.,

$$\begin{aligned}
&\text{Var}[Y_{i,j}]=\mathbb{E}[Y_{i,j}^2]-\mathbb{E}[Y_{i,j}]^2\\
&\approx\frac{1}{J}\sum_{l=1}^{J}\frac{1}{\gamma_y^{(l)}}+\frac{1}{J}\sum_{l=1}^{J}(\mathcal{H}^{\dagger}\boldsymbol{X}^{(l)})_{i,j}^2-(\frac{1}{J}\sum_{l=1}^{J}(\mathcal{H}^{\dagger}\boldsymbol{X}^{(l)})_{i,j})^2.
\end{aligned}\tag{80}$$

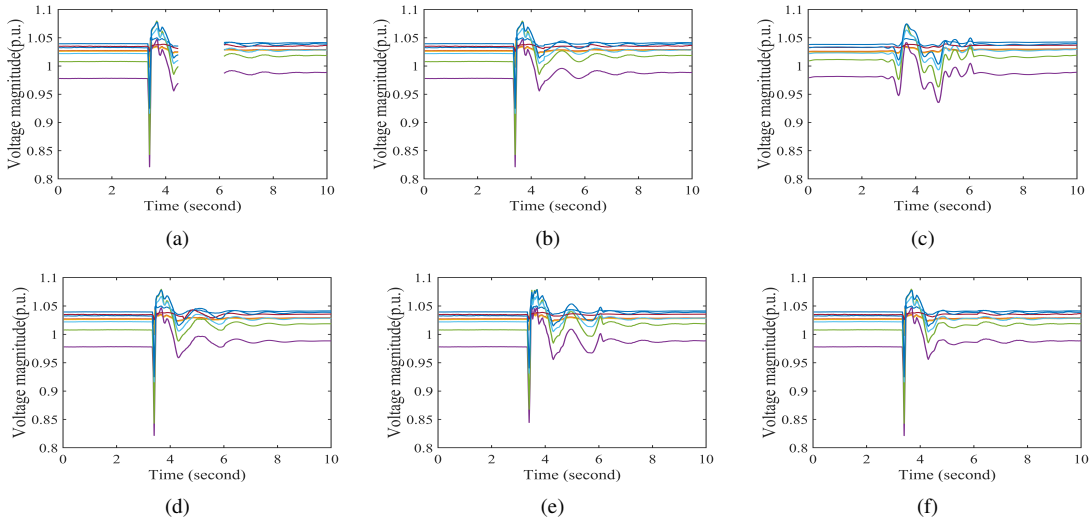


Fig. 10: The recovery performance on 16.7% M3 missing data. (a) the observed data, (b) the estimated data by the proposed Ba-NSDR method, (c) the estimated data by the BHMC method, (d) the estimated data by the BHMC-S method, (e) the estimated data by the AM-FIHT method, (f) the estimated data by the SDR-K method.

E. Parameter Settings of Baseline Methods for Synthetic Data

We listed some key parameters of baseline methods for synthetic data. The Hankel parameters $n_2 = 30$ if the missing data mode is M3 and $n_2 = 20$ for other modes if not otherwise stated. We tuned these parameters to achieve good results. Some of them may not be the optimal parameters. The key parameters are as follows:

- KMC: the rank is 100.
- AM-FIHT: the rank is 6.
- BHMC: $e_0 = 10^{-6}$, $f_0 = 10^{-6}$ the rank is 6;
- BHMC-S: $e_0 = 10^{-6}$, $f_0 = 10^{-3}$, the window length is 60, the step size is 1, the rank is 8;
- SDR-K: the window length is 20 for M3 mode and 10 for other cases, $n_2 = 8$, the rank is 5;
- SAP: the rank is 6;
- BRHMC: $e_0 = 10^{-6}$, $f_0 = 10^{-6}$, $g_0 = 10^{-6}$, $h_0 = 10^{-6}$, the rank is 10;
- BRHMC-S: $e_0 = 10^{-6}$, $f_0 = 10^{-3}$, $g_0 = 0.2$, $h_0 = 10^{-6}$, the window length is 50, the rank is 10.

F. Computational Complexity

The computational complexities for computing $\mathcal{K}_{XU}$ and $\mathcal{K}_{UU}$ are $\mathcal{O}(mn_2n_1K)$ and $\mathcal{O}(mn_2K^2)$, respectively. The most expensive parts in each iteration are computing $\boldsymbol{V}$, $\boldsymbol{U}$, and $\boldsymbol{X}$. The computational complexity for updating $\boldsymbol{V}$ is $\mathcal{O}(mn_2n_1K + mn_2K^2 + K^3 + K^2n_1)$, and because n_1 is usually much larger than K, the complexity could be reduced to $\mathcal{O}(mn_2n_1K)$. The computational complexity for updating $\boldsymbol{U}$ is $\mathcal{O}(Lmn_2n_1Kt_2^{\max} + Lmn_2K^2t_2^{\max})$. The computational complexity for updating $\boldsymbol{X}$ is $\mathcal{O}(Lmn_2n_1Kt_4^{\max} + Lmn_2n_1t_4^{\max})$. Because $t_2^{\max}$ and $t_4^{\max}$ are set as the same value in this paper, we use $t^{\max}$ to represent the maximum iterations $t_2^{\max}$ and $t_4^{\max}$ of inner loops for brevity. The total computational complexity is $\mathcal{O}(Lmn_2n_1Kt^{\max})$. One can see that the computational complexity scales at most linearly regarding the size of the Hankel matrix.

G. Additional Experiments

1) Additional synthetic dataset: In this section, we evaluated the data recovery performance on synthetic data with phase added to the sinusoids. Each entry $Y_{i,j}$ in $\boldsymbol{Y}$ is generated by

$$Y_{i,j} = \sum_{k=1}^{r} b_{k,j} e^{-a_i t_j} \sin(2\pi f_{k,j} t_j + \alpha_{k,j}) \quad i = 1, ..., m, j = 1, ..., n, \tag{81}$$

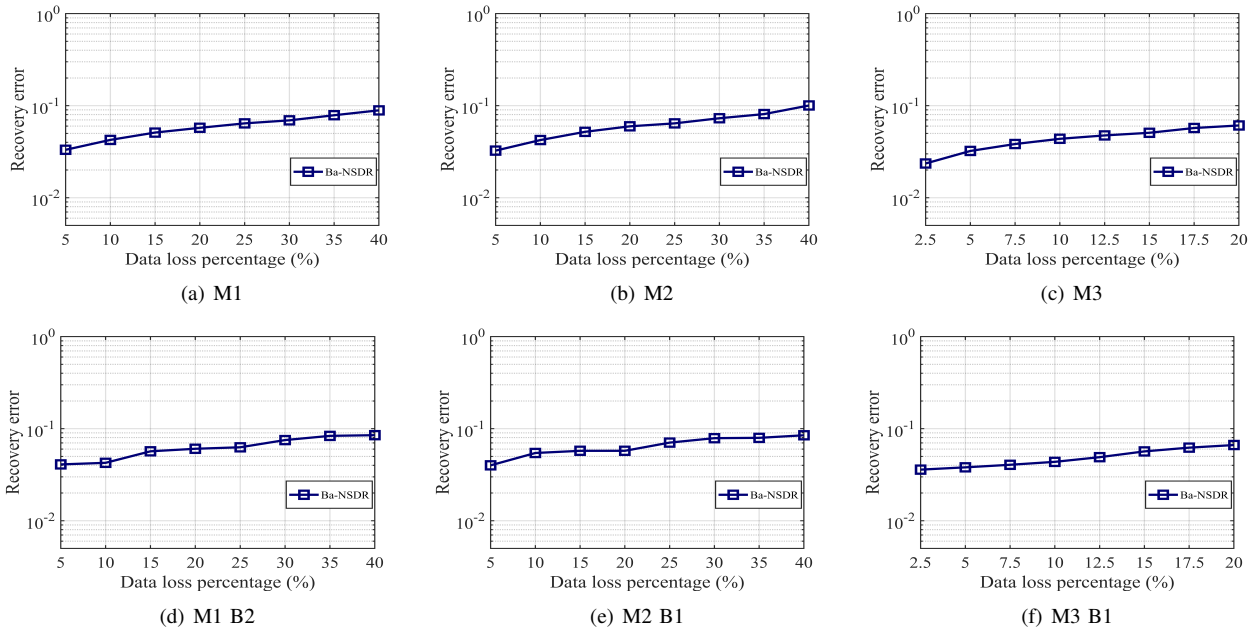
The problem setup is the same with (32) except that an extra phase is added. The phase term $\phi_{k,j}$ is also time-varying and is randomly selected from $(\frac{\pi}{12}, \frac{\pi}{6})$. Fig. 11 shows the recovery results. One can see from Fig. 11 that the recovery results of the Ba-NSDR method are comparable to Fig. 6. This verifies that our algorithm is not sensitive to the extra phase.

2) Extra recorded PMU dataset: We provide an additional case study on another recorded event, which is a transformer failure in the central New York power system. The event is shown in Fig. 12. Some parameters are set as follows: $n_2 = 80$, $c_2 = c_3 = 60$, $f_0 = 10^{-4}$.

- Case 3: 16.7% data are removed following Mode M3 on this event. The length of M3 missing data is 50 consecutive time instants, which correspond to 1.67 seconds.

Fig. 10 compares the recovery performance of our proposed Ba-NSDR method with BHMC, BHMC-S, AM-FIHT, SDR-K methods on Case 3. Table XI reports the NEE over the whole ten-second window, the NEE of a window between 3-7 seconds where missing data occur, denoted by NEE_{3-7}, and the computational time of these methods. Ba-NSDR achieves a great balance of recovery accuracy and computational cost. In Fig. 10 (d), BHMC-S can also recover the disturbances with slightly worse recovery performance than our method. However, the computational cost of BHMC-S is much higher than the proposed method. In Table XI, we can see that

This article has been accepted for publication in IEEE Transactions on Power Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2023.3254909



(a) M1 (b) M2 (c) M3

(d) M1 B2 (e) M2 B1 (f) M3 B1

Fig. 11: The recovery performance of the Ba-NSDR method on synthetic sinusoids with a phase. (a)-(c) show the missing data recovery results with three missing modes. (d)-(f) show the recovery results with both missing and bad data.

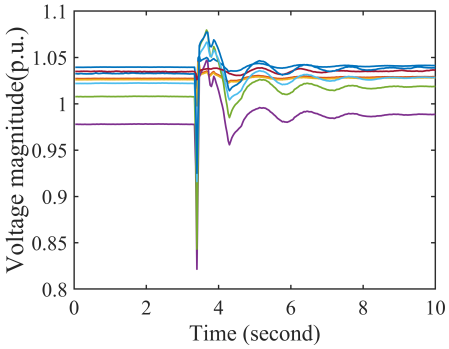


Fig. 12: The measurements of voltage magnitude

the computational time of BHMC-S is 50 times as large as required by our algorithm.

Table XI: The recovery performance on 16.7% M3 mode on the second event

method	Ba-NSDR	BHMC	BHMC-S	AM-FIHT	SDR-K
NEE	8.0×10^{-4}	7.7×10^{-3}	9.6×10^{-4}	3.2×10^{-3}	2.0×10^{-3}
NEE_{3-7}	1.3×10^{-3}	1.2×10^{-2}	1.5×10^{-3}	5.0×10^{-3}	3.2×10^{-3}
Time(sec.)	60.3	25.2	3036.1	14.7	2.1