Supplementary Material for *Block Hankel Tensor ARIMA for Multiple Short Time Series Forecasting*

Qiquan Shi,[1] Jiaming Yin,[2] * Jiajun Cai,[3] * Andrzej Cichocki,[4] Tatsuya Yokota,[5]
Lei Chen,[1] Mingxuan Yuan,[1] Jia Zeng[1]
[1]Huawei Noah's Ark Lab, Hong Kong, China, [2]Tongji Univeristy, Shanghai, China
[3]The University of Hong Kong, Hong Kong, China, [4]The Skolkovo Institute of Science and Technology, Moscow, Russia
[5]Nagoya Institute of Technology, Nagoya, Japan; RIKEN Center for Advanced Intelligence Project, Japan
[1]{shi.qiquan, lc.leichen, yuan.mingxuan, zeng.jia}@huawei.com, 14jiamingyin@tongji.edu.cn, jjcai@connect.hku.hk,
a.Cichocki@skoltech.ru, t.yokota@nitech.ac.jp

Appendix A: Derivation of BHT-ARIMA

Appendix A.1: Cases of Applying MDT

In this paper, we mainly handle two types of datasets: second-order and third-order original tensors. The situation of applying MDT on them are:

(i) $N = 1$: given a matrix $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times T}$ where each row refers to a *single univariate TS* and each column includes all the TS data points at one time point. By applying MDT along the temporal mode, we get **a third-order block Hankel tensor** $\widehat{\boldsymbol{\mathcal{X}}} \in \mathbb{R}^{J_1 \times J_2 \times \widehat{T}}$ where $M = 2, J_1 = I_1, J_2 = \tau, \widehat{T} = T - \tau + 1$, as shown in Fig. 2 in the main manuscript.

(ii) $N = 2$: given a third-order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times T}$, like a video, where each tube refers to a frame. Similarly using MDT along the temporal mode, we get **a fourth-order block Hankel tensor** $\widehat{\boldsymbol{\mathcal{X}}} \in \mathbb{R}^{J_1 \times J_2 \times J_3 \times \widehat{T}}$ where $M = 3, J_1 = I_1, J_2 = I_2, J_3 = \tau, \widehat{T} = T - \tau + 1$.

Appendix A.2: Detailed Derivation of Step 2 of BHT-ARIMA

In this step, we generalize the classical ARIMA to tensor form and incorporate it into Tucker decomposition. Specifically, with the block Hankel tensor, we compute its order-d differencing to get $\{\Delta^d \widehat{\boldsymbol{\mathcal{X}}}_t\}_{t=d}^{\widehat{T}}$. We then employ Tucker decomposition for $\{\Delta^d \widehat{\boldsymbol{\mathcal{X}}}_t\}$ by projecting it to core tensors $\{\Delta^d \widehat{\boldsymbol{\mathcal{G}}}_t\}$ using joint orthogonal factor matrices $\{\widehat{\mathbf{U}}^{(m)}\}$, i.e.,

$$\begin{aligned} \Delta^d \widehat{\boldsymbol{\mathcal{G}}}_t &= \Delta^d \widehat{\boldsymbol{\mathcal{X}}}_t \times_1 \widehat{\mathbf{U}}^{(1)^\top} \cdots \times_M \widehat{\mathbf{U}}^{(M)^\top} \\ \text{s.t. } & \widehat{\mathbf{U}}^{(m)^\top} \widehat{\mathbf{U}}^{(m)} = \mathbf{I}, m = 1, \ldots, M, \end{aligned} \tag{1}$$

where the projection matrices $\{\widehat{\mathbf{U}}^{(m)} \in \mathbb{R}^{J_m \times R_m}\}$ maximally preserve the temporal continuity between core tensors and the low-rank core tensors $\{\Delta^d \widehat{\boldsymbol{\mathcal{G}}}_t \in \mathbb{R}^{R_1 \times \cdots \times R_M}\}$ represent the most important information of original Hankel tensors and reflect the intrinsic interactions between TS.

Then, it would be more promising to train a good forecasting model directly using core tensors *explicitly* instead of whole tensors. To retain the temporal correlations among core tensors, we generalize a (p, d, q)-order ARIMA model to tensor form and use it to build connections between the current core tensor $\Delta^d \widehat{\boldsymbol{\mathcal{G}}}_t$ and the previous core tensors $\Delta^d \widehat{\boldsymbol{\mathcal{G}}}_{t-1}, \Delta^d \widehat{\boldsymbol{\mathcal{G}}}_{t-2}, \ldots, \Delta^d \widehat{\boldsymbol{\mathcal{G}}}_{t-p}$ as follows:

$$\Delta^d \widehat{\boldsymbol{\mathcal{G}}}_t = \sum_{i=1}^{p} \alpha_i \Delta^d \widehat{\boldsymbol{\mathcal{G}}}_{t-i} - \sum_{i=1}^{q} \beta_i \widehat{\boldsymbol{\mathcal{E}}}_{t-i} + \widehat{\boldsymbol{\mathcal{E}}}_t, \tag{2}$$

where the $\{\alpha_i\}$ and $\{\beta_i\}$ are the coefficients of AR and MA, respectively, and $\{\widehat{\boldsymbol{\mathcal{E}}}_{t-i}\}$ are the random errors of past q observations. In model (2), $\widehat{\boldsymbol{\mathcal{E}}}_t$ is the forecast error at the current time point, which should be minimized to optimal zero. We thus can derive the following objective function:

$$\begin{aligned} \min_{\{\widehat{\boldsymbol{\mathcal{G}}}_t, \widehat{\mathbf{U}}^{(m)}, \widehat{\boldsymbol{\mathcal{E}}}_{t-i}, \alpha_i, \beta_i\}} & \sum_{t=s+1}^{\widehat{T}} \Bigg(\frac{1}{2} \Big\| \Delta^d \widehat{\boldsymbol{\mathcal{G}}}_t - \sum_{i=1}^{p} \alpha_i \Delta^d \widehat{\boldsymbol{\mathcal{G}}}_{t-i} + \sum_{i=1}^{q} \beta_i \widehat{\boldsymbol{\mathcal{E}}}_{t-i} \Big\|_F^2 \\ & + \frac{1}{2} \Big\| \Delta^d \widehat{\boldsymbol{\mathcal{G}}}_t - \Delta^d \widehat{\boldsymbol{\mathcal{X}}}_t \times_1 \widehat{\mathbf{U}}^{(1)^\top} \cdots \times_M \widehat{\mathbf{U}}^{(M)^\top} \Big\|_F^2 \Bigg) \\ \text{s.t. } & \widehat{\mathbf{U}}^{(m)^\top} \widehat{\mathbf{U}}^{(m)} = \mathbf{I}, m = 1, \ldots, M, \end{aligned} \tag{3}$$

where $s = p + d + q$ is the sum of ARIMA orders, and is also the minimum input length of each TS. Next, we solve this problem using augmented Lagrangian methods. To facilitate the derivation of (3), we reformulate the optimization problem by unfolding each tensor variable along mode-m:

$$\begin{aligned} \min_{\{\widehat{\mathbf{G}}_t^{(m)}, \widehat{\mathbf{E}}_{t-i}^{(m)}, \widehat{\mathbf{U}}^{(m)}, \alpha_i, \beta_i\}} & \sum_{t=s+1}^{\widehat{T}} \sum_{m=1}^{M} \Bigg(\frac{1}{2} \Big\| \Delta^d \widehat{\mathbf{G}}_t^{(m)} - \sum_{i=1}^{p} \alpha_i \Delta^d \widehat{\mathbf{G}}_{t-i}^{(m)} \\ & + \sum_{i=1}^{q} \beta_i \widehat{\mathbf{E}}_{t-i}^{(m)} \Big\|_F^2 + \frac{1}{2} \Big\| \Delta^d \widehat{\mathbf{G}}_t^{(m)} - \widehat{\mathbf{U}}^{(m)^\top} \widehat{\mathbf{X}}_t^{(m)} \widehat{\mathbf{U}}^{(-m)^\top} \Big\|_F^2 \Bigg) \\ \text{s.t. } & \widehat{\mathbf{U}}^{(m)^\top} \widehat{\mathbf{U}}^{(m)} = \mathbf{I}, m = 1, \ldots, M, \end{aligned} \tag{4}$$

where $\widehat{\mathbf{U}}^{(-m)} = \widehat{\mathbf{U}}^{(M)^\top} \otimes \cdots \widehat{\mathbf{U}}^{(m+1)^\top} \otimes \widehat{\mathbf{U}}^{(m-1)^\top} \otimes \cdots \widehat{\mathbf{U}}^{(1)^\top} \in \mathbb{R}^{\prod_{j \neq m} R_j \times \prod_{j \neq m} I_j}$. In the following, we can update each target variable using closed-form solutions

*Work done during an internship at Huawei.

Update $\widehat{\mathbf{G}}_t^{(m)}$ Equation (4) with respect to $\Delta^d\widehat{\mathbf{G}}_t^{(m)}$ is:

$$\min_{\{\widehat{\mathbf{G}}_t^{(m)}\}} \sum_{t=s+1}^{\widehat{T}} \sum_{m=1}^{M} \Bigg(\frac{1}{2} \Big\| \Delta^d\widehat{\mathbf{G}}_t^{(m)} - \sum_{i=1}^{p} \alpha_i \Delta^d \widehat{\mathbf{G}}_{t-i}^{(m)} + \sum_{i=1}^{q} \beta_i \widehat{\mathbf{E}}_{t-i}^{(m)} \Big\|_F^2 + \frac{1}{2} \Big\| \Delta^d\widehat{\mathbf{G}}_t^{(m)} - \widehat{\mathbf{U}}^{(m)\top} \widehat{\mathbf{X}}_t^{(m)} \widehat{\mathbf{U}}^{(-m)\top} \Big\|_F^2 \Bigg). \tag{5}$$

Computing the partial derivation of this cost function with respect to $\widehat{\mathbf{G}}_t^{(m)}$ and equalize it to zero, we get:

$$\sum_{t=s+1}^{\widehat{T}} \sum_{m=1}^{M} \Bigg(2\Delta^d\widehat{\mathbf{G}}_t^{(m)} - \Big(\sum_{i=1}^{p} \alpha_i \Delta^d \widehat{\mathbf{G}}_{t-i}^{(m)} - \sum_{i=1}^{q} \beta_i \widehat{\mathbf{E}}_{t-i}^{(m)} \Big) - \widehat{\mathbf{U}}^{(m)\top} \widehat{\mathbf{X}}_t^{(m)} \widehat{\mathbf{U}}^{(-m)\top} \Bigg) = 0. \tag{6}$$

Thus, we can update $\widehat{\mathbf{G}}_t^{(m)}$ by:

$$\Delta^d\widehat{\mathbf{G}}_t^{(m)} = \frac{1}{2} \Big(\widehat{\mathbf{U}}^{(m)\top} \widehat{\mathbf{X}}_t^{(m)} \widehat{\mathbf{U}}^{(-m)\top} + \sum_{i=1}^{p} \alpha_i \Delta^d \widehat{\mathbf{G}}_{t-i}^{(m)} - \sum_{i=1}^{q} \beta_i \widehat{\mathbf{E}}_{t-i}^{(m)} \Big), \tag{7}$$

and then we can fold it into $\Delta^d \widehat{\boldsymbol{\mathcal{G}}}_{t-i}$ by Fold$\Big(\Delta^d\widehat{\mathbf{G}}_t^{(m)}\Big)$.

Update $\widehat{\mathbf{U}}^{(m)}$ Equation (4) with respect to $\widehat{\mathbf{U}}^{(m)}$ is:

$$\min_{\{\widehat{\mathbf{U}}^{(m)}\}} \sum_{t=s+1}^{\widehat{T}} \sum_{m=1}^{M} \frac{1}{2} \Big\| \Delta^d\widehat{\mathbf{G}}_t^{(m)} - \widehat{\mathbf{U}}^{(m)\top} \widehat{\mathbf{X}}_t^{(m)} \widehat{\mathbf{U}}^{(-m)\top} \Big\|_F^2 \tag{8}$$
$$\text{s.t. } \widehat{\mathbf{U}}^{(m)\top} \widehat{\mathbf{U}}^{(m)} = \mathbf{I}, m = 1, \cdots, M.$$

The minimization of (8) over the matrices $\widehat{\mathbf{U}}^{(m)}$ with orthonormal columns is equivalent to the maximization of the following problem (Shang, Liu, and Cheng 2014):

$$\widehat{\mathbf{U}}^{(m)} = \arg\max \sum_{t=s+1}^{\widehat{T}} \textbf{trace}\Big(\widehat{\mathbf{U}}^{(m)\top} \widehat{\mathbf{X}}_t^{(m)} \widehat{\mathbf{U}}^{(-m)\top} \Delta^d \widehat{\mathbf{G}}_t^{(m)\top} \Big), \tag{9}$$

where the problem (9) is actually the well-known orthogonality Procrustes problem (Higham and Papadimitriou 1995), whose global optimal solution is given by the SVD of $\sum_{t=s+1}^{\widehat{T}} \widehat{\mathbf{X}}_t^{(m)} \widehat{\mathbf{U}}^{(-m)\top} \Delta^d \widehat{\mathbf{G}}_t^{(m)\top}$, i.e.,

$$\widehat{\mathbf{U}}^{(m)} = \widehat{\mathbf{U}^{*}}^{(m)} (\widehat{\mathbf{V}^{*}}^{(m)})^\top \tag{10}$$

where $\widehat{\mathbf{U}^{*}}^{(m)}$ and $\widehat{\mathbf{V}^{*}}^{(m)}$ are the left and right singular vectors of SVD of $\sum_{t=s+1}^{\widehat{T}} \widehat{\mathbf{X}}_t^{(m)} \widehat{\mathbf{U}}^{(-m)\top} \Delta^d \widehat{\mathbf{G}}_t^{(m)\top}$, respectively.

Discussion: relaxed-orthogonality We empirically explored that relaxing the full-orthogonality used in Tucker decomposition by removing the orthogonal constraint along the last mode (viewed as the temporal mode of each $\widehat{\mathcal{X}}_t$ in the embedded space). This strategy probably relaxes the heavy constraints on temporal smoothness and thus would make the proposed model more flexible and robust to variability of parameters.

We relax the last mode $\widehat{\mathbf{U}}^{(M)}$ without orthogonality constraints: we compute the partial derivation of Eq. (8) with respect to $\widehat{\mathbf{U}}^{(M)}$ and equalize it to zero, we get:

$$\sum_{t=s+1}^{\widehat{T}} \Big(\widehat{\mathbf{X}}_t^{(M)} \widehat{\mathbf{U}}^{(-M)^\dagger} (\widehat{\mathbf{X}}_t^{(M)} \widehat{\mathbf{U}}^{(-M)^\dagger})^\top \widehat{\mathbf{U}}^{(M)} - \widehat{\mathbf{X}}_t^{(M)} \widehat{\mathbf{U}}^{(-M)^\dagger} \Delta^d \widehat{\mathbf{G}}_t^{(M)\top} \Big) = 0. \tag{11}$$

Thus, we can update the non-orthogonal $\widehat{\mathbf{U}}^{(M)}$ by

$$\widehat{\mathbf{U}}^{(M)} = \Big(\sum_{t=s+1}^{\widehat{T}} \widehat{\mathbf{X}}_t^{(M)} \widehat{\mathbf{U}}^{(-M)^\dagger} (\widehat{\mathbf{X}}_t^{(M)} \widehat{\mathbf{U}}^{(-M)^\dagger})^\top \Big)^{-1} \Big(\sum_{t=s+1}^{\widehat{T}} \widehat{\mathbf{X}}_t^{(M)} \widehat{\mathbf{U}}^{(-M)^\dagger} \Delta^d \widehat{\mathbf{G}}_t^{(M)\top} \Big). \tag{12}$$

Update $\widehat{\mathbf{E}}_{t-i}^{(m)}$ Equation (4) with respect to $\widehat{\mathbf{E}}_{t-i}^{(m)}$ is:

$$\min_{\{\widehat{\mathbf{E}}_{t-i}^{(m)}\}} \sum_{t=s+1}^{\widehat{T}} \sum_{m=1}^{M} \frac{1}{2} \Big\| \Delta^d\widehat{\mathbf{G}}_t^{(m)} - \sum_{i=1}^{p} \alpha_i \Delta^d \widehat{\mathbf{G}}_{t-i}^{(m)} + \sum_{i=1}^{q} \beta_i \widehat{\mathbf{E}}_{t-i}^{(m)} \Big\|_F^2. \tag{13}$$

Computing the partial derivation of Eq. (13) with respect to $\widehat{\mathbf{G}}_t^{(m)}$ and equalize it to zero, we get:

$$\sum_{t=s+1}^{\widehat{T}} \sum_{m=1}^{M} \Big(\Delta^d\widehat{\mathbf{G}}_t^{(m)} - \sum_{i=1}^{p} \alpha_i \Delta^d \widehat{\mathbf{G}}_{t-i}^{(m)} + \sum_{j\neq i}^{q} \beta_j \widehat{\mathbf{E}}_{t-j}^{(m)} + \beta_i \widehat{\mathbf{E}}_{t-i}^{(m)} \Big) = 0. \tag{14}$$

Thus, we can update $\widehat{\mathbf{E}}_{t-i}^{(m)}$ by

$$\widehat{\mathbf{E}}_{t-i}^{(m)} = \frac{\sum_{t=s+1}^{\widehat{T}} \Big(\Delta^d\widehat{\mathbf{G}}_t^{(m)} - \sum_{i=1}^{p} \alpha_i \Delta^d \widehat{\mathbf{G}}_{t-i}^{(m)} + \sum_{j\neq i}^{q} \beta_j \widehat{\mathbf{E}}_{t-j}^{(m)} \Big)}{(s + 1 - \widehat{T})\beta_i}, \tag{15}$$

and then we can fold it into $\widehat{\boldsymbol{\mathcal{E}}}_{t-i}$ by Fold$\Big(\widehat{\mathbf{E}}_{t-i}^{(m)}\Big)$.

Update $\{\alpha_i\}, \{\beta_i\}$ Regarding the coefficient parameters $\{\alpha_i\}_{i=1}^{p}, \{\beta_i\}_{i=1}^{q}$ of AR and MA respectively in the objective function (4), we follow the classical ARIMA using Yule-Walker technique to achieve it. We have revised this function to support tensorial data and estimate the AR parameters from the core tensors based on a least squares modified Yule-Walker technique. Then, we estimate the MA parameters from the residual time series.

Appendix B: Experiments

To evaluate the performance of the proposed BHT-ARIMA, we have conducted experiments on five real-world datasets. We implement the proposed methods in Python, and all experiments are performed on a PC (Intel Xeon(R) 4.0-GHz, 512-GB memory).

Appendix B.1: Experimental Setup

Datasets We conduct experiments on five real-world datasets, including:

(i) Three publicly available datasets:

- **Traffic:** The traffic data is originally collected from California department of transportation and describes the road occupy rate of Los Angeles County highway network. We here use the same subset used in (Yu, Yin, and Zhu 2017) which selects 228 sensors randomly.
- **Electricity:** The electricity data records 321 clients' hourly electricity consumption(Lai et al. 2018) from the year 2012 to the year 2014. We merge the every 24 time point to obtain a daily interval TS dataset with size 321 × 1096.
- **Smoke Video** The smoke video is publicly free available and records the Smoke from the chimney of a factory taken in Hokkaido in 2007. We sample the video every two frame and resize the images to obtain a third-order TS dataset of size $36\times64\times100$.

(ii) Two industrial datasets obtained from a real-world supply chain of Huawei:

- **PC sales:** The PC dataset includes the weekly sales records of 9 personal computers from May, 2017 to March, 2019(105 weeks in total).
- **Raw materials** The are 2246 TS items of raw materials for making a product, each item has 24 monthly demand quantities.

We illustrate these datasets in Fig. 1 and Fig. 2 by randomly selecting some samples from them.

Compared Methods

We compared *nine* competing methods:

- **ARIMA** (Box and Jenkins 1968) is autoregressive integrated moving average model .
- **VAR** (Johansen 1995) is vector autoregressive.
- **Prophet** (Taylor and Letham 2018) is developed by Facebook researchers based on an additional model where non-linear trends are fit with yearly, weekly, and daily seasonality.
- **XGBoost** (Chen and Guestrin 2016) is a widely used machine learning model.
- **DeepAR** (Salinas, Flunkert, and Gasthaus 2017) is developed by Amazon researchers a probabilistic model based on an autoregressive recurrent neural network.
- **TT-RNN** (Yu et al. 2017) is the Tensor-Train decomposition integrated RNN model for long-term time series forecasting.
- **GRU**: Gated Recurrent Units (Cho et al. 2014)[1].
- **TRMF** (temporal regularized matrix factorization) (Yu, Rao, and Dhillon 2016) introduces graph-based AR regularizer into factorization frameworks.
- **MOAR** (Jing et al. 2018) is the multilinear orthogonal autoregressive model.

In addition, we use the block Hankel tensor via MDT as input of MOAR to get **BHT+MOAR** as an improvement of MOAR. It will be used to evaluate the effectiveness of MDT together with tensor decomposition.

Parameter Setting In our experiments, all datasets are splited into training sets (90%) and testing sets (10%). We conduct grid search over parameters for each model and dataset. For XGBoost, maximum depth is chosen from $\{3, 4, ..., 10\}$, the regularization coefficient λ is chosen from $\{0, 0.25, 0.5, 0.75, 1\}$, and the subsample is chosen from $\{0.5, 0.6, 0.7, 0.8, 0.9\}$. For deep models, the hidden size is chosen from $\{2^3, 2^4, 2^5\}$ and the number of layers is chosen from $\{1, 2, 3\}$. Specially for TT-RNN, the tensor rank is selected from $\{1, 2, ..., 8\}$. For TRMF, the regularization coefficient λ is chosen from $\{0.01, 0.1, 0, 1, 10, 100, 200, 300\}$. And for all tensor-based models, we set the maximum number of iterations as 30. For other parameters of the compared methods, we have tuned the parameters based on the original papers to obtain the best results under same experimental settings. For our BHT-ARIMA, we will show its sensitivity of parameters in the following.

Appendix B.2: Analysis of Parameters and Convergence

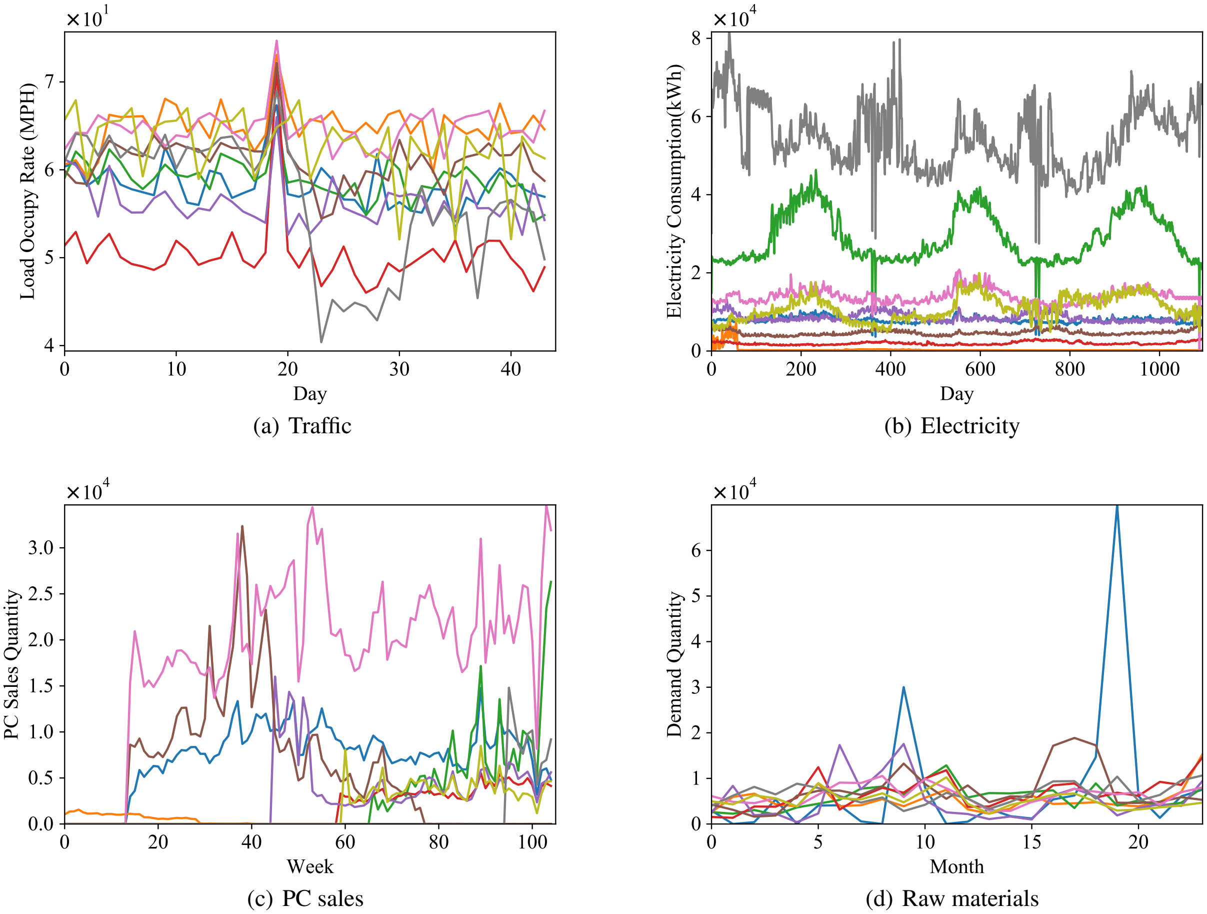
We here not only analyze the parameter sensitivity of BHT-ARIMA but also we verified the effects of that with relaxed-orthogonality by testing on the Raw material datasets with various values of parameters.

Sensitivity Analysis of Parameter τ, $\{R_n\}$ and (p, d, q) Fig. 3, 4 and 5 show the forecasting results using BHT-ARIMA with full-orthogonality (**FO**) versus (**vs.**) relaxed-orthogonality (**RO**) with different values of the parameters τ, $\{R_m\}$ and (p, d, q), respectively. Overall, BHT-ARIMA with RO is less sensitive and can achieve slightly better forecasting accuracy than that with FO. Particularly, two cases *(where BHT-ARIMA with FO can achieve better (similar) accuracy than that with RO)* are worth mentioning: 1) with very small MDT parameter τ (e.g. 1,2,5); 2) when the last mode rank R_M is equal (or near) to τ (that maximally preserves the temporal dependencies among consecutive core tensors). Besides, with the same differencing order $d = 1$ or 2, the performance of BHT-ARIMA is relatively sensitive to larger p. Moreover, the time cost of BHT-ARIMA with RO is larger than that of with FO due to the cost of computing Eq. (12) is larger than that of Eq. (10).

In short, BHT-ARIMA can be relatively robust to variability of parameters and we do not need to carefully tune the parameters, while we usually can obtain better results by setting smaller values such as $\{\ \tau = 2 - 4,\ R_M = \tau$ and small $(p = 1 - 5, d = 1, q = 1)\}$. Moreover, the Tucker-rank can be estimated automatically (Yokota, Lee, and Cichocki 2016; Shi, Lu, and Cheung 2017).

Effect of applying MDT on all modes / temporal mode As discussed in **Remark 1** in the main manuscript about why we apply MDT only along the temporal mode of TS

[1] We didn't show the comparison to Long Short-Term Memory (LSTM) as it yields similar results with GRU.



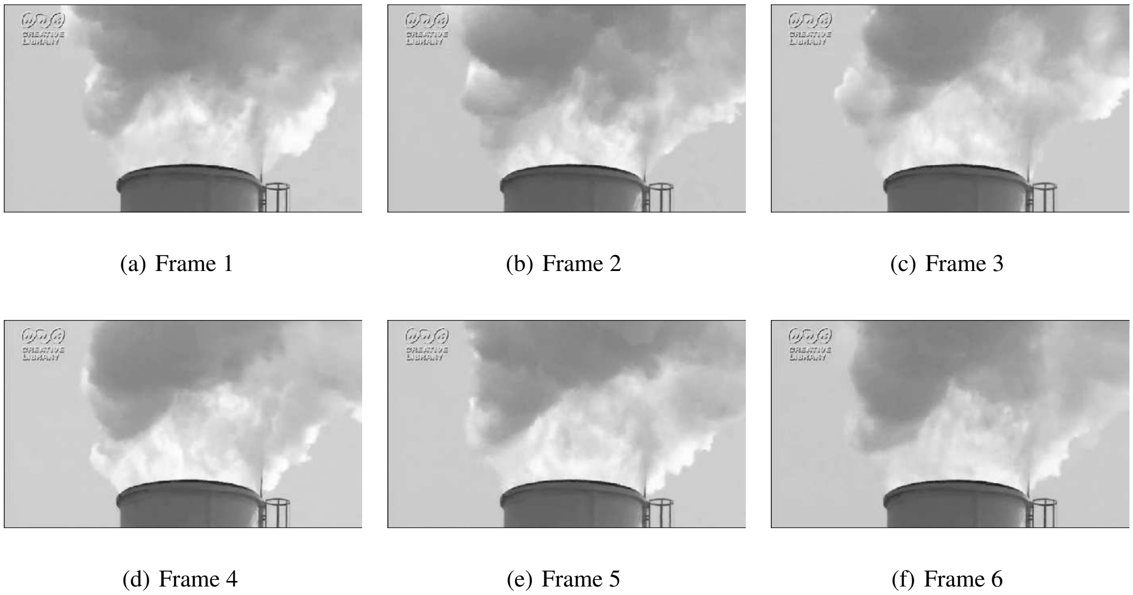
(a) Traffic

(b) Electricity

(c) PC sales

(d) Raw materials

Figure 1: Visualization of time series randomly selecting from four real-world TS datasets: Traffic, Electricity, PC sales and Raw materials, respectively.



(a) Frame 1

(b) Frame 2

(c) Frame 3

(d) Frame 4

(e) Frame 5

(f) Frame 6

Figure 2: Visualization of Smoke video time series (randomly selected frames).

data, we here verify it by testing on the Smoke video with different runs. As shown in Fig. 6: both types of applying MDT obtain similar forecasting accuracy while using MDT on all the modes costs more time due to computing higher-order tensors. This conclusion is also applicable for the cases of BHT-ARIMA with RO. These results support our assumption that these TS items usually do not have strong neighborhood relationships so it is unnecessary to applying MDT on other modes besides the temporal mode.

Convergence and maximum iteration K We study the convergence of BHT-ARIMA in terms of the relative error of projection matrices $\frac{\sum_{m=1}^{M} ||\widehat{\mathbf{U}}^{(m)^{k+1}} - \widehat{\mathbf{U}}^{(m)^{k}}||_F^2}{\sum_{m=1}^{M} ||\widehat{\mathbf{U}}^{(m)^{k+1}}||_F^2}$. Fig. 7(a) shows that both BHT-ARIMA with FO and RO converge quickly while the FO version converges more smoothly and faster within 10 iterations. Furthermore, setting maximum iteration $K > 5$ is enough to get a sufficient forecasting accuracy, as shown in Fig. 7(b). In this paper, we set $K = 10$ for BHT-ARIMA for all the tests.

Summary for Full- vs. Relaxed-orthogonality: In summary, the above-mentioned observations and analysis verify our **Discussion 1** in the main manuscript that: relaxing the orthogonality constraints of the projection matrices can consistently perform with reduced sensitivity to variability of parameters and obtain even slightly better results, although we use *BHT-ARIMA with FO by default* as it is general. Nevertheless, the performance of BHT-ARIMA (with FO) fluctuates in a narrow range. In other words, BHT-ARIMA can always outperform the SOTA methods under a wide range of parameter settings, which have been validated in the Table 1, 2 and 3, and Fig. 3 and 4 in the main manuscript.

References

Box, G. E., and Jenkins, G. M. 1968. Some recent advances in forecasting and control. *Journal of the Royal Statistical Society. Series C (Applied Statistics)* 17(2):91–109.

Chen, T., and Guestrin, C. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. ACM.

Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078.*

Higham, N., and Papadimitriou, P. 1995. Matrix Procrustes Problems. *Rapport technique, University of Manchester.*

Jing, P.; Su, Y.; Jin, X.; and Zhang, C. 2018. High-order temporal correlation model learning for time-series prediction. *IEEE Transactions on Cybernetics* 49(6):2385–2397.

Johansen, S. 1995. *Likelihood-based inference in cointegrated vector autoregressive models.* Oxford University Press on Demand.

Lai, G.; Chang, W.-C.; Yang, Y.; and Liu, H. 2018. Modeling long-and short-term temporal patterns with deep neural networks. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, 95–104. ACM.

Salinas, D.; Flunkert, V.; and Gasthaus, J. 2017. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *arXiv preprint arXiv:1704.04110.*

Shang, F.; Liu, Y.; and Cheng, J. 2014. Generalized higher-order tensor decomposition via parallel ADMM. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 1279–1285. AAAI Press.

Shi, Q.; Lu, H.; and Cheung, Y.-m. 2017. Tensor rank estimation and completion via CP-based nuclear norm. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 949–958. ACM.

Taylor, S. J., and Letham, B. 2018. Forecasting at scale. *The American Statistician* 72(1):37–45.

Yokota, T.; Lee, N.; and Cichocki, A. 2016. Robust multilinear tensor rank estimation using higher order singular value decomposition and information criteria. *IEEE Transactions on Signal Processing* 65(5):1196–1206.

Yu, R.; Zheng, S.; Anandkumar, A.; and Yue, Y. 2017. Long-term forecasting using Tensor-Train RNNs. *arXiv preprint arXiv:1711.00073.*

Yu, H.-F.; Rao, N.; and Dhillon, I. S. 2016. Temporal regularized matrix factorization for high-dimensional time series prediction. In *Advances in Neural Information Processing Systems*, 847–855.

Yu, B.; Yin, H.; and Zhu, Z. 2017. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875.*

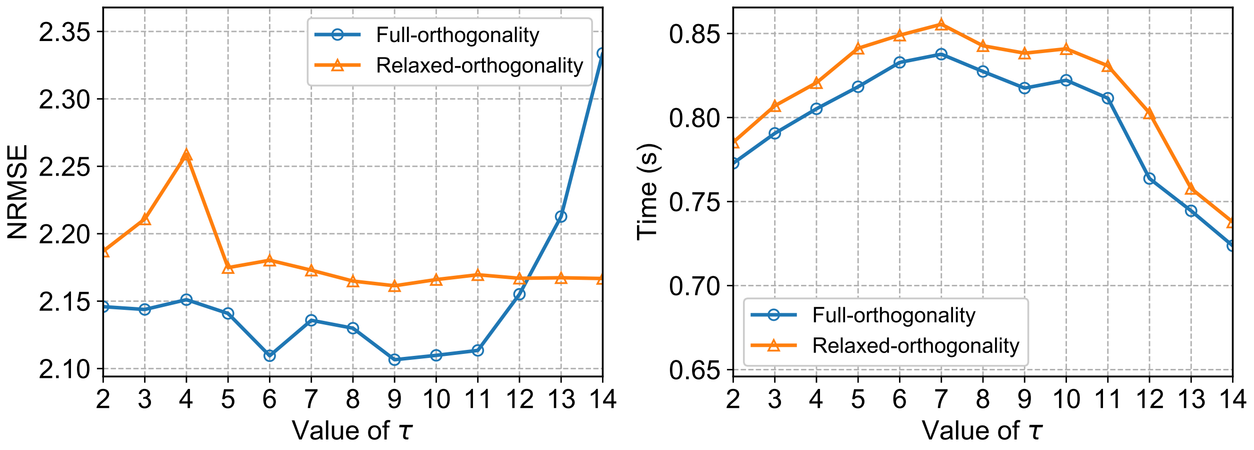


Figure 3: Effect of the parameter τ for the proposed BHT-ARIMA with full-orthogonality versus relaxed-orthogonality the Raw material datasets.

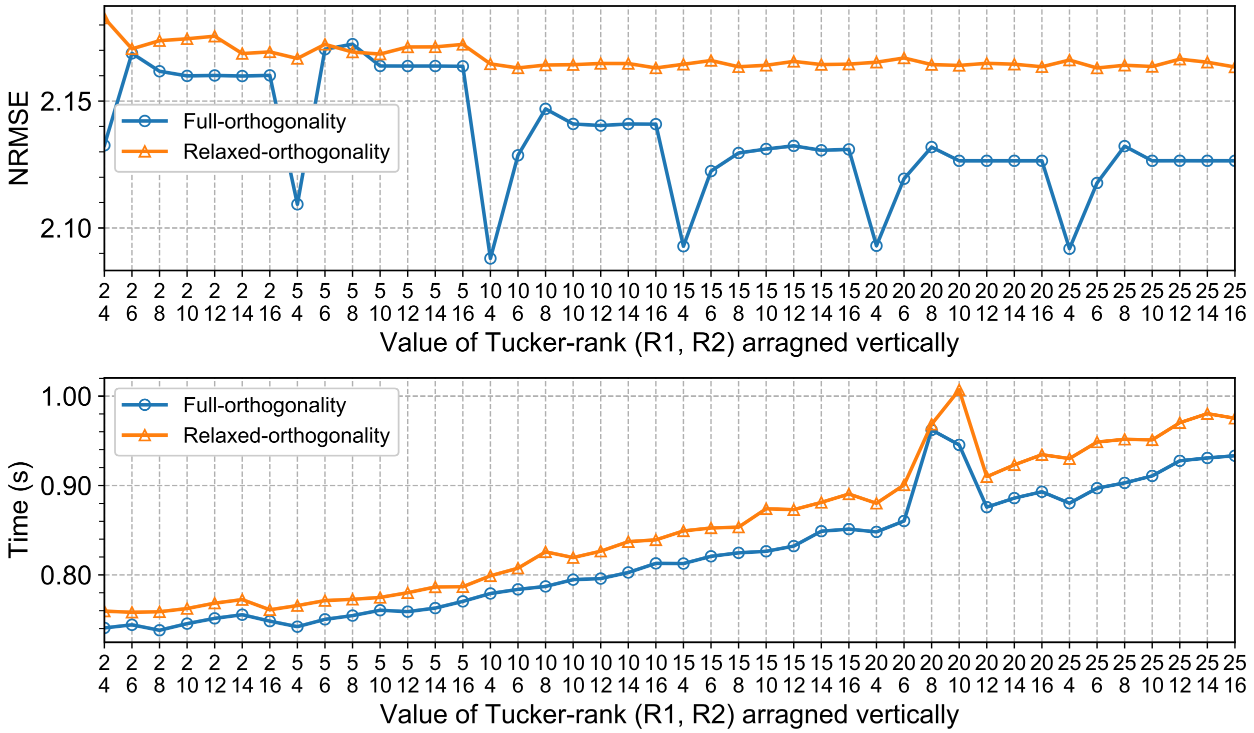


Figure 4: Effect of the parameter Tucker-rank $[R_1, \cdots R_M]$ for the proposed BHT-ARIMA with full-orthogonality versus relaxed-orthogonality on the Raw materials dataset. In this test, we fix other parameters of BHT-ARIMA: $\tau = 4$, $(p, d, q) = (3, 1, 1)$ and $K = 10$.

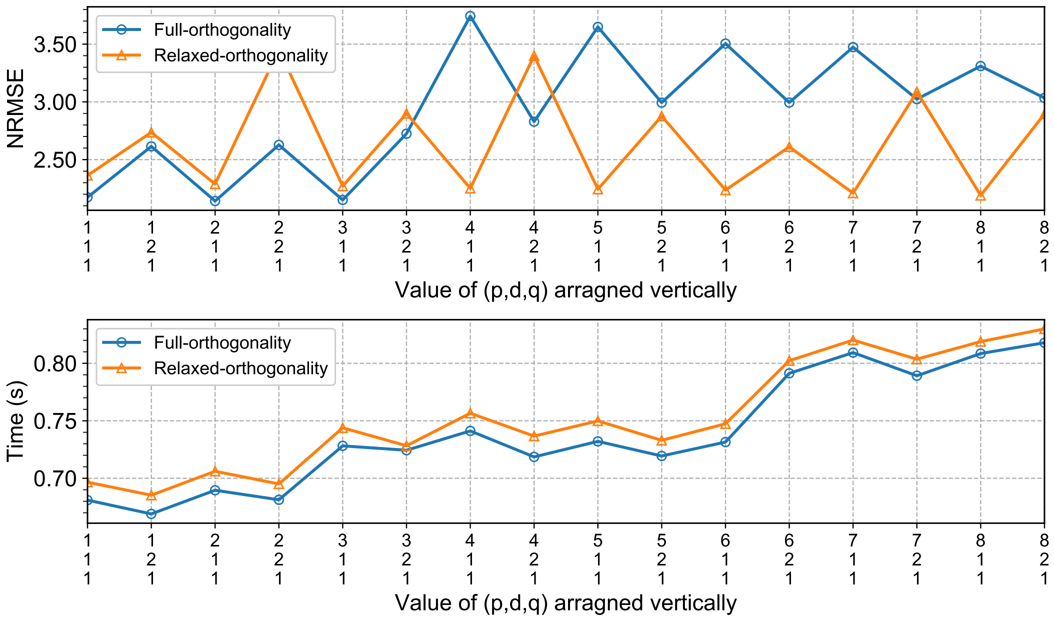


Figure 5: Effect of the parameter (p, d, q) for the proposed BHT-ARIMA with full-orthogonality versus relaxed-orthogonality on the Raw materials dataset. In this test, we fix other parameters of BHT-ARIMA: $\tau = 4$, $[R_1, R_2] = [5, 4]$, and $K = 10$.

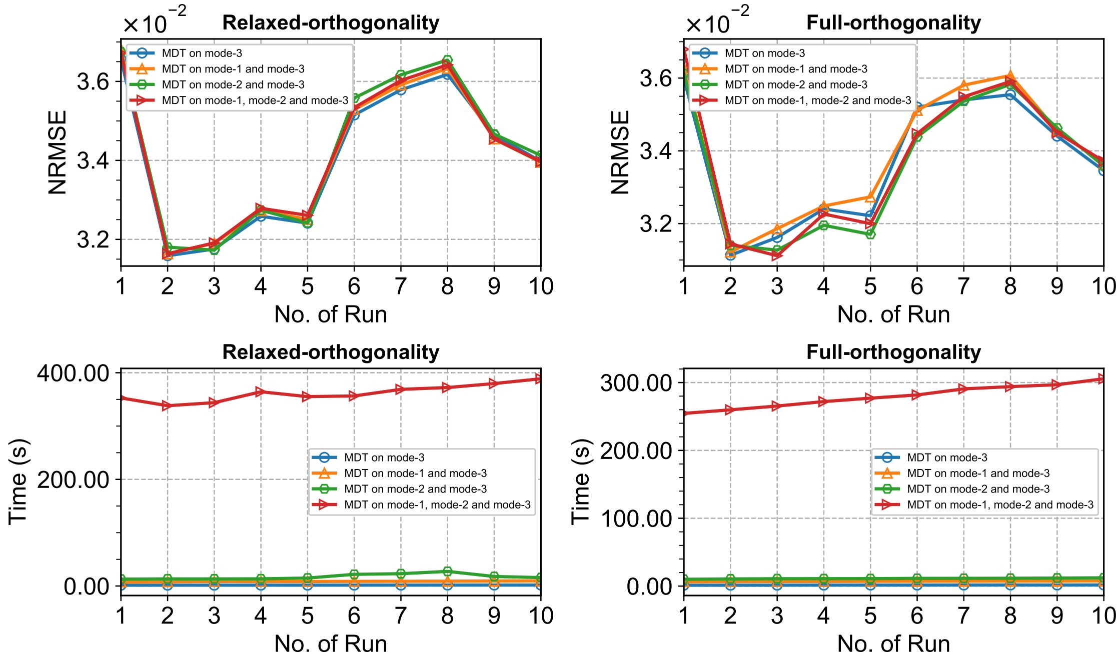
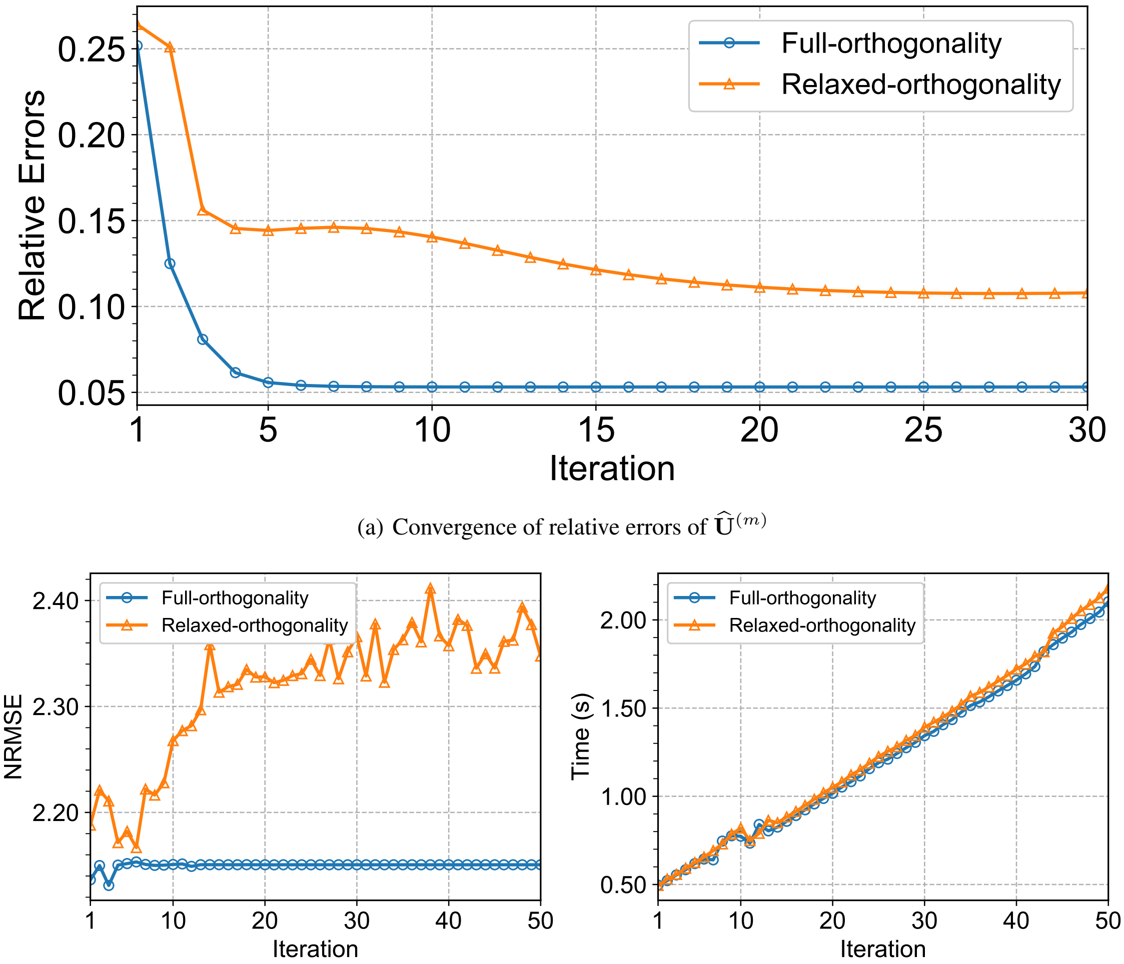


Figure 6: Effect of BHT-ARIMA by applying MDT on all the modes versus on the temporal mode only of the Smoke video (the last mode, mode-3, is the temporal mode). In this test, we fix other parameters of BHT-ARIMA: $\tau = 4$, $(p, d, q) = (3, 2, 1)$ and $K = 10$.



(a) Convergence of relative errors of $\widehat{\mathbf{U}}^{(m)}$

(b) Effect of the maximum iteration K

Figure 7: Convergence curves measured in $\frac{\sum_{m=1}^{M}||\widehat{\mathbf{U}}^{(m)^{k+1}}-\widehat{\mathbf{U}}^{(m)^{k}}||_F^2}{\sum_{m=1}^{M}||\widehat{\mathbf{U}}^{(m)^{k+1}}||_F^2}$ and effect of the maximum iteration K for the proposed BHT-ARIMA with full-orthogonality versus relaxed-orthogonality on the Raw material dataset. In this test, we fix other parameters of BHT-ARIMA: $\tau = 4$, $[R_1, R_2] = [5, 4]$ and $(p, d, q) = (3, 1, 1)$.