Taylor & Francis
Taylor & Francis Group

Journal of Computational and Graphical Statistics

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/ucgs20

Hybrid Kronecker Product Decomposition and Approximation

Chencheng Cai, Rong Chen & Han Xiao

To cite this article: Chencheng Cai, Rong Chen & Han Xiao (2022): Hybrid Kronecker Product Decomposition and Approximation, Journal of Computational and Graphical Statistics, DOI: 10.1080/10618600.2022.2134873

To link to this article: https://doi.org/10.1080/10618600.2022.2134873

Hybrid Kronecker Product Decomposition and Approximation

Chencheng Cai[a], Rong Chen[b] and Han Xiao[b,*]

[a]Temple University,
[b]Rutgers University

Chencheng Cai is a postdoctoral fellow at Department of Statistical Science, Fox School of Business, Temple University, Philadelphia, PA 19122. E-mail: chencheng.cai@temple.edu. Rong Chen is Professor, Department of Statistics, Rutgers University, Piscataway, NJ 08854. E-mail: rongchen@stat.rutgers.edu. Han Xiao is Associate Professor, Department of Statistics, Rutgers University, Piscataway, NJ 08854. E-mail: hxiao@stat.rutgers.edu.

[*]Han Xiao is the corresponding author. hxiao@stat.rutgers.edu

Abstract

Discovering underlying low dimensional structure of a high dimensional matrix is traditionally done through low rank matrix approximations in the form of a sum of rank-one matrices. In this paper, we propose a new approach. We assume a high dimensional matrix can be approximated by a sum of a small number of Kronecker products of matrices with potentially different configurations, named as a *h*ybrid **K**ronecker outer **P**roduct **A**pproximation (*h*KoPA). It provides an extremely flexible way of dimension reduction compared to the low-rank matrix approximation. Challenges arise in estimating a *h*KoPA when the configurations of component Kronecker products are different or unknown. We propose an estimation procedure when the set of configurations are given, and a joint configuration determination and component estimation procedure when the configurations are unknown. Specifically, a least squares backfitting algorithm is used when the configurations are given.

When the configurations are unknown, an iterative greedy algorithm is developed. Both simulation and real image examples show that the proposed algorithms have promising performances. Some identifiability conditions are also provided. The hybrid Kronecker product approximation may have potentially wider applications in low dimensional representation of high dimensional data.

Keywords: Dimension reduction, Identifiability, Information criterion, Kronecker product, Low dimensional structure in high dimensional data, Matrix decomposition

1 Introduction

High dimensional data often has a low dimensional structure that allows significant dimension reduction and compression. In applications such as data compression, image denoising and processing, matrix completion, high dimensional matrices of interest are often assumed to be of low ranks and can be represented as a sum of several rank-one matrices (vector outer products) in the form of the singular value decomposition (SVD),

$$\boldsymbol{X} = \sum_{k=1}^{K} \lambda_k \boldsymbol{u}_k \otimes \boldsymbol{v}_k^T, \qquad (1)$$

where $\boldsymbol{X}$ is a $P \times Q$ matrix, $\boldsymbol{u}_k$ and $\boldsymbol{v}_k$ are P and Q dimensional vectors, and $\otimes$ denotes the outer product. Eckart and Young (1936) reveals the connection between singular value decomposition and low-rank matrix approximation. Recent studies include image low-rank approximation (Freund et al., 1999), principle component analysis (Wold et al., 1987; Zou et al., 2006), factorization in high dimensional time series (Lam and Yao, 2012; Yu et al., 2016), non-negative matrix factorization (Hoyer, 2004; Cai et al., 2009), matrix factorization for community detection (Zhang and Yeung, 2012; Yang and Leskovec, 2013; Le et al., 2016), matrix completion problems (Candès and Recht, 2009; Candes and Plan, 2010; Yuan and Zhang, 2016), low rank tensor approximation (Grasedyck et al., 2013), machine learning applications (Guillamet and Vitrià, 2002; Pauca et al., 2004; Zhang et al., 2008; Sainath et al., 2013), among many others.

As an alternative to vector outer product, the Kronecker product can also be used to represent a high dimensional matrix with a potentially smaller number of elements. For any two matrices $\boldsymbol{A} \in \mathbb{R}^{p \times q}$ and $\boldsymbol{B} \in \mathbb{R}^{p^* \times q^*}$, the Kronecker product $\boldsymbol{A} \otimes \boldsymbol{B}$ is a $(pp^*) \times (qq^*)$ matrix defined by

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} a_{1,1}\boldsymbol{B} & a_{1,2}\boldsymbol{B} & \cdots & a_{1,q}\boldsymbol{B} \\ a_{2,1}\boldsymbol{B} & a_{2,2}\boldsymbol{B} & \cdots & a_{2,q}\boldsymbol{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{p,1}\boldsymbol{B} & a_{p,2}\boldsymbol{B} & \cdots & a_{p,q}\boldsymbol{B} \end{bmatrix},$$

where $a_{i,j}$ is the (*i*, *j*)-th element of $\boldsymbol{A}$. The dimensions (p, q, p^*, q^*) is called the *configuration* of the Kronecker product.

The decomposition of a high dimensional matrix into the sum of several Kronecker products of identical configuration is known as Kronecker product decomposition (Van Loan and Pitsianis, 1993), in the form of

$$\boldsymbol{X} = \sum_{k=1}^{K} \lambda_k \boldsymbol{A}_k \otimes \boldsymbol{B}_k, \quad \boldsymbol{A}_k \in \mathbb{R}^{p \times q}, \boldsymbol{B}_k \in \mathbb{R}^{p^* \times q^*} \quad (2)$$

where $\boldsymbol{X}$ is a $P \times Q$ matrix with $P = pp^*$ and $Q = qq^*$, and $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$ are of dimensions $p \times q$ and $p^* \times q^*$ respectively. In fact, any $P \times Q$ matrix can be decomposed in the form (2) with at most $K = \min\{pq, p^*q^*\}$ terms (Van Loan and Pitsianis, 1993). The formal definition of the Kronecker product decomposition can be found in Appendix D. Note that the SVD in (1) is a special case of (2) with $q = 1$ and $p^* = 1$. The form of Kronecker product appears in many fields including signal processing, image processing and quantum physics (Werner et al., 2008; Duarte and Baraniuk, 2012; Kaye et al., 2007), where the data has an intrinsic Kronecker product structure.

For a given configuration, the approximation using a sum of several Kronecker products can be turned into an approximation using a low rank matrix after a rearrangement operation of the matrix elements (Van Loan and Pitsianis, 1993). Cai et al. (2019) considers to model a high dimensional matrix with a sum of several Kronecker products of the same but unknown configuration, and uses an information criterion to determine the unknown configuration.

However, it is often the case that the **K**ronecker **o**uter **P**roduct **A**pproximation (KoPA) using a single configuration requires a large number of terms to make the approximation accurate. By allowing the use of a sum of Kronecker products of different configurations, an observed high dimensional matrix can be approximated more effectively using a much smaller number of parameters (elements). We note that often the observed matrix can have much more complex structure than what a single Kronecker product can handle. For example, representing an image in a matrix form with Kronecker products of the same configuration is often not satisfactory since the configuration dimensions determine the block structure of the recovered image, similar to the pixel size of the image. A single configuration is often not possible to provide as much details as needed. Due to these limitations, we propose to extend the KoPA approach to allow for multiple configurations. It is more flexible and may provide more accurate representation with a smaller number of parameters.

In this paper, we generalize the KoPA method in Cai et al. (2019) to a multi-term setting, where the observed high dimensional matrix is assumed to be generated from a sum of several Kronecker products of different configurations – we name the model *hybrid* KoPA (*h*KoPA). As a special case, when all the Kronecker products are vector outer products, *h*KoPA is equivalent to the low rank matrix approximation.

We consider two problems in this paper. We first propose a procedure to estimate a *h*KoPA with a set of known configurations. The procedure is based on an iterative backfitting algorithm. Each step involves finding the best one-term Kronecker product approximation to a given matrix, under a known configuration. This operation is obtained through a SVD of a rearranged matrix. Next, we consider the problem of determining the configurations in the *h*KoPA for the observed matrix. As exploiting the space of all possible configuration combinations is computationally expensive, we propose an iterative greedy algorithm similar to the forward stepwise selection. In each iteration, a single Kronecker product term is added to the model by fitting the residual matrix from the previous iteration. The configuration of the added Kronecker product is determined similar to the procedure proposed in Cai et al. (2019). This algorithm efficiently fits a *h*KoPA model with a potentially sub-optimal solution as a compromise between computation and accuracy.

The rest of the paper is organized as follows. The *h*KoPA model is introduced and discussed in Section 2, with a set of identifiability assumptions. In Sections 3 and 4, we provide the details of the iterative backfitting estimation procedure for the model with known configurations and the greed algorithm to fit a *h*KoPA with unknown configurations. Section 5 demonstrates the performance of the proposed procedures with a simulation study and a real image example. Section 6 concludes.

Notations: For a matrix $\boldsymbol{M}$, $\| \boldsymbol{M} \|_F := \sqrt{\mathrm{tr}(\boldsymbol{M}\boldsymbol{M}^T)}$ stands for its Frobenius norm and $\| \boldsymbol{M} \|_S$ its spectral norm, which is the largest singular value of $\boldsymbol{M}$. For a positive integer n, $[n]$ denotes the set of positive integers up to n such that $[n] = \{1, \dots, n\}$. We denote by $\boldsymbol{e}_{i,j}^{m,n}$ the $m \times n$ matrix with 1 at the (i, j)-th entry and 0 elsewhere.

2 Hybrid Kronecker Product Model

2.1 The Model

In this paper we consider the K-term hybrid KoPA (*h*KoPA) model, in the form

$$\boldsymbol{Y} = \boldsymbol{X} + \boldsymbol{E}, \quad (3)$$

where the observed matrix $\boldsymbol{Y}$ is the sum of a signal matrix $\boldsymbol{X}$ and a noise matrix $\boldsymbol{E}$ with i.i.d. standard Gaussian entries. We assume that the signal matrix $\boldsymbol{X}$ has the same form of (2)

$$\boldsymbol{X} = \sum_{k=1}^{K} \lambda_k \boldsymbol{A}_k \otimes \boldsymbol{B}_k, \quad \boldsymbol{A}_k \in \mathbb{R}^{p_k \times q_k}, \boldsymbol{B}_k \in \mathbb{R}^{p_k^* \times q_k^*}, \quad (4)$$

but here the matrices $(\boldsymbol{A}_k, \boldsymbol{B}_k)$ are allowed to have **different** configurations. Specifically, we assume that $\boldsymbol{Y}$ and $\boldsymbol{X}$ are of the dimension $P \times Q$, and the matrices $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$ in the k-th component are $p_k \times q_k$ and $p_k^* \times q_k^*$, respectively. We call the dimensions of $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$, (p_k, q_k, p_k^*, q_k^*), the configuration of the Kronecker product $\boldsymbol{A}_k \otimes \boldsymbol{B}_k$. Since P and Q are fixed and given by the observed matrix $\boldsymbol{Y}$, in the sequel we will simply use the pair (p_k, q_k) to denote the configuration of $\boldsymbol{A}_k \otimes \boldsymbol{B}_k$. We also assume that $1 < p_k q_k < PQ$ for all $1 \le k \le K$ so that none of $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$ are scalars. Comparing (2), we refer to (4) as a *hybrid Kronecker representation* of $\boldsymbol{X}$.

It is helpful to understand (4) as a "multi-resolution" representation of $\boldsymbol{X}$. More specifically, if $\boldsymbol{X}$ is an image, then the term $\boldsymbol{A}_k \otimes \boldsymbol{B}_k$ corresponds to a partition of the image into non-overlap $p_k^* \times q_k^*$ blocks. By allowing different configurations, i.e. different sizes of $\boldsymbol{B}_k$'s, (4) is able to extract the local patterns at different resolution (or pixel size), offering the flexibility to capture different texture of the image. This " multi-resolution" interpretation also suggests that *h*KoPA are useful for many other applications, e.g. spatial-temporal data, multi-dimensional signals analysis etc.

Define the configuration set of the *h*KoPA model (3) as the collection of individual configurations $\mathcal{C} := \{(p_k, q_k), 1 \le k \le K\}$. When the configuration set $\mathcal{C}$ is known, we need to estimate the component matrices $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$, for $k = 1, \dots, K$ in model (3). When $\mathcal{C}$ is unknown, the estimation of model (3) requires the determination of the configuration set $\mathcal{C}$ in advance.

2.2 Identifiability Conditions

The primary goal is to estimate λ_k, $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$ in (3). However, there are some obvious unidentifiability regarding them. We discuss the identifiability conditions in this section. Due to the complexity of the hKoPA models, we use a specific definition of identifiability as follows. First of all, we assume that the configuration set $\mathcal{C}$ is an ordered set, that is, the order of the configurations $\{(p_1, q_1), \dots, (p_K, q_K)\}$ is fixed. With this assumption, the following definition automatically excludes the unidentifiability due to different orderings of the terms $\{\lambda_k \boldsymbol{A}_k \otimes \boldsymbol{B}_k, 1 \le k \le K\}$ when their configurations are all distinct.

Definition 1 (Identifiability). *We say that the representation (4) is identifiable up to sign changes with respect to the ordered configuration set $\mathcal{C}$ if there are no other matrices $\{\boldsymbol{A}_k, \boldsymbol{B}_k\}$ of the same configurations $\{p_k, q_k\}$, and coefficients $\{\tilde{\lambda}_k\}$ such that*

$$\sum_{k=1}^{K} \lambda_k \boldsymbol{A}_k \otimes \boldsymbol{B}_k = \sum_{k=1}^{K} \tilde{\lambda}_k \boldsymbol{A}_k \otimes \boldsymbol{B}_k,$$

unless $\boldsymbol{A}_k = \pm \boldsymbol{A}_k$, $\boldsymbol{B}_k = \pm \boldsymbol{B}_k$ and $\tilde{\lambda}_k \boldsymbol{A}_k \otimes \boldsymbol{B}_k = \lambda_k \boldsymbol{A}_k \otimes \boldsymbol{B}_k$.

In the sequel we shall often refer to the identifiability defined above as "identifiable up to sign changes", but omit "with respect to the ordered configuration set $\mathcal{C}$" for

simplicity. Nevertheless, it should be understood that once the representation (4) is given, the associated ordered configuration set $\mathcal{C}$ is also determined, and the discussion of the identifiability will be based on this given $\mathcal{C}$.

Two more definitions are needed for the discussion of identifiability of *h*KoPA model.

Definition 2 (Conformality). *Let* $\boldsymbol{A}$ *be a matrix of dimension* (p_A, q_A) *and* $\boldsymbol{B}$ *of* (p_B, q_B). *If* p_A *is a factor of* p_B *and* q_A *is a factor of* q_B, $\boldsymbol{A}$ *is said to be conformally smaller than* $\boldsymbol{B}$, *denoted by* $\boldsymbol{A} \Subset \boldsymbol{B}$ *or* $\boldsymbol{B} \Supset \boldsymbol{A}$. *This includes the special case that* $p_A = p_B$ *and* $q_A = q_B$, *which we also say that* $\boldsymbol{A}$ *and* $\boldsymbol{B}$ *are conformally equal, denoted by* $\boldsymbol{A} \cong \boldsymbol{B}$.

Remark 1. Conformality is of interests because if $\boldsymbol{A}$ of dimension (p_A, q_A) is *strictly* conformally smaller than $\boldsymbol{B}$ of (p_B, q_B), then for any matrix $\boldsymbol{C}$ of dimension $(p_B / p_A, q_B / q_A)$ ($\boldsymbol{C}$ is not a scalar), $\boldsymbol{A} \otimes \boldsymbol{C}$ and $\boldsymbol{C} \otimes \boldsymbol{A}$ have the same dimension as $\boldsymbol{B}$, or $\boldsymbol{A} \otimes \boldsymbol{C} \cong \boldsymbol{B}$ and $\boldsymbol{C} \otimes \boldsymbol{A} \cong \boldsymbol{B}$.

Definition 3 (Orthogonality). Let $\boldsymbol{A} \in \mathbb{R}^{p_A \times q_A}$ and $\boldsymbol{B} \in \mathbb{R}^{p_B \times q_B}$ be two matrices such that $\boldsymbol{A} \Subset \boldsymbol{B}$. We say $\boldsymbol{A}$ and $\boldsymbol{B}$ are block-wise orthogonal (b-orthogonal) if

$$\underset{\boldsymbol{C} \in \mathbb{R}^{(p_B/p_A) \times (q_B/q_A)}}{\arg\min} \| \boldsymbol{B} - \boldsymbol{C} \otimes \boldsymbol{A} \|_F = \boldsymbol{0},$$

or equivalently, $\mathrm{tr}[\boldsymbol{B}^T (\boldsymbol{e}_{i,j}^{p_B/p_A, q_B/q_A} \otimes \boldsymbol{A})] = 0$ for all $i = 1, \ldots, (p_B / p_A), j = 1, \ldots, (q_B / q_A)$. Similarly, we say $\boldsymbol{A}$ and $\boldsymbol{B}$ are grid-wise orthogonal (g-orthogonal) if

$$\underset{\boldsymbol{C} \in \mathbb{R}^{(p_B/p_A) \times (q_B/q_A)}}{\arg\min} \| \boldsymbol{B} - \boldsymbol{A} \otimes \boldsymbol{C} \|_F = \boldsymbol{0},$$

or equivalently, $\mathrm{tr}[\boldsymbol{B}^T (\boldsymbol{A} \otimes \boldsymbol{e}_{i,j}^{p_B/p_A, q_B/q_A})] = 0$, for all $i = 1, \ldots, (p_B / p_A), j = 1, \ldots, (q_B / q_A)$. In particular, if $\boldsymbol{A} \cong \boldsymbol{B}$, then b-orthogonality and g-orthogonality are equivalent, and both require $\mathrm{tr}[\boldsymbol{B}^T \boldsymbol{A}] = 0$. In this case we say $\boldsymbol{A}$ and $\boldsymbol{B}$ are orthogonal.

Remark 2. If $\boldsymbol{A} \Subset \boldsymbol{B}$ and write $\boldsymbol{B} = (\boldsymbol{B}_{ij})$ as a block matrix such that each block $\boldsymbol{B}_{ij}$ has the same dimension as $\boldsymbol{A}$. Then the b-orthogonality of $\boldsymbol{A}$ and $\boldsymbol{B}$ implies $\mathrm{tr}(\boldsymbol{A}^T \boldsymbol{B}_{ij}) = 0$ for all the blocks $\boldsymbol{B}_{ij}$ of $\boldsymbol{B}$. Similarly, if $\boldsymbol{A} \Subset \boldsymbol{B}$ and $\boldsymbol{B}_{ij}^{(g)}$ is the (*i*, *j*)-th sub-grid of $\boldsymbol{B}$ (consisting of all grid elements with stride size $(p_B / p_A, q_B / q_A)$, i.e.

$b_{i+s_1(p_B/p_A),j+s_2(q_B/q_A)}$ for $s_1 = 0,\dots,p_A-1, s_2 = 0,\dots,q_A-1$), then that $\boldsymbol{A}$ and $\boldsymbol{B}$ are g-orthogonal implies $\mathrm{tr}(\boldsymbol{A}^T\boldsymbol{B}_{ij}^{(g)}) = 0$ for all the sub-grids $\boldsymbol{B}_{ij}^{(g)}$ of $\boldsymbol{B}$.

We first list the following two conditions on the signal matrix $\boldsymbol{X}$ in (4).

Assumption 1. *For all* $k = 1,\dots,K$ *,* $\|\boldsymbol{A}_k\|_F = \|\boldsymbol{B}_k\|_F = 1$, *and* $\lambda_k > 0$.

Assumption 2. *Assume* $(p_k, q_k) \neq (1, Q)$ *for all* $k = 1,\dots,K$.

Remark 3. Assumption 1 is standard and can be satisfied by re-scaling $\boldsymbol{A}$ and $\boldsymbol{B}$. For Assumption 2, note that when $(p_k, q_k) = (1, Q)$, $\boldsymbol{A}_k$ is a row vector and the corresponding $\boldsymbol{B}_k$ is a column vector of size $(P,1)$. In this case, $\boldsymbol{A}_k \otimes \boldsymbol{B}_k = \boldsymbol{B}_k \otimes \boldsymbol{A}_k$. Assumption 2 can be easily satisfied by switching so that $(p_k, q_k) = (P,1)$ when needed.

Assumption 3. For any $0 \le k, l \le K$ such that $\boldsymbol{A}_k \Subset \boldsymbol{A}_l$, $\boldsymbol{A}_k$ and $\boldsymbol{A}_l$ are g-orthogonal. For all $k \neq l$ such that $\boldsymbol{A}_k \cong \boldsymbol{A}_l$, $\boldsymbol{A}_k$ and $\boldsymbol{A}_l$ are orthogonal, and $\boldsymbol{B}_k$ and $\boldsymbol{B}_l$ are orthogonal.

Assumption 3'. *For any* $0 \le k, l \le K$ *such that* $\boldsymbol{B}_k \Subset \boldsymbol{B}_l$, $\boldsymbol{B}_k$ *and* $\boldsymbol{B}_l$ *are b-orthogonal. For all* $k \neq l$ *such that* $\boldsymbol{A}_k \cong \boldsymbol{A}_l$, $\boldsymbol{A}_k$ *and* $\boldsymbol{A}_l$ *are orthogonal, and* $\boldsymbol{B}_k$ *and* $\boldsymbol{B}_l$ *are orthogonal.*

Remark 4. This condition is to address the following identifiability situations. Suppose $\boldsymbol{A}_1 \Subset \boldsymbol{A}_2$, then for any $p_2/p_1 \times q_2/q_1$ matrix $\boldsymbol{C}$, it holds that

$$\lambda_1\boldsymbol{A}_1 \otimes (\boldsymbol{B}_1 + \lambda_2\boldsymbol{C} \otimes \boldsymbol{B}_2) + \lambda_2(\boldsymbol{A}_2 - \lambda_1\boldsymbol{A}_1 \otimes \boldsymbol{C}) \otimes \boldsymbol{B}_2 = \lambda_1\boldsymbol{A}_1 \otimes \boldsymbol{B}_1 + \lambda_2\boldsymbol{A}_2 \otimes \boldsymbol{B}_2. \quad (5)$$

Assumption 3 excludes this type of unidentifiability by requiring b-orthogonality between $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$. Such a requirement can be achieved through an orthogonalization operation. For example, let the (*i, j*)-th element of $\boldsymbol{C}$ be $[\boldsymbol{C}]_{i,j} = \mathrm{tr}\left[\boldsymbol{A}_2(\boldsymbol{A}_1 \otimes \boldsymbol{e}_{i,j}^{p_2/p_1, q_2/q_1})^T\right]$. Let

$$\begin{aligned}\lambda_1\boldsymbol{A}_1 \otimes \boldsymbol{B}_1 + \lambda_2\boldsymbol{A}_2 \otimes \boldsymbol{B}_2 &= \boldsymbol{A}_1 \otimes (\lambda_1\boldsymbol{B}_1 + \lambda_2\boldsymbol{C} \otimes \boldsymbol{B}_2) + \lambda_2(\boldsymbol{A}_2 - \boldsymbol{A}_1 \otimes \boldsymbol{C}) \otimes \boldsymbol{B}_2 \\ &=: \tilde{\lambda}_1\boldsymbol{A}_1 \otimes \boldsymbol{B}_k + \tilde{\lambda}_2\boldsymbol{A}_2 \otimes \boldsymbol{B}_2,\end{aligned}$$

with all the quantities in the last expression being rescaled to compile with Assumption 1. It is easy to show that $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ are b-orthogonal in this new representation. Algorithm in Appendix C performs such an orthogonalization for multiple terms iteratively.

Remark 5. Assumptions 3 and 3' are parallel conditions, one on $\boldsymbol{A}_i$ and another on $\boldsymbol{B}_i$. We refer to them as "Ortho-A" and "Ortho-B" conditions, respectively. Only one of them is needed.

Assumption 4. *Suppose*

(i) *For all* $k \neq l$ *such that* $\boldsymbol{B}_k$ *is a row vector and* $\boldsymbol{B}_l$ *is a column vector,* $\boldsymbol{A}_l$ *and* $\boldsymbol{B}_k$ *are b-orthogonal.*

(ii) *For all* $k \neq l$ *such that* $\boldsymbol{A}_k$ *is a row vector and* $\boldsymbol{A}_l$ *is a column vector,* $\boldsymbol{A}_l$ *and* $\boldsymbol{B}_k$ *are g-orthogonal.*

Remark 6. This condition is needed. Consider a two term representation of the form

$$\boldsymbol{A}_1 \otimes \boldsymbol{\beta}_1^T + \boldsymbol{A}_2 \otimes \boldsymbol{\beta}_2,$$

where $\boldsymbol{\beta}_i$ are column vectors. Now pick any matrix $\boldsymbol{C}$ such that $\boldsymbol{C} \otimes \boldsymbol{\beta}_2$ has the same dimension as $\boldsymbol{A}_1$, then it holds that $\boldsymbol{C} \otimes \boldsymbol{\beta}_1^T$ has the same dimension as $\boldsymbol{A}_2$, and

$$\boldsymbol{A}_1 \otimes \boldsymbol{\beta}_1^T + \boldsymbol{A}_2 \otimes \boldsymbol{\beta}_2 = (\boldsymbol{A}_1 + \boldsymbol{C} \otimes \boldsymbol{\beta}_2) \otimes \boldsymbol{\beta}_1^T + (\boldsymbol{A}_2 - \boldsymbol{C} \otimes \boldsymbol{\beta}_1^T) \otimes \boldsymbol{\beta}_2,$$

due to the fact that $\boldsymbol{\beta}_2 \otimes \boldsymbol{\beta}_1^T = \boldsymbol{\beta}_1^T \otimes \boldsymbol{\beta}_2$. Assumption 4 excludes this type of unidentifiability by requiring b-orthogonality between $\boldsymbol{A}_2$ and $\boldsymbol{\beta}_1^T$. Note that $\boldsymbol{\beta}_1^T \in \boldsymbol{A}_2$ as $\boldsymbol{\beta}_1^T$ is of $1 \times q_1^*$ and $\boldsymbol{A}_2$ is of $p_2 \times Q$, with q_1^* being a factor of Q. Such a requirement can be achieved through an orthogonalization operation in Algorithm.

Remark 7. As seen in the example given in Remark 6, Assumption 4 could also have been made on the *b*-orthogonality of $\boldsymbol{A}_1$ and $\boldsymbol{\beta}_2$. We choose the current formulation.

The following theorem states that, for any $\boldsymbol{X}$ that can be written in (4), then there is another representation such that the above conditions are satisfied. And the representation can be obtained through a sequence of orthogonalization operations.

Theorem 1. *If* $\boldsymbol{X} = \sum_{k=1}^{K} \lambda_k \boldsymbol{A}_k \otimes \boldsymbol{B}_k$ *of configuration set* $\mathcal{C}$ *satisfies Assumptions 1 and 2, then after the generalized Gram-Schmidt procedure given in Algorithm in Appendix C, the resulting representation*

$$\boldsymbol{X} = \sum_{k=1}^{\tilde{K}} \tilde{\lambda}_k \boldsymbol{A}_k \otimes \boldsymbol{B}_k. \quad (6)$$

has a configuration set $\tilde{\mathcal{C}} \subset \mathcal{C}$ *, and satisfies Assumptions 1, 2, 4 and 3 (the Ortho-A representation).*

The proof of the theorem is in Appendix D.

Remark 8. We can also obtain a representation satisfying Assumptions 1, 2, 4 and 3' (the Ortho-B representation) by slightly modifying Algorithm.

Remark 9. Algorithm outputs a representation which has a configuration set same as the original $\mathcal{C}$, but may have some zero $\tilde{\lambda}_k$. Hence the configuration set $\tilde{\mathcal{C}}$ in (6) can be a subset of $\mathcal{C}$.

We have not required any ordering of the terms $\lambda_k \boldsymbol{A}_k \otimes \boldsymbol{B}_k$, because it is assumed that the ordered configuration set $\mathcal{C}$ is given, so the terms are ordered according to $\mathcal{C}$. However, when some configurations in $\mathcal{C}$ are the same, we need to fix their orders according to the next identifiability condition. This condition is also similar to the distinct singular values condition for the identifiability of the singular vectors in the SVD of a matrix.

Assumption 5. *If* $1 \leqq k < l \leqq K$ *and* $(p_k, q_k) = (p_l, q_l)$*, then* $\lambda_k > \lambda_l$.

Remark 10. The reason that the condition is needed can be seen from the following example. If $\boldsymbol{A}_k \cong \boldsymbol{A}_l$ (and $\boldsymbol{B}_k \cong \boldsymbol{B}_l$ as well) satisfy Assumptions 1 and 3, and $\lambda_k = \lambda_l = 1$, then

$$\boldsymbol{A}_k \otimes \boldsymbol{B}_k + \boldsymbol{A}_l \otimes \boldsymbol{B}_l = \frac{\boldsymbol{A}_k + \boldsymbol{A}_l}{\sqrt{2}} \otimes \frac{\boldsymbol{B}_k + \boldsymbol{B}_l}{\sqrt{2}} + \frac{\boldsymbol{A}_k - \boldsymbol{A}_l}{\sqrt{2}} \otimes \frac{\boldsymbol{B}_k - \boldsymbol{B}_l}{\sqrt{2}} =: \boldsymbol{A}_k \otimes \boldsymbol{B}_k + \boldsymbol{A}_l \otimes \boldsymbol{B}_l,$$

but $\boldsymbol{A}_k, \boldsymbol{B}_k, \boldsymbol{A}_l, \boldsymbol{B}_l$ also satisfy Assumptions 1 and 3. When $\lambda_k \neq \lambda_l$, such an ambiguity does not occur.

So far we have given some necessary conditions for the identifiability. It is very challenging to verify whether they are sufficient due to the complexity of the hKoPA model, especially due to the fact that different configurations are present in (4). We shall leave the general sufficient conditions to the future work. In the next two sections, we give a nearly complete answer for a special case of (4) with two terms of configurations (p_1, q_1) and (p_2, q_2). We consider two scenarios depending on whether these two configurations are conformal or not.

2.3 Identifiability of the Conformal Two-Term Model

We first consider the conformal two-term representation $\boldsymbol{X} = \lambda_1 \boldsymbol{A}_1 \otimes \boldsymbol{B}_1 + \lambda_2 \boldsymbol{A}_2 \otimes \boldsymbol{B}_2$, where $\boldsymbol{A}_1 \subseteq \boldsymbol{A}_2$. We need one more technical condition.

Assumption 6. *If $\boldsymbol{A}_1 \subseteq \boldsymbol{A}_2$, assume that $\boldsymbol{A}_2$ cannot be decomposed as $\boldsymbol{C} \otimes \boldsymbol{D}$, where $\boldsymbol{C}$ has the same dimension as $\boldsymbol{A}_1$.*

Theorem 2. *If $\boldsymbol{A}_1 \subseteq \boldsymbol{A}_2$, and Assumptions 1, 3, 5 and 6 hold, then the representation*

$$\boldsymbol{X} = \lambda_1 \boldsymbol{A}_1 \otimes \boldsymbol{B}_1 + \lambda_2 \boldsymbol{A}_2 \otimes \boldsymbol{B}_2$$

is identifiable up to sign changes.

The proof of the theorem is given in Appendix D. The theorem says that for a conformal two-term model, the Ortho-A representation is unique. Similarly, under Assumptions 1, 3', and 5, 6, we also have an unique Ortho-B representation.

In the following we discuss the relationship between the Ortho-A and Ortho-B representations for the two-term model. Suppose that for the configurations (p_1, q_1) and (p_2, q_2), p_1 is a factor of p_2 and q_1 is a factor of q_2, and the matrix $\boldsymbol{X}$ is given by

$$\boldsymbol{X} = \lambda_1 \boldsymbol{A}_1 \otimes \boldsymbol{B}_1 + \lambda_2 \boldsymbol{A}_2 \otimes \boldsymbol{B}_2 + \lambda_{12} \boldsymbol{A}_1 \otimes \boldsymbol{C} \otimes \boldsymbol{B}_2, \qquad (7)$$

where $\boldsymbol{A}_1 \in \mathbb{R}^{p_1 \times q_1}$, $\boldsymbol{B}_1 \in \mathbb{R}^{p_1^* \times q_1^*}$, $\boldsymbol{A}_2 \in \mathbb{R}^{p_2 \times q_2}$, $\boldsymbol{B}_2 \in \mathbb{R}^{p_2^* \times q_2^*}$ and $\boldsymbol{C} \in \mathbb{R}^{p_2/p_1 \times q_2/q_1}$. Let's assume that $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ are orthogonal, and so are $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$. This representation

can always be obtained for any two-term model through an Ortho-A operation then an Ortho-B operation. The third term $\boldsymbol{A}_1 \otimes \boldsymbol{C} \otimes \boldsymbol{B}_2$ is conformally equal to both the first configuration (p_1, q_1) (when written as $\boldsymbol{A}_1 \otimes (\boldsymbol{C} \otimes \boldsymbol{B}_2)$) and the second configuration (p_2, q_2) (when written as $(\boldsymbol{A}_1 \otimes \boldsymbol{C}) \otimes \boldsymbol{B}_2$). By an abuse of terminology, we refer to it as the *interaction* of the two configurations. One can distribute the interaction term over the first and second Kronecker products, resulting in different representations of $\boldsymbol{X}$ under configurations (p_1, q_1) and (p_2, q_2):

$$\boldsymbol{X} = \tilde{\lambda}_1 \boldsymbol{A}_1 \otimes \boldsymbol{B}_1 + \tilde{\lambda}_2 \boldsymbol{A}_2 \otimes \boldsymbol{B}_2. \quad (8)$$

Two extreme cases are listed in (9) and (10).

$$\boldsymbol{X} = \lambda_1^c \boldsymbol{A}_1 \otimes \boldsymbol{B}_1^c + \lambda_2 \boldsymbol{A}_2 \otimes \boldsymbol{B}_2, \quad (9)$$

$$= \lambda_1 \boldsymbol{A}_1 \otimes \boldsymbol{B}_1 + \lambda_2^c \boldsymbol{A}_2^c \otimes \boldsymbol{B}_2, \quad (10)$$

where

$$\lambda_1^c = \sqrt{\lambda_1^2 + \lambda_{12}^2}, \quad \boldsymbol{B}_1^c = \frac{\lambda_1}{\lambda_1^c}\boldsymbol{B}_1 + \frac{\lambda_{12}}{\lambda_1^c}\boldsymbol{C} \otimes \boldsymbol{B}_2,$$

$$\lambda_2^c = \sqrt{\lambda_2^2 + \lambda_{12}^2}, \quad \boldsymbol{A}_2^c = \frac{\lambda_2}{\lambda_2^c}\boldsymbol{A}_2 + \frac{\lambda_{12}}{\lambda_2^c}\boldsymbol{A}_1 \otimes \boldsymbol{C}.$$

In (9), the interaction term is merged into the first Kronecker product, so that $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ are orthogonal but $\boldsymbol{B}_1^c$ and $\boldsymbol{B}_2$ are not. In other words, (9) satisfies Assumption 3 and is the Ortho-A representation. Similarly, in (10), the interaction term is merged into the second Kronecker product, where $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$ remains orthogonal but $\boldsymbol{A}_1$ and $\boldsymbol{A}_1^c$ are not. Hence it satisfies Assumption 3', and is the Ortho-B representation. Any other possible representation of $\boldsymbol{X}$ in the form (8) is an affine combination of (9) and (10).

2.4 Identifiability of the Non-conformal Two-Term Model

In this section we consider the identifiability of the non-conformal two-term model. Assume the configurations of $\boldsymbol{X} = \lambda_1 \boldsymbol{A}_1 \otimes \boldsymbol{B}_1 + \lambda_2 \boldsymbol{A}_2 \otimes \boldsymbol{B}_2$ are not conformal, and satisfy

Assumptions 1 and 2. We divide the non-conformal two-term models into two types, and treat them accordingly.

Type I non-conformal two-term model. One of $\boldsymbol{A}_1, \boldsymbol{A}_2$ is a column and the other is a row; or one of $\boldsymbol{B}_1, \boldsymbol{B}_2$ is a column and the other is a row.

Type II non-conformal two-term model. All the non-conformal two-term models that are not of type I are classified as type II.

We first point out that the type I model can be converted into a conformal model so that Theorem 2 applies for its identifiability. Without loss of generality, assume that $\boldsymbol{B}_1$ is a $p_1^* \times 1$ column vector, $\boldsymbol{B}_2$ is a $1 \times q_2^*$ row vector. To better illustrate the idea, we rewrite this two-term model as $\boldsymbol{X} = \boldsymbol{A}_1 \otimes \boldsymbol{\beta}_1 + \boldsymbol{A}_2 \otimes \boldsymbol{\beta}_2^T$. According to Assumption 2, $\boldsymbol{A}_1$ must not be a row/column vector. Write $\boldsymbol{X} = (\boldsymbol{X}_{ij})$ as a $p_1 \times q_2$ block matrix, where all the blocks $\boldsymbol{X}_{ij}$ have the same size $p_1^* \times q_2^*$. We perform the block stacking operation on $\boldsymbol{X}$ to turn it into a $(Pq_2) \times q_2^*$ matrix as

$$\boldsymbol{X} \to \mathcal{Q}_{p_1,q_2}(\boldsymbol{X}) := [\boldsymbol{X}_{11}^T, \boldsymbol{X}_{12}^T, \cdots \boldsymbol{X}_{1,q_2}^T, \boldsymbol{X}_{21}^T, \cdots \boldsymbol{X}_{p_1,q_2}^T]^T.$$

Now do a similar operation on $\boldsymbol{A}_i$: first write $\boldsymbol{A}_i$ as a $p_1 \times q_2$ block matrix with equal size blocks, then rearrange its blocks by the $\mathcal{Q}_{p_1,q_2}$ operation and denote the resulting matrix by $\mathcal{Q}_{p_1,q_2}(\boldsymbol{A}_i)$, i = 1, 2. Note that $\mathcal{Q}_{p_1,q_2}(\boldsymbol{A}_2)$ is a column vector. It follows that

$$\mathcal{Q}_{p_1,q_2}(\boldsymbol{X}) = \mathcal{Q}_{p_1,q_2}(\boldsymbol{A}_1) \otimes \boldsymbol{\beta}_1 + \mathcal{Q}_{p_1,q_2}(\boldsymbol{A}_2) \otimes \boldsymbol{\beta}_2^T = \mathcal{Q}_{p_1,q_2}(\boldsymbol{A}_1) \otimes \boldsymbol{\beta}_1 + \boldsymbol{\beta}_2^T \otimes \mathcal{Q}_{p_1,q_2}(\boldsymbol{A}_2). \quad (11)$$

The right hand side of the preceding equation gives a conformal two term representation, and the orthogonality of $\boldsymbol{A}_1$ and $\boldsymbol{\beta}_2^T$ is equivalent to the orthogonality of $\mathcal{Q}_{p_1,q_2}(\boldsymbol{A}_1)$ and $\boldsymbol{\beta}_2^T$. Therefore, the identifiability of the original type I model becomes the identifiability of the conformal two-term model in (11). We therefore have the following corollary regarding the type I model.

Corollary 1. *Consider the type I non-conformal two-term model. Suppose Assumptions 1, 2 and 4 hold. The representation* $\boldsymbol{X} = \lambda_1 \boldsymbol{A}_1 \otimes \boldsymbol{B}_1 + \lambda_2 \boldsymbol{A}_2 \otimes \boldsymbol{B}_2$ *is identifiable up to sign changes for each of the following scenarios.*

(i) *If B_1 is a column vector, B_2 is a row vector, assume A_1 cannot be decomposed as $C \otimes D$, where D is a row vector of the same length as B_2.*

(ii) *If A_1 is a column vector, A_2 is a row vector, assume B_2 cannot be decomposed as $C \otimes D$, where C is a column vector of the same length as A_1.*

For the type II model, all of Assumptions 3, 4 and 5 are not relevant. On the other hand, it is very difficult to verify whether Assumptions 1 and 2 are sufficient for the identifiability. We provide an affirmative answer when the dimensions of X are powers of 2, and when A_k and B_k are in "generic positions". It is also possible to give a set of sufficient conditions which guarantees the identifiability of any type II model. However, unlike the conformal case, these sufficient conditions are very tedious, so we choose not to spell the details out, and only discuss the identifiability for "generic" A_k and B_k, under simplified conditions.

Theorem 3. *Suppose $X = \lambda_1 A_1 \otimes B_1 + \lambda_2 A_2 \otimes B_2$ is a type II model, where A_k are $2^{m_k} \times 2^{n_k}$ matrices (k = 1, 2), and B_k are $2^{m_k^*} \times 2^{n_k^*}$ respectively. Suppose Assumptions 1 and 2 hold, and $m_1 + n_1 + m_1^* + m_2^* > 4$. Then if the elements of A_k and B_k are in generic positions, the representation $X = \lambda_1 A_1 \otimes B_1 + \lambda_2 A_2 \otimes B_2$ is identifiable up to sign changes.*

Remark 11. By "generic positions", we mean the following. If the elements of A_k and B_k are generated from some joint distribution which is absolutely continuous with respect to the Lebesgue measure, then the identifiability holds with probability one. In the proof (given in Appendix D), without loss of generality, we will assume that the elements of A_k and B_k are IID $N(0, 1)$.

Remark 12. Theorem 3 covers both the conformal and non-conformal two-term models. However, the conformal case has already been warranted by Theorem 2, so the main thrust of Theorem 3 is on the non-conformal model.

Remark 13. The condition $m_1 + n_1 + m_1^* + m_2^* > 4$ is equivalent to requiring that X has at least 32 entries. We make this technical condition due to the following reasons. First, when $m_1 + n_1 + m_1^* + m_2^* \leq 3$, all two-term models satisfying Assumption 1 and Assumption 2 are conformal or type I non-conformal. Second, when

$m_1 + n_1 + m_1^* + m_2^* = 4$, the only possible configuration sets, denoted by $\{(p_1, q_1), (p_2, q_2)\}$, of the type II non-conformal two-term model are $\{(2,2),(4,1)\}$ when $\boldsymbol{X}$ is 4 × 4, $\{(2,2),(4,1)\}$ when $\boldsymbol{X}$ is 8 × 2, and $\{(2,2),(1,4)\}$ when $\boldsymbol{X}$ is 2 × 8. We consider these cases in Examples 1 and 2 in Appendix D, and demonstrate why such non-conformal two-term models are not identifiable, even when $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$ are in generic positions.

3 Hybrid Kronecker Product Model with Known Configurations

When the configuration set $\mathcal{C} = \{(p_k, q_k), 1 \le k \le K\}$ is known, we consider the following least squares problem.

$$\min \left\| \boldsymbol{Y} - \sum_{k=1}^{K} \lambda_k \boldsymbol{A}_k \otimes \boldsymbol{B}_k \right\|_F^2. \qquad (12)$$

When K = 1, such a problem can be solved by singular value decomposition of a rearranged version of matrix $\boldsymbol{Y}$. Specifically, the rearrangement operation $\mathcal{R}_{p,q}[\cdot]$ reshapes the $P \times Q$ matrix $\boldsymbol{Y}$ to a new $pq \times p^* q^*$ matrix such that

$$\mathcal{R}_{p,q}[\boldsymbol{Y}] = [\mathrm{vec}(\boldsymbol{Y}_{1,1}^{p^*,q^*}), \ldots, \mathrm{vec}(\boldsymbol{Y}_{p,q}^{p^*,q^*})]^T,$$

where $\boldsymbol{Y}_{i,j}^{p^*,q^*}$ stands for the (i, j)-th $p^* \times q^*$ block of matrix $\boldsymbol{Y}$ and $\mathrm{vec}(\cdot)$ is the vectorization operation that flattens a matrix to a column vector. It was observed by Van Loan and Pitsianis (1993) that the rearrangement operation can transform a Kronecker product to a vector outer product such that

$$\mathcal{R}_{p,q}[\boldsymbol{A} \otimes \boldsymbol{B}] = \mathrm{vec}(\boldsymbol{A})\mathrm{vec}(\boldsymbol{B})^T.$$

This can be seen from the fact that all the elements in the matrix $\boldsymbol{A} \otimes \boldsymbol{B}$ are in the form of $a_{i,j} b_{k,\ell}$, which is exactly the same as those in $\mathrm{vec}(\boldsymbol{A})\mathrm{vec}(\boldsymbol{B})^T$, where $a_{i,j}$ is the (i, j)-th element in $\boldsymbol{A}$ and $b_{k,\ell}$ is the (k,ℓ)-th element in $\boldsymbol{B}$. The re-arrangement operation $\mathcal{R}_{p,q}[\boldsymbol{Y}]$ is also linear and preserves the Frobenius norm.

Therefore, the least squares optimization problem $\min \| \boldsymbol{Y} - \lambda \boldsymbol{A} \otimes \boldsymbol{B} \|_F^2$, is equivalent to a rank-one matrix approximation problem since

$$\| \boldsymbol{Y} - \lambda \boldsymbol{A} \otimes \boldsymbol{B} \|_F^2 = \| \mathcal{R}_{p,q}[\boldsymbol{Y}] - \lambda \mathrm{vec}(\boldsymbol{A})\mathrm{vec}(\boldsymbol{B})^T \|_F^2,$$

whose solution is given by the leading component in the SVD of $\mathcal{R}_{m,n}[\boldsymbol{Y}]$ (Eckart and Young, 1936). If the multiple terms in (3) are of the same configuration, they can be retrieved from the singular components of $\mathcal{R}_{p,q}[\boldsymbol{Y}]$ as well.

When there are multiple terms $K > 1$ in model (3), but of different configurations, we propose to solve the optimization problem (12) through a backfitting algorithm (or an alternating least squares algorithm) by iteratively estimating λ_k, $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$ through

$$\min_{\lambda_k, \boldsymbol{A}_k, \boldsymbol{B}_k} \left\| \left(\boldsymbol{Y} - \sum_{i \neq k} \hat{\lambda}_i \boldsymbol{A}_i \otimes \boldsymbol{B}_i \right) - \lambda_k \boldsymbol{A}_k \otimes \boldsymbol{B}_k \right\|_F^2,$$

using the rearrangement operator and SVD, with fixed $\hat{\lambda}_i$, $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ ($i \neq k$) from the previous iteration.

When all configurations $\{(p_k, q_k)\}_{k=1}^K$ are distinct, the backfitting procedure for *h*KoPA is depicted in Algorithm 1, where $\mathrm{vec}_{p,q}^{-1}$ is the inverse of the vectorization operation that convert a column vector back to a $p \times q$ matrix. When r terms indexed by $k_1, \ldots, k_r$ in the *h*KoPA model have the same configuration, these terms are updated simultaneously in the backfitting algorithm by keeping the first r components from the SVD of the residual matrix $\boldsymbol{E}^{(k)} = \boldsymbol{Y} - \sum_{i \neq k_1, \ldots, k_r} \hat{\lambda}_i \boldsymbol{A}_i \otimes \boldsymbol{B}_i$. We also orthonormalize the components by the Gram-Schmidt procedure (Algorithm) at the end of each backfitting round. Algorithm 1 is also referred as alternating least squares (ALS) algorithm in the subsequent context.

Algorithm 1 Backfitting Least Squares Procedure

1: Set $\hat{\lambda}_1 = \hat{\lambda}_2 = \ldots = \hat{\lambda}_K = 0$.

2: **repeat**

3: **for** $k = 1$ **to** K **do**

4: $\boldsymbol{E}^{(k)} = \boldsymbol{Y} - \sum_{i \neq k} \hat{\lambda}_i \boldsymbol{A}_i \otimes \boldsymbol{B}_i$.

5: Compute SVD of $\mathcal{R}_{p_k, q_k}[\boldsymbol{E}^{(k)}]$:

$\mathcal{R}_{p_k,q_k}[\boldsymbol{E}^{(k)}] = \sum_{j=1}^{J} s_j \boldsymbol{u}_j \boldsymbol{v}_j^T$.

6: Update $\hat{\lambda}_k = s_1$, $\boldsymbol{A}_k = \text{vec}^{-1}_{p_k,q_k}(\boldsymbol{u}_1)$ and $\boldsymbol{B}_k = \text{vec}^{-1}_{p_k^*,q_k^*}(\boldsymbol{v}_1)$.

7: **end for**

8: **until** convergence

9: Orthonormalize the components by Algorithm.

10: Return $\{(\hat{\lambda}_k, \boldsymbol{A}_k, \boldsymbol{B}_k)\}_{k=1}^{K}$.

4 Hybrid KoPA with Unknown Configurations

In this section, we consider the case when the model configuration $\mathcal{C} = \{(p_k, q_k)\}_{k=1}^{K}$ is unknown. We use a greedy method similar to forward stepwise selection to obtain the approximation by iteratively adding one Kronecker product at a time, based on the residual matrix obtained from the previous iteration. Specifically, we start the algorithm with $\boldsymbol{Y}^{(1)} = \boldsymbol{Y}$, and at iteration t, we obtain

$$\boldsymbol{Y}^{(t)} = \boldsymbol{Y} - \sum_{i=1}^{t-1} \hat{\lambda}_i \boldsymbol{A}_i \otimes \boldsymbol{B}_i,$$

where $\hat{\lambda}_i$, $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ are obtained in the previous iteration. Then we use the single-term KoPA with unknown configuration proposed in Cai et al. (2019) to obtain

$$\min_{\lambda_t, \boldsymbol{A}_t, \boldsymbol{B}_t} \| \boldsymbol{Y}^{(t)} - \lambda_t \boldsymbol{A}_t \otimes \boldsymbol{B}_t \|_F^2.$$

The procedure is repeated until a stopping criterion is reached as detailed in Algorithm 2. The algorithm without step 10 is referred later as Algorithm 2′.

Algorithm 2 Greedy Additive Algorithm for *h*KoPA Estimation

1: Set $\boldsymbol{Y}^{(1)} = \boldsymbol{Y}$, $\hat{K} = T_{max}$.

2: **for** t = 1 **to** T_{max} **do**

3: **for** all possible configuration (p, q) **do**

4: Compute SVD for $\mathcal{R}_{p,q}[\boldsymbol{Y}^{(t)}]$: $\mathcal{R}_{p,q}[\boldsymbol{Y}^{(t)}] = \sum_{j=1}^{J} s_j \boldsymbol{u}_j \boldsymbol{v}_j^T$.

5: Set $\hat{\lambda}_t^{(p,q)} = s_1$, $\boldsymbol{A}_t^{(p,q)} = \text{vec}^{-1}_{p,q}(\boldsymbol{u}_1)$ and $\boldsymbol{B}_t^{(p,q)} = \text{vec}^{-1}_{p^*,q^*}(\boldsymbol{v}_1)$.

6: Compute $\boldsymbol{S}_t^{(p,q)} = \hat{\lambda}_t^{(p,q)} \boldsymbol{A}_t^{(p,q)} \otimes \boldsymbol{B}_t^{(p,q)}$.

7: **end for**

8: Compute

$$(\hat{p}_t, \hat{q}_t) = \arg\min_{(p,q)} PQ \log \frac{\| \boldsymbol{Y}^{(t)} - \boldsymbol{S}_t^{(p,q)} \|_F^2}{PQ} + \kappa \eta.$$

9: Set $\hat{\lambda}_t = \hat{\lambda}_t^{(\hat{p}_t, \hat{q}_t)}$, $\boldsymbol{A}_t = \boldsymbol{A}_t^{(\hat{p}_t, \hat{q}_t)}$ and $\boldsymbol{B}_t = \boldsymbol{B}_t^{(\hat{p}_t, \hat{q}_t)}$.

10: (ALS Refinement) Refine $\{(\hat{\lambda}_i, \boldsymbol{A}_i, \boldsymbol{B}_i)\}_{i=1}^t$ with respect to configuration set $\{(\hat{p}_i, \hat{q}_i)\}_{i=1}^t$ using Algorithm 1.

11: **if** a stopping criterion is met **then**

12: Set $\hat{K} = t$.

13: break

14: **end if**

15: Set $\boldsymbol{Y}^{(t+1)} = \boldsymbol{Y} - \sum_{i=1}^{t} \hat{\lambda}_i \boldsymbol{A}_i \otimes \boldsymbol{B}_i$.

16: **end for**

17: Return $\{(\hat{\lambda}_t, \boldsymbol{A}_t, \boldsymbol{B}_t)\}_{t=1}^{\hat{K}}$.

Some implementation details are as follows:

Overall Objective Function and The Greedy Search Algorithm: The formulation of the data generating mechanism (3) and (4) naturally suggests an overall objective function in the form of

$$\mathrm{cIC}_\kappa(K, (p_i, q_i), i = 1, \ldots, K) = PQ \log \frac{\| \boldsymbol{Y} - \sum_{i=1}^{K} \hat{\lambda}_i \boldsymbol{A}_i \otimes \boldsymbol{B}_i \|_F^2}{PQ - \sum_{i=1}^{K} (p_i q_i + p_i^* q_i^*)} + \kappa \sum_{i=1}^{K} (p_i q_i + p_i^* q_i^*), \quad (13)$$

where $\hat{\lambda}_i, \boldsymbol{A}_i, \boldsymbol{B}_i \ (i = 1, \ldots, K)$ are the estimators obtained through Algorithm 1 in Section 3, given $K, (p_i, q_i, p_i^*, q_i^*), i = 1, \ldots, K$. Here $\sum_{i=1}^{K} (p_i q_i + p_i^* q_i^*)$ is the number of parameters in the model and κ is the penalty coefficient on model complexity. We refer to the criterion in (13) as the cumulative information criterion, denoted by cIC_κ. In particular, when $\kappa = 2$, cIC_κ corresponds to AIC and when $\kappa = \log PQ$, cIC_κ

corresponds to Bayes information criterion (BIC) (Schwarz, 1978). As shown in Cai et al. (2019), in a single-term Kronecker product case, when the signal-to-noise ratio is sufficiently large, minimizing such an information criterion produces a consistent estimate of the true configuration.

Unfortunately it may not be practical to optimize such an objective function, since it would require an exhaustive search over all possible configurations. For computational efficiency, we use a greedy algorithm (with refinement) to obtain a solution. Specifically we propose the step-wise algorithm which, at t-th step, uses

$$\mathrm{IC}_{\kappa}^{(t)}(p,q\,|\,(\hat{p}_i,\hat{q}_i),1\le i\le t-1)=PQ\log\frac{\|\boldsymbol{Y}^{(t)}-\hat{\lambda}_t^{(p,q)}\boldsymbol{A}_t^{(p,q)}\otimes\boldsymbol{B}_t^{(p,q)}\|_F^2}{PQ-\eta^{(t-1)}}+\kappa\eta^{(t-1)}+\kappa(pq+p^*q^*), \quad (14)$$

where $\eta^{(t-1)}=\sum_{i=1}^{t-1}(\hat{p}_i\hat{q}_i+\hat{p}_i^*\hat{q}_i^*)$, to detemine the "best" configuration $(\hat{p}_t,\hat{q}_t)$ of a new term to be added to the model (given the existing $(t-1)$ terms), and terminates the build-up according to the stopping rule

$$\hat{K}=\min\left\{t:\mathrm{cIC}_{\kappa}(t+1)\geqslant\mathrm{cIC}_{\kappa}(t)\right\}, \quad (15)$$

Algorithm 2 amounts to a greedy algorithm for optimizing the overall objective function in (13).

Refinement: Step 10 "ALS Refinement" in Algorithm 2 updates all the existing terms by Algorithm 1, with all the selected configurations fixed, at the end of each iteration. Without this step, Algorithm 2 is also of the boosting flavor, adding one term (a "weak " learner) in each iteration without modifying the existing terms. To distinguish the two versions, we refer to Algorithm 2 without Step 10 as Algorithm $2'$. Our simulation study in Section 5.1.4 suggests that Algorithm 2, with the refinement step, has the potential to achieve a better approximation of $\boldsymbol{\mathcal{X}}$, and select the number of terms/configurations more accurately, comparing with Algorithm $2'$. On the other hand, the refinement at each iteration will increase the computational cost significantly. Therefore, if the computation is of primary concern, we recommend Algorithm $2'$ in practice, which does not involve any intermediate refinement, but can

have a final round of refinement using Algorithm 1 after the terms/configurations have been decided.

Remark 14.

Strictly speaking, the number of parameters in (13) and (14) should be calculated under the constraint that terms of conformal configurations are orthogonal (see Definition 1 and 2 of conformality and orthogonality in Section 2.2). We choose the present formulation for several reasons. First, if all terms have the same configuration, it is easy to count how many free parameters there are under the orthogonality constraints. However, if different configurations are present, it is difficult to express this number explicitly. Second, in this paper we intend to deal with matrices of large dimensions, hence the reduction of the number of free parameters due to orthogonality constraints is of a very small fraction of the total number of parameters used, and will have very minor impact on the information criterion. So we choose the present form for simplicity.

Remark 15. Note that our current formulation of the problem and the algorithms rely on the factorization of P and Q. Such factorization provides a better and cleaner structure for model identifiability and other discussions and presentations. On the other hand, it does limit the choices of possible configurations, when P and Q do not have many factors. We briefly discuss how to alleviate this limitation in practice. In fact, for model building and estimation, any (p, q, p^*, q^*) configuration such that $p = \lceil P / p^* \rceil$ and $q = \lceil Q / q^* \rceil$ can be used, where $\lceil x \rceil$ denotes the smallest integer larger than or equal to x. In this case, the estimation step (the rearrangement and SVD given a configuration, presented in Section 3) can be done in two different ways. One is to expand the matrix $\boldsymbol{Y}$ with several rows and columns so that it becomes a $(pp^*) \times (qq^*)$ matrix. These extra rows and columns can be imputed with zeros or through an iterative EM type of procedures in the estimation step to obtain $\boldsymbol{A}$ of size $p \times q$ and $\boldsymbol{B}$ of size $p^* \times q^*$. A second approach is to truncate the matrix $\boldsymbol{Y}$ by several rows and columns so that it becomes a $((p-1)p^*) \times ((q-1)q^*)$ matrix. Using this reduced-size matrix, we can estimate $\boldsymbol{A}^*$ of size $(p-1) \times (q-1)$ and $\boldsymbol{B}$ of size $p^* \times q^*$. Each element of the missing column and row in $\boldsymbol{A}$ can be estimated by a least squares using the corresponding unused elements in $\boldsymbol{Y}$ and the estimated $\boldsymbol{B}$.

Combining $\boldsymbol{A}^{*}$ and the estimated missing row and column results in the estimated $\boldsymbol{A}$ of size $p \times q$. The evaluation of the corresponding IC criteria (13) and (14) for configuration determination need to be adjusted, so that only the observed entries of $\boldsymbol{Y}$ and the estimated matrix $\boldsymbol{A} \otimes \boldsymbol{B}$ truncated to size $P \times Q$ are involved in the evaluation. Such an approach expands the set of possible configurations significantly, creating extra flexibility and model robustness, though it also demands significantly higher computational cost for configuration selection. A compromise is to consider (p^{*}, q^{*}) being powers of 2. If $\boldsymbol{Y}$ is an image, a common practice is to super-sample or sub-sample the pixels and then apply the two aforementioned approaches respectively. Further investigation on more efficient model building procedures is needed.

5 Empirical Examples

5.1 Simulation

Intuitively, the comparison of *h*KoPA with SVD and KoPA goes like follows: *h*KoPA performs similarly to SVD if the true signal has low rank, and similarly to KoPA if the true signal is of low rank under KPD. On the other hand, *h*KoPA performs much better if the true signal is generated with terms of different configurations. This intuition has been confirmed by empirical results based on a 3-term Kronecker product model, which we choose to report in Appendix A for the interest of space.

In this section, we focus on the performance of the least squares backfitting algorithm in Algorithm 1 and the iterative algorithm in Algorithm 2 for a two-term Kronecker product model and determine the factors that affect the estimation accuracy and convergence speed of the algorithm.

In particular we focus on Model (7), as it reveals the identification issue and allows the study of the impact of interaction strength. We repeat (7) here for easy reference.

$$\boldsymbol{X} = \lambda_1 \boldsymbol{A}_1 \otimes \boldsymbol{B}_1 + \lambda_2 \boldsymbol{A}_2 \otimes \boldsymbol{B}_2 + \lambda_{12} \boldsymbol{A}_1 \otimes \boldsymbol{C} \otimes \boldsymbol{B}_2,$$

where $\boldsymbol{A}_1 \in \boldsymbol{A}_2$ and are orthogonal, and $\boldsymbol{B}_2 \in \boldsymbol{B}_1$ and are orthogonal. Recall that strictly speaking, this is a two term model with two different configurations and the

third term $\boldsymbol{A}_1 \otimes \boldsymbol{C} \otimes \boldsymbol{B}_2$ is called the interaction between the two configurations, and its strength is controlled by the coefficient λ_{12}. We first generate $\boldsymbol{A}_k, \boldsymbol{B}_k$ and $\boldsymbol{C}$ as normalized Gaussian random matrices with i.i.d. standard normal entries. We then perform the Gram-Schmidt orthogonalization so that $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ are orthogonal with each other in the sense of Assumption 3, and so are $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$. Finally all these matrices are rescaled to have Frobenius one.

In this example, we set $P = 2^M, Q = 2^N$ such that any conformable configuration (p, q) can be written as $p = 2^m, q = 2^n$ for some integers $0 \leqslant m \leqslant N$ and $0 \leqslant n \leqslant N$. To ease the notation, we simply use (m, n) to denote the configuration $(p, q) = (2^m, 2^n)$.

The observed $\boldsymbol{Y}$ is a corrupted version of $\boldsymbol{X}$ with additive Gaussian noise such that

$$\boldsymbol{Y} = \boldsymbol{X} + \frac{\sigma}{2^{(M+N)/2}} \boldsymbol{E},$$

where $\boldsymbol{E}$ is a $2^M \times 2^N$ matrix with i.i.d. standard Gaussian entries.

We express the fitted $\boldsymbol{Y}$ as

$$\boldsymbol{Y} = \hat{\lambda}_1 \boldsymbol{A}_1 \otimes \boldsymbol{B}_1 + \hat{\lambda}_2 \boldsymbol{A}_2 \otimes \boldsymbol{B}_2,$$

where $\boldsymbol{A}_1 \otimes \boldsymbol{B}_1$ and $\boldsymbol{A}_2 \otimes \boldsymbol{B}_2$ are the two Kronecker products with configurations (m_1, n_1) and (m_2, n_2) correspondingly. Recall that either Ortho-A (9) or Ortho-B (10) can be adopted to represent $\boldsymbol{Y}$ and either representation is unique. Most of the simulations are carried out under Ortho-A, which is also consistent with Assumption 3. In Section 5.1.2 we also study the impact of choosing different orthogonalizations on the estimation.

We use the following notations of various estimation errors for easier reference.

$$\begin{aligned}
&\mathrm{EY} = \| \boldsymbol{Y} - \boldsymbol{Y} \|_F^2, \\
&\mathrm{EL1} = |\hat{\lambda}_1 / \lambda_1 - 1|, \quad \mathrm{EL1c} = |\hat{\lambda}_1 / \lambda_1^c - 1|, \\
&\mathrm{EL2} = |\hat{\lambda}_2 / \lambda_2 - 1|, \quad \mathrm{EL2c} = |\hat{\lambda}_2 / \lambda_2^c - 1|, \\
&\mathrm{EA1} = \| \boldsymbol{A}_1 - \boldsymbol{A}_1 \|_F^2, \quad \mathrm{EA2} = \| \boldsymbol{A}_2 - \boldsymbol{A}_2 \|_F^2, \quad \mathrm{EA2c} = \| \boldsymbol{A}_2 - \boldsymbol{A}_2^c \|_F^2, \\
&\mathrm{EB1} = \| \boldsymbol{B}_1 - \boldsymbol{B}_1 \|_F^2, \quad \mathrm{EB1c} = \| \boldsymbol{B}_1 - \boldsymbol{B}_1^c \|_F^2, \quad \mathrm{EB2} = \| \boldsymbol{B}_2 - \boldsymbol{B}_2 \|_F^2.
\end{aligned}$$

where $\boldsymbol{A}_2^c, \boldsymbol{B}_1^c, \lambda_1^c$ and λ_2^c are defined in (9) and (10). We also define the reconstruction error (RCE),

$$\mathrm{RCE} = \frac{\|\boldsymbol{Y} - \boldsymbol{X}\|_F^2}{\|\boldsymbol{X}\|_F^2} \qquad (16)$$

which will be used later to compare the performance of different models.

5.1.1 The Benchmark Case

In the benchmark case, we use $M = N = 9, (m_1, n_1) = (4,4), (m_2, n_2) = (5,5), \lambda_1 = \lambda_2 = \lambda_{12} = 1$, $\sigma = 1$ to generate the signal matrix $\boldsymbol{X}$ in (7) and the observed matrix $\boldsymbol{Y}$. Algorithm 1 is applied to fit $\boldsymbol{Y}$ with the true configurations and the orthogonalization is done by Ortho-A. In other words, we are estimating the matrices in (9). The errors from the first 20 iterations are reported in Figure 1, where we compare $\boldsymbol{B}_1$ to $\boldsymbol{B}_1^c$ (instead of $\boldsymbol{B}_1$) under Ortho-A. The convergence of the estimators is observed at roughly the 10-th iteration.

From the middle panel of Figure 1, it is seen that the smaller matrices $\boldsymbol{A}_1$ and $\boldsymbol{B}_2$ usually have smaller estimation errors as EA1 and EB2 are smaller than EB1c and EA2 after convergence. Note that in the definitions of these estimation errors, all involved matrices are scaled to have Frobenius norm 1, so for example, EA1 essentially corresponds to the angle between $\mathrm{vec}(\boldsymbol{A}_1)$ and $\mathrm{vec}(\boldsymbol{A}_1)$. Similar phenomenon has been observed in estimating singular vectors of a low rank matrix (Cai et al., 2018). On the other hand, before convergence and especially in the first iteration, the errors EA1 and EA2 are much larger than EB1c and EB2. Here we provide two explanations.

Suppose the full Kronecker product decomposition of $\boldsymbol{A}_2$ is written as $\boldsymbol{A}_2 = \sum_{k=1}^{K} \mu_k \boldsymbol{A}_{2,k} \otimes \boldsymbol{C}_k$ where $\boldsymbol{A}_{2,k}$ has the same dimension (m_1, n_1) as $\boldsymbol{A}_1$. Then we have

$$\boldsymbol{X} = \lambda_1 \boldsymbol{A}_1 \otimes \boldsymbol{B}_1 + \lambda_2 \sum_{k=1}^{K} \mu_k \boldsymbol{A}_{2,k} \otimes (\boldsymbol{C}_k \otimes \boldsymbol{B}_2) + \lambda_{12} \boldsymbol{A}_1 \otimes \boldsymbol{C} \otimes \boldsymbol{B}_2,$$

where $\{\mathrm{vec}(\boldsymbol{A}_1), \mathrm{vec}(\boldsymbol{A}_{2,1}), \ldots, \mathrm{vec}(\boldsymbol{A}_{2,K})\}$ are orthogonal with each other. Then in the first iteration, $\boldsymbol{A}_1$ and $\boldsymbol{B}_1$ are obtained from the singular value decomposition of the re-arranged matrix (with configuration (m_1, n_1))

$$\mathcal{R}_{m_1,n_1}[\boldsymbol{X}] = \lambda_1 \mathrm{vec}(\boldsymbol{A}_1)\mathrm{vec}(\boldsymbol{B}_1)^T + \lambda_2 \sum_{k=1}^{K} \mu_k \mathrm{vec}(\boldsymbol{A}_{2,k})\mathrm{vec}(\boldsymbol{C}_k \otimes \boldsymbol{B}_2)^T + \lambda_{12}\mathrm{vec}(\boldsymbol{A}_1)\mathrm{vec}(\boldsymbol{C} \otimes \boldsymbol{B}_2)^T.$$

Then $\mathcal{R}_{m_1,n_1}[\boldsymbol{X}]^T \mathrm{vec}(\boldsymbol{A}_1) \propto \mathrm{vec}(\boldsymbol{B}_1^c)$ but $\mathcal{R}_{m_1,n_1}[\boldsymbol{X}]\mathrm{vec}(\boldsymbol{B}_1^c) \not\propto \mathrm{vec}(\boldsymbol{A}_1)$ since $\mathrm{tr}(\boldsymbol{C}^T\boldsymbol{C}_k)$ ($k = 1, \ldots, K$) are usually not zero. Therefore, in power iterations, plugging in the true value of $\boldsymbol{A}_1$ gives the true value of $\boldsymbol{B}_1^c$, but the reverse is not true.

Alternatively, one can show that the error EB1c is smaller than EA1 in the first iteration when $\lambda_2^2 < \lambda_1^2 + \lambda_{12}^2$. Let $\mathrm{vec}(\boldsymbol{A}_1) = c(\mathrm{vec}(\boldsymbol{A}_1) + \mathrm{vec}(\Delta\boldsymbol{A}_1))$ for some $\mathrm{vec}(\Delta\boldsymbol{A}_1) \perp \mathrm{vec}(\boldsymbol{A}_1)$. Then

$$\mathrm{vec}(\boldsymbol{B}_1) = \mathcal{R}_{m_1,n_1}[\boldsymbol{X}]^T \mathrm{vec}(\boldsymbol{A}_1) = c(\mathrm{vec}(\boldsymbol{B}_1^c) + \lambda_2 / \lambda_1^c \mathcal{R}_{m_1,n_1}[\boldsymbol{A}_2 \otimes \boldsymbol{B}_2]^T \mathrm{vec}(\Delta\boldsymbol{A}_1)).$$

It is easy to verify that

$$\begin{aligned} \| \lambda_2 / \lambda_1^c \mathcal{R}_{m_1,n_1}[\boldsymbol{A}_2 \otimes \boldsymbol{B}_2]^T \mathrm{vec}(\Delta\boldsymbol{A}_1) \|_2^2 &\leqslant \frac{\lambda_2^2}{\lambda_1^2 + \lambda_{12}^2} \left\| \mathcal{R}_{m_1,n_1}[\boldsymbol{A}_2 \otimes \boldsymbol{B}_2] \right\|_S^2 \| \mathrm{vec}(\Delta\boldsymbol{A}_1) \|^2 \\ &\leqslant \frac{\lambda_2^2}{\lambda_1^2 + \lambda_{12}^2} \| \mathrm{vec}(\Delta\boldsymbol{A}_1) \|^2 . \end{aligned}$$

Hence, when $\lambda_2^2 < \lambda_1^2 + \lambda_{12}^2$, EB1c is smaller than EA1 in the first iteration. The absolute errors in the coefficients λ_i, $|\mathrm{EL1c}|$ and $|\mathrm{EL2}|$, decrease and converge as expected.

5.1.2 Ortho-A and Ortho-B Representations

In this part, we investigate the influence of the choice of representation: Ortho-A and Ortho-B. In the benchmark case above, we have obtained the errors for EB1c and EA2c under Ortho-A. We will compare them with the estimation obtained under Ortho-B, in which in each iteration of Algorithm 1 we perform orthogonalization under Ortho-B. The errors are plotted in Figure 2. From the figure, it is seen that, under Ortho-A, EA2 and EB1c are smaller compared with EA2c and EB1, while EA2c and EB1 are smaller under Ortho-B. We also note that a symmetry exists between the

two representations. The component $\boldsymbol{A}_1$ and $\boldsymbol{B}_1^c$ under Ortho-A are of the same position to $\boldsymbol{A}_2^c$ and $\boldsymbol{B}_2$ under Ortho-B. The error curves of EA2 and EB1c under Ortho-A should be similar to the ones of EB1 and EA2c under Ortho-B, correspondingly. This phenomenon is observed in Figure 2 by comparing the curves in the left plot with the ones in the right plot.

5.1.3 Impact of Interaction Strength

In this part, we compare the accuracies and convergence rates of different parameter estimates under different absolute interaction strengths under Model (7). We fix the signal-to-noise ratio in order to isolate the impact of the interaction strength. Specifically, we set the value of α in the range $\alpha \in \{0.0, 0.5, 1.0, 1.5, 2.0\}$, and $\lambda_1 = 1/\sqrt{1+\alpha^2}$, $\lambda_2 = 1$, and $\lambda_{12} = \alpha/\sqrt{1+\alpha^2}$. The orthogonalization is done under Ortho-A, hence $\lambda_1^c = 1$. The value of α controls the "correlation" between the first Kronecker product and second one in (9). In particular, $\alpha^2/(1+\alpha^2)$ represents the proportion of $\| \lambda_1^c \boldsymbol{A}_1 \otimes \boldsymbol{B}_1^c \|_F^2$ that is linearly dependent to $\boldsymbol{A}_2 \otimes \boldsymbol{B}_2$.

The fitting error EY under different relative interaction strength is reported in Figure 3. A similar accuracy after convergence is observed for all different relative interaction strength α. It is seen that Algorithm 1 converges slower when higher dependence exists between the two configurations. In the absence of interaction (α = 0), Algorithm 1 converges in one iteration.

Figure 4 plots the error curves of the six fitted components. It is seen that the errors of the components converge to a similar value for different relative interaction strength α's. Again, the value of α only affects the convergence speed. We note that the intermediate errors of EA1 and EA2 are larger than the ones of EB1c and EB2 but eventually they all converge to similar values. This phenomenon is due to the potentially large estimation error of EA1 in the first iteration as discussed in the benchmark section.

5.1.4 Unknown Configurations

In this part, we simulate the data in the same way as in Section 5.1.3 and use Algorithm 2 with the stopping rule in (15) to fit *h*KoPA model without assuming the

true figuration. Algorithm 2' (without Step 10) is also considered. The results are reported in Table 1.

From the table, it is clear that although the true configuration set contains only two configurations (5, 5) and (4, 4), Algorithm $2'$ requires a third or fourth term (configuration) except for the case without the interaction ($\alpha = 0$). More terms are used as the interaction is strengthened. It is a direct consequence of the greediness of the iterative algorithm. On the other hand, Algorithm 2 stops after two iterations, selecting the two true configurations, for all levels of interaction strength.

The reconstruction errors defined in (16) are also reported in Table 1, in the rows labelled by "RCE". For Algorithm $2'$, we also try an additional ALS as a post-processing step after the algorithm stops. The corresponding RCEs are reported in the last row. The RCE reported in the second-to-last row are obtained using Algorithm $2'$ without the final ALS step. These larger RCEs (comparing to those reported in the last row of the "A-2" panel reveal that the redundant third and/or fourth configurations lead to an overfit. On the other hand, for Algorithm 2 ("A-2" panel), not only the correct number of Kronecker products is selected, but also the reconstruction error is much reduced, as seen in the last row of the upper panel "A-2 ".

5.2 Real Image Example

In this section, we demonstrate the performance of *h*KoPA on real image examples, and compare with the existing methods including SVD and KoPA. We present one example here, and leave the presentation of the other on the cameraman's image to Appendix B.

The left panel of Figure 5 is a 300×400 grayscaled image of column arcade from the Stoa of Attalos in Ancient Agora of Athens[1]. We denote this original image in grayscale by $\boldsymbol{Y}_0$, whose elements are real numbers on $[0,1]$ with 0 standing for black and 1 for white. We observe that there exist three major patterns in the image: (a) a repeated patterns for the columns; (b) a repeated patterns for the beams and shadows and (c) repeated regions for the surface textures. Specifically, pattern (a) suggests that there is a component of $\boldsymbol{Y}_0$ that can be written as $\boldsymbol{A}_a \otimes \boldsymbol{B}_a$, with $\boldsymbol{B}_a$

being the repeated vertical pattern (e.g. a matrix with a few (or one) columns and many rows for a vertical image) and $\boldsymbol{A}_a$ (a matrix with many columns and a few rows) represents its signal strength (mainly across all columns). A zero in $\boldsymbol{A}_a$ indicates that the vertical image is not present at that location.

Similarly, pattern (b) suggests a component $\boldsymbol{A}_b \otimes \boldsymbol{B}_b$, where $\boldsymbol{B}_b$ is the horizontal pattern to be repeated and $\boldsymbol{A}_b$ is the repeating strength. Pattern (c) gives a Kronecker product $\boldsymbol{A}_c \otimes \boldsymbol{B}_c$, where $\boldsymbol{B}_c$ is the repeated local texture and $\boldsymbol{A}_c$ is the repeating amplitude across the whole image. One can anticipate, from above observations, that *h*KoPA is more capable than SVD and KoPA in describing the hybrid patterns, where as the latter two methods can only utilize one configuration.

We consider a denoising problem, in which the original grayscaled image is corrupted with an additive noise of size $\sigma = 0.3$. Specifically, the image on the right panel of Figure 5, denoted by $\boldsymbol{Y}$, is generated as

$$\boldsymbol{Y} = \boldsymbol{Y}_0 + \sigma \boldsymbol{E},$$

where $\boldsymbol{E}$ is a matrix of i.i.d. standard Gaussian random variables with standard deviation σ. The goal of denoising of $\boldsymbol{Y}$ is to find a matrix $\boldsymbol{Y}$ that can ideally reveal the unknown original matrix $\boldsymbol{Y}_0$. A performance measure of $\boldsymbol{Y}$ is the reconstruction error (similar to the one defined in (16))

$$\mathrm{RCE} = \frac{\| \boldsymbol{Y} - \boldsymbol{Y}_0 \|_F^2}{\| \boldsymbol{Y}_0 \|_F^2}.$$

In this example, we examine three methods: *h*KoPA, KoPA and SVD. All of them yield a $\boldsymbol{Y}$ as a "low-rank" approximation of $\boldsymbol{Y}$: SVD decomposes $\boldsymbol{Y}_0$ through singular value decomposition, KoPA represents $\boldsymbol{Y}_0$ with respect to the Kronecker product decomposition with identical configurations, and *h*KoPA further allows the configurations of terms in KoPA to be different. Specifically, in *h*KoPA method, we apply Algorithm $2'$ proposed in Section 4 with $\kappa = \log(300 \times 400)$ (BIC). For KoPA, $(\hat{p}_1, \hat{q}_1)$ is found in the same way as in Algorithm $2'$ and $(\hat{p}_k, \hat{q}_k) \equiv (\hat{p}_1, \hat{q}_1)$ is forced for all further terms $k \geq 2$. The SVD approach can be viewed as a special case of KoPA, where $(\hat{p}_k, \hat{q}_k)$ are fixed at $(P, 1)$ (or $(1, Q)$) for all terms $k \geq 1$.

We report the configurations $(\hat{p}_k, \hat{q}_k)$, the cumulative percentage of variation ($\|\boldsymbol{Y}\|_F^2 / \|\boldsymbol{Y}\|_F^2$, denoted by c.p.v.) explained and the reconstruction error (RCE) for the first 10 terms in Table 2. From the cumulative percentage of variation explained, SVD is less capable of representing $\boldsymbol{Y}$ compared to KoPA and *h*KoPA given the same number of terms. In terms of reconstruction error, for each method, the smallest error (highlighted) is obtained when the model is about to overfit, i.e. when the $c.p.v.$ is close to $76.99 = \|\boldsymbol{Y}_0\|_F^2 / \|\boldsymbol{Y}\|_F^2$, the c.p.v. of the original image. Among all three methods, *h*KoPA achieves the smallest reconstruction error as it is capable of representing the hybrid structures of the original image. Figure 6 plots the reconstruction error against the number of parameters up to 20 terms for all three methods. It can be seen that *h*KoPA not only has the smallest reconstruction error but also uses the least number of parameters. Of course, due to its extra flexibility, when more-than-necessary number of terms are used, *h*KoPA is more likely to overfit compared to KoPA and SVD, as seen from Figure 6 when the number of parameters is greater than 6000. Such an over-fitting is prevented by the stopping rule (15).

The first 6 components fitted by *h*KoPA are plotted in Figure 7. It is seen that each additional component adds more details to the reconstructed image. The first component constructs a thumbnail image with big pixels that recovers the local surfaces. The second component is a rank-one matrix that recovers the repeated vertical patterns observed on the columns. The third and forth components further supplement the details on the shaded floor. The sixth components recovers the repeated horizontal patterns that appears on the ceiling and in the shadows. It is obvious that KoPA cannot represent the patterns from the second and the sixth component and SVD cannot capture the patterns given by components 1, 3, 4 and 5. We plot the best images reconstructed by the three methods in Figure 8. It is quite evident that the *h*KoPA provides the best approximation to the original image.

The computation time used for this example on a typical desktop[2] is reported as follows. SVD takes 9.7 milliseconds. KoPA involves one iteration of configuration selection loop and takes 0.53 seconds in total. *h*KoPA involves 20 iterations of

configuration selection loops and spends 9.63 seconds, about 0.48 seconds per iteration on average.

The implementation of *h*KoPA for this example uses $\kappa = \log(300 \times 400)$ for both IC_κ and cIC_κ, corresponding to the BIC. To compare the performance of AIC (i.e. $\kappa = 2$) and BIC, we report the selected number of terms $(\hat{K})$, the RCE without back-fitting and the RCE with back-fitting in Table 3. In the top panel of Table 3, the number of terms $\hat{K}$ is determined by the stopping criterion (13). In the bottom panel, we report the "optimal number of terms" selected by an oracle who knows the true image $\boldsymbol{\mathcal{Y}}_0$ and hence is able to calculate the RCE for the calculation of cIC_κ by replacing the observed $\boldsymbol{\mathcal{Y}}$ with the true $\boldsymbol{\mathcal{Y}}_0$ in (13). We see that the stopping criterion BIC gives the same performance as the oracle for *h*KoPA. On the other hand, the performance of AIC and BIC can be different for both KoPA and *h*KoPA, although they have been proven to have the same asymptotic performance for KoPA, as shown by Cai et al. (2019). We would recommend the use of BIC in practice, which gives a model with less complexity. We note that although it seems that BIC selects more terms than AIC for both KoPA and *h*KoPA in Table 3, the selected configurations involve less number of parameters, resulting in a smaller total number of parameters (as reported in the row "Selected # parameters"). A theoretical study and comparison of different information criteria is important but also very challenging. It is also interesting to develop a data-driven procedure for the selection of κ. More detailed investigation is needed.

6 Conclusion and Discussion

In this paper, we extend the single-term KoPA model proposed in Cai et al. (2019) to a more flexible setting, which allows multiple terms with different configurations and allows the configurations to be unknown. Identifiability conditions are introduced to ensure unique representation of the model. And we propose two iterative estimation algorithms.

With a given set of configurations, we propose a least squares backfitting algorithm that updates the Kronecker product component iteratively. The simulation study

shows the performance of the algorithm and the impact of the linear dependency between the component matrices.

When the configurations are unknown, the extra flexibility of *h*KoPA allows for more parsimonious representation of the underlying matrix, though it brings the challenge of configuration determination. An iterative greedy algorithm is proposed to jointly determine the configurations and estimate each Kronecker product component. The algorithm adds one Kronecker product term to the model at a time by finding the best one term KoPA to the residual matrix obtained from the previous iteration, using the procedure proposed in Cai et al. (2019). By analyzing a benchmark image example, we demonstrate that the proposed algorithm is able to obtain reasonable *h*KoPA and the results are significantly superior over the direct low rank matrix approximation.

The matrix $\boldsymbol{X}$ is of dimension $P \times Q$. The more factors P and Q have, the more possible configurations there are, giving more leeway to find a better approximation. On the other hand, when P and Q do not have many factors, the *h*KoPA loses much of its flexibility. We have discussed some possible approaches (Remark 15) to allowing more choices of the configurations. A comprehensive investigation of a more efficient model building process is still needed. It is also of interest to provide theoretical guarantees of the model selection and estimation procedure.

As discussed in Section 3, the greedy algorithm for configuration determination is similar to the forward stepwise selection. The theoretical properties of the proposed methods need to be further investigated. For the stopping criterion of the greedy algorithm, existing methods on the rank determination (Minka, 2001; Lam and Yao, 2012; Bai et al., 2018) may be extended for the *h*KoPA model as well.

Acknowledgement

Chen's research is supported in part by National Science Foundation grants DMS-1737857, IIS-1741390, CCF-1934924, DMS-2027855 and DMS-2052949. Xiao's research is supported in part by National Science Foundation grants DMS-1454817 and DMS-2027855, DMS-2052949, and a research grant from NEC Labs America. The authors would like to thank an AE and two referees for their insightful comments

which significantly improve the quality of this paper. The authors report there are no competing interests to declare.

Notes

[1] The original image in color and in higher resolution is credited to Ian Kershaw on Flicker https://www.flickr.com/photos/moonboots/10927753/

[2] System: Windows Subsystem for Linux version 2, CPU: 12900KF (16 cores/ 24 threads), RAM: 32GB@6000MHz, interpreter: Intel distribution for Python 3.9.

References

Bai, Z., Choi, K. P., and Fujikoshi, Y. (2018). Consistency of AIC and BIC in estimating the number of significant components in high-dimensional principal component analysis. *The Annals of Statistics*, 46:1050–1076.

Cai, C., Chen, R., and Xiao, H. (2019). KoPA: Automated Kronecker product approximation. preprint https://arxiv.org/abs/1912.02392.

Cai, D., He, X., Wang, X., Bao, H., and Han, J. (2009). Locality preserving nonnegative matrix factorization. In *Twenty-First International Joint Conference on Artificial Intelligence*.

Cai, T. T., Zhang, A., et al. (2018). Rate-optimal perturbation bounds for singular subspaces with applications to high-dimensional statistics. *The Annals of Statistics*, 46(1):60–89.

Candes, E. J. and Plan, Y. (2010). Matrix completion with noise. *Proceedings of the IEEE*, 98(6):925–936.

Candès, E. J. and Recht, B. (2009). Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717.

Duarte, M. F. and Baraniuk, R. G. (2012). Kronecker compressive sensing. *IEEE Transactions on Image Processing*, 21(2):494–504.

Eckart, C. and Young, G. (1936). The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218.

Freund, Y., Schapire, R., and Abe, N. (1999). A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780):1612.

Grasedyck, L., Kressner, D., and Tobler, C. (2013). A literature survey of low-rank tensor approximation techniques. *GAMM-Mitteilungen*, 36(1):53–78.

Guillamet, D. and Vitrià, J. (2002). Non-negative matrix factorization for face recognition. In *Catalonian Conference on Artificial Intelligence*, pages 336–344. Springer.

Hoyer, P. O. (2004). Non-negative matrix factorization with sparseness constraints. *Journal of machine learning research*, 5(Nov):1457–1469.

Kaye, P., Laflamme, R., and Mosca, M. (2007). *An introduction to quantum computing*. Oxford University Press.

Lam, C. and Yao, Q. (2012). Factor modeling for high-dimensional time series: inference for the number of factors. *The Annals of Statistics*, 40(2):694–726.

Le, C. M., Levina, E., and Vershynin, R. (2016). Optimization via low-rank approximation for community detection in networks. *The Annals of Statistics*, 44(1):373–400.

Minka, T. P. (2001). Automatic choice of dimensionality for PCA. In *Advances in neural information processing systems*, pages 598–604.

Pauca, V. P., Shahnaz, F., Berry, M. W., and Plemmons, R. J. (2004). Text mining using non-negative matrix factorizations. In *Proceedings of the 2004 SIAM International Conference on Data Mining*, pages 452–456. SIAM.

Sainath, T. N., Kingsbury, B., Sindhwani, V., Arisoy, E., and Ramabhadran, B. (2013). Low-rank matrix factorization for deep neural network training with high-dimensional output targets. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6655–6659. IEEE.

Schwarz, G. (1978). Estimating the dimension of a model. *Ann. Statist.*, 6(2):461–464.

Van Loan, C. F. and Pitsianis, N. (1993). Approximation with Kronecker products. In *Linear algebra for large scale and real-time applications*, pages 293–314. Springer.

Werner, K., Jansson, M., and Stoica, P. (2008). On estimation of covariance matrices with Kronecker product structure. *IEEE Transactions on Signal Processing*, 56(2):478–491.

Wold, S., Esbensen, K., and Geladi, P. (1987). Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52.

Yang, J. and Leskovec, J. (2013). Overlapping community detection at scale: a nonnegative matrix factorization approach. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 587–596. ACM.

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*, pages 847–855.

Yuan, M. and Zhang, C.-H. (2016). On tensor completion via nuclear norm minimization. *Foundations of Computational Mathematics*, 16(4):1031–1068.

Zhang, T., Fang, B., Tang, Y. Y., He, G., and Wen, J. (2008). Topology preserving non-negative matrix factorization for face recognition. *IEEE Transactions on Image Processing*, 17(4):574–584.

Zhang, Y. and Yeung, D.-Y. (2012). Overlapping community detection via bounded nonnegative matrix tri-factorization. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 606–614. ACM.

Zou, H., Hastie, T., and Tibshirani, R. (2006). Sparse principal component analysis. *Journal of computational and graphical statistics*, 15(2):265–286.

Fig. 1 Errors for benchmark setting

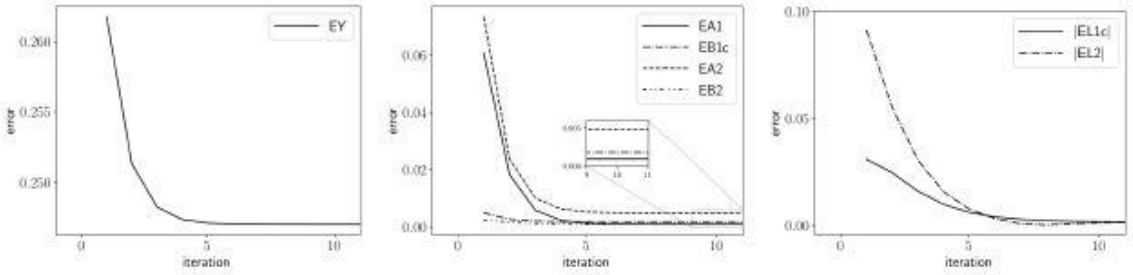

Fig. 2 Errors for benchmark setting with different orthogonalizations.

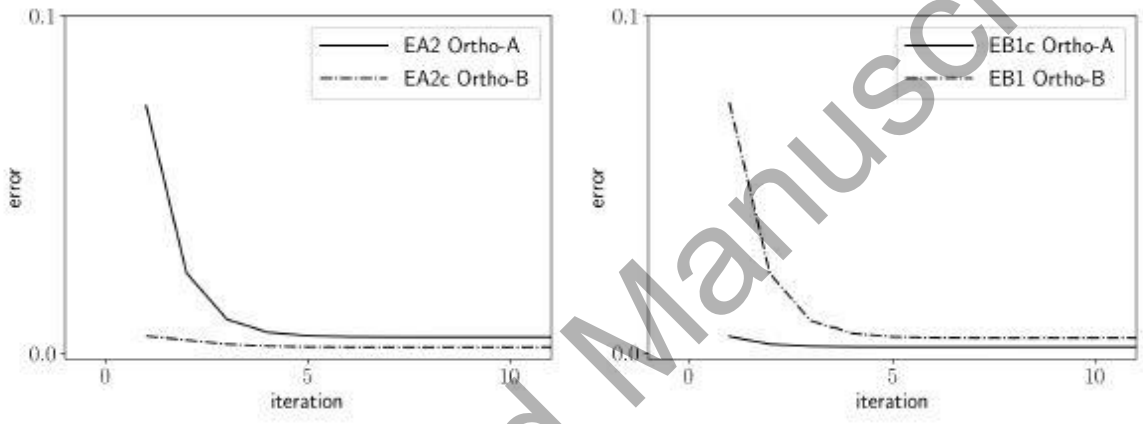

Fig. 3 Errors of Y with different relative interaction strength α's.

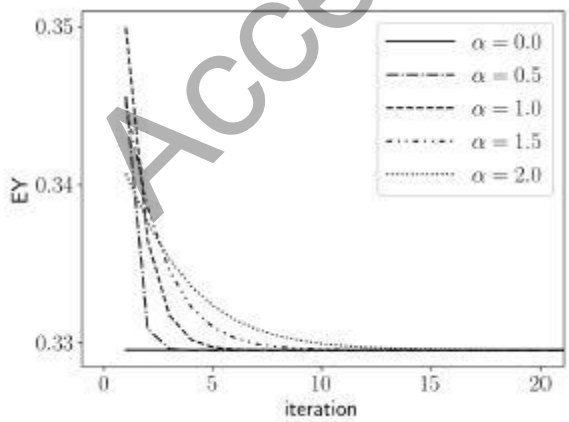

Fig. 4 Errors for components under different relative interaction strength αs.

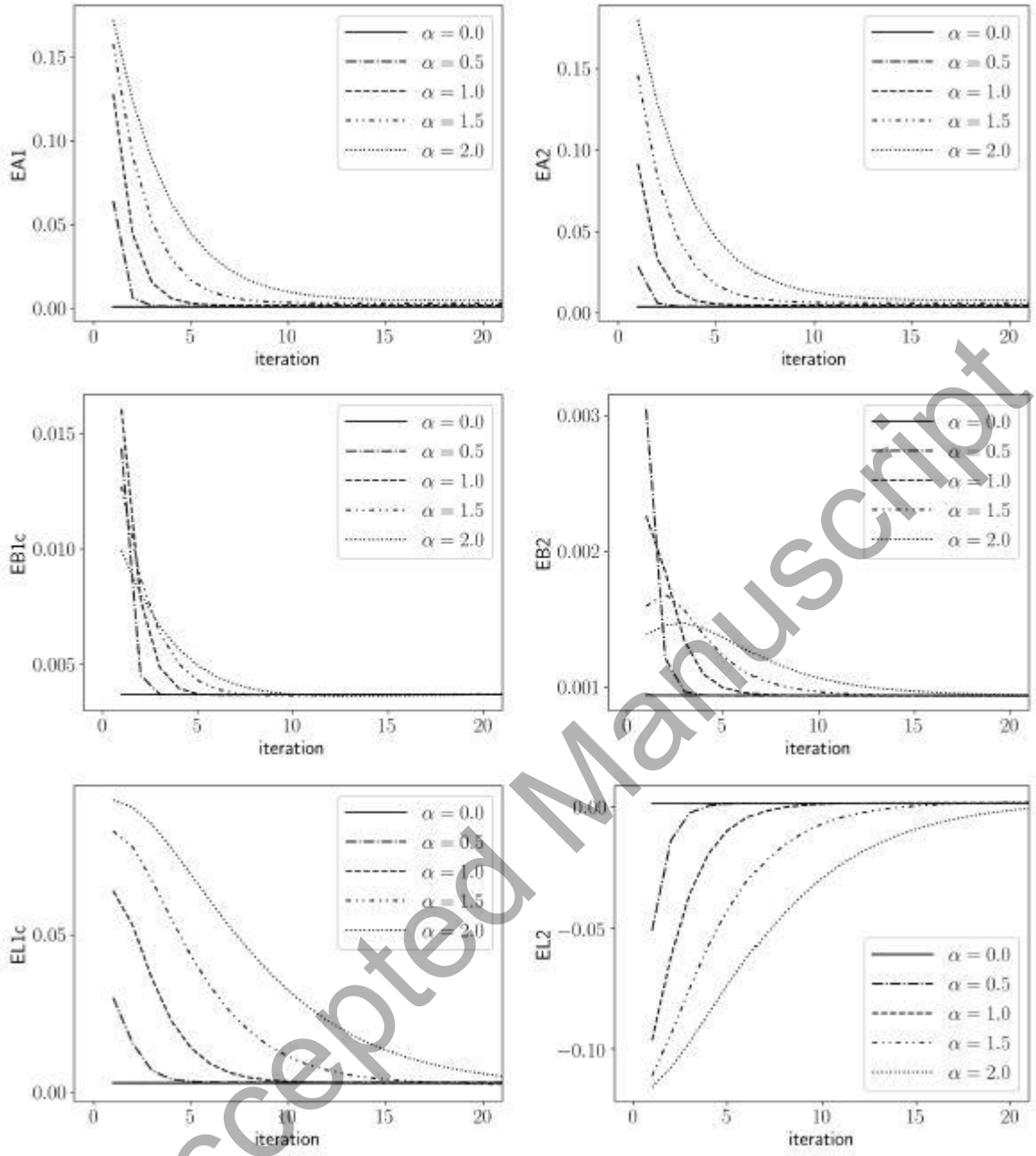

Fig. 5 The grayscaled image of Stoa of Attalos and a noisy image with additive Gaussian noise ($\sigma = 0.3$).

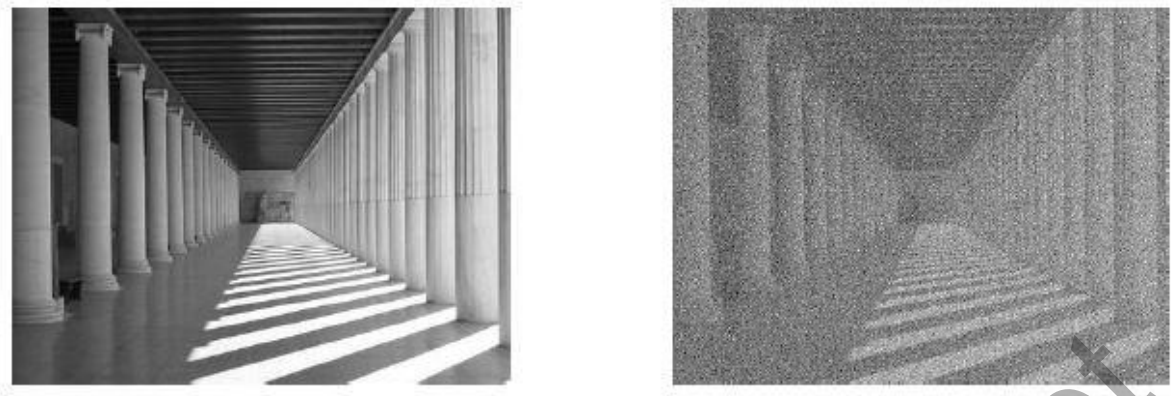

Fig. 6 Reconstruction error against number of parameters for the three methods. The optimal *h*KoPA model selected by stopping rule (15) is marked by $\star$.

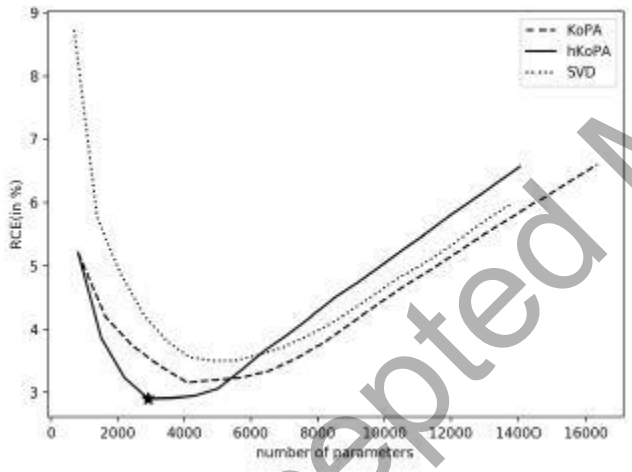

Fig. 7 Components of *h*KoPA for the first 6 iterations. (Column 1) component $\boldsymbol{A}_k$. (Column 2) component $\boldsymbol{B}_k$. (Column 3) component $\boldsymbol{A}_k \otimes \boldsymbol{B}_k$. (Column 4) cumulative components $\sum_{j=1}^{k} \boldsymbol{A}_j \otimes \boldsymbol{B}_j$. Certain components are rescaled in dimensions for better presentation.

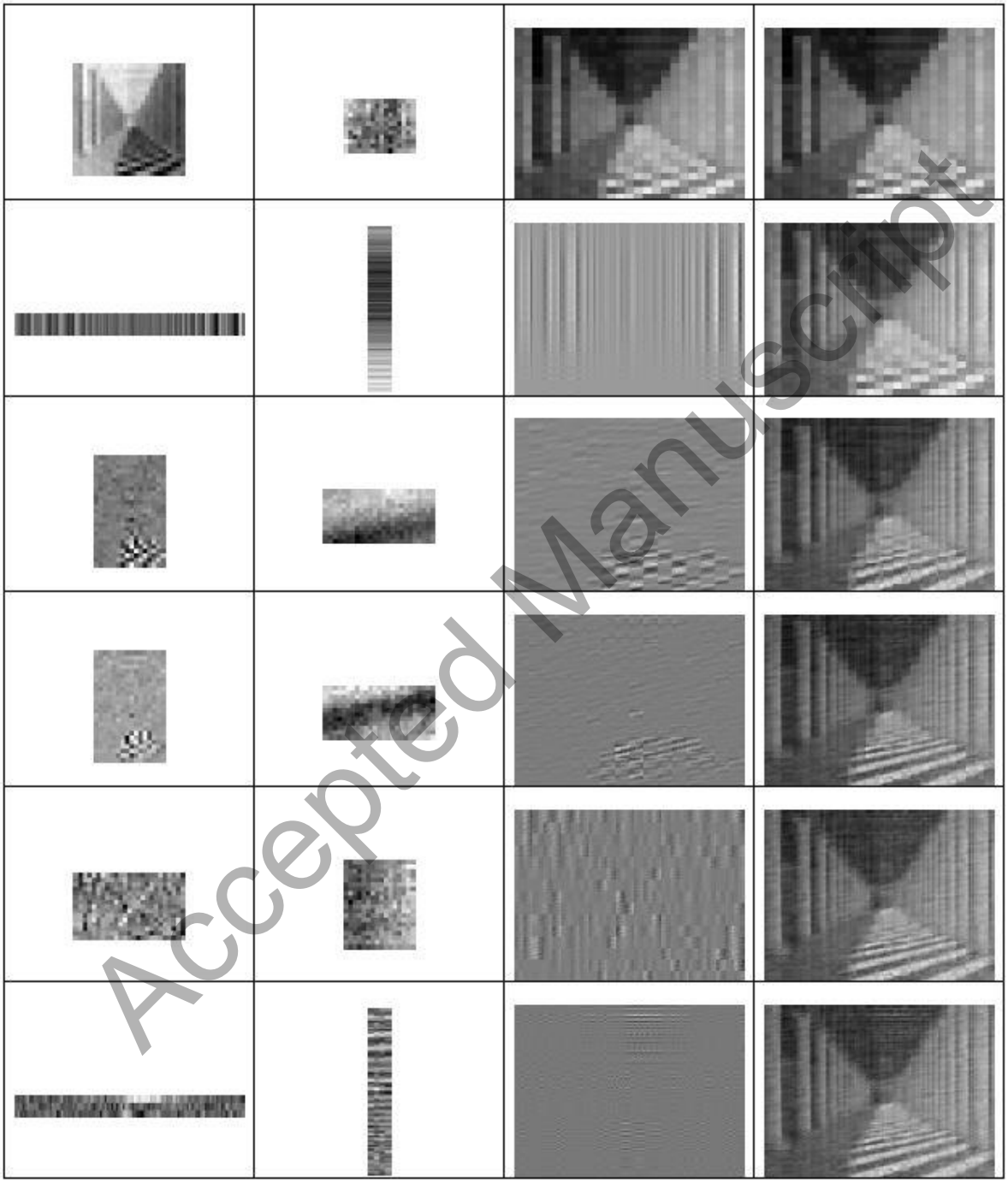

Fig. 8 The reconstructed image obtained from SVD (left), KoPA (middle), and *h*KoPA (right). Number of terms are selected to minimize the RCE.

Table 1 The selected configurations $(\hat{m}_t, \hat{n}_t)$ and the coefficients $\hat{\lambda}_t$ at each iteration for different values of α. The "A-2" and "A-$2'$" panels correspond to Algorithm 2 and Algorithm $2'$ respectively.

	t	$\alpha = 0.0$		$\alpha = 0.5$		$\alpha = 1.0$		$\alpha = 1.5$		$\alpha = 2.0$	
		$(\hat{m}, \hat{n})$	$\hat{\lambda}$	$(\hat{m}, \hat{n})$	$\hat{\lambda}$	$(\hat{m}, \hat{n})$	$\hat{\lambda}$	$(\hat{m}, \hat{n})$	$\hat{\lambda}$	$(\hat{m}, \hat{n})$	$\hat{\lambda}$
A-2	1	(4, 4)	1.003	(5, 5)	1.125	(5, 5)	1.251	(5, 5)	1.319	(5, 5)	1.354
	2	(5, 5)	1.002	(4, 4)	0.900	(4, 4)	0.713	(4, 4)	0.561	(4, 4)	0.455
	RCE	0.00475		0.00475		0.00475		0.00475		0.00476	
A-$2'$	1	(4, 4)	1.003	(5, 5)	1.113	(5, 5)	1.243	(5, 5)	1.314	(5, 5)	1.351
	2	(5, 5)	1.002	(4, 4)	0.860	(4, 4)	0.662	(4, 4)	0.515	(4, 4)	0.415
	3	-	-	(5, 5)	0.186	(5, 5)	0.176	(4, 5)	0.117	-	-
	4	-	-	-	-	-	-	(4, 5)	0.110	-	-
	RCE	0.00475		0.00737		0.00725		0.00982		0.01049	
	RCE (Post-ALS)	0.00475		0.00905		0.00891		0.01242		0.00476	

Table 2 The configurations, the cumulative percentage of variation (c.p.v.) explained, and the reconstruction error by the first 10 iterations for *h*KoPA, KoPA and SVD approaches. The smallest reconstruction error for each methods is highlighted.

k	*h*KoPA			KoPA			SVD		
	$(\hat{p}_k, \hat{q}_k)$	c.p.v.	RCE(%)	$(\hat{p}_k, \hat{q}_k)$	c.p.v.	RCE(%)	$(\hat{p}_k, \hat{q}_k)$	c.p.v.	RCE(%)
1	(25, 25)	73.66	5.21	(25, 25)	73.66	5.21	(300, 1)	70.82	8.73
2	(1, 400)	74.92	3.86	(25, 25)	74.76	4.20	(300, 1)	73.48	5.75
3	(25, 16)	75.72	3.23	(25, 25)	75.49	3.74	(300, 1)	74.42	4.88
4	(25, 16)	76.30	**2.90**	(25, 25)	76.10	3.42	(300, 1)	75.22	4.23
5	(15, 25)	76.67	2.91	(25, 25)	76.66	**3.15**	(300, 1)	75.84	3.80
6	(3, 100)	76.97	2.94	(25, 25)	77.03	3.19	(300, 1)	76.37	3.55
7	(25, 16)	77.28	3.06	(25, 25)	77.39	3.23	(300, 1)	76.78	**3.50**
8	(4, 80)	77.95	3.35	(25, 25)	77.72	3.34	(300, 1)	77.14	3.50
9	(15, 25)	78.20	3.65	(25, 25)	78.03	3.53	(300, 1)	77.44	3.71
10	(20, 16)	78.45	3.91	(25, 25)	78.32	3.38	(300, 1)	77.74	3.88

Table 3 Comparison of AIC and BIC.

Model	KoPA		*h*KoPA	
Criterion	AIC	BIC	AIC	BIC
Selected # terms	1	4	2	4
Selected # parameters	3782	3268	4482	2917
RCE (w/o bf)	3.75 %	3.42 %	2.92 %	**2.90**%
RCE (w/ bf)	3.75 %	3.42 %	2.83 %	**2.81**%
Optimal # terms	2	5	3	4
Optimal # parameters	7564	4085	6062	2917
Optimal RCE (w/o bf)	3.69%	3.15%	2.88%	2.90%
Optimal RCE (w/ bf)	3.69%	3.15%	2.90%	2.81%