EXTENDED KRYLOV SUBSPACE METHODS FOR SOLVING SYLVESTER AND STEIN TENSOR EQUATIONS

ABDESEM HAFID BENTBIB
Laboratory LAMAI
University Cadi Ayyad
Marrakesh, Morocco

SMAHANE EL-HALOY* AND EL MOSTAFA SADEK
Laboratory LAMAI
University Cadi Ayyad
Marrakesh, Morocco
Laboratory LabSIPE, ENSA d’El Jadida
University Chouaib Doukkali
El Jadida, Morocco

Abstract. This paper deals with Sylvester and Stein tensor equations with low rank right hand sides. It proposes extended Krylov-like methods for solving Sylvester and Stein tensor equations. The expressions of residual norms are presented. To show the performance of the proposed approaches, some numerical experiments are given.

1. Introduction. In this paper we are interested in approximating the solutions of the following tensor equations

\[ \mathcal{X} \times_1 A^{(1)} + \mathcal{X} \times_2 A^{(2)} + \cdots + \mathcal{X} \times_N A^{(N)} = \mathcal{B}, \]

\[ \mathcal{Z} - \mathcal{Z} \times_1 A^{(1)} \times_2 A^{(2)} \cdots \times_N A^{(N)} = \mathcal{B}, \]

known, respectively as Sylvester and Stein tensor equations. The matrices \( A^{(n)} \in \mathbb{R}^{I_n \times I_n}, n = 1, 2, \ldots, N \), the right hand side tensor \( \mathcal{B} \in \mathbb{R}^{I_1 \times \cdots \times I_N} \) are known, and \( \mathcal{X}, \mathcal{Z} \in \mathbb{R}^{I_1 \times \cdots \times I_N} \) are the unknown tensors. The product \( \times_i, i = 1, \ldots, N \), and some notations related to the concept of tensors will be specified in the next section. For simplicity, in the sequel, we define the following linear operators

\[ \mathcal{M} : \mathbb{R}^{I_1 \times \cdots \times I_N} \rightarrow \mathbb{R}^{I_1 \times \cdots \times I_N}, \quad \mathcal{X} \mapsto \mathcal{M}(\mathcal{X}) := \sum_{i=1}^{N} \mathcal{X} \times_i A^{(i)}. \]

\[ \mathcal{L} : \mathbb{R}^{I_1 \times \cdots \times I_N} \rightarrow \mathbb{R}^{I_1 \times \cdots \times I_N}, \quad \mathcal{Z} \mapsto \mathcal{L}(\mathcal{Z}) := \mathcal{Z} - \mathcal{Z} \times_1 A^{(1)} \cdots \times_N A^{(N)}. \]

It is easy to verify that equations (1) and (2) are, respectively, equivalent to the following linear systems of equations

\[ \mathcal{A} \mathcal{X} = \mathcal{B}, \]

2020 Mathematics Subject Classification. 2020 MSC.

Key words and phrases. Sylvester tensor equation, Stein tensor equation, Extended Krylov subspace, Arnoldi process, High dimensional PDEs.

* Corresponding author: Smahane El-Halouy.
\[ Dz = b, \quad (6) \]

with \( A = \sum_{n=1}^{N} I_{n} \otimes \cdots \otimes I_{n+1} \otimes A^{(n)} \otimes I_{n-1} \otimes \cdots \otimes I_{1}, \ D = I_{N} \otimes \cdots \otimes I_{1} - A^{(N)} \otimes \cdots \otimes A^{(1)}, \ x = \text{vec}(X), \ z = \text{vec}(Z) \) and \( b = \text{vec}(B) \), where \( \otimes \) denotes the Kronecker product (defined in the next section), \( I_{n} \) stands for the identity matrix of order \( I_{n} \) and the operator \( \text{vec} \) rearranges tensor’s elements in a vector. Solving those linear systems can be a real challenge, since the associated matrices are too large. For \( N=2 \), (1) and (2) are respectively reduced to

\[ A^{(1)}X +XA^{(2)T} = B, \quad (7) \]

and

\[ Z - A^{(1)}ZA^{(2)T} = B, \quad (8) \]

known, respectively, as Sylvester and Stein matrix equations. (7) has been widely used in control and communication theory, image restoration and numerical methods for ordinary differential equations, see [6] and the references therein. (8) plays an important role in filtering theory for discrete-time large-scale dynamical systems and in many problems in control [6].

The Sylvester tensor equation (1) has a unique solution if and only if \( \lambda_{1} + \lambda_{2} + \cdots + \lambda_{N} \neq 0 \), for all \( \lambda_{i} \in \sigma(A^{(i)}) \), (Lemma 4.2 [7]), and the Stein tensor equation (2) has a unique solution if and only if \( \prod_{k=1}^{N} \lambda_{ik} \neq 1 \), for all \( \lambda_{ik} \in \sigma(A^{(k)}), k \in \{1, 2, \ldots, N\} \) (Lemma 4.1 [19]), where \( \sigma(A^{(i)}) \) denotes the spectrum of \( A^{(i)} \).

Several numerical algorithms have been presented in [16] to solve the discrete Lyapunov tensor equation, i.e., (2) when the coefficient matrices are the same. In [19] the biconjugate gradients (BiCG) and biconjugate residual (BiCR) methods have been extended to solve (2). For (1), various methods have been proposed, for instance, the tensor format of the GMRES method (GMRES-BTF) has been established by Chen and Lu [7]. In [8] a gradient based iterative algorithms have been proposed for solving (1) in [9]. Kressner and Tobler proposed a tensor Krylov subspace method to solve (5) when the right hand side is given in a tensor product structure, i.e., is of rank one. The idea based on applying a standard Krylov subspace method to the coefficient matrices, in order to approximate the solution by a vector of low tensor rank [13]. Ballani and Grasedyck presented an iterative scheme similar to Krylov subspace method to solve (5), relying on truncation operator, whereas the operator is implemented by hierarchical Tucker format [12], to allow applications in high dimensions [1]. Some well-known Krylov subspace methods have been studied in their tensor format by Beik et al. in [2]. Arnoldi-based methods to solve (1) with low rank right hand sides have been proposed in [4]. Many theoretical results such as the expressions of residuals have been generalized to the tensor case. In order to construct approximate solutions, the authors have applied an Arnoldi based algorithm by projecting the coefficient matrices on block matrix Krylov subspaces. However, these methods usually need many iterations to produce an accurate approximate solution, which increases the CPU time and the memory requirements. In this work, we project onto extended block and global Krylov subspaces generated by the matrices \( A^{(i)} \) and \( A^{(i)^{-1}}, i = 1, \ldots, N \), and we obtain low-dimensional equations that are solved by recursive blocked algorithms presented in [5]. Numerical examples show that projecting onto extended Krylov subspaces is more efficient than standard Krylov subspaces.

The rest of this paper is organized as follows. In section 2 we give notations adopted in this paper, some basic definitions and properties related to tensors. The
extended Arnoldi processes are described in Section 3. In section 4 we present our approaches to solve (1) and (2) with a right hand side tensor of a specific rank, both extended block and global Arnoldi processes are applied to (1) and (2). Some numerical examples are presented in section 5 to evaluate the performance of our methods. Finally, in section 6, we give a brief conclusion.

2. Preliminaries. In this section, we present some basic definitions, tensor notations and common operations related to tensors adopted throughout the current paper, for more details see [12]. Vectors (tensors of order one), matrices (tensors of order two) and higher-order tensors (order three or higher) are signified by lower-case letter, capital letters and Euler script letters respectively. A tensor \( X \) is an element in \( \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N} \), where \( I_1, I_2, \ldots, I_N \in \mathbb{N} \), its entries are denoted by \( X_{i_1 i_2 \cdots i_N} \), for every \( 1 \leq n \leq N \). The notation \( i_1 \ldots i_N \) corresponds to a multi-index, which is obtained as follows

\[
\overline{i_1 \ldots i_N} = i_N + (i_{N-1} - 1)I_N + \ldots + (i_1 - 1)I_2 \ldots I_N.
\]

As in MATLAB, we denote by \( X(I_1:J_1, \ldots, I_N:J_N) \) the intersection of fibers \( I_i \) to \( J_i \), \( i = 1, \ldots, N \) (we mean by fibers the higher order analogue of matrix rows and columns, they are obtained by fixing every index but one).

The inner product of two same size tensors \( A, B \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N} \) is defined by

\[
\langle A, B \rangle := \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \cdots \sum_{i_N=1}^{I_N} A_{i_1 \cdots i_N} B_{i_1 \cdots i_N},
\]

and the norm induced by this inner product is

\[
\| A \| = \sqrt{\langle A, A \rangle}.
\]

The \( n \)-mode multiplication commutes with respect to the inner product, that is for a matrix \( U \) of convenient dimensions we have

\[
\langle A \times_n U, B \rangle = \langle A \times_n U^T, B \rangle.
\]

Definition 2.1 ([12], [15]). Let \( A \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N} \) be an \( N \)-th-order tensor and \( U \in \mathbb{R}^{J_i \times I_i} \) be a matrix. The \( n \)-mode product of \( A \) and \( U \), denoted by \( A \times_n U \), is a tensor of size

\[
I_1 \times I_2 \times \cdots \times I_{n-1} \times J \times I_{n+1} \times \cdots \times I_N,
\]

whose entries are given by

\[
(A \times_n U)_{i_1 \cdots i_{n-1} j i_{n+1} \cdots i_N} = \sum_{i_n=1}^{I_n} A_{i_1 \cdots i_{n-1} i_n i_{n+1} \cdots i_N} U_{j i_n}.
\]

Proposition 1 ([12], [15]). Let \( A \in \mathbb{R}^{I_1 \times \cdots \times I_N} \) be an \( N \)-th order tensor, \( U \in \mathbb{R}^{J \times I_m} \), \( V \in \mathbb{R}^{K \times I_n} \) and \( W \in \mathbb{R}^{I_m \times I_n} \) be three matrices, then for distinct modes in a series of multiplication, the order of the multiplication is irrelevant, i.e.,

\[
A \times_m U \times_n V = A \times_n V \times_m U.
\]

If the modes are the same, then

\[
A \times_n W \times_n V = A \times_n VW.
\]
Definition 2.2 ([12]). The outer product of two tensors $A \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ and $B \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_M}$ is a tensor denoted by $A \circ B = C \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N \times J_1 \times J_2 \times \cdots \times J_M}$.

Elementwise,

$$C_{i_1 \ldots i_N j_1 \ldots j_M} = A_{i_1 \ldots i_N} B_{j_1 \ldots j_M}.$$ 

If $v_1, v_2, \ldots, v_N$ are $N$ vectors of sizes $I_i$, $i = 1, \ldots, N$, their outer product is an $N^{th}$-order tensor of size $I_1 \times \cdots \times I_N$ and we have

$$v_1 \circ \cdots \circ v_N = v_1(i_1) \cdots v_N(i_N).$$

Definition 2.3 ([12]). An $N^{th}$ order tensor $A \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is of rank one if it can be written as the outer product of $N$ vectors $v_k \in \mathbb{R}^{I_k}$, $k = 1, 2, \ldots, N$, i.e.,

$$A = v_1 \circ v_2 \circ \cdots \circ v_N.$$ 

A tensor is of rank $R \in \mathbb{N}$ if it could be written as the sum of $R$ rank one tensors.

Definition 2.4 ([12], [15]). The Kronecker product of two matrices $A \in \mathbb{R}^{I_1 \times I_2}$ and $B \in \mathbb{R}^{J_1 \times J_2}$ is a matrix of size $I_1 J_1 \times I_2 J_2$ denoted by $A \otimes B$, where

$$A \otimes B = \begin{pmatrix} a_{11} B & \cdots & a_{12} B \\ \vdots & \ddots & \vdots \\ a_{I_1 1} B & \cdots & a_{I_1 I_2} B \end{pmatrix}.$$ 

The Kronecker product of two tensors $A \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ and $B \in \mathbb{R}^{J_1 \times \cdots \times J_M}$ is defined by

$$C = A \otimes B \in \mathbb{R}^{I_1 J_1 \times \cdots \times I_N J_M},$$

where

$$C_{i_1 \ldots i_N j_1 \ldots j_M} = A_{i_1 \ldots i_N} B_{j_1 \ldots j_M},$$

for $i_n = 1, \ldots, I_n$, $j_n = 1, \ldots, J_n$, $n = 1, \ldots, N$.

Proposition 2 ([15]). Let $A = a_1 \circ a_2 \circ \cdots \circ a_N$ and $B = b_1 \circ b_2 \circ \cdots \circ b_N$ denote two rank one tensors, then, the Kronecker product $A \otimes B$ can be expressed by

$$A \otimes B = (a_1 \otimes b_1) \circ \cdots \circ (a_N \otimes b_N).$$

It is well known that if $A, B, C$ and $D$ are four matrices of convenient dimensions, we have

$$(A \otimes B)(C \otimes D) = (AC) \otimes (BD).$$

An elegant generalization of this property to tensors is given as follow

Proposition 3 ([4]). Let $A \in \mathbb{R}^{I_1 \times \cdots \times I_N}$, $B \in \mathbb{R}^{J_1 \times \cdots \times J_M}$ be two tensors, and $A \in \mathbb{R}^{K_1 \times \cdots \times K_n}$ and $B \in \mathbb{R}^{L_1 \times \cdots \times L_m}$ be two matrices, then

$$(A \otimes B) \times_n (A \otimes B) = (A \times_n A) \otimes (B \times_n B).$$

It is easy to verify the following result

Proposition 4. Let $A = a_1 \circ a_2 \circ \cdots \circ a_N$ a rank one tensor and $N$ matrices $V_i$, $i = 1, \ldots, N$, then we have

$$A \times_1 V_1 \cdots \times_N V_N = V_1 a_1 \circ \cdots \circ V_N a_N.$$ 

3. Extended Arnoldi processes. We review in the following the extended block and global processes. We give the algorithms that described those processes as well.
3.1. Extended block Arnoldi process. The extended block Arnoldi (EBA) algorithm [10, 11, 18] constructs an orthonormal basis \( V_m = [V_1, \ldots, V_m] \) of the extended block Krylov subspace associated with the pair \((A, V)\):

\[
\mathbb{K}^e_m(A, V) := \text{Range} \{ V, A^{-1}V, AV, A^{-2}V, A^2V, \ldots, A^{m-1}V, A^{-m}V \}.
\]

Since \( \mathbb{K}^e_m(A, V) \) is enriched by \( A^{-1} \), it contains more information than the classical subspace \( \mathbb{K}^e_m(A, V) := \text{Range} \{ V, AV, A^2V, \ldots, A^{m-1}V \} \). The extended block Arnoldi process is summarized in algorithm 1.

**Algorithm 1 Extended Block Arnoldi (EBA)**

1: Input: \( n \times n \) matrix \( A \), \( n \times r \) matrix \( B \) and an integer \( m \).
2: Compute the QR decomposition of \([B, A^{-1}B] = U_1\Lambda, U_0 = [] \) for \( j = 1, \ldots, m \).
3: Set \( U_j^{(1)} := \text{first } r \text{ columns of } U_j, U_j^{(2)} := \text{second } r \text{ columns of } U_j \).
4: Orthogonalize \( W \) w.r. to \( U_j^{(2)} \) to get \( U_j^{(1)} \) i.e.,
5: for \( i = 1, \ldots, j \),
6: Compute the QR decomposition of \( W = U_{j+1} H_{i,j} \).

The restriction of the matrix \( A \) to the extended block Krylov subspace \( \mathbb{K}^e_m(A, V) \) is the matrix \( T_m = V_m^T A V_m \). It can be computed from the upper block Hessenberg matrix \( H_m = [H_{i,j}] \) generated by EBA algorithm, without requiring additional matrix-vector products with \( A \). Let \( V_{m+1} \) be the matrix generated by EBA algorithm and \( T_m = [T_m^T E_m T_{m+1,m}^T] \), where \( E_m \) is the \( 2mr \times 2r \) matrix corresponding to the last \( 2r \) columns of the identity matrix \( I_{2mr} \). Then the following relation is satisfied [10]

\[
A V_m = V_{m+1} T_m = V_m T_m + V_{m+1} T_{m+1,m} E_m^T.
\]

3.2. Extended global Arnoldi process. As in the block case, the classical Krylov subspace is enriched by \( A^{-1} \) to get

\[
\mathbb{K}^e_m(A, W) := \text{span} \{ V, A^{-1}V, AV, A^{-2}V, A^2V, \ldots, A^{m-1}V, A^{-m}V \}.
\]

The extended global Arnoldi (EGA) process built an orthonormal basis \( W_m = [W_1, \ldots, W_m] \) of \( \mathbb{K}^e_m(A, W) \), where the first block \( W_1 \) can be obtained by the global QR decomposition of \([W_1, A^{-1}W_1] \). The extended global Arnoldi process is summarized in algorithm 2. Let \( T_m = [T_m^T E_m T_{m+1,m}^T] \), where \( T_m \) is the restriction of the matrix \( A \) to the extended global Krylov subspace \( \mathbb{K}^e_m(A, W) \), \( E_m \) is the \( 2m \times 2 \) matrix corresponding to the last \( 2 \) columns of the identity matrix \( I_{2m} \) and \( W_{m+1} \) be the basis generated by EGA algorithm, we have

\[
A W_m = W_{m+1} (T_m \otimes I_r) = W_m (T_m \otimes I_r) + W_{m+1} (T_{m+1,m} E_m^T \otimes I_r).
\]
For more details about the EBA and EGA processes and the construction of the restriction matrices $T_m$ and $T_m$, we refer the reader to [11, 3, 18] and the references therein.

4. **The EBA and EGA processes for Sylvester and Stein tensor equations.**

In this section, we show how to extract approximate solutions to (1) and (2) with low rank right hand side. We assume in the following that the right hand side is of rank $R$, i.e.,

$$B = \sum_{r=1}^{R} b_i^{(r)} \circ \cdots \circ b_N^{(r)},$$

where $b_i^{(r)} \in \mathbb{R}^{l_i}$, for $i \in \{1, \ldots, N\}$, and $r \in \{1, \ldots, R\}$. We set for $i = 1, 2, \ldots, N$

$$B^{(i)} = \left[ b_i^{(1)}, b_i^{(2)}, \ldots, b_i^{(R)} \right].$$

Straightforward computations show that the right hand side tensor can also be written as follow

$$B = I_{R} \times_{1} B^{(1)} \cdots \times_{N} B^{(N)},$$

where $I_R$, called identity tensor, is the $N^{th}$ order tensor of size $R \times \ldots \times R$ with ones along the super-diagonal.

4.1. **Sylvester tensor EBA process.** We seek approximate solutions of the form

$$X_m = \mathcal{Y}_m \times_1 V_{m_1} \times_2 \cdots \times_N V_{m_N},$$

where $\mathcal{Y}_m \in \mathbb{R}^{2R_{m_1} \times \ldots \times 2R_{m_N}}$ and $V_{m_i} = \left[ V_{1}^{(i)}, \ldots, V_{m_i}^{(i)} \right]$, $i = 1, \ldots, N$, are the matrices obtained by applying simultaneously $m_i$ steps of the EBA algorithm to the pairs $(A^{(i)}, B^{(i)})$, $i = 1, 2, \ldots, N$, then the following relation holds, for $i = 1, \ldots, N$,

$$A^{(i)} V_{m_i} = V_{m_i+1} \Gamma_{m_i} = V_{m_i} \Gamma_{m_i+1} + V_{m_i+1} \Gamma_{m_i+1, m_i} E_{v_{m_i}}^T.$$

(9)

The matrices $\Gamma_{m_i}$, $i = 1, \ldots, N$, are the restrictions of the matrices $A^{(i)}$, $i = 1, \ldots, N$, to the extended Krylov subspaces $K_{m_i}(A^{(i)}, B^{(i)})$, $i = 1, \ldots, N$. The first blocks of $V_{m_i}$, $i = 1, \ldots, N$, are given by the QR factorization (step 2 in algorithm 1) of

$$B^{(i)} A^{(i)} B^{(i)} = V_{1}^{(i)} \Gamma^{(i)}, i = 1, \ldots, N,$$

where $\Gamma^{(i)} = \left( \gamma_{k,l}^{(i)} \right)_{k,l=1,2}$, $i = 1, \ldots, N$, are $2R \times 2R$ upper triangular matrices, and $\gamma_{k,l}^{(i)} \in \mathbb{R}^{R \times R}$. These QR decompositions lead to the following relation, for $i = 1, \ldots, N$:

$$V_{m_i}^T B^{(i)} = E_{v_{m_i}}^T \gamma_{i11}^{(i)} = \tilde{z}_{i11}^{(i)},$$

(10)

where $E_{v_{m_i}}^T$ is the $2m_i R \times R$ matrix corresponding to the first $R$ columns of the identity matrix $I_{2m_i R}$.

The tensor $\mathcal{Y}_m$ is determined by imposing the following Petrov-Galerkin condition of orthogonality in its tensor format:

$$\mathcal{R}_m \times_1 V_{m_1}^T \times_2 \cdots \times_N V_{m_N}^T = 0,$$

(11)

where $\mathcal{R}_m = B - \mathcal{M}(\mathcal{X}_m)$ is the residual tensor associated to $\mathcal{X}_m$. Using the relation (9) and the condition (11) we obtain

$$0 = B \times_1 V_{m_1}^T \times_2 \cdots \times_N V_{m_N}^T - \sum_{i=1}^{N} \mathcal{Y}_m \times_i T_{m_i}.$$

Using the expression of the right hand side tensor and the relation (10) we have
\[
\mathcal{B} \times_1 \mathcal{V}_{m_1}^T \times_N \mathcal{V}_{m_N}^T = I_{R} \times_1 \mathcal{V}_{m_1}^T \mathcal{B}^{(1)} \times_N \mathcal{V}_{m_N}^T \mathcal{B}^{(N)}
\]

\[
= I_{R} \times_1 \mathcal{V}_{m_1}^T \mathcal{Z}_{(N)}^{(1)} \times_N \mathcal{V}_{m_N}^T \mathcal{Z}_{(N)}^{(N)}
\]

\[
:= \mathcal{B}_m
\]

Then \( \mathcal{Y}_m \) is the solution of the following low dimensional Sylvester tensor equation

\[
\sum_{i=1}^{N} \mathcal{Y}_m \times_i \mathcal{T}_{m_i} = \mathcal{B}_m.
\]  

(12)

Now we establish the link between the right hand sides of (1) and (12). According to the QR factorization computed in step 2 of algorithm 1, we have

\[
\mathcal{B}_i = \mathcal{V}_{m_i}^{(1)} \mathcal{Z}_{(N)}^{(1)}(i), \quad i = 1, \ldots, N,
\]

where \( \mathcal{V}_{m_i}^{(1)} \) denotes the first \( R \) columns of \( \mathcal{V}_{m_i} \), i.e., \( \mathcal{V}_{m_i}^{(1)} = \mathcal{V}_{m_i} \mathcal{E}_{m_i}^{(1)} \), it follows then

\[
\mathcal{B} = I_{R} \times_1 \mathcal{B}^{(1)} \times_N \mathcal{B}^{(N)}
\]

\[
= I_{R} \times_1 \mathcal{V}_{m_1} \mathcal{E}_{m_1}^{(1)} \mathcal{Z}_{(N)}^{(1)}(1) \times_N \mathcal{V}_{m_N} \mathcal{E}_{m_N}^{(N)} \mathcal{Z}_{(N)}^{(N)}(N)
\]

\[
= I_{R} \times_1 \mathcal{V}_{m_1} \mathcal{Z}_{(N)}^{(1)} \times_N \mathcal{V}_{m_N} \mathcal{Z}_{(N)}^{(N)}
\]

\[
= \mathcal{B}_m \times_1 \mathcal{V}_{m_1} \ldots \times_N \mathcal{V}_{m_N}.
\]

The following result provides an inexpensive way to compute the norm of the residual tensor \( \mathcal{R}_m \).

**Theorem 4.1.** Let \( \mathcal{X}_m = \mathcal{Y}_m \times_1 \mathcal{V}_{m_1} \times_N \mathcal{V}_{m_N} \) be the approximate solution obtained after applying the Sylvester EBA process, where \( \mathcal{Y}_m \) is the exact solution of (12). Then the norm of the residual tensor \( \mathcal{R}_m \) associated to \( \mathcal{X}_m \) is given by

\[
\| \mathcal{R}_m \| = \left( \frac{1}{2} \sum_{i=1}^{N} \| \mathcal{Y}_m \times_i \mathcal{T}_{m_i+1,m_i} \mathcal{E}_{m_i}^T \|_2^2 \right)^{1/2}.
\]

**Proof.** We have

\[
\mathcal{R}_m = \mathcal{B} - \mathcal{M}(\mathcal{X}_m)
\]

\[
= \mathcal{B} - \sum_{i=1}^{N} \mathcal{Y}_m \times_1 \mathcal{V}_{m_1} \ldots \times_i \mathcal{A}^{(i)} \mathcal{V}_{m_i} \ldots \times_N \mathcal{V}_{m_N}.
\]

Using the expression of the right hand side tensor \( \mathcal{B} \) and the relation (9), we obtain

\[
\mathcal{R}_m = \left( \mathcal{B}_m - \sum_{i=1}^{N} \mathcal{Y}_m \times_1 \mathcal{V}_{m_1} \ldots \times_i \mathcal{A}^{(i)} \mathcal{V}_{m_i} \ldots \times_N \mathcal{V}_{m_N} \right)
\]

\[
\times_1 \mathcal{V}_{m_1} \ldots \times_N \mathcal{V}_{m_N}
\]

Taking in consideration the fact that \( \mathcal{V}_{m_i}, i = 1, \ldots, N \), are orthonormal, we compute the norm \( \| \mathcal{R}_m \|^2 = \langle \mathcal{R}_m, \mathcal{R}_m \rangle \), then the result achieved. \( \square \)
4.2. Stein tensor EBA process. In this section, we extract approximate solutions to (2) by applying the EBA process in the same way as in section 4.1. We look for approximate solutions of the form
\[ X_m = Z_m \times_1 V_{m_1} \ldots \times_N V_{m_N}, \]
where \( Z_m \in \mathbb{R}^{2R_{m_1} \times \ldots \times 2R_{m_N}} \) is the exact solution of the following low dimensional Stein tensor equation
\[ Z_m - Z_m \times_1 T_{m_1} \ldots \times_N T_{m_N} = B_m, \tag{13} \]
which is obtained in the same way as in the previous section. The associated residual tensor to \( X_m \) is given by
\[ Q_m = B - L(X_m) = B - Z_m \times_1 V_{m_1} \ldots \times_N V_{m_N} + Z_m \times_1 A^{(1)} V_{m_1} \ldots \times_N A^{(N)} V_{m_N}. \]
The following result provides an inexpensive way to compute the residual norm

**Theorem 4.2.** Let \( X_m = Z_m \times_1 V_{m_1} \ldots \times_N V_{m_N} \) be the approximate solution obtained after applying the Stein EBA process, where \( Z_m \) is the exact solution of (13). Then the norm of the residual tensor \( Q_m \) associated to \( X_m \) is given by
\[ \| Q_m \| = \left( \alpha_1^2 + \sum_{i=1}^{\alpha_2} (\alpha_2^{(i)})^2 + \sum_{i=1}^N (\alpha_3^{(i)})^2 \right)^{1/2}, \]
where
\[ \alpha_1 = \| Z_m \times_1 T_{m_1+1,m_1} E_{m_1}^T \ldots \times_N T_{m_N+1,m_N} E_{m_N}^T \|, \]
\[ \alpha_2^{(i)} = \| Z_m \times_1 T_{m_1} \ldots \times_i T_{m_1+1,m_i} E_{m_i}^T \ldots \times_N T_{m_N} \| \]
and
\[ \alpha_3^{(i)} = \| Z_m \times_1 T_{m_1+1,m_i} E_{m_i}^T \ldots \times_i T_{m_1} \ldots \times_n T_{m_N+1,m_N} E_{m_N}^T \|. \]

**Proof.** We pursue in the same way as in the previous section. \( \square \)

The Sylvester and Stein EBA process is summarized in the following algorithm

**Algorithm 3** Sylvester / Stein EBA Process

1. **Input:** Coefficient matrices \( A^{(i)}, i = 1, \ldots, N \), and the right hand side in low rank representation \( B = [B^{(1)}, \ldots, B^{(N)}] \).
2. **Output:** Approximate solutions, \( X_m \), to the equations (1) / (2).
3. Choose a tolerance \( \epsilon > 0 \), a maximum number of iterations \( \text{itermax} \), a step-size parameter \( k \), set for \( i = 1, \ldots, N, m_i = k \).
4. For \( m_i = k \), construct the orthonormal basis \( V_{m_i} \) and the restriction matrices \( T_{m_i} \) by EBA algorithm (1).
5. Compute \( Y_m / Z_m \) the solution of the low dimensional equation (12) / (13).
6. Compute the residual norms \( r_m = \| R_m \| \) as in theorem (4.1) / \( q_m = \| Q_m \| \) as in theorem (4.2).
7. If \( r_m \geq \epsilon / q_m \geq \epsilon \), set for \( i = 1, \ldots, N, m_i = m_i + k \) and go to step 2.
8. The approximate solutions are given by \( X_m = Y_m \times_1 V_{m_1} \ldots \times_N V_{m_N} / X_m = Z_m \times_1 V_{m_1} \ldots \times_N V_{m_N} \).

4.3. Sylvester tensor EGA process. In this section, we apply algorithm 2 to the pairs \( (A^{(i)}, B^{(i)}), i = 1, \ldots, N \), in order to construct the extended Krylov subspaces \( W_{m_i}, i = 1, \ldots, N \), and the restrictions \( T_{m_i} \) of the matrices \( A^{(i)}, i = 1, \ldots, N \). We have then
\[ A^{(i)} W_{m_i} = W_{m_i+1} (\hat{T}_{m_i} \otimes I_R) = W_{m_i} (T_{m_i+1} \otimes I_R) + W_{m_i+1,m_i} (T_{m_i+1,m_i} E_{m_i}^T \otimes I_R), \tag{14} \]
Notice that $W_i^{(i)}$, $i = 1, \ldots, N$, the first blocks of the matrices $\mathbb{W}_{m_i}$, $i = 1, \ldots, N$, are obtained by computing the global QR decomposition of $[B^{(i)} A^{(i)-1} B^{(i)}] = W_i^{(i)} (\Omega^{(i)} \otimes I_R)$, $i = 1, \ldots, N$, where $\Omega^{(i)} = [\omega_{k,l}]_{k,l=1,2}$. We have then for $i = 1, \ldots, N$,

$$B^{(i)} = \omega_{11} W_i^{(i)(1)} I_R,$$

(15)

where $W_i^{(i)(1)}$ denotes the first $R$ columns of $\mathbb{W}_{m_i}$, $i = 1, \ldots, N$.

We look for approximate solutions to (1) of the following form

$$X_m = (Y_m \otimes I_R) \times_1 W_{m_1} \times_2 \cdots \times_N W_{m_N},$$

where $Y_m \in \mathbb{R}^{2m_1 \times \cdots \times 2m_N}$ is the exact solution of the following low dimensional Sylvester tensor equation

$$\sum_{i=1}^N Y_m \times_i T_{m_i} = \omega E_m,$$

(16)

with $\omega = \prod_{i=1}^N \omega_{i(1)}^{(1)}$ and $E_m = e_1^{(2m_1)} \circ \cdots \circ e_1^{(2m_N)}$, where $e_1^{(2m_i)} = [1, 0, \ldots, 0]^T \in \mathbb{R}^{2m_i}$, $i = 1, \ldots, N$.

Next, we give an upper bound for the norm of the residual tensor, which will be used as a stopping criterion in the Sylvester EGA algorithm without computing the whole residual tensor at each iteration. We first recall the following lemma [4] to be used later

**Lemma 4.3.** Let $\mathbb{W}_{m_i}$ be the basis generated by the EGA algorithm applied to the pairs $(A^{(i)}, B^{(i)})$ and $X \in \mathbb{R}^{J_1 \times \cdots \times J_N}$ with $J_i = 2Rm_i$ and $Z \in \mathbb{R}^{K_1 \times \cdots \times K_N}$ with $K_i = 2m_i$. Then

$$\|X \times_i \mathbb{V}_{m_i}\| \leq \|X\|,$$

(17)

$$\|(Z \otimes I_R) \times_i \mathbb{V}_{m_i}\| = \|Z\|.$$

(18)

In the following theorem, we give an upper bound for the residual norm

**Theorem 4.4.** Let $X_m$ be the approximate solution obtained by applying the Sylvester EGA process and let $R_m = B - \mathcal{M}(X_m)$ be the corresponding residual tensor. Then

$$\|R_m\| \leq \left( \sum_{i=1}^N \|Y_m \times_i T_{m_i+1,m_i} E_{m_i}^T\|^2 \right)^{1/2}.$$

**Proof.** We first establish the link between the right hand sides of (1) and (16). Using the relation (15) we obtain

$$B = \sum_{r=1}^R [b_r^{(1)} \circ \cdots \circ b_N^{(r)}]$$

$$= \sum_{r=1}^R \omega_{11}^{(1)} W_i^{(1)(1)} (;, r) \circ \cdots \circ \omega_{11}^{(N)} W_i^{(N)(1)} (;, r)$$

$$= \sum_{r=1}^R \omega_{11}^{(1)} \mathbb{W}_{m_1} e_r^{(2Rm_1)} \circ \cdots \circ \omega_{11}^{(N)} \mathbb{W}_{m_N} e_r^{(2Rm_N)}.$$

Using the proposition (4), the fact that $e_r^{(2Rm_1)} = e_1^{(2m_1)} \otimes e_r^{(R)}$ and the proposition (2), we obtain

$$B = \omega E_m \otimes I_R \times_1 \mathbb{W}_{m_1} \times_2 \cdots \times_N \mathbb{W}_{m_N}.$$
We have
\[
\mathcal{R}_m = \mathcal{B} - \mathcal{M}(\mathcal{X}_m)
\]
\[
= \mathcal{B} - \sum_{i=1}^{N} (\mathcal{Y}_m \otimes \mathcal{I}_R) \times_1 \mathbb{W}_{m_1} \times_2 \ldots \times_i A^{(i)} \mathbb{W}_{m_i} \times_{i+1} \ldots \times_N \mathbb{W}_{m_N},
\]

Invoking the relation (14) and using the proposition (3), we obtain
\[
\mathcal{R}_m = \omega \mathcal{E}_m \otimes \mathcal{I}_R \times_1 \mathbb{W}_{m_1} \times_N \mathbb{W}_{m_N} \sum_{i=1}^{N} \mathcal{Y}_m \times_i \tilde{\mathcal{T}}_{m_i} \otimes \mathcal{I}_R \times_1 \mathbb{W}_{m_1} \times_2 \ldots \times_i \mathbb{W}_{m_i+1} \times_{i+1} \ldots \times_N \mathbb{W}_{m_N}.
\]

Now, let \(Z^{(i)}_m \in \mathbb{R}^{2(m_1+1) \times \ldots \times 2(m_N+1)}\), \(i = 1, \ldots, N\), be the tensor defined by
\[
\begin{align*}
Z^{(i)}_m(1:2m_1, \ldots, 1:2m_i-1, 2m_i+1:2(m_i+1), 1:2m_{i+1}, \ldots, 1:2m_N) &= \mathcal{Y}_m \times_i T_{m_i}, \\
Z^{(i)}_m(2Rm_1+1:2R(m_1+1), \ldots, 1:2R(m_i+1), 2Rm_N+1:2R(m_N+1)) &= 0,
\end{align*}
\]
and we complete the tensor \(\tilde{\mathcal{E}}_m\) by zeros in order to obtain a tensor \(\tilde{\mathcal{E}}_m \in \mathbb{R}^{2(m_1+1) \times \ldots \times 2(m_N+1)}\), then
\[
\mathcal{R}_m = \omega \tilde{\mathcal{E}}_m \otimes \mathcal{I}_R \times_1 \mathbb{W}_{m_1+1} \times_N \mathbb{W}_{m_N+1} - \sum_{i=1}^{N} Z^{(i)}_m \otimes \mathcal{I}_R \times_1 \mathbb{W}_{m_1+1} \times_2 \ldots \times_i \mathbb{W}_{m_i+1} \times_{i+1} \ldots \times_N \mathbb{W}_{m_N+1}
\]
where \(\mathcal{R}^{(i)}_m = \omega \tilde{\mathcal{E}}_m - \sum_{i=1}^{N} Z^{(i)}_m\).

By applying the relation (17) of lemma 4.3 \((N-1)\) times to \(\|\mathcal{R}_m\|\) we obtain
\[
\|\mathcal{R}_m\|^2 \leq \|\mathcal{R}^{(0)}_m \otimes \mathcal{I}_R \times_1 \mathbb{W}_{m_1+1}\|^2.
\]
Then the relation (18) of lemma 4.3 leads to
\[
\|\mathcal{R}_m\|^2 \leq \|\mathcal{R}^{(0)}_m\|^2.
\]
Notice that by the construction of the tensor \(\mathcal{R}^{(i)}_m\), we have
\[
\|\mathcal{R}^{(i)}_m\|^2 \leq \sum_{i=1}^{N} \|\mathcal{Y}_m \times_i T_{m_i+1, m_i} E_{m_i}^T\|^2.
\]
\[\square\]

4.4. Stein tensor EGA process. We are interested in applying algorithm 2 to the coefficient matrices in (2), in order to extract approximate solutions to (2) in the same way as in section 4.1. We look for approximate solutions of the form
\[
\mathcal{X}_m = (\mathcal{Z}_m \otimes \mathcal{I}_R) \times_1 \mathbb{W}_{m_1} \times_2 \ldots \times_N \mathbb{W}_{m_N},
\]
where \(\mathcal{Z}_m \in \mathbb{R}^{2m_1 \times \ldots \times 2m_N}\) is the exact solution of the following low dimensional Stein tensor equation
\[
\mathcal{Z}_m - \mathcal{Z}_m \times_1 T_{m_1} \times_2 \ldots \times_N T_{m_N} = \omega \mathcal{E}_m,
\]
where \(\omega = \prod_{i=1}^{N} \omega^{(i)}_{11}\) and \(\mathcal{E}_m = e^{(2m_1)}_1 \circ \ldots \circ e^{(2m_N)}_1\).

The associated residual tensor associated to \(\mathcal{X}_m\) is given by
\[
\mathcal{Q}_m = \mathcal{B} - \mathcal{L}(\mathcal{X}_m)
\]
\[
= \mathcal{B} - (\mathcal{Z}_m \otimes \mathcal{I}_R) \times_1 \mathbb{W}_{m_1} \times_2 \ldots \times_N \mathbb{W}_{m_N}
\]
\[
+ (\mathcal{Z}_m \otimes \mathcal{I}_R) \times_1 A^{(1)} \mathbb{W}_{m_1} \times_2 \ldots \times_N A^{(N)} \mathbb{W}_{m_N}
\]
In the following result, an upper bound of the residual norm is given.

**Theorem 4.5.** Let $X_m$ be the approximate solution obtained by applying the Stein EGA process and let $Q_m = B - L(X_m)$ be the corresponding residual tensor. Then

$$||Q_m|| \leq \left( \beta_1^2 + \sum_{i=1}^{N} \beta_2^{(i)2} + \sum_{i=1}^{N} \beta_3^{(i)2} \right)^{1/2},$$

where $\beta_1 = \|Z_m \times_1 T_m_{1\scriptscriptstyle +1,m_1}E^T \times \cdots \times_N T_m_{N+1,m_N}E^T\|$, $\beta_2^{(i)} = \|Z_m \times_1 T_m{1\scriptscriptstyle +1,m_1}E^T_{1\scriptscriptstyle +1,m_1} \times_i T_m{1\scriptscriptstyle +1,m_1}E^T_{1\scriptscriptstyle +1,m_1} \times \cdots \times_i T_m{1\scriptscriptstyle +1,m_1}E^T_{1\scriptscriptstyle +1,m_1} \times N T_m_{N+1,m_N}E^T\|$, and $\beta_3^{(i)} = \|Z_m \times_1 T_m{1\scriptscriptstyle +1,m_1}E^T_{1\scriptscriptstyle +1,m_1} \times_i T_m{1\scriptscriptstyle +1,m_1}E^T_{1\scriptscriptstyle +1,m_1} \times \cdots \times_i T_m{1\scriptscriptstyle +1,m_1}E^T_{1\scriptscriptstyle +1,m_1} \times N T_m_{N+1,m_N}E^T\|$. 

**Proof.** By reasoning in the same way as in the previous section, we obtain

$$Q_m = \omega E_m \otimes I_R \times_1 W_{m_1} \times \cdots \times_N W_{m_N} - Z_m \otimes I_R \times_1 W_{m_1} \times \cdots \times_N W_{m_N}$$

$$+ (Z_m \times_1 \tilde{T}_{m_1} \times \cdots \times_N \tilde{T}_{m_N}) \otimes I_R \times_1 W_{m_1+1} \times \cdots \times_N W_{m_N+1}.$$

We complete the tensors $E_m$ and $Z_m$ by zeros in order to obtain tensors $\tilde{E}_m$ and $\tilde{Z}_m \in \mathbb{R}^{2(m_1+1) \times \cdots \times 2(m_N+1)}$, respectively. Then we have

$$Q_m = Q_m^{(0)} \otimes I_R \times_1 W_{m_1+1} \times \cdots \times_N W_{m_N+1},$$

where $Q_m^{(0)} = \omega E_m - \tilde{Z}_m + (Z_m \times_1 \tilde{T}_{m_1} \times \cdots \times_N \tilde{T}_{m_N})$. By applying lemma (4.3) to $Q_m$ and invoking (19), the result achieved.

The Sylvester and Stein EGA process is summarized in the following algorithm.

**Algorithm 4 Sylvester / Stein EGA Process**

1: **Input:** Coefficient matrices $A^{(i)}$, $i = 1, \ldots, N$, and the right hand side in low rank representation $B = [B^{(1)}, \ldots, B^{(N)}]$.

2: **Output:** Approximate solutions, $X_m$, to the equations (1) / (2).

3: Choose a tolerance $\epsilon > 0$, a maximum number of iterations $\text{itermax}$, a step-size parameter $k$, set for $i = 1, \ldots, N$, $m_i = k$.

4: For $i = 1, \ldots, N$ do
   
5: Compute $Y_m \otimes Z_m$ the solutions of the low dimensional equations (16) / (19).

6: Compute the upper bound for the residual norms $r_m = \left(\sum_{i=1}^{N} \|Y_m \times_i T_{m_{1\scriptscriptstyle +1},m_1}E_{1\scriptscriptstyle +1,m_1}^T\|^2\right)^{1/2}$ as in theorem (4.4) / $q_m = \left(\beta_1^2 + \sum_{i=1}^{N} \beta_2^{(i)2} + \sum_{i=1}^{N} \beta_3^{(i)2}\right)^{1/2}$ as in theorem (4.5).

7: If $r_m \geq \epsilon / q_m \geq \epsilon$, set for $i = 1, \ldots, N$, $m_i = m_i + k$ and go to step 2.

8: The approximate solutions are given by $X_m = (Y_m \otimes I_R) \times_1 W_{m_1} \times \cdots \times_N W_{m_N} / X_m = (Z_m \otimes I_R) \times_1 W_{m_1} \times \cdots \times_N W_{m_N}$.

5. **Numerical examples.** In this section, we present four numerical examples to show the effectiveness of our methods for solving (1) and (2). The low dimensional tensor equations (12), (13), (16) and (19) will be solved by the recursive blocked algorithms presented in [5]. The numerical results were performed on a 2.8 GHz Intel Core i5 and 4 Go of RAM with Matlab R2016a.

In all examples, the right hand side tensor is either constructed randomly or constructed so that the exact solution $X^*$ is a tensor where all of its elements are equal to one. Note that each cycle corresponds to a step size parameter $k = 5$ iterations in all examples. The used stopping criterion is $||R_m|| < \epsilon$, and $||Q_m|| < \epsilon$, where $\epsilon$ is a given tolerance, $R_m$ and $Q_m$ are the $m^{th}$ residuals associated to
the approximated solutions $X_m$. Notice that the larger CPU time needed for the extended block process is caused by the computational expenses of the extended block Arnoldi algorithm and to the computation of the solution of the reduced tensor equation of order $mR \times mR \times mR$ for increasing $m$.

Example 1. We consider the following 3-dimensional Poisson equation

$$-\Delta u = f \text{ in } \Omega, \quad u/\partial \Omega = 0,$$

where $\Omega = [0, 1]^3$ is the 3-dimensional hyper-cube, and with sufficiently smooth right-hand side $f$ to be well approximate by a short sum of separable functions [1]. A standard finite-difference discretization on equidistant nodes leads to (1) with the following coefficient matrices

$$A(i) = \frac{1}{h^2} \left( \begin{array}{ccc} 2 & -1 & \cdot \cdot \cdot \\ -1 & 2 & -1 \\ \cdot \cdot \cdot & -1 & 2 \\ \cdot \cdot \cdot & \cdot \cdot \cdot & \cdot \cdot \cdot & -1 \\ 2 & 2 & \cdot \cdot \cdot \\ \end{array} \right), \quad i = 1, 2, 3,$$

and a rank $R$ tensor as the the right hand side. For simplicity, we construct the right hand side of uniformly distributed pseudorandom numbers.

Example 2. In this example, the coefficient matrices of Sylvester tensor equation (1) are obtained by standard finite difference discretization of the following convection diffusion equation on a uniform grid

$$-\nu \Delta u + c^T \nabla u = f \text{ in } \Omega, \quad u/\partial \Omega = 0,$$

where $\Omega = [0, 1]^3$ is the 3-dimensional hyper-cube, and $h = 1/n + 1$,

$$A(i) = \frac{\nu}{h^2} \left( \begin{array}{ccc} 2 & -1 & -1 \\ -1 & 2 & -1 \\ \cdot \cdot \cdot & -1 & 2 \\ \cdot \cdot \cdot & \cdot \cdot \cdot & -1 & 2 \\ \end{array} \right) + \frac{c_i}{4h} \left( \begin{array}{ccc} 3 & -5 & 1 \\ 1 & 3 & -5 \\ \cdot \cdot \cdot & \cdot \cdot \cdot & 1 \\ \cdot \cdot \cdot & \cdot \cdot \cdot & 1 \\ \end{array} \right), \quad i = 1, 2, 3,$$

For $\nu = 1$ and $c_1 = c_2 = c_3 = 1$, (case 1) and for $\nu = 10$ and $c_1 = 1, c_2 = 2$ and $c_3 = 3$, (case 2), we construct the right hand side so that the exact solution of (1) is $X^* = \text{ones}(n, n, n)$, which is a tensor of rank 1. Notice in this case that the right hand side is of rank 3. We run an example where the rank $R = 5$, where the right hand side is constructed randomly.
Example 2. The methods presented in the current paper are more efficient comparing to the ones presented in [4]. Figures 1 and 2 show the efficiency of the proposed methods in term of convergence speed, table 3 shows that the Sylvester EGA process is more efficient in terms of precision of the approximate solution.

Example 3. Now we use the same examples as in [4] (example 2 and 3). The coefficient matrices \( A^{(i)}, i = 1, 2, 3 \), are generated using the following Matlab command

\[
A^{(i)} = \text{gallery}('poisson', n_0),
\]

and the other set of coefficient matrices \( A^{(i)}, i = 1, 2, 3 \), are extracted from the Lyapack package [17], using the command \text{fdm}, denoted as

\[
A^{(i)} = \text{fdm}(n_0, f_1(x, y), f_2(x, y), g(x, y)),
\]

with \( f_1(x, y) = e^{xy}, \ f_2(x, y) = \sin(xy), \ g(x, y) = y^2 - x^2 \). For both sets, the right hand sides are constructed randomly. We point out that this example shows that the methods presented in the current paper are more efficient comparing to the ones presented in [4]. Figures 1 and 2 show the efficiency of the proposed methods in term of convergence speed, table 3 shows that the Sylvester EGA process is more efficient in terms of precision of the approximate solution.

Example 4. We first give an example of an application of (2). We consider the following PDE

\[
u u - \nu \Delta u + c^T \nabla u = f \quad \text{in} \ \Omega, \quad u/\partial \Omega = 0,
\]

where \( \Omega = [0, 1]^3 \). A standard finite difference discretization on a uniform grid combined with a second-order convergent scheme (Fromm’s scheme) for the convection term with the mesh-size \( h = 1/n + 1 \), leads to the discrete system matrix of the form (6) such that

\[
\mathbb{D} = I_n \otimes I_n \otimes I_n - I_n \otimes I_n \otimes A^{(1)} - I_n \otimes A^{(2)} \otimes I_n - A^{(3)} \otimes I_n \otimes I_n,
\]

where \( A^{(i)}, i = 1, 2, 3 \), are the same as in example 2. The corresponding tensor equation is given as follows

\[
\mathcal{Z} - \mathcal{Z} \times_1 A^{(1)} - \mathcal{Z} \times_2 A^{(2)} - \mathcal{Z} \times_3 A^{(3)},
\]

As mentioned in [19], we can find matrices \( C^{(i)}, i = 1, 2, 3 \), by using Langville and Stewart method [14] such that

\[
\mathbb{D} = I_n \otimes I_n \otimes I_n - C^{(1)} \otimes C^{(2)} \otimes C^{(3)}
\]
then (20) can be written approximately as follows

\[ Z - Z \times_1 C^{(1)} \times_2 C^{(2)} \times_3 C^{(3)} \approx B. \]  

(21)

We are interested in evaluating the efficiency of our methods, thus, we constructed the matrices \( C^{(i)}, i = 1, 2, 3 \), randomly, i.e.,

\[ C^{(i)} = \text{rand}(n) \quad \text{for } i = 1, 2, 3, \]

where \( \text{rand} \) is a Matlab-command. The numerical results for this example are reported in table 4. The methods act in the same way as for Sylvester tensor
Table 4. Example 4

| Algorithm | $R$ | $n$ | Cycles | $R_m$ | CPU Times (Second) |
|-----------|-----|-----|--------|-------|--------------------|
| Algorithm 3 | 3   | 300 | 6      | $1.58 \cdot 10^{-6}$ | 27.49  |
| Algorithm 4 | 3   | 300 | 4      | $1.56 \cdot 10^{-7}$ | 0.83   |
|           | 5   | 300 | 4      | $8.77 \cdot 10^{-7}$ | 0.85   |

equation, figure 3 shows that Algorithm 4 converges quickly comparing to Algorithm 3 due to the computation of the solution of the reduced Stein tensor equation of order $mR \times mR \times mR$ for increasing $m$.

**Figure 3.** Residual norm of the extended block and global Arnoldi algorithms for example 4 with $n = 300$

6. **Conclusion.** In this paper, we have presented new methods, using extended Krylov subspaces projections, to extract approximate solutions to (1) and (2) with low rank right hand sides. We have applied extended block and global Arnoldi processes to the coefficient matrices, which lead to low dimensional tensor equations. We gave inexpensive ways to compute the residual norms. Numerical examples show that our methods lead to satisfactory results.

**REFERENCES**

[1] J. Ballani and L. Grasedyck, A projection method to solve linear systems in tensor format, *Numerical Linear Algebra with Applications*, 20 (2013), 27–43.

[2] F. P. A. Beik, F. S. Movahed and S. Ahmadi-Asli, On the Krylov subspace methods based on tensor format for positive definite Sylvester tensor equations, *Numerical Linear Algebra with Applications*, 23 (2016), 444–466.

[3] A. H. Bentbib, K. Jbilou and El M. Sadek, On some extended block Krylov based methods for large scale nonsymmetric Stein matrix equations, *Mathematics*, 5 (2017), 21.

[4] A. H. Bentbib, S. El-Halouy and El M. Sadek, Krylov subspace projection method for Sylvester tensor equation with low rank right-hand side, *Numerical Algorithms*, (2020), 1–20.

[5] M. Chen and D. Kressner, Recursive blocked algorithms for linear systems with Kronecker product structure, *Numerical Algorithms*, (2019), 1–18.
[6] D. Calvetti and L. Reichel, Application of ADI iterative methods to the restoration of noisy images, *SIAM Journal on Matrix Analysis and Applications*, 17 (1996), 165–186.

[7] Z. Chen and L. Lu, A projection method and Kronecker product pre-conditioner for solving Sylvester tensor equations, *Science China Mathematics*, 55 (2012), 1281–1292.

[8] Z. Chen and L. Lu, A gradient based iterative solutions for Sylvester tensor equations, *Mathematical Problems in Engineering*, (2013).

[9] F. Ding and T. Chen, Gradient based iterative algorithms for solving a class of matrix equations, *IEEE Transactions on Automatic Control*, 50 (2005), 1216–1221.

[10] M. Heyouni and K. Jbilou, An extended block Arnoldi algorithm for large-scale solutions of the continuous-time algebraic Riccati equation, *Electronic Transactions on Numerical Analysis*, 33 (2009), 53–62.

[11] M. Heyouni, Extended Arnoldi methods for large low-rank Sylvester matrix equations, *Applied Numerical Mathematics*, 60 (2010), 1171–1182.

[12] T. G. Kolda and B. W. Bader, Tensor decompositions and applications, *SIAM Review*, 51 (2009), 455–500.

[13] D. Kressner and C. Tobler, Krylov subspace methods for linear systems with tensor product structure, *SIAM Journal on Matrix Analysis and Applications*, 31 (2010), 1688–1714.

[14] A. N. Langville and W. J. Stewart, A Kronecker product approximate preconditioner for SANs, *Numerical Linear Algebra with Applications*, 11 (2004), 723–752.

[15] N. Lee and A. Cichocki, Fundamental tensor operations for large-scale data analysis using tensor network formats, *Multidimensional Systems and Signal Processing*, 29 (2018), 921–960.

[16] T. Li, Q. W. Wang and X. F. Duan, Numerical algorithms for solving discrete Lyapunov tensor equation, *Journal of Computational and Applied Mathematics*, 370 (2020), 112676.

[17] T. Stykel, et al., A Matlab toolbox for large Lyapunov and Riccati equations, model reduction problems, and linear quadratic optimal control problems, https://www.tu-chemnitz.de/sfb393/lyapack/, (2000).

[18] T. Stykel and V. Simoncini, Krylov subspace methods for projected Lyapunov equations, *Applied Numerical Mathematics*, 62 (2012), 35–50.

[19] X. Xu and W. Q. Wang, Extending BiCG and BiCR methods to solve the Stein tensor equation, *Computers & Mathematics with Applications*, 77 (2019), 3117–3127.

Received August 2020; revised January 2021.

E-mail address: a.bentbib@uca.ac.ma
E-mail address: elhalouysmahane@gmail.com
E-mail address: sadek.maths@gmail.com