Iranian Journal of Numerical Analysis and Optimization
Vol. 12, No. 3 (Special Issue), 2022, pp 658–679
DOI:10.22067/ijnao.2022.78966.1186
https://ijnao.um.ac.ir/

How to cite this article
Research Article

Global and extended global Hessenberg processes for solving Sylvester tensor equation with low-rank right-hand side

T. Cheraghzadeh, F. Toutounian* , and R. Khoshsiar Ghaziani

Abstract

In this paper, we introduce two new schemes based on the global Hessenberg processes for computing approximate solutions to low-rank Sylvester tensor equations. We first construct bases for the matrix and extended matrix Krylov subspaces by applying the global and extended global Hessenberg processes. Then the initial problem is projected into the matrix or extended matrix Krylov subspaces with small dimensions. The reduced Sylvester tensor equation obtained by the projection methods can be solved by using a recursive blocked algorithm. Furthermore, we present the upper bounds for the residual tensors without requiring the computation of the approximate solutions in any iteration. Finally, we illustrate the performance of the proposed methods with some numerical examples.

AMS subject classifications (2020): 65F10.

Keywords: Low-rank Sylvester tensor equation; Global Hessenberg process; Extended Global Hessenberg process; CP decomposition.

---

* Corresponding author

Received 27 September 2022; revised 29 October 2022; accepted 31 October 2022

T. Cheraghzadeh
Department of Applied Mathematics, Faculty of Mathematical Science, Shahrekord University, Shahrekord, Iran. e-mail: heraghzadeh@stu.sku.ac.ir

F. Toutounian
Department of Applied Mathematics, Faculty of Mathematical Science, The Center of Excellence on Modeling and Control Systems, Ferdowsi University of Mashhad, Iran. e-mail: toutouni@math.um.ac.ir

R. Khoshsiar
Department of Applied Mathematics, Faculty of Mathematical Science, Shahrekord University, Shahrekord, Iran. e-mail: Khoshsiar@sku.ac.ir

1 Introduction

Let $I_1, I_2, \ldots, I_N \in \mathbb{N}$. The multidimensional array $\mathcal{X} = (\mathcal{X}_{i_1 i_2 \cdots i_N})\,(1 \le i_j \le I_j, j = 1, \ldots, N)$ is called an N-mode tensors with $I_1 I_2 \cdots I_N$ entries. There has been increasing research on tensors in recent years. For instance, Chang, Pearson, and Zhang [8] generalized the Perron–Frobenius theorem for nonnegative matrices to the nonnegative tensors. Eigenvalues, eigenvectors, symmetric hyperdeterminants were defined by Qi [31] for the real supersymmetric tensors, and their properties were described. In [30], the restart techniques are described for the tensor infinite Arnoldi method.

In this work, we introduce two new projection methods for solving the low-rank Sylvester tensor equation

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

where the matrices $A^{(n)} \in \mathbb{R}^{I_n \times I_n}$, $n = 1, 2, \ldots, N$, and right-hand side tensor $\mathcal{B} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ are given, and $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is an unknown tensor. The Sylvester tensor equation (1) has a unique solution if and only if $\lambda_1 + \lambda_2 + \cdots + \lambda_N \neq 0$, for all $\lambda_i \in \sigma(A^{(i)}),\ i = 1, 2, \ldots, N$, where $\sigma(A^{(i)})$ is the spectral of matrix $A^{(i)}$ [9]. In this study, it is assumed that the Sylvester tensor equation has a unique solution. The Sylvester tensor equations are one of the famous problems arising from the discretization of a linear partial differential equation in high dimensions by the use of finite elements, finite differences, and spectral methods [27, 28, 37]. The Sylvester matrix equation

$$A^{(1)} X + X A^{(2)T} = B,$$

is a special case of the Sylvester tensor equation (1), where X is a 2-mode tensor. Many iteration methods for computing approximate solutions for the Sylvester tensor equations (1) have been introduced in recent years. For example, Chen and Lu [9] proposed the GMRES method based on tensor form (GMRES-BTF) to solve the Sylvester tensor equation. Also, to speed up the convergence of the GMRES-BTF method, they proposed preconditioned GMRES-BTF. Beik, Saberi Movahed, and Ahmadi-Asl [4] presented some iterative methods based on the tensor format to solve the Sylvester tensor equations (1). In [33, 34], Saberi–Movahed et al. introduced the tensor format of restarted Simpler GMRES, (SGMRES-BTF(m)), to solve the Sylvester tensor equation and described an accelerating method in accordance with a modification of the generalized conjugate residual with inner orthogonalization (GCRO) method based on the tensor format. Bi-conjugate gradient (BiCG) and bi-conjugate residual (BiCR) methods as well as their preconditioned versions based on the tensor format, have been presented in [39]. The tensor form of the global least squares method is proposed in [24]. Huang, Xie, and Ma [22] proposed the tensor form of the GMRES method for solving a class of tensor equations via the Einstein product. Furthermore, for the case in which the coefficient tensor is symmetric, they proposed the MINRES and SYMMLQ methods based on the tensor format. Dehdezi and Karimi [15] extended the conjugate gradient squared and the conjugate residual squared methods to solve the generalized coupled Sylvester tensor equations. In [16], the authors proposed a gradient based iterative method version for solving the tensor equations and presented a new preconditioner to accelerate the convergence rate of the proposed iterative methods. A projection method has been introduced in [3] to find approximations of linear systems in low-rank tensor format. Kressner and Tobler [25] proposed the Krylov subspace for the case in which the right-hand side tensor has a low-rank. Recently, Bentbib, El-Halouy, and Sadek [5] introduced a new projection method to compute approximate solutions for the low-rank Sylvester tensor equations. The extended Krylov-like methods were proposed in [6] to find the solutions for the low-rank Sylvester and Stein tensor equations. The block and extended block Hessenberg algorithms for solving the Sylvester tensor equation with low-rank right-hand side (1) were presented in [12]. Hessenberg based methods are among the popular methods in terms of the Krylov subspace methods, with less need for arithmetic operations and less storage space compared to the Arnoldi-based methods. The Hessenberg process constructs nonorthogonal bases for the associated Krylov subspace. The schemes based on the Hessenberg process have recently received great attention; see, for instance, [32, 35, 19, 17, 21, 12]. This motivated us to introduce two new projection schemes, employing the global Hessenberg process on the matrix Krylov subspaces. The main idea of this scheme is to project the problem onto a matrix or an extended matrix Krylov subspace. Then the reduced problem can be solved by using the recursive blocked algorithm [11]. Complexity consideration is given to show that the global and extended global Hessenberg processes are less expensive than the global and extended global Arnoldi ones.

We use the following notations. For the matrices X and Y in $\mathbb{R}^{n\times n}$, we consider the following inner product $\langle X, Y\rangle_F = \mathrm{tr}(X^T Y)$, where $\mathrm{tr}(\cdot)$ denotes the trace. The associated norm is the Frobenius norm denoted by $\|E\|_F$. The notation $X \perp_F Y$ means that $\langle X, Y\rangle_F = 0$. The $n \times n$ identity matrix is denoted by $I^{(n)}$. Moreover, $e_j^{(k)}$ denotes the jth canonical vector of $\mathbb{R}^k$, and $0_{m\times n}$ denotes the $m \times n$ zero matrix.

The remainder of this paper is organized as follows. In section 2, we review some basic notations and definitions. In section 3, the global Hessenberg process with maximum strategy and an approach for solving (1) with a right-hand side tensor of a specific rank is described. The extended global Hessenberg approach is presented in section 4. The complexity of the new methods is considered in section 5. Some numerical examples for evaluating the performance of our approaches are given in section 6. Finally, section 7 gives a brief conclusion.

2 Preliminaries

In this part, the notations and basic definitions of tensors are presented. Throughout this paper, we denote tensors by Euler script letters. Matrices and vectors are denoted by capital and lowercase letters, respectively. Also, the Kronecker product of matrices A and B is denoted by $A \otimes B$ and the Kronecker product of tensors $\mathcal{A}$ and $\mathcal{B}$, is denoted by $\mathcal{A} \otimes \mathcal{B}$. Norm of an Nth order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is denoted by $\|\mathcal{X}\|_F$ and is defined as follows:

$$\|\mathcal{X}\|_F = \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle} = \sqrt{\sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \cdots \sum_{i_N=1}^{I_N} \mathcal{X}_{i_1 i_2 \cdots i_N}^2}.$$

Definition 1 ([13])**.** Denote the N-mode (matrix) product of a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ and a matrix $U \in \mathbb{R}^{J \times I_n}$ by $\mathcal{X} \times_n U$. It is of dimension $I_1 \times I_2 \times \cdots \times I_{n-1} \times J \times I_{n+1} \times \cdots \times I_N$ and defined as

$$(\mathcal{X} \times_n U)_{i_1 \cdots i_{n-1} j i_{n+1} \cdots i_N} = \sum_{i_n=1}^{I_n} \mathcal{X}_{i_1 i_2 \cdots i_N} u_{j i_n}.$$

Proposition 1 ([13])**.** Let $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ be an Nth order tensor, let $B \in \mathbb{R}^{J \times I_m}, C \in \mathbb{R}^{K \times I_n}$, and let $W \in \mathbb{R}^{I_n \times I_n}$. Then

$$\mathcal{A} \times_m B \times_n C = \mathcal{A} \times_n C \times_m B,$$
$$\mathcal{A} \times_n W \times_n C = \mathcal{A} \times_n CW.$$

Definition 2 ([14])**.** Assume that $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is an Nth order tensor and that $\{U\}$ is a set of matrices $U_n \in \mathbb{R}^{I_n \times I_n} (n = 1, \ldots, N)$. Then their product in all possible modes $(n = 1, 2, \ldots, N)$ is of size $I_1 \times I_2 \times \cdots \times I_N$ and defined as follows:

$$\mathcal{X} \times \{U\} = \mathcal{X} \times_1 U_1 \times_2 U_2 \cdots \times_N U_N,$$

and

$$\mathcal{X} \times \{U\}^T = \mathcal{X} \times_1 U_1^T \times_2 U_2^T \cdots \times_N U_N^T.$$

Definition 3 ([13])**.** . The outer product of two tensors $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_M}$ and $\mathcal{B} \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_N}$ is denoted by $\mathcal{A} \circ \mathcal{B} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_M \times J_1 \times J_2 \times \cdots \times J_N}$, with entries

$$\mathcal{C}_{i_1 \cdots i_M j_1 \cdots j_N} = \mathcal{A}_{i_1 \cdots i_M} \mathcal{B}_{j_1 \cdots j_N}.$$

If $v_1, v_2, \ldots, v_N$ are N vectors of sizes $I_i, i = 1, \ldots, N$, then their outer product is an Nth order tensor of size $I_1 \times I_2 \times \cdots \times I_N$ and is given by

$$v_1 \circ \cdots \circ v_{N i_1,\ldots,i_N} = v_1(i_1) \cdots v_N(i_N).$$

Definition 4 ([13])**.** An Nth order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is called a rank one tensor if it can be written as the outer product of N vectors $a_i \in \mathbb{R}^{I_i}$ $(i = 1, \ldots, N)$ as follows:

$$\mathcal{X} = a_1 \circ a_2 \circ \cdots \circ a_N.$$

If a tensor can be written as a sum of R rank one tensors, then it is called a rank R tensor.

Definition 5 ([26])**.** The Kronecker product of two tensor $\mathcal{A} = a_1 \circ a_2 \circ \cdots \circ a_N$ and $\mathcal{B} = b_1 \circ b_2 \circ \cdots \circ b_N$ is defined as

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

Proposition 2 ([5])**.** Assume that $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ and $\mathcal{B} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ are Nth order tensors, that $A \in \mathbb{R}^{k_n \times I_n}$, and that $B \in \mathbb{R}^{I_n \times J_n}$. Then

$$(\mathcal{A} \otimes \mathcal{B}) \times_n (A \otimes B) = (\mathcal{A} \times_n A) \otimes (\mathcal{B} \times_n B).$$

Proposition 3 ([5])**.** The product of a rank one tensor $\mathcal{A} = a_1 \circ a_2 \circ \cdots \circ a_N \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ and a set of matrices $U_n \in \mathbb{R}^{I_n \times I_n}, (n = 1, \ldots, N)$ is defined as follows:

$$\mathcal{A} \times \{U\} = U_1 a_1 \circ \cdots \circ U_N a_N. \tag{2}$$

Definition 6 ([13])**.** The CP decomposition of an Nth order tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is written as follows:

$$\mathcal{A} = \sum_{r=1}^{R} a_r^{(1)} \circ a_r^{(2)} \circ \cdots \circ a_r^{(N)},$$

where $R \in \mathbb{N}$ and $a_r^{(i)} \in \mathbb{R}^{I_i}$, $(i = 1, \ldots, N)$. Assume that $a_r^{(i)}$, $(i = 1, \ldots, N)$, are normalized. Then the CP decomposition is given by

$$\mathcal{A} = \sum_{r=1}^{R} \lambda_r a_r^{(1)} \circ a_r^{(2)} \circ \cdots \circ a_r^{(N)},$$

where $\lambda_r \in \mathbb{R}$.

Definition 7 (Left inverse[35])**.** Consider $Z_k \in \mathbb{R}^{n \times k}$ as a matrix partitioned as follows:

$$Z_k = \begin{bmatrix} Z_{1k} \\ \\ Z_{2k} \end{bmatrix},$$

where Z_{1k} is a $k \times k$ matrix. If the matrix Z_{1k} is nonsingular, then a left inverse of Z_k is defined as follow

$$Z_k^L = \left[Z_{1k}^{-1}, 0_{k\times(n-k)}\right].$$

Definition 8 ([7]). Let $A = [A_1, A_2, \dots, A_p]$ and $B = [B_1, B_2, \dots, B_l]$ be matrices of dimension $n \times ps$ and $n \times ls$, respectively, where A_i and B_j ($i = 1, \dots, p; j = 1, \dots, l$) are $n \times s$ matrices. Then the $\diamond$-product of matrices A and B denoted by $A^T \diamond B$ is the $p \times l$ matrix defined by:

$$(A^T \diamond B)_{i,j} = \langle A_i, B_j \rangle_F.$$

Some properties that are verified by the $\otimes$- and $\diamond$-products are as follows:

1. $(DA)^T \diamond B = A^T \diamond (D^T B)$.
2. $A^T \diamond (B(L \otimes I^{(p)}) = (A^T \diamond B)L$.

In what follows, we assume that the right-hand side $\mathcal{B}$ in (1) is of rank R. As known [13], by using the CP decomposition, $\mathcal{B}$ can be written as

$$\mathcal{B} = \sum_{r=1}^{R} b_1^{(r)} \circ \cdots \circ b_N^{(r)}, \tag{3}$$

where $B^{(i)} = \left[b_i^{(1)}, b_i^{(2)}, \dots, b_i^{(R)}\right] \in \mathbb{R}^{I_i \times R}, i = 1, \dots, N$, are the factor matrices. By simple calculations, we can rewrite the relation (3) as

$$\mathcal{B} = \mathcal{I}_R \times_1 B^{(1)} \cdots \times_N B^{(N)}, \tag{4}$$

in which $\mathcal{I}_R$ denotes the identity tensor of Nth order of size $R \times R \times \cdots \times R$ with ones along the super-diagonal.

3 Global Hessenberg process with maximum strategy

The global Hessenberg process with maximum strategy was first presented in [17] by Heyouni to build a basis of the matrix Krylov subspace

$$\mathcal{K}_m(A, V) = \left\{ \sum_{i=0}^{m-1} \gamma_i A^i V, \quad \text{where } \gamma_i \in \mathbb{R} \text{ for } i = 0, 1, \dots, m-1 \right\},$$

where $A \in \mathbb{R}^{n\times n}$ and $V \in \mathbb{R}^{n\times s}$. The global Hessenberg process with maximum strategy can be summarized in Algorithm 1 [17].

By employing Algorithm 1 with $m = m_i$ and $s = R$ for the pair $(A^{(i)}, B^{(i)})$, we obtain $\mathbb{V}_{m_i+1} = [V_1^{(i)}, \dots, V_{m_i+1}^{(i)}] \in \mathbb{R}^{n\times(m_i+1)R}$ with $V_k^{(i)} \in \mathbb{R}^{n\times R}$, for $k = 1, \dots, m_i+1$, and the upper Hessenberg matrix $\bar{H}_{m_i} = (h_{i,j}^{(i)}) \in \mathbb{R}^{(m_i+1)\times m_i}$, which satisfy

$$A^{(i)}\mathbb{V}_{m_i} = \mathbb{V}_{m_i+1}(\bar{H}_{m_i} \otimes I^{(R)}), \tag{5}$$

Algorithm 1 The Global Hessenberg process with Maximum Strategy

1. **Input:** Nonsingular matrix A, initial block V, and an integer m.
2. Determine i_0 and j_0 such that $|V_{i_0,j_0}| = \max\{|V_{i,j}|\}_{1\le i\le n}^{1\le j\le s}$; $\beta = V_{i_0,j_0}$;
   $V_1 = V/\beta$; $l_1 = i_0$; $c_1 = j_0$.
3. For $k = 1, 2, \ldots, m$
4. $U = AV_k$.
5. For $j = 1, 2, \ldots, k$
6. $h_{j,k} = U_{l_j,c_j}$; $U = U - h_{j,k}V_j$.
7. End For.
8. Determine i_0 and j_0 such that $|U_{i_0,j_0}| = \max\{|U_{i,j}|\}_{1\le i\le n}^{1\le j\le s}$
   $h_{k+1,k} = U_{i_0,j_0}$; $V_{k+1} = U/h_{k+1,k}$; $l_{k+1} = i_0$; $c_{k+1} = j_0$.
9. End For.

$$= \mathbb{V}_{m_i}(H_{m_i} \otimes I^{(R)}) + h^{(i)}_{m_i+1,m_i} V^{(i)}_{m_i+1}(e^{(m_i)^T}_{m_i} \otimes I^{(R)}), \tag{6}$$

where H_{m_i} denotes the matrix obtained from $\bar{H}_{m_i}$ by deleting its last row. As [5], we consider an approximate solution of (1) as

$$\mathcal{X}_m = (\mathcal{Y}_m \otimes \mathcal{I}_R) \times \{\mathbb{V}_m\}, \tag{7}$$

where $\{\mathbb{V}_m\}$ denotes a set of matrices $\{\mathbb{V}_{m_1}, \mathbb{V}_{m_2}, \ldots, \mathbb{V}_{m_N}\}$ and $\mathcal{Y}_m$ is an $m_1 \times \cdots \times m_N$ tensor satisfying the low-dimensional Sylvester tensor equation

$$\sum_{i=1}^{N} \mathcal{Y}_m \times_i H_{m_i} = \beta \mathcal{E}_m, \tag{8}$$

where $\beta = \prod_{i=1}^{N} \beta_i$ and $\mathcal{E}_m = (e_1^{(m_1)} \circ \cdots \circ e_1^{(m_N)})$.

Proposition 4. Let $\mathcal{R}_m$ be the residual tensor corresponding to the approximate solution $\mathcal{X}_m$ of (1). Then

$$\mathcal{R}_m = -\sum_{i=1}^{N} h_{m_i+1,m_i}(\mathcal{Y}_m \times_i e^{(m_i)^T}_{m_i}) \otimes \mathcal{I}_R \times_1 \mathbb{V}_{m_1} \cdots \times_i V^{(i)}_{m_i+1} \cdots \times_N \mathbb{V}_{m_N}, \tag{9}$$

where $\mathcal{Y}_m$ is the solution to (8).

Proof. The proof is similar to that of Proposition 6 in [12]. □

Theorem 1. Let $\mathcal{X}_m$ be an approximate solution of (1). Then the corresponding residual $\mathcal{R}_m$ satisfies

$$\|\mathcal{R}_m\| \le \sqrt{((2nR - (m-1))\frac{m}{2})^N} \sqrt{\sum_{i=1}^{N} |\, h_{m_i+1,m_i} \,|^2 \, \|\mathcal{Y}_m \times_i e^T_{m_i}\|^2}, \tag{10}$$

where $m = \max_{1\le i\le N} m_i$.

Proof. The proof is similar to that of Theorem 7 in [12]. □

Furthermore, from the fact that

$$\|\mathbb{V}_{m_j}\|^2 \leq nm_jR, \quad i = 1, \dots, N,$$

we have

$$\|\mathcal{R}_m\| \leq \sqrt{(nmR)^N} \sqrt{\sum_{i=1}^{N} \mid h_{m_i+1,m_i} \mid^2 \|\mathcal{Y}_m \times_i e_{m_i}^T\|^2}. \tag{11}$$

The upper bounds (10) and (11) are pessimistic. We propose the following approximation, which is derived heuristically,

$$\|\mathcal{R}_m\| \approx E_m := \sqrt[N]{(nmR)} \sqrt{\sum_{i=1}^{N} \mid h_{m_i+1,m_i} \mid^2 \|\mathcal{Y}_m \times_i e_{m_i}^T\|^2}. \tag{12}$$

Similar to Algorithm 2 in [5], the global Hessenberg process with the maximum strategy for the Sylvester tensor equation (1) can be summarized in Algorithm 2.

Algorithm 2

---

1. **Input**: Coefficient matrices $A^{(i)}, i = 1, \dots N$, and the right-hand side in low-rank representation,
   $\mathcal{B} = \left[B^{(1)}, B^{(2)}, \dots, B^{(N)}\right]$.
2. **Output**: An approximate solution $\mathcal{X}_m$ for equation (1).
3. Choose a tolerance $\epsilon > 0$, integer parameters $k_i^{'}$, $i = 1, \dots, N$. Set $k_i = 0, m_i = k_i^{'}$.
4. For $i = 1 : N$
5. For $j = k_i + 1 : k_i + k_i^{'}$
6. Construct the basis $\left[V_{k_i+1}, \dots, V_{k_i+k_i^{'}}\right]$ and the matrix $\mathbb{H}_{m_i}$ by Algorithm 1.
7. End For
8. End For
9. Solve the low-dimensional equation $\sum_{i=1}^{N} \mathcal{Y}_m \times_i \mathbb{H}_{m_i} = \beta \mathcal{E}_m$ by the recursive blocked algorithms presented in [11].
10. Compute the estimated residual norm of $\mathcal{R}_m$,
    i.e., $E_m = \sqrt[N]{(nmR)} \sqrt{\sum_{i=1}^{N} \mid h_{m_i+1,m_i} \mid^2 \|\mathcal{Y}_m \times_i e_{m_i}^T\|^2}$.
11. If $E_m > \epsilon$, set $k_i = k_i + k_i^{'}$, $m_i = k_i + k_i^{'}$ for $i = 1, \dots, N$, and go to step 4.
12. Compute the approximate solution by $\mathcal{X}_m = (\mathcal{Y}_m \otimes \mathcal{I}^{(R)}) \times_1 \mathbb{V}_{m_1} \cdots \times_N \mathbb{V}_{m_N}$.

---

4 The extended global Hessenberg process

We first recall the extended matrix Krylov subspace. Let $A \in \mathbb{R}^{n\times n}$ and $V \in \mathbb{R}^{n\times s}$. The extended global Hesssenberg process corresponding to the pair (A, V) is defined as follows [17]:

$$\begin{aligned}\mathcal{K}^e_m(A,V) &= \operatorname{span}(V, A^{-1}V, AV, \dots, A^{m-1}V, A^{-m}V),\\ &= \mathcal{K}_m(A,V) + \mathcal{K}_m(A^{-1}, A^{-1}V).\end{aligned}$$

The algorithm proceeds by running one step of the Global Hessenberg process with A and one step with A^{-1}, while maintaining orthogonalization among all generated vectors and the $n \times s$ matrices $Y_j = e^{(n)}_{l_j} e^{(s)^T}_{c_j}$ whose entries are zero except $(Y_j)_{l_j,c_j} = 1$. The first two block vectors $V^{(1)}_1$ and $V^{(2)}_1$ are obtained as follows:

$$V^{(1)}_1 = V/r_{11}, \tag{13}$$

where $r_{11} = V_{l_1,c_1}$ and $|V_{l_1,c_1}| = \max\{|V_{i,j}|\}^{1\le j\le s}_{1\le i\le n}$, and

$$V^{(2)}_2 = W/r_{2,2}, \tag{14}$$

where $W = A^{-1}V - r_{1,2}V^{(1)}_1, r_{1,2} = (A^{-1}V)_{l_1,c_1}, r_{2,2} = W_{l_2,c_2}$, and $|W_{l_2,c_2}| = \max\{|W_{i,j}|\}^{1\le j\le s}_{1\le i\le n}$.

Let $V_i = [V^{(1)}_i, V^{(2)}_i]$ be the ith $n \times 2s$ block vector of $\mathbb{V}_m = [V_1, \dots, V_m]$ and let

$$H_{i,j} = \begin{bmatrix} h_{2i-1,2j-1} & h_{2i-1,2j} \\ h_{2i,2j-1} & h_{2i,2j} \end{bmatrix},$$

be the 2×2 block matrix (i, j) of the upper block Hessenberg matrix $\overline{\mathbb{H}}_m \in \mathbb{R}^{2(m+1)\times 2m}$. Then we compute the two block vectors $V^{(1)}_{k+1}$ and $V^{(2)}_{k+1}$ by the relation

$$\left[V^{(1)}_{k+1}\ V^{(2)}_{k+1}\right](H_{k+1,k} \otimes I^{(s)}) = [AV^{(1)}_k, A^{-1}V^{(2)}_k] - \sum_{j=1}^{k}[V^{(1)}_j, V^{(2)}_j](H_{j,k} \otimes I^{(s)}), \tag{15}$$

where the entries of coefficients matrices $H_{k+1,k}$ and $H_{i,k}$, for $i = 1, \dots, k$, will be determined such that the relations

$$V^{(1)}_{k+1} \perp_F Y_1, \dots, Y_{2k} \qquad \text{and} \qquad (V^{(1)}_{k+1})_{l_{2k+1},c_{2k+1}} = 1,$$

and

$$V^{(2)}_{k+1} \perp_F Y_1, \dots, Y_{2k+1} \qquad \text{and} \qquad (V^{(2)}_{k+1})_{l_{2k+2},c_{2k+2}} = 1$$

hold for $k = 1,\dots,m$. The determination of indices l_{2k+1}, c_{2k+1} and l_{2k+2}, c_{2k+2} is similar to that of indices l_1, c_1 and l_2, c_2, respectively. The main steps of the extended global Hessenberg process algorithm to generate $\mathbb{V}_m$ and $\overline{\mathbb{H}}_m$ may be summarized as follows.

Algorithm 3 The Extended Global Hessenberg process with Maximum Strategy

1. **Input:** Nonsingular matrix A, initial block V, and an integer m.
2. Determine i_0 and j_0 such that $|V_{i_0,j_0}| = \max\{|V_{i,j}|\}_{1\le i\le n}^{1\le j\le s}$; $r_{1,1} = V_{i_0,j_0}$; $V_1^{(1)} = V/r_{1,1}$; $l_1 = i_0$; $c_1 = j_0$; .
3. $W = A^{-1}V$; $r_{1,2} = W_{l_1,c_1}$.
4. $W = W - r_{1,2}V_1^{(1)}$, $|W_{i_0,j_0}| = \max\{|W_{i,j}|\}_{1\le i\le n}^{1\le j\le s}$; $r_{2,2} = W_{i_0,j_0}$; $V_1^{(2)} = W/r_{2,2}$; $l_2 = i_0$, $c_2 = j_0$.
5. For $k = 1,2,\dots,m$
6. $\quad W = AV_k^{(1)}$.
7. $\quad$ For $i = 1,\dots,k$
8. $\quad\quad h_{2i-1,2k-1} = W_{l_{2i-1},c_{2i-1}}$, $W = W - h_{2i-1,2k-1}V_i^{(1)}$; $h_{2i,2k-1} = W_{l_{2i},c_{2i}}$, $W = W - h_{2i,2k-1}V_i^{(2)}$.
9. $\quad$ End For.
10. $\quad$ Determine i_0 and j_0 such that $|W_{i_0,j_0}| = \max\{|W_{i,j}|\}_{1\le i\le n}^{1\le j\le s}$; $h_{2k+1,2k-1} = W_{i_0,j_0}$; $V_{k+1}^{(1)} = W/h_{2k+1,2k-1}$; $l_{2k+1} = i_0$; $c_{2k+1} = j_0$.
11. $\quad W = A^{-1}V_k^{(2)}$.
12. $\quad$ For $i = 1,\dots,k$.
13. $\quad\quad h_{2i-1,2k} = W_{l_{2i-1},c_{2i-1}}$, $W = W - h_{2i-1,2k}V_i^{(1)}$; $h_{2i,2k} = W_{l_{2i},c_{2i}}$; $W = W - h_{2i,2k}V_i^{(2)}$.
14 $\quad$ End For.
15 $\quad h_{2k+1,2k} = W_{l_{2k+1},c_{2k+1}}$, $W = W - h_{2k+1,2k}V_{k+1}^{(1)}$.
16. $\quad$ Determine i_0 and j_0 such that $|W_{i_0,j_0}| = \max\{|W_{i,j}|\}_{1\le i\le n}^{1\le j\le s}$; $h_{2k+2,2k} = W_{i_0,j_0}$; $V_{k+1}^{(2)} = W/h_{2k+2,2k}$; $l_{2k+2} = i_0$; $c_{2k+2} = j_0$.
17. End For.

Suppose that the matrix $\mathbb{P}_m$ is defined by $[Y_1, Y_2, \dots, Y_{2m}]$. Then

$$\mathbb{P}_m^T \diamond \mathbb{V}_m = \mathbb{L}_m,$$

where $\mathbb{L}_m \in \mathbb{R}^{2m\times 2m}$ is a unit lower triangular matrix. So, we have $\mathbb{L}_{m_i}^{-1}(\mathbb{P}_{m_i}^T \diamond \mathbb{V}_{m_i}) = I^{(2m_i)}$. As in [1], we consider $\mathbb{V}_m^L = (\mathbb{P}_m(\mathbb{L}_m^{-T} \otimes I^{(s)}))^T = (\mathbb{L}_m^{-1} \otimes I^{(s)})\mathbb{P}_m^T$, as a left inverse for the $\diamond$-product, which verifies the relation $\mathbb{V}_m^L \diamond \mathbb{V}_m = I^{(2ms)}$. Using this matrix, we can state the following proposition.

Proposition 5. Let $\overline{\mathbb{T}}_m = \mathbb{V}_{m+1}^L \diamond (A\mathbb{V}_m)$, and suppose that m steps of Algorithm 3 have been carried out. Then

$$A\mathbb{V}_m = \mathbb{V}_{m+1}(\overline{\mathbb{T}}_m \otimes I^{(s)}), \tag{16}$$

$$= \mathbb{V}_m(\mathbb{T}_m \otimes I^{(s)}) + V_{m+1}(T_{m+1,m}E_m^T \otimes I^{(s)}), \tag{17}$$

where $T_{i,j}$ is the 2×2 block (i,j) of $\mathbb{T}_m$ and $E_m^T = [0_{2\times 2(m-1)}, I^{(2)}]$, and $\mathbb{T}_m$ is obtained by removing the two last rows of $\overline{\mathbb{T}}_m$.

Proof. The proof is similar to the case for the classical Arnoldi process in [20]. □

As [36], in the following proposition, we derive some recursive relations, which can be used to significantly reduce the computational cost of the basic algorithm.

Proposition 6. Let $\overline{\mathbb{T}}_m = [t_{:,1}, \dots, t_{:,2m}]$ and $\overline{\mathbb{H}}_m = [h_{:,1}, \dots, h_{:,2m}]$ be two $2(m+1) \times 2m$ block upper Hessenberg matrices, let $\ell^{(k+1)} = (\ell_{i,j}) = H_{k+1,k}^{-1}$, and let $r_{1,1}, r_{1,2}, r_{2,2}$ be as defined in Algorithm 3. Then for the odd columns, we have

$$t_{:,2j-1} = h_{:,2j-1}, \qquad j = 1, \dots, m,$$

and for the even columns, we have

$$\begin{aligned}
(k=1) \qquad t_{:,2} &= \frac{1}{r_{2,2}}(r_{1,1} e_1^{2(m+1)} - r_{1,2} t_{:,1}),\\
t_{:,4} &= (e_2^{2(m+1)} - \begin{bmatrix} \overline{\mathbb{T}}_1 h_{1:2,2} \\ 0_{(2m-2)\times 2} \end{bmatrix}) \ell_{22}^{(2)},\\
\rho^{(2)} &= (\ell_{11}^{(2)})^{-1} \ell_{12}^{(2)},\\
(1 < k \le m) \qquad t_{:,2k} &= t_{:,2k} + t_{:,2k-1} \rho^{(k)},\\
t_{:,2k+2} &= (e_{2k}^{2(m+1)} - \begin{bmatrix} \overline{\mathbb{T}}_k h_{1:2k,2k} \\ 0_{(2m-2k)\times 2} \end{bmatrix}) \ell_{22}^{(k+1)},\\
\rho^{(k+1)} &= (\ell_{11}^{(k+1)})^{-1} \ell_{12}^{(k+1)}.
\end{aligned}$$

Proof. Starting from (15), we have

$$\begin{aligned}
AV_k^{(1)} &= V_{k+1}(H_{k+1,k} e_1^{(2)} \otimes I^{(s)}) + \mathbb{V}_k(\mathbb{H}_k e_{2k-1}^{(2k)} \otimes I^{(s)})\\
&= \mathbb{V}_{k+1}(\overline{\mathbb{H}}_k e_{2k-1}^{(2k)} \otimes I^{(s)}).
\end{aligned}$$

Pre-multiplying the above relation by $\mathbb{V}_{m+1}^L$, we get

$$\begin{aligned}
\mathbb{V}_{m+1}^L \diamond AV_k^{(1)} &= \mathbb{V}_{m+1}^L \diamond \mathbb{V}_{k+1}(\overline{\mathbb{H}}_k e_{2k-1}^{(2k)} \otimes I^{(s)})\\
&= (\mathbb{V}_{m+1}^L \diamond \mathbb{V}_{k+1}) \overline{\mathbb{H}}_k e_{2k-1}^{(2k)}\\
&= \begin{bmatrix} I^{(2k+2)} \\ 0_{(2m-2k)\times(2k+2)} \end{bmatrix} \overline{\mathbb{H}}_k e_{2k-1}^{(2k)}\\
&= \begin{bmatrix} \overline{\mathbb{H}}_k \\ 0_{(2m-2k)\times(2k+2)} \end{bmatrix} e_{2k-1}^{(2k)}.
\end{aligned}$$

Hence,

$$t_{:,2k-1} = h_{:,2k-1}, \qquad k = 1, \dots, m.$$

From the lines 2 and 3 of Algorithm 3, we have

$$r_{2,2}V_1^{(2)} = r_{1,1}A^{-1}V_1^{(1)} - r_{1,2}V_1^{(1)}.$$

Pre-multiplying this relation by A, we get

$$r_{2,2}AV_1^{(2)} = r_{1,1}V_1^{(1)} - r_{1,2}AV_1^{(1)}.$$

Pre-multiplying the above relation by $\mathbb{V}_{m+1}^L$, we have

$$(\mathbb{V}_{m+1}^L \diamond AV_1^{(2)}) = \frac{1}{r_{2,2}}(r_{1,1}(\mathbb{V}_{m+1}^L \diamond V_1^{(1)}) - r_{1,2}(\mathbb{V}_{m+1}^L \diamond AV_1^{(1)})).$$

Consequently,

$$t_{:,2} = \frac{1}{r_{2,2}}(r_{1,1}e_1^{2(m+1)} - r_{1,2}h_{:,1}),$$

In addition, from (15), one gets

$$V_k^{(2)} = AV_{k+1}(H_{k+1,k}e_2^{(2)} \otimes I^{(s)}) + A\mathbb{V}_k(\mathbb{H}_k e_{2k}^{(2k)} \otimes I^{(s)}).$$

This relation implies that

$$\begin{aligned}
&\mathbb{V}_{m+1}^L \diamond AV_{k+1}(H_{k+1,k}e_2^{(2)} \otimes I^{(s)}) \\
&\qquad = \mathbb{V}_{m+1}^L \diamond V_k^{(2)} - \mathbb{V}_{m+1}^L \diamond (A\mathbb{V}_k(\mathbb{H}_k e_{2k}^{(2k)} \otimes I^{(s)}) \\
&\qquad = e_{2k}^{2(m+1)} - (\mathbb{V}_{m+1}^L \diamond AV_k)\mathbb{H}e_{2k}^{(2k)} \\
&\qquad = e_{2k}^{2(m+1)} - \begin{bmatrix} \overline{\mathbb{T}}_k h_{1:2k,2k} \\ 0_{(2m-2k)\times 2k} \end{bmatrix}.
\end{aligned}$$

On the other hand, for the left-hand side of this relation, we deduce

$$\begin{aligned}
&\mathbb{V}_{m+1}^L \diamond AV_{k+1}(H_{k+1,k}e_2^{(2)} \otimes I^{(s)}) \\
&\qquad = \mathbb{V}_{m+1}^L \diamond [AV_{k+1}^{(1)} \quad AV_{k+1}^{(2)}] \begin{bmatrix} h_{2k+1,2k}I^{(s)} \\ h_{2k+2,2k}I^{(s)} \end{bmatrix} \\
&\qquad = h_{2k+1,2k}\mathbb{V}_{m+1}^L \diamond AV_{k+1}^{(1)} + h_{2k+2,2k}\mathbb{V}_{m+1}^L \diamond AV_{k+1}^{(2)} \\
&\qquad = h_{2k+1,2k}t_{:,2k+1} + h_{2k+2,2k}t_{:,2k+2}.
\end{aligned}$$

Hence

$$t_{:,2k+2} = \frac{1}{h_{2k+2,2k}}(-h_{2k+1,2k}t_{:,2k+1} + e_{2k}^{2(m+1)} - \begin{bmatrix} \overline{\mathbb{T}}_k h_{1:2k,2k} \\ 0_{(2m-2k)\times 2k} \end{bmatrix}).$$

By using the inverse of the 2×2 upper triangular matrix $H_{k+1,k}$ and defining $\rho^{(k+1)} = (\ell_{11}^{(k+1)})^{-1}\ell_{12}^{(k+1)}$, this relation can be written as follows:

$$t_{:,2k+2} = t_{:,2k+1}\rho^{(k+1)} + (e_{2k}^{2(m+1)} - \begin{bmatrix} \overline{\mathbb{T}}_k h_{1:2k,2k} \\ 0_{(2m-2k)\times 2k} \end{bmatrix})\ell_{22}^{(k+1)},$$

which completes the proof. □

4.1 Extended global Hessenberg process for low-rank Sylvester tensor equation

In this subsection, we consider the extended global Hessenberg process derived in the previous subsection for the pair $(A^{(i)}, B^{(i)}),\ i = 1, \dots, N$. By applying Algorithm 3 with $s = R$ to the pair $(A^{(i)}, B^{(i)}),\ i = 1, \dots, N$, the block matrices $\mathbb{V}_{m_i} = [V_1^{(i)}, \dots, V_{m_i}^{(i)}], i = 1, \dots, N$, are obtained and the following relation holds, for $i = 1, \dots, N$,

$$\begin{aligned} A^{(i)}\mathbb{V}_{m_i} &= \mathbb{V}_{m_i+1}(\overline{\mathbb{T}}_{m_i} \otimes I^{(R)}) \\ &= \mathbb{V}_{m_i}(\mathbb{T}_{m_i} \otimes I^{(R)}) + V_{m_i+1}^{(i)}(T_{m_i+1,m_i}^{(i)} E_{m_i}^T \otimes I^{(R)}), \end{aligned} \tag{18}$$

where $E_{m_i}^T = \left[0_{2\times 2}, \dots, 0_{2\times 2}, I^{(2)}\right] \in \mathbb{R}^{2\times 2m_i}$, and $\overline{\mathbb{T}}_{m_i} = (T_{i,j}^{(i)}) \in \mathbb{R}^{2(m_i+1)\times 2m_i}$ is the restriction of $A^{(i)}$ to the extended global Krylov subspace $\mathcal{K}_{m_i}^e(A^{(i)}, B^{(i)})$. Using Line 1 of Algorithm 3, we have

$$B^{(i)} = r_{11}^{(i)}(V_1^{(i)})^{(1)}, \qquad \text{for} \ \ i = 1, 2, \dots, N.$$

As in the case of the global Hessenberg process, for the low-rank Sylvester tensor equation (1), we seek an approximate solution of the form

$$\mathcal{X}_m = (\mathcal{Y}_m \otimes \mathcal{I}_R) \times \{\mathbb{V}_m\}, \tag{19}$$

where $\{\mathbb{V}_m\}$ denotes a set of matrices $\mathbb{V}_{m_i} \in \mathbb{R}^{n\times 2Rm_i},\ i = 1, \dots, N$, and $\mathcal{Y}_m \in \mathbb{R}^{2m_1\times\cdots\times 2m_N}$ satisfies the low-dimensional Sylvester tensor equation

$$\sum_{i=1}^{N} \mathcal{Y}_m \times_i \mathbb{T}_{m_i} = \beta_m \mathcal{E}_m, \tag{20}$$

where $\beta_m = \prod_{i=1}^{N} r_{11}^{(i)}$ and $\mathcal{E}_m = (e_1^{(2m_1)} \circ \cdots \circ e_1^{(2m_N)})$. In this case, the residual corresponding to $\mathcal{X}_m$ can be written as

$$\mathcal{R}_m = -\sum_{i=1}^{N} (\mathcal{Y}_m \times_i T^{(i)}_{m_{i+1},m_i} E^T_{m_i}) \otimes \mathcal{I}_R \times_1 \mathbb{V}_{m_1} \ldots \times_i V^{(i)}_{m_i+1} \ldots \times_N \mathbb{V}_{m_N}. \tag{21}$$

We can easily obtain

$$\|\mathcal{R}_m\| \leq \sqrt{((2nR-2m+1)m)^N} \sqrt{\sum_{i=1}^{N} \|\mathcal{Y}_m \times_i T^{(i)}_{m_{i+1},m_i} E^T_{m_i}\|} \tag{22}$$

and

$$\|\mathcal{R}_m\| \leq \sqrt{(2nmR)^N} \sqrt{\sum_{i=1}^{N} \|\mathcal{Y}_m \times_i T^{(i)}_{m_{i+1},m_i} E^T_{m_i}\|}, \tag{23}$$

where $m = \max_{1 \leq i \leq N} m_i$. Finally, the following estimate is derived heuristically:

$$\|\mathcal{R}_m\| \approx E_m := \sqrt[N]{(2nmR)} \sqrt{\sum_{i=1}^{N} \|\mathcal{Y}_m \times_i T^{(i)}_{m_i+1,m_i} E^T_{m_i}\|}. \tag{24}$$

For the extended global Hessenberg process, the main part of Algorithm 2 remains the same except that the lines 6, 9, and 10 must be changed as follows:

6. Construct the basis $\left[V_{k_i+1}, \ldots, V_{k_i+k'_i}\right]$ and the matrix $\mathbb{T}_{m_i}$ by Algorithm 3 and the formulas of Proposition 6.
9. Solve the low-dimensional equation $\sum_{i=1}^{N} \mathcal{Y}_m \times_i \mathbb{T}_{m_i} = \beta_m \mathcal{E}_m$ by the recursive blocked algorithms presented in [11].
10. Compute the estimated residual norm of $\mathcal{R}_m$, that is, $E_m = \sqrt[N]{(2nmR)} \sqrt{\sum_{i=1}^{N} \|\mathcal{Y}_m \times_i T^{(i)}_{m_i+1,m_i} E^T_{m_i}\|^2}$.

5 Complexity consideration

In this section, we present the required number of operations to solve the low-rank Sylvester tensor equation (1) for $I_1 = I_2 = \cdots = I_N$. Let Nnz denote the number of nonzero elements of matrix A, and suppose that the LU decomposition of A is available for computing the block matrix $W = A^{-1}V$. We compare the required operations for the extended global Hessenberg process and the extended global Arnoldi process [18]. Algorithm 3 requires $(2n^2s + 4ns)$ operations for computing the block matrices $V_1^{(1)}$ and $V_1^{(2)}$. In addition, the iteration k of this algorithm involves

- $V_{k+1}^{(1)}$, which requires $2sNnz + ns(4k+1) - 4k^2$ operations,
- $V_{k+1}^{(2)}$, which requires $2n^2s + ns(4k+3) - (2k+1)^2$ operations.

Note that the global Arnoldi process (Algorithm 2 in [18]) requires $2n^2s+10ns$ operations for computing the global QR decomposition $[V, A^{-1}V]$, and the iteration k of this process involves

- $U = [AV_k^{(1)}, A^{-1}V_k^{(2)}]$, which requires $2sNnz + 2n^2s$ operations.
- $H_{i,j} = V_i^T \diamond U,\ U = U - V_i(H_{i,j} \otimes I^{(s)}),\ i = 1, 2, \dots, k$, which require $16nsk$ operations.
- the global decomposition of U, that is, $U = V_{k+1}(H_{k+1,k} \otimes I^{(s)})$, which requires $10ns$ operations.

Therefore, for computing an approximation of the solution of Sylvester tensor equation (1), the total cost number of operations required to perform m iterations of the extended global versions of Arnoldi and Hessenberg processes is approximately shown in Table 1. In addition, the total cost number of operations required to perform m iterations of the global Hessenberg process (Algorithm 1) and the modified global Arnoldi process (Algorithm 2.2 in [23]) is presented in this table. According to Table 1, when solving the low-rank Sylvester tensor equation (1), the global and extended global Hessenberg processes are less expensive than the global and extended global Arnoldi ones. On the other hand, these Hessenberg processes use the maximum strategy. Hence they involve some data movement. However, these processes need slightly less storage than the Arnoldi processes per iteration.

Table 1: Operation count for the global and extended global versions of Hessenberg and Arnoldi processes.

Process	Number of operations
Global Arnoldi	$N(2mRNnz + (m+1)(2m+3)nR - (m(m+1))/2)$
Global Hessenberg	$N(2mRNnz + (m+1)^2nR - (m(m+1)(2m+1))/6)$
Extended Global Arnoldi	$N(2mRNnz + 2(m+1)n^2R + (m+1)(8m+10)nR)$
Extended Global Hessenberg	$N(2mRNnz + 2(m+1)n^2R + 4(m+1)^2nR - m(8m^2+18m+13)/3)$

6 Numerical experiments

In this section, some test problems with $N = 3$ are used to examine the robustness of two new presented methods for solving the low-rank Sylvester equation (1). All the numerical experiments were performed in double-precision floating-point arithmetic in MATLAB 2021a. The machine we have used is an Intel(R) Xeon(R) CPU E5-2680 v4@2.40 GHz, 128 GB of RAM, using the Tensor Toolbox [2]. We employ the recursive blocked algorithms introduced in [11] to solve the low-dimensional Sylvester tensor equations (8) and (20). The step size parameter k' associated with one cycle is equal to 3. The algorithms stopped whenever $E_m \leq 10^{-7}$, where E_m is the estimate of $\|\mathcal{R}_m\|$. We also compare the numerical behavior of the methods in terms of the number of cycles (Cycle), the norm of residual $\|\mathcal{R}_m\|$, the norm of error $\|\mathcal{X}^* - \mathcal{X}_m\|$, where $\mathcal{X}^*$ is the exact solution, and the CPU time in seconds (CPU time) required only for constructing the Krylov subspace basis and the solution of reduced Sylvester tensor equation. Note that we use the procedure $cp\_als(\mathcal{B}, R)$ from the toolbox [2] to compute the CP decomposition of the right-hand side $\mathcal{B}$. In Table 2, we report $\|\mathcal{B} - \mathcal{B}_{cp}\|$, where the $\mathcal{B}_{cp}$ is the CP decomposition corresponding to the right-hand side tensor $\mathcal{B}$, using the procedure $cp\_als(\mathcal{B}, R)$. The results of examples are reported in Table 2. For each example, the rank R and the dimension n are presented in this table. In Figure 1 , by plotting the norm of residual $\|R_m\|_F$ versus the number of cycles, we display the convergence history of the global and extended global Arnoldi and Hessenberg algorithms for Examples 1–5.

Example 1. In this example, as in [5], we consider the matrices $A^{(i)}, i = 1, 2, 3$, corresponding to discretization of the operator

$$L(u) = \Delta u - f_1(x, y)\frac{\partial u}{\partial x} + f_2(x, y)\frac{\partial u}{\partial y} + g(x, y),$$

in the unit square $[0, 1] \times [0, 1]$ with Dirichlet homogeneous boundary conditions. The number of inner grid points in each direction is n_0 for the operator L. The discretization of the operator L yields matrices extracted from the Lyapack package [29], using the command fdm and denoted as

$$A^{(i)} = \text{fdm}(n_0, f_1(x, y), f_2(x, y), g(x, y)), \qquad i = 1, 2, 3,$$

with $f_1(x, y) = e^{xy}, f_2(x, y) = sin(x, y), g(x, y) = y^2 - x^2$, $n = n_0^2$. The right-hand side tensor is chosen in such a way that the exact solution of the Sylvester tensor equation (1) has the form $\mathcal{X}^* = x_1 \circ x_2 \circ x_3$, with $x_i = \text{rand}(n, 1)$, for $i = 1, 2, 3$.

Example 2. Assume that in the Sylvester tensor equation (1), the coefficient matrices are presented as [5]

$$A^{(i)} = \text{gallery}('\text{poisson}', n_0), \quad i = 1, 2, 3,$$

where $n = n_0^2$. The right-hand side tensor is constructed such that the exact solution $\mathcal{X}$ of the Sylvester tensor equation (1) is a tensor with entries equal to one.

Example 3. Let $A^{(i)}$, $i = 1, 2, 3$, be defined as [10]

$$A^{(i)} = \text{rand(n,n)} + \text{diag(ones(n,1)} * \text{alfa)},$$

where alfa = 8 and the right-hand side tensor is constructed as in Example 1.

Example 4. Consider the Sylvester equation (1) with the coefficient matrices generated by [38]

$$A^{(i)} = \text{diag(rand(n-1,1),-1)} + \text{diag}(2 + \text{diag(rand(n,n))}), \quad i = 1, 2, 3,$$

and the right-hand side tensor is constructed as in Example 1.

Example 5. The coefficient matrices $A^{(i)},\ i = 1, 2, 3$, for the Sylvester tensor equation (1) are defined as

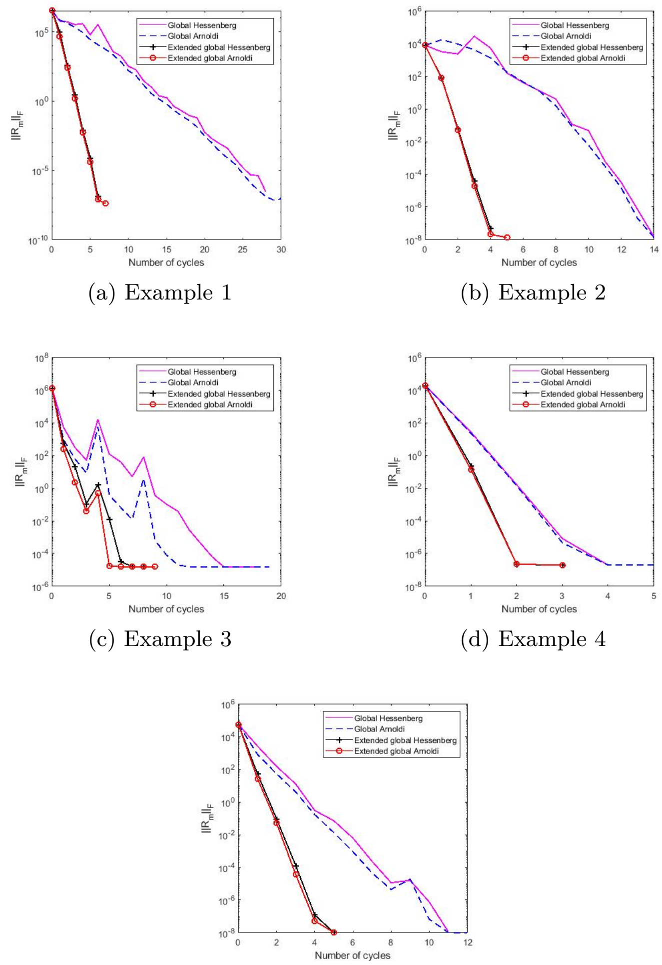
$$A^{(i)}(l, j) = \frac{1}{1 + |l - j|},$$

and the right-hand side tensor is constructed as in Example 1.

Table 2: Results of Examples 1–5.

Example	Algorithm	$\|\mathcal{B} - \mathcal{B}_{cp}\|$	$\|\mathcal{R}_m\|$	$\|\mathcal{X}^* - \mathcal{X}_m\|$	Cycle	CPU time
Example 1 $n = 400$,$R = 4$	Global Arnoldi	$3.655e-08$	$8.549e-08$	$9.903e-11$	30	2.879
	Global Hessenberg	$3.655e-08$	$2.901e-07$	$2.667e-10$	28	2.558
	Extended Global Arnoldi	$3.655e-08$	$4.197e-08$	$3.173e-11$	7	0.261
	Extended Global Hessenberg	$3.655e-08$	$1.411e-07$	$1.162e-10$	6	0.110
Example 2 $n = 400$, $R = 3$	Global Arnoldi	$1.355e-08$	$1.406e-08$	$1.560e-08$	14	0.138
	Global Hessenberg	$1.355e-08$	$1.573e-08$	$1.735e-08$	14	0.229
	Extended Global Arnoldi	$1.355e-08$	$1.375e-08$	$1.603e-08$	5	0.079
	Extended Global Hessenberg	$1.355e-08$	$4.528e-08$	$2.652e-08$	4	0.058
Example 3 $n = 500$, $R = 3$	Global Arnoldi	$1.532e-05$	$1.531e-05$	$3.731e-07$	19	0.612
	Global Hessenberg	$1.532e-05$	$1.530e-05$	$3.729e-07$	18	0.479
	Extended Global Arnoldi	$1.532e-05$	$1.531e-05$	$3.731e-07$	9	0.429
	Extended Global Hessenberg	$1.532e-05$	$1.531e-05$	$3.731e-07$	8	0.267
Example 4 $n = 500$,$R = 3$	Global Arnoldi	$1.980e-07$	$1.980e-07$	$2.698e-08$	5	0.049
	Global Hessenberg	$1.980e-07$	$1.984e-07$	$2.704e-08$	4	0.046
	Extended Global Arnoldi	$1.980e-07$	$1.980e-07$	$2.698e-08$	3	0.082
	Extended Global Hessenberg	$1.980e-07$	$1.980e-07$	$2.698e-08$	3	0.077
Example 5 $n = 500$, $R = 3$	Global Arnoldi	$1.038e-08$	$1.034e-08$	$2.567e-09$	12	0.120
	Global Hessenberg	$1.038e-08$	$1.161e-08$	$2.622e-09$	11	0.144
	Extended Global Arnoldi	$1.038e-08$	$1.042e-08$	$2.567e-09$	5	0.115
	Extended Global Hessenberg	$1.038e-08$	$1.034e-08$	$2.566e-09$	5	0.112

As can be seen from Table 2 and Figure 1, Global Arnoldi, Extended Global Arnoldi, and Global Hessenberg, Extended Global Hessenberg methods are shown a similar behavior. In addition, for all examples, the number of cycles of Extended Global Hessenberg is less than or equal to that of the other methods. In Examples 1, 2, 3, and 5, the CPU time of Extended Global Hessenberg method is less than the others. The results of Example 4 show that when the required number of cycles is small for Global Hessenberg method, this method outperforms the other methods in terms of CPU times.



(a) Example 1 (b) Example 2

(c) Example 3 (d) Example 4

(e) Example 5

Figure 1: Convergence history of the global and extended global Arnoldi and Hessenberg algorithms for Examples 1–5.

7 Conclusion

In this study, for computing the approximate solutions of the Sylvester tensor equation (1) with the low-rank right-hand side, two new projection methods based on the Hessenberg process were proposed. The theoretical results of these methods were presented and analyzed as well. The global and extended global Hessenberg algorithms were compared, in terms of CPU times, cycles, and the number of operations, with the global and extended global Arnoldi algorithms, respectively. Numerical examples showed that the global and extended global Hessenberg algorithms are efficient and feasible for solving the low-rank Sylvester tensor equation (1).

Acknowledgment

The authors would like to thank the anonymous referees for their comments and suggestions, which considerably improved the paper. All computations of this paper were carried out in the High-Performance Computing Center of Shahrekord university. The authors of this paper would like to express their gratitude for this support.

References

1. Addam, M., Elbouyahyaoui, L. and Heyouni, M. *On Hessenberg type methods for low-rank Lyapunov matrix equations,* Applicationes Mathematicae 45 (2018) 255–273.

2. Bader, B.W. and Kolda, T.G. *MATLAB tensor toolbox version 3.2.* http://www.sandia.gov/tgkolda/TensorToolbox/.

3. Ballani, J. and Grasedyck, L. *A projection method to solve linear systems in tensor format,* Numer. Linear Algebra Appl. 20 (1) (2013) 27–43.

4. Beik, F.P.A., Saberi Movahed, F. and Ahmadi-Asl, S. *On the krylov subspace methods based on tensor format for positive definite Sylvester tensor equations,* Numer. Linear Algebra Appl. 23 (3) (2016) 444–466.

5. Bentbib, A.H., El-Halouy, S. and Sadek, El M. *Krylov subspace projection method for Sylvester tensor equation with low rank right-hand side,* Numer. Algorithms 84 (4) (2020) 1411–1430.

6. Bentbib, A.H., El-Halouy, S. and Sadek, El M. *Extended Krylov subspace methods for solving Sylvester and Stein tensor equations,* Discrete Continuous Dyn. Syst.-s 15 (1) (2022) 41–56.

7. Bouyouli, R., Jbilou, K., Sadaka, R. and Sadok, H. *Convergence properties of some block Krylov subspace methods for multiple linear systems,* J. Comp. Appl. Math. 196 (2) (2006) 498–511.

8. Chang, K.C., Pearson, K. and Zhang, T. *Perron-Frobenius theorem for nonnegative tensors,* Commun. Math. Sci. 6 (2) (2008) 507–520.

9. Chen, Z. and Lu, L.Z. *A projection method and Kronecker product preconditioner for solving Sylvester tensor equations,* Sci. China Math. 55 (6) (2012) 1281–1292.

10. Chen, Z. and Lu, L.Z. *A gradient based iterative solutions for Sylvester tensor equations,* Math. Probl. Eng. (2013) Article ID 819479, 7 pp.

11. Chen, M. and Kressner, D. *Recursive blocked algorithms for linear systems with Kronecker product structure,* Numer. Algorithms 84 (3) (2020) 1199–1216.

12. Cheraghzadeh, T., Khoshsiar Ghaziani, R. and Toutounian, F. *Projection schemes based on Hessenberg process for Sylvester tensor equation with low-rank right-hand side,* Comput. Appl. Math. 41 (2022) Article number 311.

13. Colda, T.G. and Bader, B.W. *Tensor decompositions and applications,* SIAM Rev. 51 (3) (2009) 455–500.

14. Cichocki, A., Zdunek, R., Phan, A.H. and Amari, S.-I. *Nonnegative matrix and tensor factorizations: Applications to exploratory multi-way data analysis and blind source separation,* John Wiley and Sons, 2009.

15. Dehdezi, E.K. and Karimi, S. *Extended conjugate gradient squared and conjugate residual squared methods for solving the generalized coupled Sylvester tensor equations,* T. I. Meas. Control. 9(4)(2021) 645–664.

16. Dehdezi, E.K. and Karimi, S. *A gradient based iterative method and associated preconditioning technique for solving the large multilinear systems,* Calcolo. 58(4) (2021) 1–19.

17. Heyouni, M. *The global Hessenberg and global CMRH methods for linear systems with multiple right-hand sides,* Numer. Algorithms 26 (4) (2001) 317–332.

18. Heyouni, M. *Extended Arnoldi methods for large low-rank Sylvester matrix equations,* Appl. Numer. Math. 60 (11) (2010) 1171–1182.

19. Heyouni, M. and Essai, A. *Matrix Krylov subspace methods for linear systems with multiple right-hand sides,* Numer. Algorithms 40 (2) (2005) 137–156.

20. Heyouni M. and Jbilou, K. *An extended block Arnoldi algorithm for large-scale solution of the continuous-time algebraic Riccati equation,* Electron. Trans. Numer. Anal. 33 (2009) 53–62.

21. Heyouni, M., Saberi-Movahed, F. and Tajaddini, A. *A tensor format for the generalized Hessenberg method for solving Sylvester tensor equations,* J. Comput. Appl. Math. 377 (2020) 112878.

22. Huang, B., Xie, Y. and Ma, C. *Krylov subspace methods to solve a class of tensor equations via the Einstein product,* Numer. Linear Algebra Appl. 26 (2019) e2254.

23. Jbilou, K., Messaoudi, A. and Sadok, H. *Global FOM and GMRES algorithms for matrix equations,* Appl. Numer. Math. 31 (1) (1999) 49–63.

24. Karimi, S. and Dehghan, M. *Global least squares method based on tensor form to solve linear systems in Kronecker format,* Trans. Inst. Measur. Contr. 40 (7) (2018) 2378–2386.

25. Kressner, D. and Tobler, C. *Low-rank tensor Krylov subspace methods for parametrized linear systems,* SIAM J. Matrix Anal. Appl. 32 (4) (2011) 1288–1316.

26. Lee, N. and Cichocki, *A. Fundamental tensor operations for large-scale data analysis using tensor network formats,* Multidim. Syst. Sign. Process. 29 (3) (2018) 921–960.

27. Li, B.W., Tian, S., Sun, Y.S. and Hu, Z.M. *Schur decomposition for 3d matrix equations and its application in solving radiative discrete ordinates equations discretized by Chebyshev collocation spectral method,* J. Comput. Phys. 229 (4) (2010) 1198–1212 .

28. Malek, A. and Momeni-Masuleh, S.H. *A mixed collocation finite difference method for 3d microscopic heat transport problems,* J. Comput. Appl. Math. 217 (1) (2008) 137–147.

29. Penzl, T. 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).

30. Mele, G. and Jarlebring, E. *On restarting for the tensor infinite Arnoldi method,* BIT 58 (1) (2018) 133–162.

31. Qi, L. *Eigenvalues of a real supersymmetric tensor,* J. Symbolic Comput. 40 (6) (2005) 1302–1324.

32. Ramezani, Z. and Toutounian, F. *Extended and rational Hessenberg methods for the evaluation of matrix functions,* BIT Numer. Math. 59, (2019) 523–545.

33. Saberi-Movahed, F., Tajaddini, A., Heyouni, M. and Elbouyahyaoui, L. *Some iterative approaches for Sylvester tensor equations, PartI: A tensor format of truncated Loose Simpler GMRES,* Appl. Numer. Math. 172 (2022) 428–445.

34. Saberi-Movahed, F., Tajaddini, A., Heyouni, M. and Elbouyahyaoui, L. *Some iterative approaches for Sylvester tensor equations, Part II: A tensor format of truncated Loose Simpler GMRES,* Appl. Numer. Math. 172 (2022) 428–445.

35. Sadok, H. *CMRH: A new method for solving nonsymmetric linear systems based on the Hessenberg reduction algorithm,* Numer. Algorithms 20 (4) (1999) 303–321.

36. Simoncini, V. *A new iterative method for solving large-scale Lyapunov matrix equations,* SIAM J. Sci. Comp. 29 (3) (2007) 1268–1288.

37. Sun, Y., Ma, J. and Li, B.W. *Chebyshev collocation spectral method for three-dimensional transient coupled radiative conductive heat transfer,* J. Heat Transfer 134 (9) (2012):092701.

38. Xu, X. and Wang, Q.-W. *Extending Bi-CG and Bi-CR methods to solve the Stein tensor equation,* Comput. Math. Appl. 77 (12) (2019) 3117–3127.

39. Zhang, X.-F.and Wang, Q.-W. *Developing iterative algorithms to solve Sylvester tensor equations,* Appl. Math. Comput. 409 (2021) 126403.

How to cite this article

T. Cheraghzadeh, F. Toutounian and R. Khoshsiar Ghaziani Global and extended global Hessenberg processes for solving Sylvester tensor equation with low-rank right-hand side. *Iranian Journal of Numerical Analysis and Optimization*, 2022; 12(3 (Special Issue), 2022): 658-679. doi: 10.22067/ijnao.2022.78966.1186.