Article

Not peer-reviewed version

The BiCG Algorithm for Solving the Minimal Frobenius Norm Solution of Generalized Sylvester Tensor Equation over the Quaternions

Mengyan Xie , Qing-Wen Wang * , Yang Zhang

Posted Date: 14 August 2024

doi: 10.20944/preprints202408.0929.v1

Keywords: Quaternion tensor; Sylvester tensor equation; Iterative algorithm; Color video restoration

doi:10.20944/preprints202408.0929.v1

Article

The BiCG Algorithm for Solving the Minimal Frobenius Norm Solution of Generalized Sylvester Tensor Equation over the Quaternions

Mengyan Xie [1,2], Qing-Wen Wang [2,3,*] and Yang Zhang [4]

1 College of Information Technology, Shanghai Ocean University, Shanghai 201306, P.R. China; myxie@shou.edu.cn
2 Department of Mathematics, Shanghai University, Shanghai 200444, P.R. China; wqw@t.shu.edu.cn
3 Collaborative Innovation Center for the Marine Artificial Intelligence, Shanghai 200444, China
4 Department of Mathematics, University of Manitoba, Winnipeg, MB, Canada; yang.zhang@umanitiba.ca
* Correspondence: wqw@t.shu.edu.cn

Abstract: In this paper, we develop an effective iterative algorithm to solve a generalized Sylvester tensor equation over quaternions which includes several well-studied matrix/tensor equations as special cases. We discuss the convergence of this algorithm within a finite number of iterations, assuming negligible round-off errors for any initial tensor. Moreover, we demonstrate the unique minimal Frobenius norm solution achievable by selecting specific types of initial tensors. Additionally, numerical examples are presented to illustrate the practicality and validity of our proposed algorithm. These examples include demonstrating the algorithm's effectiveness in addressing three-dimensional microscopic heat transport and color video restoration problems.

Keywords: quaternion tensor; sylvester tensor equation; iterative algorithm; color video restoration

---

1. Introduction

An order N tensor $\mathcal{A} = (a_{i_1 \ldots i_N})_{1 \leqslant i_j \leqslant I_j}$ $(j = 1, \ldots, N)$ over a field $\mathbb{F}$ is a multidimensional array with $I_1 I_2 \cdots I_N$ entries in $\mathbb{F}$, where N is a positive integer ([27,44]). The set of all such N tensors is denoted by $\mathbb{F}^{I_1 \times \cdots \times I_N}$. In the past decades, there have been active researches on tensors due to various applications in many areas such as physics, computer vision, data mining, etc. (see, e.g., [9,11,15–17,21,25–27,29–33,35–37,41–43,54,56]).

In this paper, we investigate the solvability of some tensor equations over quaternions. This is motivated by the recent researches on tensor equations as well as a long research history of solving matrix equations, which are briefly outlined as follows. It is well-known that the following Sylvester matrix equation

$$AX + YB = C \tag{1}$$

and its generalized forms have been widely investigated and found numerous applications in many areas.

During the past decades, many methods have been developed for solving Sylvester-type matrix equation over the quaternion algebra. For example, Kyrchei [28] gave explicit determinantal representation formulas of solutions to the Equation (1). Heyouni et al. [20] presented the SGl-CMRH method and preconditioned framework of this method to solve matrix Equation (1) when $X = Y$. Zhang [53] investigated the general system of generalized Sylvester quaternion matrix equations. Ahmadi-Asl and Beik [1,2,7] presented effective iterative algorithms for solving different quaternion matrix equations. Song [47] investigated the general solution to a system of quaternion matrix equation by Cramer's rule. Wang et al. [48] considered the solvability of a system of constrained two-sided coupled generalized Sylvester quaternion matrix equations. Zhang et al. [52] obtained some special least squares solutions of the quaternion matrix equation $AXB + CXD = E$. Huang et al. [22]

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 August 2024 doi:10.20944/preprints202408.0929.v1

considered the modified conjugate gradient method to solve the generalized coupled Sylvester conjugate matrix equations.

Quaternions offer greater versatility and flexibility compared to real and complex numbers, particularly in addressing multidimensional problems. This unique property has attracted growing interest among scholars, leading to numerous valuable achievements in quaternion-related research (see, e.g., [13,19,23,34,38,50,51]). The tensor equation is a natural extension of the matrix equation.

In this paper, we consider the following generalized Sylvester tensor equation over $\mathbb{H}$:

$$\mathcal{X} \times_1 A^{(1)} + \mathcal{X} \times_2 A^{(2)} + \cdots + \mathcal{X} \times_N A^{(N)} + \mathcal{Y} \times_1 B^{(1)} + \mathcal{Y} \times_2 B^{(2)} + \cdots + \mathcal{Y} \times_N B^{(N)} = \mathcal{C}, \quad (2)$$

where the tensors $A^{(n)}$, $B^{(n)} \in \mathbb{H}^{I_n \times I_n}$ $(n = 1, 2, \ldots, N)$, $\mathcal{C} \in \mathbb{H}^{I_1 \times \cdots \times I_N}$ are given, and the tensors $\mathcal{X}$, $\mathcal{Y} \in \mathbb{H}^{I_1 \times \cdots \times I_N}$ are unknown. The n-mode product of a tensor $\mathcal{X} \in \mathbb{H}^{I_1 \times \cdots \times I_N}$ with a matrix $A \in \mathbb{H}^{I_n \times I_n}$ is defined as

$$(\mathcal{X} \times_n A)_{i_1 \cdots i_{n-1} j i_{n+1} \cdots i_N} = \sum_{i_n=1}^{I_n} a_{j i_n} x_{i_1 \cdots i_{n-1} i_n i_{n+1} \cdots i_N}.$$

Note that the n-mode product can also be expressed in terms of unfolded quaternion tensors:

$$\mathcal{Y} = \mathcal{X} \times_n A \Longleftrightarrow \mathcal{Y}_{[n]} = A \mathcal{X}_{[n]},$$

where $\mathcal{X}_{[n]}$ is the mode-n unfolding of $\mathcal{X}$ ([27]). We consider the following problems corresponding to (2):

Problem 1.1: For given the tensor $\mathcal{C} \in \mathbb{H}^{I_1 \times I_2 \times \cdots \times I_N}$, the matrices $A^{(n)}, B^{(n)} \in \mathbb{H}^{I_n \times I_n} (n = 1, 2, \ldots, N)$, find the tensors $\widetilde{\mathcal{X}}, \widetilde{\mathcal{Y}} \in \mathbb{H}^{I_1 \times I_2 \times \cdots \times I_N}$ such that

$$\left\| \sum_{k=1}^{N} \widetilde{\mathcal{X}} \times_k A^{(k)} + \widetilde{\mathcal{Y}} \times_k B^{(k)} - \mathcal{C} \right\| = \min_{\mathcal{X}} \left\| \sum_{k=1}^{N} \mathcal{X} \times_k A^{(k)} + \mathcal{Y} \times_k B^{(k)} - \mathcal{C} \right\|.$$

Problem 1.2: Let the solution set of Problem 1.1 be denoted by S_{XY}. For a given tensor $\mathcal{X}_0, \mathcal{Y}_0 \in \mathbb{H}^{I_1 \times I_2 \times \cdots \times I_N}$, find the tensor $\check{\mathcal{X}}, \check{\mathcal{Y}} \in \mathbb{H}^{I_1 \times I_2 \times \cdots \times I_N}$ such that

$$\|\check{\mathcal{X}} - \mathcal{X}_0\| + \|\check{\mathcal{Y}} - \mathcal{Y}_0\| = \min_{\mathcal{X} \in S_L} \|\mathcal{X} - \mathcal{X}_0\| + \|\mathcal{Y} - \mathcal{Y}_0\|.$$

It is worth emphasizing that the tensor equation (2) includes several well-studied matrix/tensor equations as special cases. For example, if $\mathcal{X}$ and $\mathcal{Y}$ in (2) are order 2 tensors, i.e., matrices, then Equation (2) can be reduced to the following extended Sylvester matrix equation

$$A^{(1)} \mathcal{X} + \mathcal{X} (A^{(2)})^{\mathrm{T}} + B^{(1)} \mathcal{Y} + \mathcal{Y} (B^{(2)})^{\mathrm{T}} = \mathcal{C}.$$

In the case of $B^{(n)} = 0$ $(n = 1, 2 \cdots, N)$, Equation (2) becomes the following equation

$$\mathcal{X} \times_1 A^{(1)} + \mathcal{X} \times_2 A^{(2)} + \cdots + \mathcal{X} \times_N A^{(N)} = \mathcal{C}, \quad (3)$$

which has been studied extensively in recent years. For instance, Saberi et al. [45,46] investigated the SGMRES-BTF method and SGCRO-BTF method to solve Equation (3) over $\mathbb{R}$. Wang et al. [49] gave the conjugate gradient least squares method to solve Equation (3) over $\mathbb{H}$. Zhang and Wang [55] presented the tensor forms of the bi-conjugate gradient (BiCG-BTF) and bi-conjugate residual (BiCR-BTF) methods for solving tensor Equation (3) over real number field $\mathbb{R}$. Chen and Lu [10] considered a projection method with a Kronecker product preconditioner to solve Equation (3) over $\mathbb{R}$. Karimi and Dehghan [24] presented the tensor form of global least squares method for finding an approximate solution of (3). Furthermore, Najafi-Kalyani et al. [39] derived some iterative algorithms on global

Hessenberg process in their tensor forms to solve Equation (3). Considering Equation (3) over $\mathbb{R}$ and $N = 3$, that is,

$$\mathcal{X} \times_1 A^{(1)} + \mathcal{X} \times_2 A^{(2)} + \mathcal{X} \times_3 A^{(3)} = \mathcal{C}. \tag{4}$$

It has been shown that Equation (4) plays an important role in finite difference [3], thermal radiation [29], information retrieval [33], finite elements [14], and microscopic heat transport problem [37]. Therefore, our study on Equation (2) will provide a unified treatment for these matrix/tensor equations.

The rest of this paper is organized as follows. In Section 2, we recall some definitions and notations, and prove several lemmas for changing Equation (2). In Section 3, we develop the BiCG iterative algorithm for solving the quaternion tensor Equation (2), and prove that our algorithm is correct. We also show that the minimal Frobenious norm solution can be obtained by choosing special kinds of initial tensors. Some numerical examples are presented in Section 4 for illustrating the efficiency and applications of the proposed algorithm. Finally, we summarize our contributions in Section 5.

2. Preliminaries

We first recall some notations and definitions. For two complex matrices $U = (u_{ij}) \in \mathbb{C}^{m\times n}$, $V = (v_{ij}) \in \mathbb{C}^{p\times q}$, the symbol $U \otimes V = (u_{ij}V) \in \mathbb{C}^{mp\times nq}$ denotes the Kronecker product of U and V.

The operator $\text{vec}(\cdot)$ is defined as: for a matrix A and a tensor $\mathcal{X}$,

$$\text{vec}(A) = \left(a_1^{\text{T}}, a_2^{\text{T}}, \ldots, a_n^{\text{T}}\right)^{\text{T}}, \quad \text{and} \quad \text{vec}(\mathcal{X}) = \text{vec}(\mathcal{X}_{[1]}),$$

respectively, where a_k is the kth column of A and $\mathcal{X}_{[1]}$ is the mode-1 unfolding of the tensor $\mathcal{X}$. The inner product of two tensors $\mathcal{X}, \mathcal{Y} \in \mathbb{H}^{I_1\times\cdots\times I_N}$ is defined by

$$\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i_1=1}^{I_1}\sum_{i_2=1}^{I_2}\cdots\sum_{i_N=1}^{I_N} x_{i_1i_2\cdots i_N}\bar{y}_{i_1i_2\cdots i_N},$$

where $\bar{y}_{i_1i_2\cdots i_N}$ represents the quaternion conjugate of $y_{i_1i_2\cdots i_N}$. If $\langle \mathcal{X}, \mathcal{Y} \rangle = 0$, then we say that tensors $\mathcal{X}$ and $\mathcal{Y}$ are orthogonal. The Frobenius norm of tensor $\mathcal{X}$ is defined as $\|\mathcal{X}\| = \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle}$.

For any $\mathcal{X} \in \mathbb{H}^{I_1\times\cdots\times I_N}$, it is well-known that $\mathcal{X}$ can be uniquely expressed as $\mathcal{X} = \mathcal{X}_1 + \mathcal{X}_2\mathbf{i} + \mathcal{X}_3\mathbf{j} + \mathcal{X}_4\mathbf{k}$, where $\mathcal{X}_i \in \mathbb{R}^{I_1\times\cdots\times I_N}, i = 1,2,3,4$. Next we define n-mode operators for $\mathcal{X}_i$.

Let $A^{(n)} = A_1^{(n)} + A_2^{(n)}\mathbf{i} + A_3^{(n)}\mathbf{j} + A_4^{(n)}\mathbf{k}, B^{(n)} = B_1^{(n)} + B_2^{(n)}\mathbf{i} + B_3^{(n)}\mathbf{j} + B_4^{(n)}\mathbf{k} \in \mathbb{H}^{I_n\times I_n}$, where $A_i^{(n)}, B_i^{(n)} \in \mathbb{R}^{I_n\times I_n}, i = 1,2,3,4$. For $\mathcal{W} \in \mathbb{R}^{I_1\times\cdots\times I_N}$, we define

$$\begin{aligned}
\mathcal{L}_{A_i^{(n)}}(\mathcal{W}) &= \mathcal{W} \times_1 A_i^{(1)} + \mathcal{W} \times_2 A_i^{(2)} + \cdots + \mathcal{W} \times_N A_i^{(N)}, \quad i = 1,2,3,4,\\
\mathcal{L}_{B_i^{(n)}}(\mathcal{W}) &= \mathcal{W} \times_1 B_i^{(1)} + \mathcal{W} \times_2 B_i^{(2)} + \cdots + \mathcal{W} \times_N B_i^{(N)}, \quad i = 1,2,3,4.
\end{aligned}$$

Next, replacing $\mathcal{W}$ in above equations by $\mathcal{X}_i$'s, we define the following notations:

$$\begin{aligned}
\Gamma_1[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4] &= \mathcal{L}_{A_1^{(n)}}(\mathcal{X}_1) - \mathcal{L}_{A_2^{(n)}}(\mathcal{X}_2) - \mathcal{L}_{A_3^{(n)}}(\mathcal{X}_3) - \mathcal{L}_{A_4^{(n)}}(\mathcal{X}_4),\\
\Gamma_2[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4] &= \mathcal{L}_{A_2^{(n)}}(\mathcal{X}_1) + \mathcal{L}_{A_1^{(n)}}(\mathcal{X}_2) + \mathcal{L}_{A_4^{(n)}}(\mathcal{X}_3) - \mathcal{L}_{A_3^{(n)}}(\mathcal{X}_4),\\
\Gamma_3[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4] &= \mathcal{L}_{A_3^{(n)}}(\mathcal{X}_1) - \mathcal{L}_{A_4^{(n)}}(\mathcal{X}_2) + \mathcal{L}_{A_1^{(n)}}(\mathcal{X}_3) + \mathcal{L}_{A_2^{(n)}}(\mathcal{X}_4),\\
\Gamma_4[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4] &= \mathcal{L}_{A_4^{(n)}}(\mathcal{X}_1) + \mathcal{L}_{A_3^{(n)}}(\mathcal{X}_2) - \mathcal{L}_{A_2^{(n)}}(\mathcal{X}_3) + \mathcal{L}_{A_1^{(n)}}(\mathcal{X}_4),
\end{aligned} \tag{5}$$

$$\begin{aligned}
\Phi_1[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4] &= \mathcal{L}_{B_1^{(n)}}(\mathcal{X}_1) - \mathcal{L}_{B_2^{(n)}}(\mathcal{X}_2) - \mathcal{L}_{B_3^{(n)}}(\mathcal{X}_3) - \mathcal{L}_{B_4^{(n)}}(\mathcal{X}_4),\\
\Phi_2[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4] &= \mathcal{L}_{B_2^{(n)}}(\mathcal{X}_1) + \mathcal{L}_{B_1^{(n)}}(\mathcal{X}_2) + \mathcal{L}_{B_4^{(n)}}(\mathcal{X}_3) - \mathcal{L}_{B_3^{(n)}}(\mathcal{X}_4),\\
\Phi_3[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4] &= \mathcal{L}_{B_3^{(n)}}(\mathcal{X}_1) - \mathcal{L}_{B_4^{(n)}}(\mathcal{X}_2) + \mathcal{L}_{B_1^{(n)}}(\mathcal{X}_3) + \mathcal{L}_{B_2^{(n)}}(\mathcal{X}_4),\\
\Phi_4[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4] &= \mathcal{L}_{B_4^{(n)}}(\mathcal{X}_1) + \mathcal{L}_{B_3^{(n)}}(\mathcal{X}_2) - \mathcal{L}_{B_2^{(n)}}(\mathcal{X}_3) + \mathcal{L}_{B_1^{(n)}}(\mathcal{X}_4),
\end{aligned} \tag{6}$$

The following lemma shows that the quaternion tensor Equation (2) is equivalent to a system of four real tensor equations.

Lemma 1. *In Equation (2), we assume that* $A^{(n)} = A_1^{(n)} + A_2^{(n)}\mathbf{i} + A_3^{(n)}\mathbf{j} + A_4^{(n)}\mathbf{k}$, $B^{(n)} = B_1^{(n)} + B_2^{(n)}\mathbf{i} + B_3^{(n)}\mathbf{j} + B_4^{(n)}\mathbf{k} \in \mathbb{H}^{I_n\times I_n}, n = 1,2,\ldots,N$, *and* $\mathcal{C} = \mathcal{C}_1 + \mathcal{C}_2\mathbf{i} + \mathcal{C}_3\mathbf{j} + \mathcal{C}_4\mathbf{k}$, $\mathcal{X} = \mathcal{X}_1 + \mathcal{X}_2\mathbf{i} + \mathcal{X}_3\mathbf{j} + \mathcal{X}_4\mathbf{k}$, $\mathcal{Y} = \mathcal{Y}_1 + \mathcal{Y}_2\mathbf{i} + \mathcal{Y}_3\mathbf{j} + \mathcal{Y}_4\mathbf{k} \in \mathbb{H}^{I_1\times\cdots\times I_N}$. *Then the quaternion Sylvester tensor Equation (2) is equivalent to the following system of real tensor equations*

$$\begin{aligned}
\Gamma_1[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4] + \Phi_1[\mathcal{Y}_1,\mathcal{Y}_2,\mathcal{Y}_3,\mathcal{Y}_4] &= \mathcal{C}_1,\\
\Gamma_2[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4] + \Phi_2[\mathcal{Y}_1,\mathcal{Y}_2,\mathcal{Y}_3,\mathcal{Y}_4] &= \mathcal{C}_2,\\
\Gamma_3[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4] + \Phi_3[\mathcal{Y}_1,\mathcal{Y}_2,\mathcal{Y}_3,\mathcal{Y}_4] &= \mathcal{C}_3,\\
\Gamma_4[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4] + \Phi_4[\mathcal{Y}_1,\mathcal{Y}_2,\mathcal{Y}_3,\mathcal{Y}_4] &= \mathcal{C}_4,
\end{aligned} \tag{7}$$

where $\Gamma_i[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4]$, $\Phi_i[\mathcal{Y}_1,\mathcal{Y}_2,\mathcal{Y}_3,\mathcal{Y}_4] (i = 1,2,3,4)$ *are defined by (5) and (6). Furthermore, the system of real tensor Equations (7) is equivalent to the following linear system*

$$[\mathcal{M}_A, \mathcal{M}_B]z = c, \tag{8}$$

where

$$\mathcal{M}_A = \begin{pmatrix}
\mathrm{Kro}\left(\mathcal{L}_{A_1^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{A_2^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{A_3^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{A_4^{(n)}}\right)\\
\mathrm{Kro}\left(\mathcal{L}_{A_2^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}_{A_1^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}_{A_4^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{A_3^{(n)}}\right)\\
\mathrm{Kro}\left(\mathcal{L}_{A_3^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{A_4^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}_{A_1^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}_{A_2^{(n)}}\right)\\
\mathrm{Kro}\left(\mathcal{L}_{A_4^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}_{A_3^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{A_2^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}_{A_1^{(n)}}\right)
\end{pmatrix},$$

$$\mathcal{M}_B = \begin{pmatrix}
\mathrm{Kro}\left(\mathcal{L}_{B_1^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{B_2^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{B_3^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{B_4^{(n)}}\right)\\
\mathrm{Kro}\left(\mathcal{L}_{B_2^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}_{B_1^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}_{B_4^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{B_3^{(n)}}\right)\\
\mathrm{Kro}\left(\mathcal{L}_{B_3^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{B_4^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}_{B_1^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}_{B_2^{(n)}}\right)\\
\mathrm{Kro}\left(\mathcal{L}_{B_4^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}_{B_3^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{B_2^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}_{B_1^{(n)}}\right)
\end{pmatrix},$$

$$z = \begin{pmatrix}
\mathrm{vec}(\mathcal{X}_1)\\ \mathrm{vec}(\mathcal{X}_2)\\ \mathrm{vec}(\mathcal{X}_3)\\ \mathrm{vec}(\mathcal{X}_4)\\ \mathrm{vec}(\mathcal{Y}_1)\\ \mathrm{vec}(\mathcal{Y}_2)\\ \mathrm{vec}(\mathcal{Y}_3)\\ \mathrm{vec}(\mathcal{Y}_4)
\end{pmatrix}, \quad c = \begin{pmatrix}
\mathrm{vec}(\mathcal{C}_1)\\ \mathrm{vec}(\mathcal{C}_2)\\ \mathrm{vec}(\mathcal{C}_3)\\ \mathrm{vec}(\mathcal{C}_4)
\end{pmatrix},$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 August 2024 doi:10.20944/preprints202408.0929.v1

$$\mathrm{Kro}\left(\mathcal{L}_{A_i^{(n)}}\right) = \sum_{n=1}^{N} I^{(I_N)} \otimes \cdots \otimes I^{(I_{n+1})} \otimes A_i^{(n)} \otimes I^{(I_{n-1})} \otimes \cdots \otimes I^{(I_1)}, \quad i = 1,2,3,4,$$
$$\mathrm{Kro}\left(\mathcal{L}_{B_i^{(n)}}\right) = \sum_{n=1}^{N} I^{(I_N)} \otimes \cdots \otimes I^{(I_{n+1})} \otimes B_i^{(n)} \otimes I^{(I_{n-1})} \otimes \cdots \otimes I^{(I_1)}, \quad i = 1,2,3,4,$$

and $I^{(n)}$ stands for the identity matrix of order n.

Proof of Lemma 1. We apply the definition of n-mode product of the quaternion tensor for (2).

$$\begin{aligned}
&\sum_{n=1}^{N} \left(\mathcal{X} \times_n A^{(n)} + \mathcal{Y} \times_n B^{(n)}\right) \\
=&\sum_{n=1}^{N} \Big((\mathcal{X}_1 + \mathcal{X}_2\mathbf{i} + \mathcal{X}_3\mathbf{j} + \mathcal{X}_4\mathbf{k}) \times_n \left(A_1^{(n)} + A_2^{(n)}\mathbf{i} + A_3^{(n)}\mathbf{j} + A_4^{(n)}\mathbf{k}\right) \\
&+ (\mathcal{Y}_1 + \mathcal{Y}_2\mathbf{i} + \mathcal{Y}_3\mathbf{j} + \mathcal{Y}_4\mathbf{k}) \times_n \left(B_1^{(n)} + B_2^{(n)}\mathbf{i} + B_3^{(n)}\mathbf{j} + B_4^{(n)}\mathbf{k}\right)\Big) \\
=&\sum_{n=1}^{N} \Big(\mathcal{X}_1 \times_n A_1^{(n)} - \mathcal{X}_2 \times_n A_2^{(n)} - \mathcal{X}_3 \times_n A_3^{(n)} - \mathcal{X}_4 \times_n A_4^{(n)} \\
&+\mathcal{Y}_1 \times_n B_1^{(n)} - \mathcal{Y}_2 \times_n B_2^{(n)} - \mathcal{Y}_3 \times_n B_3^{(n)} - \mathcal{Y}_4 \times_n B_4^{(n)}\Big) \\
&+ \sum_{n=1}^{N} \Big(\mathcal{X}_1 \times_n A_2^{(n)} + \mathcal{X}_2 \times_n A_1^{(n)} + \mathcal{X}_3 \times_n A_4^{(n)} - \mathcal{X}_4 \times_n A_3^{(n)} \\
&+\mathcal{Y}_1 \times_n B_2^{(n)} + \mathcal{Y}_2 \times_n B_1^{(n)} + \mathcal{Y}_3 \times_n B_4^{(n)} - \mathcal{Y}_4 \times_n B_3^{(n)}\Big)\,\mathbf{i} \\
&+ \sum_{n=1}^{N} \Big(\mathcal{X}_1 \times_n A_3^{(n)} - \mathcal{X}_2 \times_n A_4^{(n)} + \mathcal{X}_3 \times_n A_1^{(n)} + \mathcal{X}_4 \times_n A_2^{(n)} \\
&+\mathcal{Y}_1 \times_n B_3^{(n)} - \mathcal{Y}_2 \times_n B_4^{(n)} + \mathcal{Y}_3 \times_n B_1^{(n)} + \mathcal{Y}_4 \times_n B_2^{(n)}\Big)\,\mathbf{j} \\
&+ \sum_{n=1}^{N} \Big(\mathcal{X}_1 \times_n A_4^{(n)} + \mathcal{X}_2 \times_n A_3^{(n)} - \mathcal{X}_3 \times_n A_2^{(n)} + \mathcal{X}_4 \times_n A_1^{(n)} \\
&+\mathcal{Y}_1 \times_n B_4^{(n)} + \mathcal{Y}_2 \times_n B_3^{(n)} - \mathcal{Y}_3 \times_n B_2^{(n)} + \mathcal{Y}_4 \times_n B_1^{(n)}\Big)\,\mathbf{k} \\
=&\mathcal{C}_1 + \mathcal{C}_2\mathbf{i} + \mathcal{C}_3\mathbf{j} + \mathcal{C}_4\mathbf{k}.
\end{aligned}$$

By the definitions of Γ_i and Φ_i, the Equations (7) hold. To show (8), we make use of operator "vec" to $\Gamma_1[\mathcal{X}_1, \mathcal{X}_2, \mathcal{X}_3, \mathcal{X}_4]$ and $\Phi_1[\mathcal{Y}_1, \mathcal{Y}_2, \mathcal{Y}_3, \mathcal{Y}_4]$, that is,

$$\begin{aligned}
&\mathrm{vec}\left(\Gamma_1[\mathcal{X}_1, \mathcal{X}_2, \mathcal{X}_3, \mathcal{X}_4]\right) \\
=&\mathrm{vec}\left(\mathcal{L}_{A_1^{(n)}}(\mathcal{X}_1) - \mathcal{L}_{A_2^{(n)}}(\mathcal{X}_2) - \mathcal{L}_{A_3^{(n)}}(\mathcal{X}_3) - \mathcal{L}_{A_4^{(n)}}(\mathcal{X}_4)\right) \\
=&\mathrm{vec}\left(\mathcal{L}_{A_1^{(n)}}(\mathcal{X}_1)\right) - \mathrm{vec}\left(\mathcal{L}_{A_2^{(n)}}(\mathcal{X}_2)\right) - \mathrm{vec}\left(\mathcal{L}_{A_3^{(n)}}(\mathcal{X}_3)\right) - \mathrm{vec}\left(\mathcal{L}_{A_4^{(n)}}(\mathcal{X}_4)\right) \\
=&\mathrm{Kro}\left(\mathcal{L}_{A_1^{(n)}}\right)\mathrm{vec}(\mathcal{X}_1) - \mathrm{Kro}\left(\mathcal{L}_{A_2^{(n)}}\right)\mathrm{vec}(\mathcal{X}_2) - \mathrm{Kro}\left(\mathcal{L}_{A_3^{(n)}}\right)\mathrm{vec}(\mathcal{X}_3) - \mathrm{Kro}\left(\mathcal{L}_{A_4^{(n)}}\right)\mathrm{vec}(\mathcal{X}_4),
\end{aligned}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 August 2024 doi:10.20944/preprints202408.0929.v1

$$
\begin{aligned}
&\mathrm{vec}\left(\Phi_1[\mathcal{Y}_1,\mathcal{Y}_2,\mathcal{Y}_3,\mathcal{Y}_4]\right)\\
=&\mathrm{vec}\left(\mathcal{L}_{B_1^{(n)}}(\mathcal{Y}_1)-\mathcal{L}_{B_2^{(n)}}(\mathcal{Y}_2)-\mathcal{L}_{B_3^{(n)}}(\mathcal{Y}_3)-\mathcal{L}_{B_4^{(n)}}(\mathcal{Y}_4)\right)\\
=&\mathrm{vec}\left(\mathcal{L}_{B_1^{(n)}}(\mathcal{Y}_1)\right)-\mathrm{vec}\left(\mathcal{L}_{B_2^{(n)}}(\mathcal{Y}_2)\right)-\mathrm{vec}\left(\mathcal{L}_{B_3^{(n)}}(\mathcal{Y}_3)\right)-\mathrm{vec}\left(\mathcal{L}_{B_4^{(n)}}(\mathcal{Y}_4)\right)\\
=&\mathrm{Kro}\left(\mathcal{L}_{B_1^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_1)-\mathrm{Kro}\left(\mathcal{L}_{B_2^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_2)-\mathrm{Kro}\left(\mathcal{L}_{B_3^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_3)-\mathrm{Kro}\left(\mathcal{L}_{B_4^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_4).
\end{aligned}
$$

Similarly, we have the following results for the rest of Γ_i's and Φ_i's:

$$
\begin{aligned}
&\mathrm{vec}\left(\Gamma_2[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4]\right)\\
=&\mathrm{Kro}\left(\mathcal{L}_{A_2^{(n)}}\right)\mathrm{vec}(\mathcal{X}_1)+\mathrm{Kro}\left(\mathcal{L}_{A_1^{(n)}}\right)\mathrm{vec}(\mathcal{X}_2)+\mathrm{Kro}\left(\mathcal{L}_{A_4^{(n)}}\right)\mathrm{vec}(\mathcal{X}_3)-\mathrm{Kro}\left(\mathcal{L}_{A_3^{(n)}}\right)\mathrm{vec}(\mathcal{X}_4),\\
&\mathrm{vec}\left(\Phi_2[\mathcal{Y}_1,\mathcal{Y}_2,\mathcal{Y}_3,\mathcal{Y}_4]\right)\\
=&\mathrm{Kro}\left(\mathcal{L}_{B_2^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_1)+\mathrm{Kro}\left(\mathcal{L}_{B_1^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_2)+\mathrm{Kro}\left(\mathcal{L}_{B_4^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_3)-\mathrm{Kro}\left(\mathcal{L}_{B_3^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_4),\\
&\mathrm{vec}\left(\Gamma_3[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4]\right)\\
=&\mathrm{Kro}\left(\mathcal{L}_{A_3^{(n)}}\right)\mathrm{vec}(\mathcal{X}_1)-\mathrm{Kro}\left(\mathcal{L}_{A_4^{(n)}}\right)\mathrm{vec}(\mathcal{X}_2)+\mathrm{Kro}\left(\mathcal{L}_{A_1^{(n)}}\right)\mathrm{vec}(\mathcal{X}_3)+\mathrm{Kro}\left(\mathcal{L}_{A_2^{(n)}}\right)\mathrm{vec}(\mathcal{X}_4),\\
&\mathrm{vec}\left(\Phi_3[\mathcal{Y}_1,\mathcal{Y}_2,\mathcal{Y}_3,\mathcal{Y}_4]\right)\\
=&\mathrm{Kro}\left(\mathcal{L}_{B_3^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_1)-\mathrm{Kro}\left(\mathcal{L}_{B_4^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_2)+\mathrm{Kro}\left(\mathcal{L}_{B_1^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_3)+\mathrm{Kro}\left(\mathcal{L}_{B_2^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_4),\\
&\mathrm{vec}\left(\Gamma_4[\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3,\mathcal{X}_4]\right)\\
=&\mathrm{Kro}\left(\mathcal{L}_{A_4^{(n)}}\right)\mathrm{vec}(\mathcal{X}_1)+\mathrm{Kro}\left(\mathcal{L}_{A_3^{(n)}}\right)\mathrm{vec}(\mathcal{X}_2)-\mathrm{Kro}\left(\mathcal{L}_{A_2^{(n)}}\right)\mathrm{vec}(\mathcal{X}_3)+\mathrm{Kro}\left(\mathcal{L}_{A_1^{(n)}}\right)\mathrm{vec}(\mathcal{X}_4),\\
&\mathrm{vec}\left(\Phi_4[\mathcal{Y}_1,\mathcal{Y}_2,\mathcal{Y}_3,\mathcal{Y}_4]\right)\\
=&\mathrm{Kro}\left(\mathcal{L}_{B_4^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_1)+\mathrm{Kro}\left(\mathcal{L}_{B_3^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_2)-\mathrm{Kro}\left(\mathcal{L}_{B_2^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_3)+\mathrm{Kro}\left(\mathcal{L}_{B_1^{(n)}}\right)\mathrm{vec}(\mathcal{Y}_4),
\end{aligned}
$$

By writing up above equations, we obtain the system (8). □

Lemma 2. *[21,40] Suppose that $A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m$ and the linear matrix equation $Ax = b$ has a solution $\tilde{x} \in \mathcal{R}\left(A^\top\right)$, then, $\tilde{x}$ is the unique minimum norm solution of $Ax = b$.*

From Lemma 1 and 2, it is easy to see that the uniqueness of the solution of equation (2) can be described as follows:

Theorem 1. *The tensor Equation (2) has the unique minimal Frobenius norm solution if and only if the matrix Equation (8) has a solution $\tilde{z} \in \mathcal{R}\left([\mathcal{M}_A, \mathcal{M}_B]^\top\right)$, then, $\tilde{z}$ is the unique minimum norm solution of matrix Equation (8).*

Given fixed matrices $A^{(n)} \in \mathbb{R}^{I_n\times I_n}, n = 1, 2, \ldots, N$, we define the following linear operators

$$\mathcal{L}_{A^{(n)}}(\mathcal{X}) = \mathcal{X}\times_1 A^{(1)} + \mathcal{X}\times_2 A^{(2)} + \cdots + \mathcal{X}\times_N A^{(N)},\quad \text{for any } \mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}.$$

Using the property $\left\langle \mathcal{X}, \mathcal{Y}\times_n A^{(n)}\right\rangle = \left\langle \mathcal{X}\times_n (A^{(n)})^{\mathrm{T}}, \mathcal{Y}\right\rangle$ in [26], it is easy to prove the following lemma.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 August 2024 doi:10.20944/preprints202408.0929.v1

Lemma 3. *Let* $A^{(n)} \in \mathbb{R}^{I_n \times I_n}, n = 1, 2, \ldots, N,\ \mathcal{X},\ \mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$. *Then*

$$\langle \mathcal{L}_{A^{(n)}}(\mathcal{X}), \mathcal{Y} \rangle = \left\langle \mathcal{X}, \mathcal{L}^*_{A^{(n)}}(\mathcal{Y}) \right\rangle,$$

where

$$\mathcal{L}^*_{A^{(n)}}(\mathcal{Y}) = \mathcal{Y} \times_1 \left(A^{(1)}\right)^{\mathrm{T}} + \mathcal{Y} \times_2 \left(A^{(2)}\right)^{\mathrm{T}} + \cdots + \mathcal{Y} \times_N \left(A^{(N)}\right)^{\mathrm{T}}.$$

Clearly, $\mathcal{L}_{A^{(n)}}$ defined above is a linear mapping. The following lemma provides the uniqueness of the dual mapping for these kinds of linear mappings.

Lemma 4. *([49]) Let* $\mathcal{N}$ *be a linear mapping from tensor space* $\mathbb{R}^{I_1 \times \cdots \times I_N}$ *to tensor space* $\mathbb{R}^{J_1 \times \cdots \times J_N}$. *For any tensors* $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ *and* $\mathcal{Y} \in \mathbb{R}^{J_1 \times \cdots \times J_N}$, *there exists a unique linear mapping* $\mathcal{M}$ *from tensor space* $\mathbb{R}^{J_1 \times \cdots \times J_N}$ *to tensor space* $\mathbb{R}^{I_1 \times \cdots \times I_N}$ *such that*

$$\langle \mathcal{N}(\mathcal{X}), \mathcal{Y} \rangle = \langle \mathcal{X}, \mathcal{M}(\mathcal{Y}) \rangle.$$

Finally, we use linear operators $\mathcal{L}$ and $\mathcal{L}^*$ to describe the inner products involving Γ_i and Φ_i, which we will use in next sections.

Lemma 5. *Let* $\Gamma_i[\mathcal{Z}_1, \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Z}_4]$, $\Phi_i[\mathcal{Z}_1, \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Z}_4]$ $(i = 1, 2, 3, 4)$ *be defined by (5) and (6),* $\mathcal{W}_i \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ $(i = 1, 2, 3, 4)$. *Then*

$$\begin{aligned}
\sum_{i=1}^{4} \langle \Gamma_i[\mathcal{Z}_1, \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Z}_4], \mathcal{W}_i \rangle &= \sum_{i=1}^{4} \langle \mathcal{Z}_i, \Gamma_i^*[\mathcal{W}_1, \mathcal{W}_2, \mathcal{W}_3, \mathcal{W}_4] \rangle, \\
\sum_{i=1}^{4} \langle \Phi_i[\mathcal{Z}_1, \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Z}_4], \mathcal{W}_i \rangle &= \sum_{i=1}^{4} \langle \mathcal{Z}_i, \Phi_i^*[\mathcal{W}_1, \mathcal{W}_2, \mathcal{W}_3, \mathcal{W}_4] \rangle,
\end{aligned} \tag{9}$$

where

$$\begin{aligned}
\Gamma_1^*[\mathcal{Z}_1, \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Z}_4] &= \mathcal{L}^*_{A_1^{(n)}}(\mathcal{Z}_1) + \mathcal{L}^*_{A_2^{(n)}}(\mathcal{Z}_2) + \mathcal{L}^*_{A_3^{(n)}}(\mathcal{Z}_3) + \mathcal{L}^*_{A_4^{(n)}}(\mathcal{Z}_4), \\
\Gamma_2^*[\mathcal{Z}_1, \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Z}_4] &= -\mathcal{L}^*_{A_2^{(n)}}(\mathcal{Z}_1) + \mathcal{L}^*_{A_1^{(n)}}(\mathcal{Z}_2) - \mathcal{L}^*_{A_4^{(n)}}(\mathcal{Z}_3) + \mathcal{L}^*_{A_3^{(n)}}(\mathcal{Z}_4), \\
\Gamma_3^*[\mathcal{Z}_1, \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Z}_4] &= -\mathcal{L}^*_{A_3^{(n)}}(\mathcal{Z}_1) + \mathcal{L}^*_{A_4^{(n)}}(\mathcal{Z}_2) + \mathcal{L}^*_{A_1^{(n)}}(\mathcal{Z}_3) - \mathcal{L}^*_{A_2^{(n)}}(\mathcal{Z}_4), \\
\Gamma_4^*[\mathcal{Z}_1, \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Z}_4] &= -\mathcal{L}^*_{A_4^{(n)}}(\mathcal{Z}_1) - \mathcal{L}^*_{A_3^{(n)}}(\mathcal{Z}_2) + \mathcal{L}^*_{A_2^{(n)}}(\mathcal{Z}_3) + \mathcal{L}^*_{A_1^{(n)}}(\mathcal{Z}_4),
\end{aligned} \tag{10}$$

$$\begin{aligned}
\Phi_1^*[\mathcal{Z}_1, \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Z}_4] &= \mathcal{L}^*_{B_1^{(n)}}(\mathcal{Z}_1) + \mathcal{L}^*_{B_2^{(n)}}(\mathcal{Z}_2) + \mathcal{L}^*_{B_3^{(n)}}(\mathcal{Z}_3) + \mathcal{L}^*_{B_4^{(n)}}(\mathcal{Z}_4), \\
\Phi_2^*[\mathcal{Z}_1, \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Z}_4] &= -\mathcal{L}^*_{B_2^{(n)}}(\mathcal{Z}_1) + \mathcal{L}^*_{B_1^{(n)}}(\mathcal{Z}_2) - \mathcal{L}^*_{B_4^{(n)}}(\mathcal{Z}_3) + \mathcal{L}^*_{B_3^{(n)}}(\mathcal{Z}_4), \\
\Phi_3^*[\mathcal{Z}_1, \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Z}_4] &= -\mathcal{L}^*_{B_3^{(n)}}(\mathcal{Z}_1) + \mathcal{L}^*_{B_4^{(n)}}(\mathcal{Z}_2) + \mathcal{L}^*_{B_1^{(n)}}(\mathcal{Z}_3) - \mathcal{L}^*_{B_2^{(n)}}(\mathcal{Z}_4), \\
\Phi_4^*[\mathcal{Z}_1, \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Z}_4] &= -\mathcal{L}^*_{B_4^{(n)}}(\mathcal{Z}_1) - \mathcal{L}^*_{B_3^{(n)}}(\mathcal{Z}_2) + \mathcal{L}^*_{B_2^{(n)}}(\mathcal{Z}_3) + \mathcal{L}^*_{B_1^{(n)}}(\mathcal{Z}_4),
\end{aligned} \tag{11}$$

$$\mathcal{L}^*_{A_i^{(n)}}(\mathcal{X}) = \mathcal{X} \times_1 (A_i^{(1)})^{\mathrm{T}} + \mathcal{X} \times_2 (A_i^{(2)})^{\mathrm{T}} + \cdots + \mathcal{X} \times_N (A_i^{(N)})^{\mathrm{T}}, \quad i = 1, 2, 3, 4,$$

$$\mathcal{L}^*_{B_i^{(n)}}(\mathcal{X}) = \mathcal{X} \times_1 (B_i^{(1)})^{\mathrm{T}} + \mathcal{X} \times_2 (B_i^{(2)})^{\mathrm{T}} + \cdots + \mathcal{X} \times_N (B_i^{(N)})^{\mathrm{T}}, \quad i = 1, 2, 3, 4.$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 August 2024 doi:10.20944/preprints202408.0929.v1

Proof of Lemma 5. For the first part of the equalities, we divide $\sum_{i=1}^{4}\langle\Gamma_i[\mathcal{Z}_1,\mathcal{Z}_2,\mathcal{Z}_3,\mathcal{Z}_4],\mathcal{W}_i\rangle$ into 4 parts by i, and then apply Lemma 3 to each part, that is,

$$
\begin{aligned}
&\langle\Gamma_1[\mathcal{Z}_1,\mathcal{Z}_2,\mathcal{Z}_3,\mathcal{Z}_4],\mathcal{W}_1\rangle\\
&=\langle\mathcal{L}_{A_1^{(n)}}(\mathcal{Z}_1),\mathcal{W}_1\rangle-\langle\mathcal{L}_{A_2^{(n)}}(\mathcal{Z}_2),\mathcal{W}_1\rangle-\langle\mathcal{L}_{A_3^{(n)}}(\mathcal{Z}_3),\mathcal{W}_1\rangle-\langle\mathcal{L}_{A_4^{(n)}}(\mathcal{Z}_4),\mathcal{W}_1\rangle\\
&=\langle\mathcal{Z}_1,\mathcal{L}^*_{A_1^{(n)}}(\mathcal{W}_1)\rangle-\langle\mathcal{Z}_2,\mathcal{L}^*_{A_2^{(n)}}(\mathcal{W}_1)\rangle-\langle\mathcal{Z}_3,\mathcal{L}^*_{A_3^{(n)}}(\mathcal{W}_1)\rangle-\langle\mathcal{Z}_4,\mathcal{L}^*_{A_4^{(n)}}(\mathcal{W}_1)\rangle,
\end{aligned}
$$

$$
\begin{aligned}
&\langle\Gamma_2[\mathcal{Z}_1,\mathcal{Z}_2,\mathcal{Z}_3,\mathcal{Z}_4],\mathcal{W}_2\rangle\\
&=\langle\mathcal{L}_{A_2^{(n)}}(\mathcal{Z}_1),\mathcal{W}_2\rangle+\langle\mathcal{L}_{A_1^{(n)}}(\mathcal{Z}_2),\mathcal{W}_2\rangle+\langle\mathcal{L}_{A_4^{(n)}}(\mathcal{Z}_3),\mathcal{W}_2\rangle-\langle\mathcal{L}_{A_3^{(n)}}(\mathcal{Z}_4),\mathcal{W}_2\rangle\\
&=\langle\mathcal{Z}_1,\mathcal{L}^*_{A_2^{(n)}}(\mathcal{W}_2)\rangle+\langle\mathcal{Z}_2,\mathcal{L}^*_{A_1^{(n)}}(\mathcal{W}_1)\rangle+\langle\mathcal{Z}_3,\mathcal{L}^*_{A_3^{(n)}}(\mathcal{W}_2)\rangle-\langle\mathcal{Z}_4,\mathcal{L}^*_{A_4^{(n)}}(\mathcal{W}_2)\rangle,
\end{aligned}
$$

$$
\begin{aligned}
&\langle\Gamma_3[\mathcal{Z}_1,\mathcal{Z}_2,\mathcal{Z}_3,\mathcal{Z}_4],\mathcal{W}_3\rangle\\
&=\langle\mathcal{L}_{A_3^{(n)}}(\mathcal{Z}_1),\mathcal{W}_3\rangle-\langle\mathcal{L}_{A_4^{(n)}}(\mathcal{Z}_2),\mathcal{W}_3\rangle+\langle\mathcal{L}_{A_1^{(n)}}(\mathcal{Z}_3),\mathcal{W}_3\rangle+\langle\mathcal{L}_{A_2^{(n)}}(\mathcal{Z}_4),\mathcal{W}_3\rangle\\
&=\langle\mathcal{Z}_1,\mathcal{L}^*_{A_3^{(n)}}(\mathcal{W}_3)\rangle-\langle\mathcal{Z}_2,\mathcal{L}^*_{A_4^{(n)}}(\mathcal{W}_3)\rangle+\langle\mathcal{Z}_3,\mathcal{L}^*_{A_1^{(n)}}(\mathcal{W}_2)\rangle+\langle\mathcal{Z}_4,\mathcal{L}^*_{A_2^{(n)}}(\mathcal{W}_3)\rangle,
\end{aligned}
$$

$$
\begin{aligned}
&\langle\Gamma_4[\mathcal{Z}_1,\mathcal{Z}_2,\mathcal{Z}_3,\mathcal{Z}_4],\mathcal{W}_4\rangle\\
&=\langle\mathcal{L}_{A_4^{(n)}}(\mathcal{Z}_1),\mathcal{W}_4\rangle+\langle\mathcal{L}_{A_3^{(n)}}(\mathcal{Z}_2),\mathcal{W}_4\rangle-\langle\mathcal{L}_{A_2^{(n)}}(\mathcal{Z}_3),\mathcal{W}_4\rangle+\langle\mathcal{L}_{A_1^{(n)}}(\mathcal{Z}_4),\mathcal{W}_4\rangle\\
&=\langle\mathcal{Z}_1,\mathcal{L}^*_{A_4^{(n)}}(\mathcal{W}_4)\rangle+\langle\mathcal{Z}_2,\mathcal{L}^*_{A_3^{(n)}}(\mathcal{W}_4)\rangle-\langle\mathcal{Z}_3,\mathcal{L}^*_{A_2^{(n)}}(\mathcal{W}_4)\rangle+\langle\mathcal{Z}_4,\mathcal{L}^*_{A_1^{(n)}}(\mathcal{W}_4)\rangle.
\end{aligned}
$$

By adding up the above four parts, we have

$$\sum_{i=1}^{4}\langle\Gamma_i[\mathcal{Z}_1,\mathcal{Z}_2,\mathcal{Z}_3,\mathcal{Z}_4],\mathcal{W}_i\rangle=\sum_{i=1}^{4}\langle\mathcal{Z}_i,\Gamma_i^*[\mathcal{W}_1,\mathcal{W}_2,\mathcal{W}_3,\mathcal{W}_4]\rangle,$$

where $\Gamma_i^*[\mathcal{W}_1,\mathcal{W}_2,\mathcal{W}_3,\mathcal{W}_4]$'s are defined by (10). Using a similar process, we can get the second equality

$$\sum_{i=1}^{4}\langle\Phi_i[\mathcal{Z}_1,\mathcal{Z}_2,\mathcal{Z}_3,\mathcal{Z}_4],\mathcal{W}_i\rangle=\sum_{i=1}^{4}\langle\mathcal{Z}_i,\Phi_i^*[\mathcal{W}_1,\mathcal{W}_2,\mathcal{W}_3,\mathcal{W}_4]\rangle,$$

where $\Phi_i^*[\mathcal{W}_1,\mathcal{W}_2,\mathcal{W}_3,\mathcal{W}_4]$'s are defined by (11). □

3. An Iterative Algorithm For Solving The Problem 1.1 And 1.2

The purpose of this section is to propose an iterative algorithm for obtaining the solution of Sylvester tensor Equation (2). As well known that the classical bi-conjugate gradient (BiCG) methods for solving nonsymmetric linear system of equations are feasible and efficient, one may refer to [5,6,12,18,55]. We extend the BiCG method based on tensor format (BTF) for solving Equation (2) and discuss its convergence. Clearly, the tensor Equation (2) and (8) have the same solution from Lemma 1. However, the size of $[\mathcal{M}_A,\mathcal{M}_B]$ in Equation (8) is usually too large to save computation time and memory space. Beik et al. [8] showed that the algorithms based on tensor format are more efficient than their classical forms in general. Inspired by these issues, we develop the following least squares algorithm based on tensor format for solving tensor Equation (2):

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 August 2024 doi:10.20944/preprints202408.0929.v1

Algorithm 1. *BiCG-BTF method for solving equation (2).*

Input: $A_i^{(n)}, B_i^{(n)} \in \mathbb{R}^{I_n \times I_n}, \mathcal{C}_i \in \mathbb{R}^{I_1 \times \cdots \times I_N}, n = 1, 2, \ldots, N; i = 1, 2, 3, 4.$

Output: *The norm* $\sum_{i=1}^4 \|\mathcal{R}_i(\cdot)\|$ *and the solutions* $\mathcal{X}_i(\cdot), \mathcal{Y}_i(\cdot), i = 1, 2, 3, 4.$

Initialization: $\mathcal{X}_i(1), \mathcal{Y}_i(1) \in \mathbb{R}^{I_1 \times \cdots \times I_N}, i = 1, 2, 3, 4;$

(i) *Compute*

$\mathcal{R}_i(1) := \mathcal{C}_i - \Gamma_i[\mathcal{X}_1(1), \mathcal{X}_2(1), \mathcal{X}_3(1), \mathcal{X}_4(1)] - \Phi_i[\mathcal{Y}_1(1), \mathcal{Y}_2(1), \mathcal{Y}_3(1), \mathcal{Y}_4(1)], \ i = 1, 2, 3, 4;$

Set $\mathcal{R}_i^*(1) := \mathcal{R}_i(1); \mathcal{P}_i(1) := \mathcal{R}_i^*(1); \mathcal{P}_i^*(1) := \mathcal{P}_i(1);$

Compute the norm: $Rnorm := \sum_{i=1}^4 \|\mathcal{R}_i(1)\|, i = 1, 2, 3, 4;$

Set $k := 1;$

(ii) *If* $Rnorm = 0$, *then stop;*

(iii) *Otherwise, compute*

$\mathcal{Q}_{ix}(k) := \Gamma_i[\mathcal{P}_1(k), \mathcal{P}_2(k), \mathcal{P}_3(k), \mathcal{P}_4(k)], \ i = 1, 2, 3, 4;$

$\mathcal{Q}_{iy}(k) := \Phi_i[\mathcal{P}_1(k), \mathcal{P}_2(k), \mathcal{P}_3(k), \mathcal{P}_4(k)], \ i = 1, 2, 3, 4;$

$\alpha(k) := \left(\sum_{i=1}^4 \langle \mathcal{R}_i(k), \mathcal{R}_i^*(k)\rangle\right) / \left(\sum_{i=1}^4 \langle \mathcal{P}_i^*(k), \mathcal{Q}_{ix}(k) + \mathcal{Q}_{iy}(k)\rangle\right);$

$\mathcal{X}_i(k+1) := \mathcal{X}_i(k) + \alpha(k)\mathcal{P}_i(k), \ i = 1, 2, 3, 4;$

$\mathcal{Y}_i(k+1) := \mathcal{Y}_i(k) + \alpha(k)\mathcal{P}_i(k), \ i = 1, 2, 3, 4;$

$\mathcal{R}_i(k+1) := \mathcal{R}_i(k) - \alpha(k)(\mathcal{Q}_{ix}(k) + \mathcal{Q}_{iy}(k)), \ i = 1, 2, 3, 4;$

$\mathcal{Q}_{ix}^*(k) := \Gamma_i^*[\mathcal{P}_1^*(k), \mathcal{P}_2^*(k), \mathcal{P}_3^*(k), \mathcal{P}_4^*(k)], \ i = 1, 2, 3, 4;$

$\mathcal{Q}_{iy}^*(k) := \Phi_i^*[\mathcal{P}_1^*(k), \mathcal{P}_2^*(k), \mathcal{P}_3^*(k), \mathcal{P}_4^*(k)], \ i = 1, 2, 3, 4;$

$\mathcal{R}_i^*(k+1) := \mathcal{R}_i^*(k) - \alpha(k)(\mathcal{Q}_{ix}^*(k) + \mathcal{Q}_{iy}^*(k)), \ i = 1, 2, 3, 4;$

$\beta(k) = \left(\sum_{i=1}^4 \langle \mathcal{R}_i(k+1), \mathcal{R}_i^*(k+1)\rangle\right) / \left(\sum_{i=1}^4 \langle \mathcal{R}_i(k), \mathcal{R}_i^*(k)\rangle\right);$

$\mathcal{P}_i(k+1) := \mathcal{R}_i(k+1) + \beta(k)\mathcal{P}_i(k), \ i = 1, 2, 3, 4;$

$\mathcal{P}_i^*(k+1) := \mathcal{R}_i^*(k+1) + \beta(k)\mathcal{P}_i^*(k), \ i = 1, 2, 3, 4;$

$Rnorm = \sum_{i=1}^4 \|\mathcal{R}_i(k)\|, i = 1, 2, 3, 4;$

(iv) *Set* $k := k+1$, *go to (ii);*

Note that Γ_i, Φ_i and Γ_i^*, Φ_i^* are defined by (5), (6) and (10), (11), respectively. Next, we discuss some bi-orthogonality properties of Algorithm 1.

Theorem 2. *Assume that iterative sequences* $\{\mathcal{R}_i(k)\}$, $\{\mathcal{R}_i^*(k)\}$, $\{\mathcal{P}_i(k)\}$, $\{\mathcal{P}_i^*(k)\}$ $\{\mathcal{Q}_{ix}(k)\}$ *and* $\{\mathcal{Q}_{iy}(k)\}$ $(i = 1, 2, 3, 4)$ *are generated by Algorithm* **??**. *Then we have*

$$\sum_{i=1}^4 \langle \mathcal{R}_i(l), \mathcal{R}_i^*(m)\rangle = 0, \quad l \neq m, \tag{12}$$

$$\sum_{i=1}^4 \langle \mathcal{Q}_{ix}(l) + \mathcal{Q}_{iy}(l), \mathcal{P}_i^*(m)\rangle = 0, \quad l \neq m, \tag{13}$$

$$\sum_{i=1}^4 \langle \mathcal{R}_i(l), \mathcal{P}_i^*(m)\rangle = 0, \quad l > m. \tag{14}$$

Proof of Theorem 2. We apply mathematics induction on k. Let's consider $1 \leq m < l \leq k$ first.

 doi:10.20944/preprints202408.0929.v1

When $k = 2$. The conclusion holds as the following calculations show:

$$\begin{aligned}
&\sum_{i=1}^{4} \langle \mathcal{R}_i(2), \mathcal{R}_i^*(1) \rangle \\
&= \sum_{i=1}^{4} \left\langle \mathcal{R}_i(1) - \alpha_1 \left(\mathcal{Q}_{ix}(1) + \mathcal{Q}_{iy}(1) \right), \mathcal{R}_i^*(1) \right\rangle \\
&= \sum_{i=1}^{4} \langle \mathcal{R}_i(1), \mathcal{R}_i^*(1) \rangle - \frac{\sum_{i=1}^{4} \langle \mathcal{R}_i(1), \mathcal{R}_i^*(1) \rangle}{\sum_{i=1}^{4} \langle \mathcal{P}_i^*(1), \mathcal{Q}_{ix}(1) + \mathcal{Q}_{iy}(1) \rangle} \sum_{i=1}^{4} \langle \mathcal{P}_i^*(1), \mathcal{Q}_{ix}(1) + \mathcal{Q}_{iy}(1) \rangle \\
&= 0,
\end{aligned}$$

and

$$\begin{aligned}
&\sum_{i=1}^{4} \langle \mathcal{Q}_{ix}(2) + \mathcal{Q}_{iy}(2), \mathcal{P}_i^*(1) \rangle \\
&= \sum_{i=1}^{4} \langle \Gamma_i[\mathcal{P}_1(2), \mathcal{P}_2(2), \mathcal{P}_3(2), \mathcal{P}_4(2)], \mathcal{P}_i^*(1) \rangle + \sum_{i=1}^{4} \langle \Phi_i[\mathcal{P}_1(2), \mathcal{P}_2(2), \mathcal{P}_3(2), \mathcal{P}_4(2)], \mathcal{P}_i^*(1) \rangle \\
&= \sum_{i=1}^{4} \langle \mathcal{P}_i(2), \Gamma_i^*[\mathcal{P}_1^*(1), \mathcal{P}_2^*(1), \mathcal{P}_3^*(1), \mathcal{P}_4^*(1)] \rangle + \sum_{i=1}^{4} \langle \mathcal{P}_i(2), \Phi_i^*[\mathcal{P}_1^*(1), \mathcal{P}_2^*(1), \mathcal{P}_3^*(1), \mathcal{P}_4^*(1)] \rangle \\
&= \sum_{i=1}^{4} [\langle \mathcal{R}_i(2), \mathcal{Q}_{ix}^*(1) \rangle + \beta(1) \langle \mathcal{P}_i(1), \mathcal{Q}_{ix}^*(1) \rangle] + \sum_{i=1}^{4} \left[\langle \mathcal{R}_i(2), \mathcal{Q}_{iy}^*(1) \rangle + \beta(1) \langle \mathcal{P}_i(1), \mathcal{Q}_{iy}^*(1) \rangle \right] \\
&= \sum_{i=1}^{4} \left[\langle \mathcal{R}_i(2), \mathcal{Q}_{ix}^*(1) \rangle + \frac{\sum_{i=1}^{4} \langle \mathcal{R}_i(2), \mathcal{R}_i^*(1) - \alpha(1)(\mathcal{Q}_{ix}^*(1) + \mathcal{Q}_{iy}^*(1)) \rangle}{\sum_{i=1}^{4} \langle \mathcal{R}_i(1), \mathcal{R}_i^*(1) \rangle} \langle \mathcal{P}_i(1), \mathcal{Q}_{ix}^*(1) \rangle \right] \\
&\quad + \sum_{i=1}^{4} \left[\langle \mathcal{R}_i(2), \mathcal{Q}_{iy}^*(1) \rangle + \frac{\sum_{i=1}^{4} \langle \mathcal{R}_i(2), \mathcal{R}_i^*(1) - \alpha(1)(\mathcal{Q}_{ix}^*(1) + \mathcal{Q}_{iy}^*(1)) \rangle}{\sum_{i=1}^{4} \langle \mathcal{R}_i(1), \mathcal{R}_i^*(1) \rangle} \langle \mathcal{P}_i(1), \mathcal{Q}_{iy}^*(1) \rangle \right] \\
&= \sum_{i=1}^{4} \left[\langle \mathcal{R}_i(2), \mathcal{Q}_{ix}^*(1) \rangle - \frac{\sum_{i=1}^{4} \langle \mathcal{R}_i(2), \mathcal{Q}_{ix}^*(1) + \mathcal{Q}_{iy}^*(1) \rangle}{\sum_{i=1}^{4} \langle \mathcal{P}_i^*(1), \mathcal{Q}_{ix}(1) + \mathcal{Q}_{iy}(1) \rangle} \langle \mathcal{P}_i(1), \mathcal{Q}_{ix}^*(1) \rangle \right] \\
&\quad + \sum_{i=1}^{4} \left[\langle \mathcal{R}_i(2), \mathcal{Q}_{iy}^*(1) \rangle - \frac{\sum_{i=1}^{4} \langle \mathcal{R}_i(2), \mathcal{Q}_{ix}^*(1) + \mathcal{Q}_{iy}^*(1) \rangle}{\sum_{i=1}^{4} \langle \mathcal{P}_i^*(1), \mathcal{Q}_{ix}(1) + \mathcal{Q}_{iy}(1) \rangle} \langle \mathcal{P}_i(1), \mathcal{Q}_{iy}^*(1) \rangle \right] \\
&= \sum_{i=1}^{4} \langle \mathcal{R}_i(2), \mathcal{Q}_{ix}^*(1) + \mathcal{Q}_{iy}^*(1) \rangle - \frac{\sum_{i=1}^{4} \langle \mathcal{R}_i(2), \mathcal{Q}_{ix}^*(1) + \mathcal{Q}_{iy}^*(1) \rangle}{\sum_{i=1}^{4} \langle \mathcal{P}_i^*(1), \mathcal{Q}_{ix}(1) + \mathcal{Q}_{iy}(1) \rangle} \sum_{i=1}^{4} \langle \mathcal{P}_i^*(1), \mathcal{Q}_{ix}(1) + \mathcal{Q}_{iy}(1) \rangle \\
&= 0.
\end{aligned}$$

It has been found $\sum_{i=1}^{4} \langle \mathcal{R}_i(2), \mathcal{P}_i^*(1) \rangle = 0$ clearly. Now, assume that (12) and (13) hold for $1 \leq m < l \leq k$ ($k > 2$). Then

$$\begin{aligned}
\sum_{i=1}^{4} \langle \mathcal{R}_i(k+1), \mathcal{P}_i^*(m) \rangle &= \sum_{i=1}^{4} \left\langle \mathcal{R}_i(k) - \alpha(k) \left(\mathcal{Q}_{ix}(k) + \mathcal{Q}_{iy}(k) \right), \mathcal{P}_i^*(m) \right\rangle \\
&= \sum_{i=1}^{4} \left[\langle \mathcal{R}_i(k), \mathcal{P}_i^*(m) \rangle - \alpha(k) \langle \mathcal{Q}_{ix}(k) + \mathcal{Q}_{iy}(k), \mathcal{P}_i^*(m) \rangle \right] \\
&= 0,
\end{aligned}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 August 2024 doi:10.20944/preprints202408.0929.v1

and

$$\begin{aligned}\sum_{i=1}^{4}\langle\mathcal{R}_i(k+1),\mathcal{P}_i^*(k)\rangle &= \sum_{i=1}^{4}\langle\mathcal{R}_i(k)-\alpha(k)\left(\mathcal{Q}_{ix}(k)+\mathcal{Q}_{iy}(k)\right),\mathcal{P}_i^*(k)\rangle \\ &= \sum_{i=1}^{4}\left[\langle\mathcal{R}_i(k),\mathcal{P}_i^*(k)\rangle-\alpha(k)\langle\mathcal{Q}_{ix}(k)+\mathcal{Q}_{iy}(k),\mathcal{P}_i^*(k)\rangle\right] \\ &= \sum_{i=1}^{4}[\mathcal{R}_i(k),\mathcal{R}_i^*(k)+\beta(k-1)\langle\mathcal{R}_i(k),\mathcal{P}_i^*(k-1)\rangle] \\ &\quad -\frac{\sum_{i=1}^{4}\langle\mathcal{R}_i(k),\mathcal{R}_i^*(k)\rangle}{\sum_{i=1}^{4}\langle\mathcal{P}_i^*(k),\mathcal{Q}_{ix}(k)+\mathcal{Q}_{iy}(k)\rangle}\sum_{i=1}^{4}\langle\mathcal{P}_i^*(k),\mathcal{Q}_{ix}(k)+\mathcal{Q}_{iy}(k)\rangle \\ &= 0.\end{aligned}$$

The equality (14) holds for all $l > m$. Next, we will prove that the equalities (12) and (13) hold for all $l > m$.

$$\begin{aligned}\sum_{i=1}^{4}\langle\mathcal{R}_i(k+1),\mathcal{R}_i^*(m)\rangle &= \sum_{i=1}^{4}\langle\mathcal{R}_i(k+1),\mathcal{P}_i^*(m)-\beta(m-1)\mathcal{P}_i^*(m-1)\rangle \\ &= \sum_{i=1}^{4}[\langle\mathcal{R}_i(k+1),\mathcal{P}_i^*(m)\rangle-\beta(m-1)\langle\mathcal{R}_i(k+1),\mathcal{P}_i^*(m-1)\rangle] \\ &= 0,\end{aligned}$$

$$\begin{aligned}\sum_{i=1}^{4}\langle\mathcal{R}_i(k+1),\mathcal{R}_i^*(k)\rangle &= \sum_{i=1}^{4}\langle\mathcal{R}_i(k+1),\mathcal{P}_i^*(m)-\beta(m-1)\mathcal{P}_i^*(m-1)\rangle \\ &= \sum_{i=1}^{4}[\langle\mathcal{R}_i(k+1),\mathcal{P}_i^*(m)\rangle-\beta(m-1)\langle\mathcal{R}_i(k+1),\mathcal{P}_i^*(m-1)\rangle] \\ &= 0,\end{aligned}$$

and

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 August 2024 doi:10.20944/preprints202408.0929.v1

$$
\begin{aligned}
&\sum_{i=1}^{4}\langle \mathcal{Q}_{ix}(k+1)+\mathcal{Q}_{iy}(k+1),\mathcal{P}_i^*(m)\rangle \\
&=\sum_{i=1}^{4}\langle \mathcal{Q}_{ix}(k+1),\mathcal{P}_i^*(m)\rangle+\sum_{i=1}^{4}\langle \mathcal{Q}_{iy}(k+1),\mathcal{P}_i^*(m)\rangle \\
&=\sum_{i=1}^{4}[\langle \mathcal{R}_i(k+1),\Gamma_i^*[\mathcal{P}_1^*(m),\mathcal{P}_2^*(m),\mathcal{P}_3^*(m),\mathcal{P}_4^*(m)]\rangle \\
&\quad+\beta(k)\langle \mathcal{P}_i(k),\Gamma_i^*[\mathcal{P}_1^*(m),\mathcal{P}_2^*(m),\mathcal{P}_3^*(m),\mathcal{P}_4^*(m)]\rangle] \\
&\quad+\sum_{i=1}^{4}[\langle \mathcal{R}_i(k+1),\Phi_i^*[\mathcal{P}_1^*(m),\mathcal{P}_2^*(m),\mathcal{P}_3^*(m),\mathcal{P}_4^*(m)]\rangle \\
&\quad+\beta(k)\langle \mathcal{P}_i(k),\Phi_i^*[\mathcal{P}_1^*(m),\mathcal{P}_2^*(m),\mathcal{P}_3^*(m),\mathcal{P}_4^*(m)]\rangle] \\
&=\sum_{i=1}^{4}[\langle \mathcal{R}_i(k+1),\mathcal{Q}_{ix}^*(m)\rangle+\beta(k)\langle \mathcal{P}_i(k),\mathcal{Q}_{ix}^*(m)\rangle \\
&\quad+\langle \mathcal{R}_i(k+1),\mathcal{Q}_{iy}^*(m)\rangle+\beta(k)\langle \mathcal{P}_i(k),\mathcal{Q}_{iy}^*(m)\rangle\Big] \\
&=\sum_{i=1}^{4}\left[\left\langle \mathcal{R}_i(k+1),-\frac{1}{\alpha(m)}(\mathcal{R}_i^*(m+1)-\mathcal{R}_i(m))\right\rangle\right. \\
&\quad\left.+\beta(k)\left\langle \mathcal{P}_i(k),-\frac{1}{\alpha(m)}(\mathcal{R}_i^*(m+1)-\mathcal{R}_i(m))\right\rangle\right] \\
&=0,
\end{aligned}
$$

$$
\begin{aligned}
&\sum_{i=1}^{4}\langle \mathcal{Q}_{ix}(k+1)+\mathcal{Q}_{iy}(k+1),\mathcal{P}_i^*(k)\rangle \\
&=\sum_{i=1}^{4}\langle \mathcal{Q}_{ix}(k+1),\mathcal{P}_i^*(k)\rangle+\sum_{i=1}^{4}\langle \mathcal{Q}_{iy}(k+1),\mathcal{P}_i^*(k)\rangle \\
&=\sum_{i=1}^{4}[\langle \mathcal{R}_i(k+1),\Gamma_i^*[\mathcal{P}_1^*(k),\mathcal{P}_2^*(k),\mathcal{P}_3^*(k),\mathcal{P}_4^*(k)]\rangle \\
&\quad+\beta(k)\langle \mathcal{P}_i(k),\Gamma_i^*[\mathcal{P}_1^*(k),\mathcal{P}_2^*(k),\mathcal{P}_3^*(k),\mathcal{P}_4^*(k)]\rangle] \\
&\quad+\sum_{i=1}^{4}[\langle \mathcal{R}_i(k+1),\Phi_i^*[\mathcal{P}_1^*(k),\mathcal{P}_2^*(k),\mathcal{P}_3^*(k),\mathcal{P}_4^*(k)]\rangle \\
&\quad+\beta(k)\langle \mathcal{P}_i(k),\Phi_i^*[\mathcal{P}_1^*(k),\mathcal{P}_2^*(k),\mathcal{P}_3^*(k),\mathcal{P}_4^*(k)]\rangle] \\
&=\sum_{i=1}^{4}\Big[\langle \mathcal{R}_i(k+1),\mathcal{Q}_{ix}^*(k)\rangle+\beta(k)\langle \mathcal{P}_i(k),\mathcal{Q}_{ix}^*(k)\rangle+\langle \mathcal{R}_i(k+1),\mathcal{Q}_{iy}^*(k)\rangle+\beta(k)\langle \mathcal{P}_i(k),\mathcal{Q}_{iy}^*(k)\rangle\Big] \\
&=\sum_{i=1}^{4}\left[\left\langle \mathcal{R}_i(k+1),\mathcal{Q}_{ix}^*(k)+\mathcal{Q}_{iy}^*(k)\right\rangle\right. \\
&\quad\left.-\frac{\sum_{i=1}^{4}\left\langle \mathcal{R}_i(k+1),\mathcal{Q}_{ix}^*(k)+\mathcal{Q}_{iy}^*(k)\right\rangle}{\sum_{i=1}^{4}\left\langle \mathcal{P}_i(k),\mathcal{Q}_{ix}^*(k)+\mathcal{Q}_{iy}^*(k)\right\rangle}+\left\langle \mathcal{P}_i(k),\mathcal{Q}_{ix}^*(k)+\mathcal{Q}_{iy}^*(k)\right\rangle\right] \\
&=0.
\end{aligned}
$$

Similarly, the equalities (12) and (13) also hold for the case $1 \le m < l \le k$. Therefore, the facts illustrate that (12) and (13) are satisfied for $l \neq m$. □

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 August 2024 doi:10.20944/preprints202408.0929.v1

Corollary 1. *Assume the conditions in Theorem 2 are satisfied. Then*

$$\sum_{i=1}^{4}\langle\mathcal{R}_i(k),\mathcal{P}_i^*(k)\rangle=\sum_{i=1}^{4}\langle\mathcal{R}_i(k),\mathcal{R}_i^*(k)\rangle, \tag{15}$$

$$\sum_{i=1}^{4}\langle\mathcal{Q}_{ix}(k)+\mathcal{Q}_{iy}(k),\mathcal{R}_i^*(k)\rangle=\sum_{i=1}^{4}\langle\mathcal{Q}_{ix}(k)+\mathcal{Q}_{iy}(k),\mathcal{P}_i^*(k)\rangle. \tag{16}$$

Proof of Corollary 1. From Algorithm 1 and Theorem 2, we have

$$\begin{aligned}\sum_{i=1}^{4}\langle\mathcal{R}_i(k),\mathcal{P}_i^*(k)\rangle&=\sum_{i=1}^{4}\langle\mathcal{R}_i(k),\mathcal{R}_i^*(k)+\beta(k-1)\mathcal{P}_i^*(k-1)\rangle\\&=\sum_{i=1}^{4}\langle\mathcal{R}_i(k),\mathcal{R}_i^*(k)\rangle,\end{aligned}$$

and

$$\begin{aligned}\sum_{i=1}^{4}\langle\mathcal{Q}_{ix}(k)+\mathcal{Q}_{iy}(k),\mathcal{R}_i^*(k)\rangle&=\sum_{i=1}^{4}\Big\langle\mathcal{Q}_{ix}(k)+\mathcal{Q}_{iy}(k),\mathcal{P}_i^*(k)-\beta(k-1)\mathcal{P}_i^*(k-1)\Big\rangle\\&=\sum_{i=1}^{4}\langle\mathcal{Q}_{ix}(k)+\mathcal{Q}_{iy}(k),\mathcal{P}_i^*(k)\rangle.\end{aligned}$$

□

Theorem 3. *Let tensor sequences $\{\mathcal{X}_i(k)\}$, $\{\mathcal{Y}_i(k)\}$ $(i=1,2,3,4)$ be generated by Algorithm* **??**. *If Algorithm 1 does not break down, then the tensor sequences*

$$\begin{aligned}\{[\mathcal{X}(k),\mathcal{Y}(k)]\mid\mathcal{X}(k)&=\mathcal{X}_1(k)+\mathcal{X}_2(k)\mathbf{i}+\mathcal{X}_3(k)\mathbf{j}+\mathcal{X}_4(k)\mathbf{k},\\ \mathcal{Y}(k)&=\mathcal{Y}_1(k)+\mathcal{Y}_2(k)\mathbf{i}+\mathcal{Y}_3(k)\mathbf{j}+\mathcal{Y}_4(k)\mathbf{k},k=1,2,\dots\}\end{aligned}$$

converge to the solution of Equation (2) within a finite iteration steps in the absence of round-off errors.

Proof of Theorem 3. We will prove that there exists a $k\le 4S_N$ such that $\mathcal{R}_i(k)=0$. By contradiction, assume that $\mathcal{R}_i(k)\neq 0, i=1,2,3,4$, for all $k\le 4S_N$, and thus we can compute $\mathcal{R}_i(4S_N+1)$. Suppose that $\mathcal{R}_i(1),\mathcal{R}_i(2),\dots,\mathcal{R}_i(4S_N)$ is a dependent sequence, then there exist real numbers $\lambda_{i,1},\dots,\lambda_{i,4S_N}$, not all zero, such that

$$\lambda_{i,1}\mathcal{R}_i(1)+\cdots+\lambda_{i,4S_N}\mathcal{R}_i(4S_N)=0,$$

for $i=1,2,3,4$. Then

$$\begin{aligned}\sum_{i=1}^{4}\langle\mathcal{R}_i(l),0\rangle&=\sum_{i=1}^{4}\langle\mathcal{R}_i(l),\lambda_{i,1}\mathcal{R}_i(1)+\cdots+\lambda_{i,4S_N}\mathcal{R}_i(4S_N)\rangle\\&=\sum_{i=1}^{4}\lambda_{i,l}\langle\langle\mathcal{R}_i(l),\langle\mathcal{R}_i(l)\rangle\\&=0,\end{aligned}$$

which implies $\sum_{i=1}^{4}\langle\langle\mathcal{R}_i(l),\langle\mathcal{R}_i(l)\rangle=0$. This is a contradiction since we can not calculate $\mathcal{R}_i(4S_N+1)$ in this case. Therefore, there must exists a $k\le 4S_N$ such that $\mathcal{R}_i(k)=0$, that is, the exact solution of tensor Equation (2) can be computed by Algorithm 1 within a finite iteration steps in the absence of round-off errors. □

In the following theorem, we show that if we choose special kinds of initial tensor, then Algorithm 1 can yield the unique minimal Frobenius norm solution of the tensor Equation (2).

Theorem 4. *If we choose the initial tensors as*

$$\mathcal{X}_j(1) = \Gamma_j^*[\mathcal{H}_1(1), \mathcal{H}_2(1), \mathcal{H}_3(1), \mathcal{H}_4(1)],\ \mathcal{Y}_j(1) = \Phi_j^*[\mathcal{H}_1(1), \mathcal{H}_2(1), \mathcal{H}_3(1), \mathcal{H}_4(1)], \tag{17}$$

where $\Gamma_j^*, \Phi_j^* (j = 1,2,3,4)$ are defined by (10) and (11), $\mathcal{H}_i(1) \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}, (i = 1,2,3,4)$ are arbitrary tensors (in particular, we take $\mathcal{X}_j(1) = \mathcal{O}, \mathcal{Y}_j(1) = \mathcal{O}, j = 1,2,3,4$), then the solution group $\widetilde{\mathcal{X}}_j, \widetilde{\mathcal{Y}}_j (j = 1,2,3,4)$ given by Algorithm 1 is the unique minimal Frobenius norm solution of the tensor Equations (2).

Proof of Theorem 4. If we choose the initial tensors as (17), it is not difficult to verify that $\widetilde{\mathcal{X}}_j, \widetilde{\mathcal{Y}}_j (j = 1,2,3,4)$ obtained by Algorithm 1 have the following form

$$\widetilde{\mathcal{X}}_j = \Gamma_j^*[\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3, \mathcal{H}_4],\ \widetilde{\mathcal{Y}}_j = \Phi_j^*[\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3, \mathcal{H}_4],$$

where tensors $\mathcal{H}_i \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N} (i = 1,2,3,4)$. Now, we show that $\widetilde{\mathcal{X}} = \widetilde{\mathcal{X}}_1 + \widetilde{\mathcal{X}}_2\mathbf{i} + \widetilde{\mathcal{X}}_3\mathbf{j} + \widetilde{\mathcal{X}}_4\mathbf{k}$, $\widetilde{\mathcal{Y}} = \widetilde{\mathcal{Y}}_1 + \widetilde{\mathcal{Y}}_2\mathbf{i} + \widetilde{\mathcal{Y}}_3\mathbf{j} + \widetilde{\mathcal{Y}}_4\mathbf{k}$ is the unique minimal Frobenius norm solution of tensor Equation (2). Let

$$\tilde{z} = \begin{pmatrix} \mathrm{vec}\left(\widetilde{\mathcal{X}}_1\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{X}}_2\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{X}}_3\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{X}}_4\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{Y}}_1\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{Y}}_2\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{Y}}_3\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{Y}}_4\right) \end{pmatrix} = \begin{pmatrix} \mathrm{Kro}\left(\mathcal{L}^T_{A_1^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{A_2^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{A_3^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{A_4^{(n)}}\right) \\ -\mathrm{Kro}\left(\mathcal{L}^T_{A_2^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{A_1^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}^T_{A_4^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{A_3^{(n)}}\right) \\ -\mathrm{Kro}\left(\mathcal{L}^T_{A_3^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{A_4^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{A_1^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}^T_{A_2^{(n)}}\right) \\ -\mathrm{Kro}\left(\mathcal{L}^T_{A_4^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}^T_{A_3^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{A_2^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{A_1^{(n)}}\right) \\ \mathrm{Kro}\left(\mathcal{L}^T_{B_1^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{B_2^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{B_3^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{B_4^{(n)}}\right) \\ -\mathrm{Kro}\left(\mathcal{L}^T_{B_2^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{B_1^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}^T_{B_4^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{B_3^{(n)}}\right) \\ -\mathrm{Kro}\left(\mathcal{L}^T_{B_3^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{B_4^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{B_1^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}^T_{B_2^{(n)}}\right) \\ -\mathrm{Kro}\left(\mathcal{L}^T_{B_4^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}^T_{B_3^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{B_2^{(n)}}\right) & +\mathrm{Kro}\left(\mathcal{L}^T_{B_1^{(n)}}\right) \end{pmatrix}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 August 2024 doi:10.20944/preprints202408.0929.v1

$$
\times \begin{pmatrix} \mathrm{vec}\left(\widetilde{\mathcal{H}_1}\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{H}_2}\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{H}_3}\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{H}_4}\right) \end{pmatrix},
$$

$$
= \begin{pmatrix} \mathrm{Kro}\left(\mathcal{L}_{A_1^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{A_2^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{A_3^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{A_4^{(n)}}\right) \\ \mathrm{Kro}\left(\mathcal{L}_{A_2^{(n)}}\right) & \mathrm{Kro}\left(\mathcal{L}_{A_1^{(n)}}\right) & \mathrm{Kro}\left(\mathcal{L}_{A_4^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{A_3^{(n)}}\right) \\ \mathrm{Kro}\left(\mathcal{L}_{A_3^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{A_4^{(n)}}\right) & \mathrm{Kro}\left(\mathcal{L}_{A_1^{(n)}}\right) & \mathrm{Kro}\left(\mathcal{L}_{A_2^{(n)}}\right) \\ \mathrm{Kro}\left(\mathcal{L}_{A_4^{(n)}}\right) & \mathrm{Kro}\left(\mathcal{L}_{A_3^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{A_2^{(n)}}\right) & \mathrm{Kro}\left(\mathcal{L}_{A_1^{(n)}}\right) \end{pmatrix}
$$

$$
\begin{pmatrix} \mathrm{Kro}\left(\mathcal{L}_{B_1^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{B_2^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{B_3^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{B_4^{(n)}}\right) \\ \mathrm{Kro}\left(\mathcal{L}_{B_2^{(n)}}\right) & \mathrm{Kro}\left(\mathcal{L}_{B_1^{(n)}}\right) & \mathrm{Kro}\left(\mathcal{L}_{B_4^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{B_3^{(n)}}\right) \\ \mathrm{Kro}\left(\mathcal{L}_{B_3^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{B_4^{(n)}}\right) & \mathrm{Kro}\left(\mathcal{L}_{B_1^{(n)}}\right) & \mathrm{Kro}\left(\mathcal{L}_{B_2^{(n)}}\right) \\ \mathrm{Kro}\left(\mathcal{L}_{B_4^{(n)}}\right) & \mathrm{Kro}\left(\mathcal{L}_{B_3^{(n)}}\right) & -\mathrm{Kro}\left(\mathcal{L}_{B_2^{(n)}}\right) & \mathrm{Kro}\left(\mathcal{L}_{B_1^{(n)}}\right) \end{pmatrix}^T \times \begin{pmatrix} \mathrm{vec}\left(\widetilde{\mathcal{H}_1}\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{H}_2}\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{H}_3}\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{H}_4}\right) \end{pmatrix},
$$

$$
= [\mathcal{M}_A, \mathcal{M}_B]^T \times \begin{pmatrix} \mathrm{vec}\left(\widetilde{\mathcal{H}_1}\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{H}_2}\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{H}_3}\right) \\ \mathrm{vec}\left(\widetilde{\mathcal{H}_4}\right) \end{pmatrix} \in \mathcal{R}([\mathcal{M}_A, \mathcal{M}_B]^T).
$$

According to Theorem 1, we can conclude that $\widetilde{\mathcal{X}} = \widetilde{\mathcal{X}_1} + \widetilde{\mathcal{X}_2}\mathbf{i} + \widetilde{\mathcal{X}_3}\mathbf{j} + \widetilde{\mathcal{X}_4}\mathbf{k}$, $\widetilde{\mathcal{Y}} = \widetilde{\mathcal{Y}_1} + \widetilde{\mathcal{Y}_2}\mathbf{i} + \widetilde{\mathcal{Y}_3}\mathbf{j} + \widetilde{\mathcal{Y}_4}\mathbf{k}$ generated by Algorithm 1 is the unique minimal Frobenius norm solution of the tensor Equations (2). □

Now, we solve Problem 1.2. If the tensor Equation (2) is consistent, the solution pair set S_{XY} of Problem 1.1 is non-empty, for given tensors $\bar{\mathcal{X}}, \bar{\mathcal{Y}} \in \mathbb{H}^{I_1 \times I_2 \times \cdots \times I_N}$, we have

$$
\begin{aligned}
&\min_{\mathcal{X},\mathcal{Y} \in \mathbb{H}_1 \times I_2 \times \cdots \times I_N} \left\| \sum_{k=1}^{N} \mathcal{X} \times_k A^{(k)} + \mathcal{Y} \times_k B^{(k)} - \mathcal{C} \right\| \\
&= \min_{\mathcal{X},\mathcal{Y} \in \mathbb{H}^{I_1 \times I_2 \times \cdots \times I_N}} \left\| \sum_{k=1}^{N} (\mathcal{X} - \bar{\mathcal{X}}) \times_k A^{(k)} + (\mathcal{Y} - \bar{\mathcal{Y}}) \times_k B^{(k)} \right. \\
&\left. - \left(\mathcal{C} - \sum_{k=1}^{N} \bar{\mathcal{X}} \times_k A^{(k)} - \bar{\mathcal{Y}} \times_k B^{(k)} \right) \right\|.
\end{aligned}
$$

Let $\widetilde{\mathcal{X}} = \mathcal{X} - \bar{\mathcal{X}}, \widetilde{\mathcal{Y}} = \mathcal{Y} - \bar{\mathcal{Y}}$, and $\widetilde{\mathcal{C}} = \mathcal{C} - \sum_{k=1}^{N} \bar{\mathcal{X}} \times_k A^{(k)} - \bar{\mathcal{Y}} \times_k B^{(k)}$, then the tensor nearness Problem 1.2 is equivalent to first finding the minimal Frobenius norm sulution of tensor equation

$$
\sum_{k=1}^{N} \widetilde{\mathcal{X}} \times_k A^{(k)} + \widetilde{\mathcal{Y}} \times_k B^{(k)} = \mathcal{C}. \tag{18}
$$

By using Algorithm **??**, and letting the initial tensors be $\widetilde{\mathcal{X}} = \widetilde{\mathcal{X}_1} + \widetilde{\mathcal{X}_2}\mathbf{i} + \widetilde{\mathcal{X}_3}\mathbf{j} + \widetilde{\mathcal{X}_4}\mathbf{k}$, $\widetilde{\mathcal{Y}} = \widetilde{\mathcal{Y}_1} + \widetilde{\mathcal{Y}_2}\mathbf{i} + \widetilde{\mathcal{Y}_3}\mathbf{j} + \widetilde{\mathcal{Y}_4}\mathbf{k}$, where $\mathcal{X}_j(1) = \Gamma_j^*[\mathcal{H}_1(1), \mathcal{H}_2(1), \mathcal{H}_3(1), \mathcal{H}_4(1)]$, $\mathcal{Y}_j(1) = \Phi_j^*[\mathcal{H}_1(1), \mathcal{H}_2(1), \mathcal{H}_3(1), \mathcal{H}_4(1)](j = 1, 2, 3, 4)$, where $\mathcal{H}_i(1) \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}(i = 1, 2, 3, 4)$ are arbitrary tensors (in particular, we

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 August 2024 doi:10.20944/preprints202408.0929.v1

take $\mathcal{X}_j(1) = \mathcal{O}, \mathcal{Y}_j(1) = \mathcal{O}, j = 1,2,3,4)$, we can obtain the unique minimal Frobenius norm solution $\widetilde{\mathcal{X}}^*, \widetilde{\mathcal{Y}}^*$ of Equation (18). Once the above tensor $\widetilde{\mathcal{X}}^*, \widetilde{\mathcal{Y}}^*$ are achieved, the unique solution $\check{\mathcal{X}}, \check{\mathcal{Y}}$ of Problem 1.2 can be computed. In this case, $\check{\mathcal{X}}, \check{\mathcal{Y}}$ can be expressed as $\check{\mathcal{X}} = \widetilde{\mathcal{X}}^* + \bar{\mathcal{X}}, \check{\mathcal{Y}} = \widetilde{\mathcal{Y}}^* + \bar{\mathcal{Y}}$.

4. Numerical Examples

In this section, we will give some numerical examples to support the efficiency and applications of Algorithm 1. The codes in our computation are written in Matlab R2018a with 2.3 GHz central processing unit (Intel(R) Core(TM) i5), 8GB memory. Moreover, we implemented all the operations based on tensor toolbox (version 3.2.1) proposed by Bader and Kolda [27]. For all of the examples, the iterations begin with the initial values $\mathcal{X}_i = \mathcal{Y}_i = 0,\ i = 1,2,3,4$ in Algorithm 1, and the stopping criteria is Res $\leq 10^{-5}$ or the number of iteration steps exceeding 2000. We describe some notations that appear in the following examples in Table 1.

Table 1. Some denotations in numerical examples

IT	The number of iteration steps
CPU time	The Elapsed CPU time in seconds
Res	$\sum_{i=1}^{4} \|\mathcal{R}_i(k)\|$, $\mathcal{R}_i(k)$ is the residual at kth iteration.
$tenrand(n,n,n)$	The order three tensor with pseudo-random values drawn from a uniform distribution on the unit interval
$triu(hilb(n))$	The upper triangular portion of Hilbert matrix
$triu(ones(n,n))$	The upper triangular portion of matrix with all 1
$eye(n)$	Identity matrix
$zeros(n)$	Zero matrix
$tridiag(a,b,c,n)$	The tridiagonal matrix with a, b, c

Example 1. We consider the tensor Equation (2) with:

$$A^{(1)} = \begin{pmatrix} 2+4\mathbf{i}+4\mathbf{j}+5\mathbf{k} & 1-\mathbf{i}+\mathbf{j}+2\mathbf{k} & -2\mathbf{i}-2\mathbf{j}+\mathbf{k} \\ -2+\mathbf{j} & 3+2\mathbf{i}+5\mathbf{j}+\mathbf{k} & 1-2\mathbf{j}+\mathbf{k} \\ 2+2\mathbf{i}+\mathbf{j} & 1-2\mathbf{i}+2\mathbf{j}-\mathbf{k} & 3+4\mathbf{i}+4\mathbf{j}+\mathbf{k} \end{pmatrix},$$

$$A^{(2)} = \begin{pmatrix} 3+4\mathbf{j} & -1+\mathbf{j} & -1-2\mathbf{j} \\ -2+\mathbf{j} & 4+3\mathbf{j} & -\mathbf{j} \\ -\mathbf{j} & 1 & 2+\mathbf{j} \end{pmatrix}, \quad A^{(3)} = \begin{pmatrix} 2\mathbf{i}+3\mathbf{j} & \mathbf{i} & 1+\mathbf{j} \\ -2\mathbf{i}-\mathbf{j} & 5\mathbf{i}+3\mathbf{j} & -\mathbf{i} \\ 2\mathbf{i}-\mathbf{j} & -\mathbf{i}-\mathbf{j} & 5\mathbf{i}+4\mathbf{j} \end{pmatrix},$$

$$B^{(1)} = \begin{pmatrix} 3+3\mathbf{i} & 2+\mathbf{i} & -2\mathbf{i} \\ -2+\mathbf{i} & 4+4\mathbf{i} & -2-2\mathbf{i} \\ 2+2\mathbf{i} & 2 & 6+4\mathbf{i} \end{pmatrix}, \quad B^{(2)} = \begin{pmatrix} 3\mathbf{i}+5\mathbf{j} & 2\mathbf{i} & 2\mathbf{i}+2\mathbf{j} \\ 0 & 2\mathbf{i}+3\mathbf{j} & \mathbf{j} \\ -2\mathbf{i}+2\mathbf{j} & -2\mathbf{i}-2\mathbf{j} & 6\mathbf{i}+6\mathbf{j} \end{pmatrix},$$

$$B^{(3)} = \begin{pmatrix} 1+3\mathbf{i}+3\mathbf{k} & -1-2\mathbf{i} & \mathbf{i}+2\mathbf{k} \\ -\mathbf{i} & 2+4\mathbf{i}+2\mathbf{k} & 2+2\mathbf{i} \\ 2+\mathbf{i}+2\mathbf{k} & 2+\mathbf{i}-2\mathbf{k} & 5+4\mathbf{i}+6\mathbf{k} \end{pmatrix},$$

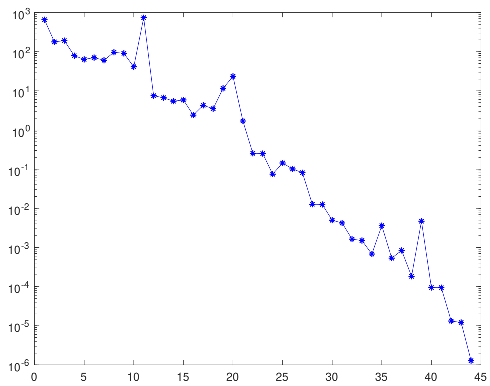
$$\mathcal{C}(:,:,1) = \begin{pmatrix} 3-11\mathbf{i}+31\mathbf{j}+19\mathbf{k} & 6-8\mathbf{i}+\mathbf{j}+28\mathbf{k} & 2-6\mathbf{i}+20\mathbf{j}+24\mathbf{k} \\ 4+36\mathbf{j}+8\mathbf{k} & 7+3\mathbf{i}+27\mathbf{j}+17\mathbf{k} & 3+5\mathbf{i}+25\mathbf{j}+13\mathbf{k} \\ 24-2\mathbf{i}+52\mathbf{j}+22\mathbf{k} & 27+\mathbf{i}+43\mathbf{j}+31\mathbf{k} & 23+3\mathbf{i}+41\mathbf{j}+27\mathbf{k} \end{pmatrix},$$

$$\mathcal{C}(:,:,2) = \begin{pmatrix} 11-13\mathbf{i}+31\mathbf{j}+13\mathbf{k} & 14-10\mathbf{i}+22\mathbf{j}+22\mathbf{k} & 10-8\mathbf{i}+20\mathbf{j}+18\mathbf{k} \\ 12-2\mathbf{i}+36\mathbf{j}+2\mathbf{k} & 15+\mathbf{i}+27\mathbf{j}+11\mathbf{k} & 11+3\mathbf{i}+25\mathbf{j}+7\mathbf{k} \\ 32-4\mathbf{i}+52\mathbf{j}+16\mathbf{k} & 35-\mathbf{i}+43\mathbf{j}+25\mathbf{k} & 31+\mathbf{i}+41\mathbf{j}+21\mathbf{k} \end{pmatrix},$$

$$\mathcal{C}(:,:,3) = \begin{pmatrix} 9-9\mathbf{i}+45\mathbf{j}+25\mathbf{k} & 12-6\mathbf{i}+36\mathbf{j}+34\mathbf{k} & 8-4\mathbf{i}+34\mathbf{j}+30\mathbf{k} \\ 10+2\mathbf{i}+50\mathbf{j}+14\mathbf{k} & 13+5\mathbf{i}+41\mathbf{j}+23\mathbf{k} & 9+7\mathbf{i}+39\mathbf{j}+19\mathbf{k} \\ 30+66\mathbf{j}+28\mathbf{k} & 33+3\mathbf{i}+57\mathbf{j}+3\mathbf{k} & 29+5\mathbf{i}+55\mathbf{j}+33\mathbf{k} \end{pmatrix}.$$

Applying for Algorithm 1, the IT is 46, the costed CPU time is 3.9496s and the Res is 1.2885e-06. Besides that, the Figure 1 reported here illustrates Algorithm 1 is feasible.

Figure 1. Convergence history of Example 1.



Example 2 (Test matrices from [1]). In this example, we consider the quaternion tensor Equation (2) such that

$$A^{(m)} = A_{m1} + A_{m2}\mathbf{i} + A_{m3}\mathbf{j} + A_{m4}\mathbf{k}, \quad B^{(m)} = B_{m1} + B_{m2}\mathbf{i} + B_{m3}\mathbf{j} + B_{m4}\mathbf{k}, \quad m = 1,2,3,$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 August 2024 doi:10.20944/preprints202408.0929.v1

where

$$A_{11} = triu(hilb(n)), A_{12} = triu(ones(n,n)), A_{13} = eye(n), A_{14} = ones(n),$$
$$A_{21} = zeros(n), A_{22} = zeros(n), A_{23} = tridiag(-1,2,-1,n), A_{24} = zeros(n),$$
$$A_{31} = zeros(n), A_{32} = tridiag(0.5,6,-0.5,n), A_{33} = eye(n), A_{34} = zeros(n),$$

$$B_{11} = eye(n), B_{12} = ones(n), B_{13} = zeros(n), B_{14} = zeros(n),$$
$$B_{21} = zeros(n), B_{22} = tridiag(0.5,6,-0.5,n), B_{23} = eye(n), B_{24} = zeros(n),$$
$$B_{31} = tridiag(0.5,6,-0.5,n), B_{32} = eye(n), B_{33} = zeros(n), B_{34} = ones(n).$$

$$\mathcal{C} = tenrand(n,n,n) + tenrand(n,n,n)\mathbf{i} + tenrand(n,n,n)\mathbf{j} + tenrand(n,n,n)\mathbf{k}.$$

Choosing the initial tensor $\mathcal{X} = \mathcal{Y} = 0$, we depict the convergence curves of Algorithm 1 for different n in Figure 2. For $n = 20$, $n = 40$ and $n = 60$, we list the costed CPU time, residual norms after finite steps and the relative errors of approximate solutions by Algorithm 1 in Table 2.

Figure 2. Convergence history of Example 2

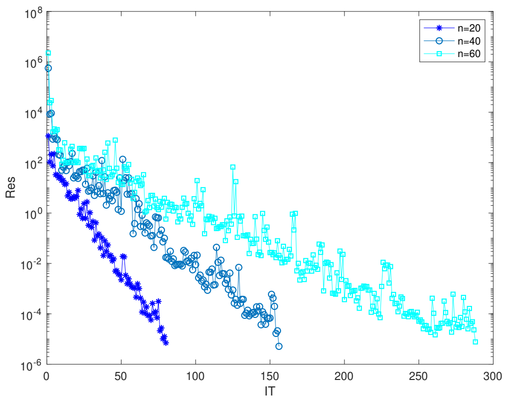


Table 2. Numerical results for Example 2

	IT	CPU time	Res
n=20	82	8.3272	7.7182e-06
n=40	156	24.8187	5.1233e-06
n=60	288	91.4387	7.6898e-06

Example 3. We consider the solution of the following convection-diffusion equation over the quaternion algebra [4]

$$\begin{aligned} -v\Delta u + c^T \nabla u &= f \quad \text{in } \Gamma = [0,1] \times [0,1] \times [0,1] \\ u &= 0 \quad \text{on } \partial\Gamma. \end{aligned}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 August 2024 doi:10.20944/preprints202408.0929.v1

Based on a standard finite difference discretization on a uniform grid for the diffusion term and a second-order convergent scheme (Fromm's scheme) for the convection term with the mesh-size $h = \frac{1}{p+1}$, we solve the quaternion tensor Equation (3) with

$$A^{(n)} = A_1^{(n)} + A_2^{(n)}\mathbf{i} + A_3^{(n)}\mathbf{j} + A_4^{(n)}\mathbf{k}, \quad n = 1,2,3,$$

where

$$A_i^{(n)} = \frac{v_i^{(n)}}{h^2}\begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}_{p\times p} + \frac{c_i^{(n)}}{4h}\begin{bmatrix} 3 & -5 & 1 & & \\ 1 & 3 & -5 & & \\ & \ddots & \ddots & \ddots & 1 \\ & & 1 & 3 & -5 \\ & & & 1 & 3 \end{bmatrix}_{p\times p}, \quad i = 1,2,3,4.$$

The right hand side tensor $\mathcal{C}$ is contructed such that the exact solution of Equation (3) is $\mathcal{X}^* = \mathcal{X}_1^* + \mathcal{X}_2^*\mathbf{i} + \mathcal{X}_3^*\mathbf{j} + \mathcal{X}_4^*\mathbf{k} = tenones(p,p,p) + tenones(p,p,p)\mathbf{i} + tenones(p,p,p)\mathbf{j} + tenones(p,p,p)\mathbf{k}$.

We consider two cases in order to compare Algorithm 1 with the CGLS algorithm in [49]. In case I, we choose different $v_i^{(n)}$ and $c_i^{(n)}$ to obtain the results. In Table 3, we set

$$v_i^{(1)} = 1,\ c_i^{(1)} = 1;\ v_i^{(2)} = 1,\ c_i^{(2)} = 2;\ v_i^{(3)} = 1,\ c_i^{(2)} = 3,\ i = 1,2,3,4 \tag{19}$$

to get $\mathcal{A}^{(n)}$ $(n = 1,2,3)$ with same real part and imaginary part. In Table 4, we set

$$\begin{aligned} &v_i^{(1)} = 1,\ c_1^{(1)} = 1,\ c_2^{(1)} = -1,\ c_3^{(1)} = 0,\ c_4^{(1)} = 1; \\ &v_i^{(2)} = 0.1,\ c_1^{(2)} = -1,\ c_2^{(2)} = -1,\ c_3^{(2)} = 1,\ c_4^{(2)} = 0; \\ &v_i^{(3)} = 0.01,\ c_1^{(3)} = 1,\ c_2^{(3)} = 1,\ c_3^{(3)} = 0,\ c_4^{(3)} = 0,\ i = 1,2,3,4 \end{aligned} \tag{20}$$

to get $\mathcal{A}^{(n)}$ $(n = 1,2,3)$ with different real parts and imaginary parts.

Table 3. CPU time (IT) for Example 3 with parameter setup in (19)

	p=10	p=25	p=30
Algorithm 1	25.6884(320)	140.8233(1270)	202.7709(1580)
CGLS [49]	15.5685(219)	144.3166(1266)	230.7340(1832)

Table 4. CPU time (IT) for Example 3 with parameter setup in (20)

	p=10	p=15	p=20
Algorithm **??**	45.4157(576)	104.4370(1095)	163.6110(1700)
CGLS [49]	44.0701(579)	118.2419(1296)	230.7978(2298)

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 August 2024 doi:10.20944/preprints202408.0929.v1

In case II, we set $c_i^{(1)} = 1$; $c_i^{(2)} = 2$; $c_i^{(2)} = 3$, $i = 1,2,3,4$, we apply Algorithm 1 and the CGLS algorithm in [49] with $v_i^{(j)} = 1e-03, 1, 100$, $i = 1,2,3,4$, $j = 1,2,3$ for grid $10 \times 10 \times 10$. The corresponding relative errors of approximate solution $\sum_{i=1}^{i=4} \|\mathcal{X}_i(k) - \mathcal{X}_i^*\| / \|\mathcal{X}_i^*\|$ computed by these methods are presented in Table 5.

Table 5. The relative errors of the solution (IT) for Example 3

	$v_i^{(1)} = 1e-03$	$v_i^{(2)} = 1$	$v_i^{(3)} = 100$
Algorithm **??**	5.6210e-09(248)	6.0527e-09(188)	4.1925e-09(248)
CGLS [49]	9.9300e-09(106)	9.7035e-09(178)	8.9552e-09(167)

The previous results show that Algorithm 1 has faster convergent rates than the CGLS algorithm in [49] as p increases.

Example 4. In this example, we employ Algorithm 1 to compare its performance with the CGLS algorithm in restoring a color video comprised of a sequence of RGB images (slices). The video, named 'rhinos', is sourced from Matlab and stored in AVI format. Each frontal slice of this color video is depicted by a pure quaternion matrix measuring 240×320 pixels. For $\mathcal{C} = \hat{\mathcal{C}} + \mathcal{N} = \mathcal{X} \times_1 A$, we consider $\mathcal{X}$ to represent the orginal colour video, A as the blurred matrix, and $\mathcal{N}$ as a noise tensor. When $\mathcal{N} = 0$, $\mathcal{C}$ is referred to as the blurred and noise-free color video. In this scenario, we select a blurred matrix $A = A_1 \otimes A_2 \in \mathbb{R}^{240\times 320}$, where $A_1 = \left(a_{ij}^{(1)}\right)_{1\le i,j\le 16}$ and $A_2 = \left(a_{ij}^{(2)}\right)_{1\le i\le 15, 1\le j\le 20}$ are Toeplitz matrices with entries defined as:

$$a_{ij}^{(1)} = \begin{cases} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{(i-j)^2}{2\sigma^2}\right), & |i-j| \le r, \\ 1, & \text{otherwise.} \end{cases} \quad ; \quad a_{ij}^{(2)} = \begin{cases} \frac{1}{2s-1}, & |i-j| \le s, \\ 0, & \text{otherwise.} \end{cases}$$

We denote $\mathcal{X}_{\text{restored}}$ as the resulting restored color video. The algorithm's performance is assessed using the peak signal-to-noise ratio (PSNR) measured in decibels (dB):

$$\text{PSNR}(\mathcal{X}) = 10 \log_{10}\left(\frac{I_1 I_2 d^2}{\|\mathcal{X} - \mathcal{X}_{\text{restored}}\|^2}\right),$$

where d represents the maximum possible pixel value of the image. RE($\mathcal{X}$) represents the relative error defined as

$$\text{RE}(\mathcal{X}) = \frac{\|\mathcal{X} - \mathcal{X}_{\text{restored}}\|}{\|\mathcal{X}\|}.$$

Here, we set $d = 255$ and the variance $\sigma = 1$. The peak signal-to-noise ratio (PSNR) and the relative error (RE) of Algorithm 1 and the CGLS algorithm with different parameters are presented in Table 6. As indicated, the PSNR and the relative error of our algorithms are significantly superior to those of CGLS. For the case where $r = 6$ and $s = 6$ with slice No. 7 of the color video, we illustrate the original image, blurred image, and the restored image by CGLS and Algorithm 1 in Figure 3. This figure demonstrates that our algorithm can effectively restore blurred and noise-free color video with high quality.

 doi:10.20944/preprints202408.0929.v1

Table 6. The numerical results for Example 4

Algorithm([r,s])	Algorithm **??**(PSNR/RE)	CGLS(PSNR/RE)
[3,3]	38.6181(0.0235)	13.8404(0.3769)
[6,6]	37.3721(0.0300)	14.0529(0.3694)
[8,8]	33.8073(0.0338)	14.7958(0.3551)

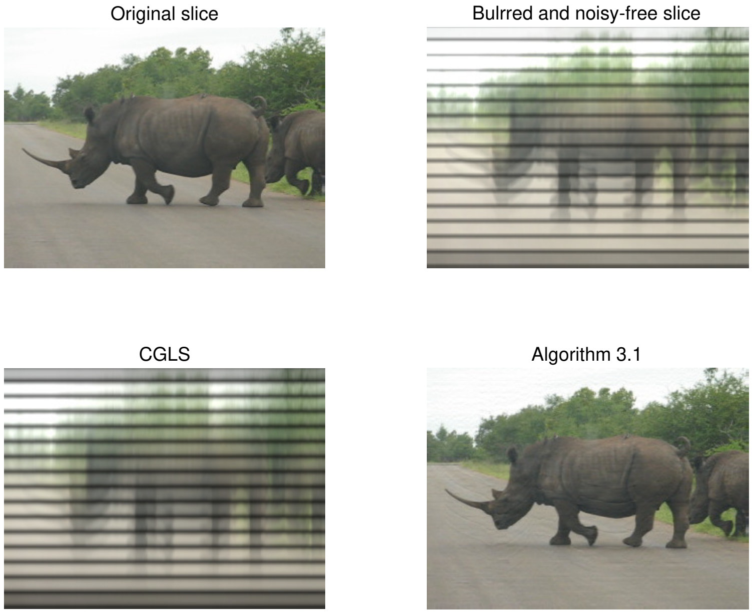


Figure 3. The restored colour video for the case $r = 6, s = 6$ with slice No. 7

5. Conclusions

The main goal of this paper is to solve the generalized quaternion tensor Equation (2). We herewith develop a BiCG iterative algorithm based on the tensor format to solve Equation (2) efficiently, and we also prove the convergence of our proposed method. Moreover, we demonstrate that the solution with minimal Frobenius norm can be achieved by initializing specific types of tensors. We present several examples to effectively demonstrate the efficiency of our algorithm. Furthermore, our algorithm is successfully applied to the restoration of color videos. This contribution significantly advances the current understanding of quaternion tensor equations by introducing a practical iterative approach.

Author Contributions: All authors have equal contributions in conceptualization, formal analysis, investigation, methodology, software, validation, writing an original draft, writing a review, and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research gratelfully acknowledges financial support from National Natural Science Foundation of China (#12371023, #12301028) and Canada NSERC grant RGPIN-2020-06746.

Data Availability Statement: Data are contained within the article.

 doi:10.20944/preprints202408.0929.v1

Acknowledgments: The authors would like to thank the editor and reviewers for their valuable suggestions, comments, and the Natural Science Foundation of China under Grant Nos.#12371023, #12301028, the Canada NSERC under Grant No. RGPIN-2020-06746.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. S. Ahmadi-Asl, F. P. A. Beik. An efficient iterative algorithm for quaternionic least-squares problems over the generalized η-(anti-)bi-Hermitian matrices. *Linear Multilinear Algebra*, **2017**, *65(9)*, 1743–1769.
2. S. Ahmadi-Asl, F. P. A. Beik. Iterative algorithms for least-squares solutions of a quaternion matrix equation. *J. Appl. Math. Comput.*, **2017**, *53*, 95–127.
3. Z.-Z. Bai, G. H. Golub, M. K. Ng. Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems. *SIAM J. Matrix Anal. Appl.*, **2003**, *24(3)*, 603–626.
4. J. Ballani, L. Grasedyck. A projection method to solve linear systems in tensor format. *Numer. Linear Algebra Appl.*, **2013**, *20(1)*, 27–43.
5. R. E. Bank , T.-F. Chan. An analysis of the composite step biconjugate gradient method. *Numer. Math.*, **1993**, *66 (1)*, 295–319.
6. R. E. Bank , T.-F. Chan. A composite step bi-conjugate gradient algorithm for nonsymmetric linear systems. *Numer. Algorithms*, **1994**, *7 (1)*,1–16.
7. F. Beik, S. Ahmadi-Asl. An iterative algorithmfor η-(anti)-Hermitian least-squares solutions of quaternion matrix equations. *Electron. J. Linear Algebra*, **2015**, *30*, 372–401.
8. F. Beik, F. Panjeh, F. Movahed, and S. Ahmadi-Asl. On the Krylov subspace methods based on tensor format for positive definite Sylvester tensor equations. *Numer. Linear Algebra Appl.*, **2016**, *23(3)*, 444–466.
9. F. Beik, K. Jbilou, M. Najafi-Kalyani, et al. Golub–Kahan bidiagonalization for ill-conditioned tensor equations with applications. *Numer. Algorithms*, **2020**, *84*, 1535-1563.
10. Z. Chen, L. Lu. A projection method and Kronecker product preconditioner for solving Sylvester tensor equations. *Sci. China Math.*, **2012**, *55(6)*, 1281–1292.
11. X. F. Duan, Y. S. Zhang, Q. W. Wang. An efficient iterative method for solving a class of constrained tensor least squares problem. *Appl. Numer. Math.*, **2024**, *196*, 104–117.
12. R. W. Freund, G. H. Golub, and N. M. Nachtigal. Iterative solution of linear systems. *Acta Numer.*, **1992**, *1*, 44.
13. Z.-H. Gao, Q.-W. Wang, L. Xie. The (anti-)η-Hermitian solution to a novel system of matrix equations over the split quaternion algebra. *Math. Meth. Appl. Sci.*, **2024**, 1–18, DOI 10.1002/mma.10245.
14. L. Grasedyck. Existence and computation of low Kronecker-rank approximations for large linear systems of tensor product structure. *Computing*, **2004**, *72*, 247–265.
15. Y. Guan, D. Chu. Numerical computation for orthogonal low-rank approximation of tensors. *SIAM J. Matrix Anal. Appl.*, **2019**, *40(3)*, 1047–1065.
16. Y. Guan, M. T. Chu, D. Chu. Convergence analysis of an SVD-based algorithm for the best rank-1 tensor approximation. *Linear Algebra Appl.*, **2018**, *555*, 53–69.
17. Y. Guan, M. T. Chu, D. Chu. SVD-based algorithms for the best rank-1 approximation of a symmetric tensor. *SIAM J. Matrix Anal.*, **2018**, *39(3)*, 1095–1115.
18. M. Hajarian. Developing Bi-CG and Bi-CR methods to solve generalized Sylvester-transpose matrix equations. *Int. J. Auto. Comput.*, **2014**, *11 (1)*, 25–29.
19. Z.-H. He, X.-X. Wang, Y.-F. Zhao. Eigenvalues of quaternion tensors with applications to color video processing. *J. Sci. Comput.*, **2023**, *94(1)*, 1.
20. M. Heyouni, F. Saberi-Movahed, A. Tajaddini. On global Hessenberg based methods for solving Sylvester matrix equations. *Comput. Math. Appl.*, **2019**, *77(1)*, 77–92.
21. J. Hu, Y. Ke, C. Ma. Efficient iterative method for generalized Sylvester quaternion tensor equation. *Comput. Appl. Math.*, **2023**, *42(5)*, 237.
22. N. Huang, C.-F. Ma. Modified conjugate gradient method for obtaining the minimum-norm solution of the generalized coupled Sylvester-conjugate matrix equations. *Appl. Math. Model.*, **2016**, *40(2)*, 1260–1275.
23. Z. Jia, M. Wei, M.-X. Zhao, et al. A new real structure-preserving quaternion QR algorithm. *J. Comput. Appl. Math.*, **2018**, *343*, 26–48.

24. S. Karimi, M. Dehghan. Global least squares method based on tensor form to solve linear systems in Kronecker format. *Trans. Inst. Measure. Control*, **2018**, *40(7)*, 2378–2386.
25. Y. Ke. Finite iterative algorithm for the complex generalized Sylvester tensor equations. *J. Appl. Anal. Comput.*, **2020**, *10(3)*, 972–985.
26. T. G. Kolda. *Multilinear operators for higher-order decompositions*, Sandia National Laboratory(SNL),Albuquerque, NM, and Livermore, CA, 2006.
27. T. G. Kolda, B. W. Bader. Tensor decompositions and applications. *SIAM Rev.*, **2009**, *51(3)*, 455–500.
28. I. Kyrchei. Cramer's rules for Sylvester quaternion matrix equation and its special cases. *Adv. Appl. Clifford Algebras*, **2018**, *28*, 1–26.
29. B.-W. Li, S. Tian, Y.-S. Sun, et al. Schur-decomposition for 3D matrix equations and its application in solving radiative discrete ordinates equations discretized by Chebyshev collocation spectral method. *J. Comput. Phys.*, **2010**, *229(4)*, 1198–1212.
30. T. Li, Q.-W. Wang, X.-F. Duan. Numerical algorithms for solving discrete Lyapunov tensor equation. *J. Comput. Appl. Math.*, **2020**, *370*, 112676.
31. T. Li, Q.-W. Wang, X.-F. Zhang. Gradient based iterative methods for solving symmetric tensor equations. *Numer. Linear Algebra Appl.*, **2022**, *29 (2)*, DOI: 10.1002/nla.2414.
32. T. Li, Q.-W. Wang, X.-F. Zhang. A Modified conjugate residual method and nearest kronecker product preconditioner for the generalized coupled Sylvester tensor equations. *Mathematics*, **2022**, *10(10)*, 1730.
33. X. Li, M. K. Ng. Solving sparse non-negative tensor equations: Algorithms and applications. *Front. Math. China*, **2015**, *10(3)*, 649–680.
34. Y. Li, M. Wei, F. Zhang, et al. Real structure-preserving algorithms of Householder based transformations for quaternion matrices. *J. Comput. Appl. Math.*, **2016**, *305*, 82–91.
35. Y. Liang, S. D. Silva, Y. Zhang. The tensor rank problem over the quaternions. *Linear Algebra Appl.*, **2021**, *620*, 37–60.
36. C. Lv, C. Ma. A modified CG algorithm for solving generalized coupled Sylvester tensor equations. *Appl. Math. Comput.*, **2020**, *365*, 124699.
37. A. Malek, S. H. Momeni-Masuleh. A mixed collocation–finite difference method for 3D microscopic heat transport problems. *J. Comput. Appl. Math.*, **2008**, *217(1)*, 137–147.
38. M.S. Mehany, Q.-W. Wang, L. Liu. A System of Sylvester-like quaternion tensor equations with an application. *Front. Math.*, **2024**, 1–20.
39. M. Najafi-Kalyani, F. P. A. Beik, K. Jbilou. On global iterative schemes based on Hessenberg process for (ill-posed) Sylvester tensor equations. *J. Comput. Appl. Math.*, **2020**, *373*, 112216.
40. Y. Peng, X. Hu, L. Zhang. An iteration method for the symmetric solutions and the optimal approximation solution of the matrix equation $AXB = C$. *Appl. Math. Comput.*, **2005**, *160*, 763–777.
41. L. Qi. Eigenvalues of a real supersymmetric tensor. *J. Symbolic Comput.*, **2005**, *40(6)*, 1302–1324.
42. L. Qi. Symmetric nonnegative tensors and copositive tensors. *Linear Algebra Appl.*, **2013**, *439(1)*, 228–238.
43. L. Qi, H. Chen, Y. Chen. *Tensor eigenvalues and their applications*; J. Symbolic Comput., Springer, 2018.
44. L. Qi, Z. Luo. *Tensor analysis: Spectral theory and special tensors*; SIAM, Philadelphia, 2017.
45. F. Saberi-Movahed, A. Tajaddini, M. Heyouni, et al. Some iterative approaches for Sylvester tensor equations, Part I: A tensor format of truncated Loose Simpler GMRES. *Appl. Numer. Math.*, **2022**, *172*, 428–445.
46. F. Saberi-Movahed, A. Tajaddini, M. Heyouni, et al. Some iterative approaches for Sylvester tensor equations, Part II: A tensor format of Simpler variant of GCRO-based methods. *Appl. Numer. Math.*, **2022**, *172*, 413–427.
47. G. Song, Q.-W. Wang, S. Yu. Cramer's rule for a system of quaternion matrix equations with applications. *Applied Math. Comput.*, **2018**, *336*, 490–499.
48. Q.-W. Wang, Z.-H. He, Y. Zhang. Constrained two-sided coupled Sylvester-type quaternion matrix equations. *Automatica*, **2019**, *101*, 207–213.
49. Q.-W. Wang, X. Xu, X. Duan. Least squares solution of the quaternion Sylvester tensor equation. *Linear Multilinear Algebra*, **2021**, *69(1)*, 104–130.
50. M. Xie, Q.-W. Wang. Reducible solution to a quaternion tensor equation. *Front. Math. China*, **2020**, *15(5)*, 1047–1070.
51. M. Xie, Q.-W. Wang, Z.-H. He, et al. A system of Sylvester-type quaternion matrix equations with ten variables. *Acta Math. Sin. (Engl. Ser.)*, **2022**, *38(8)*, 1399C–1420.

 doi:10.20944/preprints202408.0929.v1

52. F. Zhang, M. Wei, Y. Li, et al. Special least squares solutions of the quaternion matrix equation $AXB + CXD = E$. *Comput. Math. Appl.*, **2016**, *72(5)*, 1426–1435.
53. X. Zhang. A system of generalized Sylvester quaternion matrix equations and its applications. *Appl. Math. Comput.*, **2016**, *273*, 74–81.
54. X.-F. Zhang, T. Li, Y.-G. Ou. Iterative solutions of generalized Sylvester quaternion tensor equations. *Linear Multilinear Algebra*, **2024**, *72(8)*, 1259–1278.
55. X.-F. Zhang, Q.-W. Wang. Developing iterative algorithms to solve Sylvester tensor equations. *Appl. Math. Comput.*, **2021**, *409*, 126403.
56. X.-F. Zhang, Q.-W. Wang. On RGI Algorithms for Solving Sylvester Tensor Equations. *Taiwan. J. Math.*, **2022**, *1(1)*, 1–19.